

# **EXHIBIT 30**

**IN THE UNITED STATES DISTRICT COURT  
FOR THE EASTERN DISTRICT OF TEXAS  
MARSHALL DIVISION**

<b>AGIS SOFTWARE DEVELOPMENT LLC,</b>	§	
	§	
	§	
<b>Plaintiff,</b>	§	<b>Civil Action No. 2:17-CV-514-JRG (Lead Case)</b>
	§	
v.	§	
	§	<b><u>JURY TRIAL DEMANDED</u></b>
<b>HTC CORPORATION.,</b>	§	
	§	
<b>Defendant.</b>	§	
	§	

<b>AGIS SOFTWARE DEVELOPMENT LLC,</b>	§	
	§	
	§	
<b>Plaintiff,</b>	§	<b>Civil Action No. 2:17-CV-517-JRG (Consolidated Case)</b>
	§	
v.	§	
	§	<b><u>JURY TRIAL DEMANDED</u></b>
<b>ZTE CORPORATION ET AL.,</b>	§	
	§	
<b>Defendants.</b>	§	
	§	

**PLAINTIFF’S DISCLOSURE OF ASSERTED CLAIMS  
AND INFRINGEMENT CONTENTIONS**

Plaintiff AGIS Software Development LLC (“AGIS”) hereby makes the following infringement disclosures under the Patent Local Rules with respect to United States Patent Nos. 9,467,838 (the “838 Patent”), 9,445,251 (the “251 Patent”), 9,408,055 (the “055 Patent”), and 8,213,970 (the “970 Patent”) (collectively, “patents-in-suit”). AGIS’s investigation is ongoing and discovery has not yet commenced. Accordingly, these disclosures are based on information available to AGIS at this time. AGIS reserves the right to supplement this disclosure after further

discovery from the defendants and non-parties, particularly documents and other discovery regarding the HTC Accused Products set forth below. AGIS also reserves the right to assert additional claims of the patents-in-suit, accuse different products, or find alternative literal and/or equivalent infringing elements in the HTC Accused Products.

**I. DISCLOSURE OF ASSERTED CLAIMS AND INFRINGEMENT CONTENTIONS PURSUANT TO PATENT LOCAL RULE 3-1**

**A. ASSERTED CLAIMS**

Defendant HTC Corporation (“Defendant” or “HTC”) has infringed and continues to infringe at least the following claims of the patents-in-suit in connection with the HTC Accused Products set forth below:

- Claims 1–54 of the ’838 Patent;
- Claims 1–35 of the ’251 Patent;
- Claims 1–54 of the ’055 Patent; and
- Claims 1 and 3–9 of the ’970 Patent.

AGIS reserves the right to seek leave of court to add, delete, substitute, or otherwise amend this list of asserted claims should further discovery, the Court’s claim construction, or other circumstances so merit.

**B. ACCUSED INSTRUMENTALITIES**

AGIS is currently aware that the following HTC Accused Products infringe each of the patents-in-suit either alone or in concert with one or more other HTC Accused Products:

- 10
- 10 evo
- 10 Lifestyle
- 2125 / 2100 (Faraday)
- 3125 / Smartflip / 8500 (Star Trek)

- 5800 / Fusion / S720
- 7 Mozart
- 7 Pro
- 7 Surround
- 7 Trophy
- 8125 / 8100 / MDA (USA) / K-JAM / P4300 (Wizard)
- 8XT
- AD phones continues below...
- Amaze 4G
- Aria
- Arrive
- Arrive / 7 Pro (CDMA)
- Bolt
- Butterfly
- Butterfly 2
- Butterfly 3
- Butterfly S
- ChaCha
- Dash / S620 / S621 (Excalibur)
- Dash 3G / Snap (GSM)
- Desire

- Desire (CDMA)
- Desire / Desire 601 (CDMA)
- Desire 10 Compact
- Desire 10 Lifestyle
- Desire 10 Pro
- Desire 200
- Desire 210 dual sim
- Desire 300
- Desire 310
- Desire 310 dual sim
- Desire 320
- Desire 326G dual sim
- Desire 400 dual sim
- Desire 500
- Desire 501
- Desire 501 dual sim
- Desire 510
- Desire 510 (CDMA)
- Desire 510 (GSM)
- Desire 516 dual sim
- Desire 520

- Desire 526
- Desire 526 (CDMA)
- Desire 526G+ dual sim
- Desire 530
- Desire 555
- Desire 600 dual sim
- Desire 601
- Desire 601 dual sim
- Desire 610
- Desire 610 (GSM)
- Desire 612
- Desire 612 (CDMA)
- Desire 616 dual sim
- Desire 620
- Desire 620G dual sim
- Desire 625
- Desire 626
- Desire 626 (CDMA)
- Desire 626 (GSM)
- Desire 626 (USA)
- Desire 626G+

- Desire 626s
- Desire 626s (CDMA)
- Desire 626s (GSM)
- Desire 628
- Desire 630
- Desire 650
- Desire 700
- Desire 700 dual sim
- Desire 728 dual sim
- Desire 728 Ultra
- Desire 816
- Desire 816 dual sim
- Desire 816G dual sim
- Desire 820
- Desire 820 dual sim
- Desire 820G+ dual sim
- Desire 820q dual sim
- Desire 820s dual sim
- Desire 825
- Desire 826 dual sim
- Desire 828 dual sim



- Desire 830
- Desire C
- Desire C (CDMA)
- Desire Eye
- Desire HD
- Desire L
- Desire P
- Desire Q
- Desire S
- Desire SV
- Desire U
- Desire V
- Desire VC
- Desire VT
- Desire X
- Desire XC
- Desire Z
- Dream
- DROID DNA
- DROID ERIS
- Droid Incredible

- DROID Incredible 2
- DROID Incredible 4G LTE
- EVO 3D
- EVO 3D CDMA
- Evo 4G
- Evo 4G LTE
- Evo 4G+
- EVO Design 4G
- EVO Design 4G / Hero S (CDMA)
- EVO Shift 4G
- EVO V 4G / EVO 3D (CDMA)
- EVO View 4G
- Explorer
- First
- Flyer
- Flyer Wi-Fi
- Freestyle
- Fuze / Touch Pro (GSM)
- G1
- G2
- Glacier

- Gratia
- HD mini
- HD2
- HD7
- HD7 / HD7S
- HD7S
- Hero
- Hero (CDMA)
- Hero CDMA
- Hero S
- Imagio
- Incredible S
- Inspire 4G
- J
- JAMin / S200 (Prophet)
- Jetstream
- Lead
- Legend
- Magic
- MAX 4G
- MDA Compact / xda II mini / JAM (Magician)

- Merge
- Mogul / XV6800 / PPC6800 / P4000
- myTouch 3G / Magic
- myTouch 3G Slide
- myTouch 4G
- myTouch 4G Slide
- One
- One (E8)
- One (E8) CDMA
- One (M7 / CDMA)
- One (M7 / GSM)
- One (M8 Eye)
- One (M8)
- One (M8) (CDMA)
- One (M8) (GSM)
- One (M8) CDMA
- One (M8) dual sim
- One A9
- One A9s
- One Dual Sim
- One E9

- One E9+
- One M8s
- One M9
- One M9 (CDMA)
- One M9 (GSM)
- One M9 Prime Camera
- One M9+
- One M9+ Supreme Camera
- One M9s
- One Max
- One max (CDMA)
- One ME
- One mini
- One mini 2
- One mini 2 (GSM)
- One Remix
- One Remix / One mini 2 (CDMA)
- One S
- One S C2
- One S9
- One SC

- One ST
- One SV
- One SV CDMA
- One V
- One VX
- One X
- One X AT&T
- One X+
- One X10
- One X9
- One XC
- One XL
- Ozone
- Ozone XV6175
- Panache
- Paradise
- Prime
- Pure
- Pure / Touch Diamond2
- Radar
- Raider 4G

- Rezound
- Rhyme
- Rhyme CDMA
- Rider
- S710 (Vox)
- S730
- S740
- Salsa
- Schubert
- SDA (USA) / SP5m (Tornado)
- Sensation
- Sensation 4G
- Sensation XE
- Sensation XL
- Shadow
- Shadow (2009)
- Smart
- Snap
- Snap S511 (CDMA)
- SP3i / SDA (Europe) (Feeler)
- SPV C550 (Hurricane)

- SPV E200 / XPhone (Voyager)
- Status
- Surround
- Tattoo
- ThunderBolt
- ThunderBolt 4G
- Tilt 8925 / TyTN II
- Tilt2
- Titan
- Titan II
- Touch (CDMA) / XV6900
- Touch 3G
- Touch Cruise
- Touch Cruise 09
- Touch Diamond (CDMA)
- Touch Diamond2
- Touch Diamond2 CDMA
- Touch Dual
- Touch HD
- Touch HD T8285
- Touch Pro



- Touch Pro (CDMA)
- Touch Pro CDMA
- Touch Pro2
- Touch Pro2 (CDMA)
- Touch Pro2 (GSM) / Tilt 2
- Touch Pro2 CDMA
- Touch Viva
- Touch2
- Trophy
- Trophy (CDMA)
- TyTN / 8525 / JasJam (Hermes)
- U Play
- U Ultra
- U Ultra
- U11
- U11
- U11 Eyes
- U11 Life
- U11 Plus
- U11+
- Velocity 4G

- Vivid
- Wildfire
- Wildfire (CDMA)
- Wildfire CDMA
- Wildfire S
- Wildfire S (CDMA)
- Wildfire S (GSM)
- Wing / P4350 (Herald)
- xda II / MDA II
- FLYER
- JETSTREAM
- FLYER WI-FI
- EVO View 4G
- FLYER CDMA

AGIS reserves the right to amend this list of accused instrumentalities, as well as other information contained in this document and the exhibits hereto, to incorporate new information learned during the course of discovery, including, but not limited to, the inclusion of newly-released products or any other equivalent devices ascertained through discovery.

**C. CLAIM CHARTS**

Claim charts identifying a location of every element of every asserted claim of the patents-in-suit within HTC Accused Products are attached hereto as Exhibits A–D. AGIS believes that

the citations in the claim charts are representative of all HTC Accused Products. For example, where AGIS cites reference material or images representing a phone or tablet, that citation is representative for all other such phones or tablets including all prior and future versions unless otherwise noted. AGIS reserves the right to amend these claim charts as well as other information contained in this document and the exhibits hereto, to incorporate new information learned during the course of discovery, including, but not limited to, information that is not publically available or readily discernible without discovery. AGIS further reserves the right to amend these claim charts, as well as other information contained in this document and the exhibits attached hereto, pursuant to Patent Local Rules 3-1(g) and 3-6.

**D. LITERAL INFRINGEMENT AND DOCTRINE OF EQUIVALENTS**

AGIS asserts that, under the proper construction of the asserted claims and their claim terms, the limitations of the asserted claims of the patents-in-suit are literally present in the HTC Accused Products as set forth in the claim charts attached hereto as Exhibits A–D. AGIS contends that any and all elements found not to be literally infringed are infringed under the doctrine of equivalents because the differences between the claimed inventions and the accused instrumentalities, if any, are insubstantial.

AGIS contends that HTC directly infringes the asserted claims by making, using, offering for sale, selling, and importing in to the United States the accused instrumentalities as well as indirectly infringe by contributing to and/or inducing others (e.g., HTC customers or its HTC customers’ customers) to directly infringe those claims by making, using, offering for sale or selling the HTC Accused Products. AGIS contends that HTC directly infringes the asserted claims by testing the HTC Accused Products in the United States. AGIS contends that HTC

infringes the asserted claims by contributing to and/or inducing Mobile Network Operators (MNOs) and/or jointly-and-directly infringing the asserted claims with MNOs by making, using, offering for sale, selling, and importing in to the United States the accused instrumentalities.

Pursuant to Patent Local Rule 3-6(a)(1), AGIS reserves the right to amend its Infringement Contentions as to literal infringement or infringement under the doctrine of equivalents, e.g., in light of the Court's claim construction.

**E. PRIORITY DATES**

Under P.R. 3-1(e), each of the asserted claims of the patents-in-suit are entitled to a priority date of at least as early as September 21, 2004. AGIS reserves the right to establish an earlier date of invention based upon actions related to conception and reduction to practice of the claimed inventions.

**F. AGIS'S OWN PRODUCTS**

At the present time, AGIS does not intend to rely on the assertion that its own apparatuses, products, devices, processes, methods, acts, or other instrumentalities practice the claimed inventions. AGIS reserves the right to amend this statement to identify covered products as discovery progresses.

**II. PRODUCTION OF DOCUMENTS PURSUANT TO PATENT LOCAL RULE 3-2**

AGIS is producing or making available for inspection documents that are in AGIS'S possession, custody or control as set forth in Patent Local Rule 3-2. An AGIS 3-2 Production Index identifying these documents is attached hereto.

This preliminary identification of documents is for convenience and is not an admission that each document falls within any exemplary categories in the Patent Local Rules, or that any

document qualifies as prior art. AGIS is continuing with its investigation, particularly with respect to ESI. Thus, AGIS reserves its right to add to, delete from, or otherwise modify its disclosures in this section as its investigation proceeds.

Production of these documents is governed by Patent Local Rule 2-2, and, with the exception of documents produced pursuant to P.R. 3.2(c) and public documents listed in the infringement charts, are considered “Confidential – Outside Attorneys Eyes Only” and disclosure of the confidential document or information shall be limited to each party’s outside attorney(s) of record and the employees of such outside attorney(s).

Dated: January 19, 2018

Respectfully submitted,

**BROWN RUDNICK LLP**

*/s/ Vincent J. Rubino*

---

Alfred R. Fabricant  
NY Bar No. 2219392  
Email: afabricant@brownrudnick.com  
Peter Lambrianakos  
NY Bar No. 2894392  
Email: plambrianakos@brownrudnick.com  
Vincent J. Rubino, III  
NY Bar No. 4557435  
Email: vrubino@brownrudnick.com  
Alessandra C. Messing  
NY Bar No. 5040019  
Email: amessing@brownrudnick.com  
John A. Rubino  
NY Bar No. 5020797  
Email: jrubino@brownrudnick.com  
Enrique W. Iturralde  
NY Bar No. 5526280  
eiturralde@brownrudnick.com

**BROWN RUDNICK LLP**

7 Times Square  
New York, NY 10036

PLAINTIFF’S DISCLOSURE OF ASSERTED CLAIMS  
AND INFRINGEMENT CONTENTIONS

Telephone: (212) 209-4800

Facsimile: (212) 209-4801

Samuel F. Baxter

Texas State Bar No. 01938000

[sbaxter@mckoolsmith.com](mailto:sbaxter@mckoolsmith.com)

Jennifer L. Truelove

Texas State Bar No. 24012906

[jtruelove@mckoolsmith.com](mailto:jtruelove@mckoolsmith.com)

**MCKOOL SMITH, P.C.**

104 e. Houston Street, Suite 300

Marshall, Texas 75670

Telephone : (903) 923-9000

Facsimile (903) 923-9099

***ATTORNEYS FOR PLAINTIFF,  
AGIS SOFTWARE DEVELOPMENT LLC***

**CERTIFICATE OF SERVICE**

I hereby certify that on January 19, 2018 a true and correct copy of the above and foregoing document has been served by email on:

Matthew C. Bernstein  
**Perkins Coie LLP**  
11988 El Camino Real, Suite 350  
San Diego, CA 92130-2594  
mbernstein@perkinscoie.com

Miguel J. Bombach  
Perkins Coie LLP  
11988 El Camino Real, Suite 350  
San Diego, CA 92130-2594  
mbombach@perkinscoie.com

James Young Hurt  
**Perkins Coie LLP**  
11988 El Camino Real, Suite 350  
San Diego, CA 92130-2594  
jhurt@perkinscoie.com

*Attorneys of Record for HTC Corporation*

/s/ Vincent J. Rubino, III

Vincent J. Rubino, III

**AGIS 3-2 PRODUCTION INDEX**

**3-2(a)**

<b>Bates Start</b>	<b>Bates End</b>
AGISTX_00007059	AGISTX_00007071

**3-2(b)**

<b>Bates Start</b>	<b>Bates End</b>
AGISTX_00006047	AGISTX_00006191
AGISTX_00007035	AGISTX_00007058

**3-2(c)**

<b>Bates Start</b>	<b>Bates End</b>
AGISTX_00000001_	AGISTX_00006046

**Public Documents Listed in Infringement Charts<sup>1</sup>**

<b>Bates Start</b>	<b>Bates End</b>
AGISTX_00007106	AGISTX_00007788
AGISTX_00007789	AGISTX_00008277
AGIS_HTC00000001	AGIS_HTC00000160

---

<sup>1</sup> Exemplary Source Code excerpts were taken from the Android 7 Nougat Release 1.0 at <https://android.googleusercontent.com/> as set forth in Exhibits A–D. AGIS expects that Defendant will produce this release, and any additional relevant releases, for the Accused Products in its entirety.



## **Exhibit A for US Patent No. 8,213,970 Against HTC Accused Products**

In these Infringement Contentions, AGIS Software Development LLC (“AGIS”) contends that at least the following claims of U.S. Patent No. 8,213,970 (the “’970 Patent”) identified below are infringed by the Accused Products (*e.g.*, phones and tablets) which are manufactured, sold, offered for sale, and/or used by HTC Corporation (“HTC”).

The Accused Products comprise HTC products running the Android mobile operating system and manufactured, used, or sold during and after 2011. For example, the Accused Products comprise the following Android-based phones and tablets: 10, 10 evo, 10 Lifestyle, 2125 / 2100 (Faraday), 3125 / Smartflip / 8500 (Star Trek), 5800 / Fusion / S720, 7 Mozart, 7 Pro, 7 Surround, 7 Trophy, 8125 / 8100 / MDA (USA) / K-JAM / P4300 (Wizard), 8XT, Amaze 4G, Aria, Arrive, Arrive / 7 Pro (CDMA), Bolt, Butterfly, Butterfly 2, Butterfly 3, Butterfly S, ChaCha, Dash / S620 / S621 (Excalibur), Dash 3G / Snap (GSM), Desire, Desire (CDMA), Desire / Desire 601 (CDMA), Desire 10 Compact, Desire 10 Lifestyle, Desire 10 Pro, Desire 200, Desire 210 dual sim, Desire 300, Desire 310, Desire 310 dual sim, Desire 320, Desire 326G dual sim, Desire 400 dual sim, Desire 500, Desire 501, Desire 501 dual sim, Desire 510, Desire 510 (CDMA), Desire 510 (GSM), Desire 516 dual sim, Desire 520, Desire 526, Desire 526 (CDMA), Desire 526G+ dual sim, Desire 530, Desire 555, Desire 600 dual sim, Desire 601, Desire 601 dual sim, Desire 610, Desire 610 (GSM), Desire 612, Desire 612 (CDMA), Desire 616 dual sim, Desire 620, Desire 620G dual sim, Desire 625, Desire 626, Desire 626 (CDMA), Desire 626 (GSM), Desire 626 (USA), Desire 626G+, Desire 626s, Desire 626s (CDMA), Desire 626s (GSM), Desire 628, Desire 630, Desire 650, Desire 700, Desire 700 dual sim, Desire 728 dual sim, Desire 728 Ultra, Desire 816, Desire 816 dual sim, Desire 816G dual sim, Desire 820, Desire 820 dual sim, Desire 820G+ dual sim, Desire 820q dual sim, Desire 820s dual sim, Desire 825, Desire 826 dual sim, Desire 828 dual sim, Desire 830, Desire C, Desire C (CDMA), Desire Eye, Desire HD, Desire L, Desire P, Desire Q, Desire S, Desire SV, Desire U, Desire V, Desire VC, Desire VT, Desire X, Desire XC, Desire Z, Dream, DROID DNA, DROID ERIS, Droid Incredible, DROID Incredible 2, DROID Incredible 4G LTE, EVO 3D, EVO 3D CDMA, Evo 4G, Evo 4G LTE, Evo 4G+, EVO Design 4G, EVO Design 4G / Hero S (CDMA), EVO Shift 4G, EVO V 4G / EVO 3D (CDMA), EVO View 4G, Explorer, First, Flyer, Flyer Wi-Fi, Freestyle, Fuze / Touch Pro (GSM), G1, G2, Glacier, Gratia, HD mini, HD2, HD7, HD7 / HD7S, HD7S, Hero, Hero (CDMA), Hero S, Imagio, Incredible S, Inspire 4G, J, JAMin / S200 (Prophet), Jetstream, Lead, Legend, Magic, MAX 4G, MDA Compact / xda II mini / JAM (Magician), Merge, Mogul / XV6800 / PPC6800 / P4000, myTouch 3G / Magic, myTouch 3G Slide, myTouch 4G, myTouch 4G Slide, One, One (E8), One (E8) CDMA, One (M7 / CDMA), One (M7 / GSM), One (M8 Eye), One (M8), One (M8) (CDMA), One (M8) (GSM), One (M8) CDMA, One (M8) dual sim, One A9, One A9s, One Dual Sim, One E9, One E9+, One M8s, One M9, One M9 (CDMA), One M9 (GSM), One M9 Prime Camera, One M9+, One M9+ Supreme Camera, One M9s, One Max, One max (CDMA), One ME, One mini, One mini 2, One mini 2 (GSM), One Remix, One Remix / One mini 2 (CDMA), One S, One S C2, One S9, One SC, One ST, One SV, One SV CDMA, One V, One VX, One X, One X

## **Exhibit A for US Patent No. 8,213,970 Against HTC Accused Products**

AT&T, One X+, One X10, One X9, One XC, One XL, Ozone, Ozone XV6175, Panache, Paradise, Prime, Pure, Pure / Touch Diamond2, Radar, Raider 4G, Rezound, Rhyme, Rhyme CDMA, Rider, S710 (Vox), S730, S740, Salsa, Schubert, SDA (USA) / SP5m (Tornado), Sensation, Sensation 4G, Sensation XE, Sensation XL, Shadow, Shadow (2009), Smart, Snap, Snap S511 (CDMA), SP3i / SDA (Europe) (Feeler), SPV C550 (Hurricane), SPV E200 / XPhone (Voyager), Status, Surround, Tattoo, ThunderBolt, ThunderBolt 4G, Tilt 8925 / TyTN II, Tilt2, Titan, Titan II, Touch (CDMA) / XV6900, Touch 3G, Touch Cruise, Touch Cruise 09, Touch Diamond (CDMA), Touch Diamond2, Touch Diamond2 CDMA, Touch Dual, Touch HD, Touch HD T8285, Touch Pro, Touch Pro (CDMA), Touch Pro2, Touch Pro2 (CDMA), Touch Pro2 (GSM) / Tilt 2, Touch Viva, Touch2, Trophy, Trophy (CDMA), TyTN / 8525 / JasJam (Hermes), U Play, U Ultra, U Ultra , U11, U11, U11 Eyes, U11 Life, U11 Plus, U11+, Velocity 4G, Vivid, Wildfire, Wildfire (CDMA), Wildfire S, Wildfire S (CDMA), Wildfire S (GSM), Wing / P4350 (Herald), xda II / MDA II, FLYER, JETSTREAM, FLYER WI-FI, EVO View 4G, FLYER CDMA, and any variants thereof. AGIS reserves the right to amend this list of Accused Products as discovery progresses. For example, AGIS reviewed Android-based products from multiple Android-based handset manufacturers, including two HTC phones (serial numbers FA73J1500645, FA73E1500899) which are available for inspection at HTC's request. For example, the Accused Products comprise HTC products, including but not limited to the phones and tablets as described herein, running the following versions (and all intervening updates and sub-versions) of the Android mobile operating system: Android 2.3, 4.0, 4.1, 4.2, 4.3, 4.4, 5.0, 5.1, 6.0, 7.0, 7.1, 8.0, and 8.1. For example, the Accused Products comprise HTC products, including but not limited to the phones and tablets as described herein, running any versions of the following Android-based applications and/or software: Android Device Manager, Find My Phone, Find My Device, Google Latitude, Google Plus, Google Hangouts, Google Maps, Google Assistant, Google Search, Google Messages, Android Messenger, Google Allo, Google Duo, Gmail, and Google Chrome. For example, the Accused Products comprise HTC products, including but not limited to the phones and tablets described herein, participating in any networks and/or services related to the execution and/or use of the Android mobile operating system versions and Android-based applications and/or software described herein.

AGIS does not concede that any claims of the '970 Patent that are not listed below are not infringed by the identified products. Moreover, the citations to certain documents and other information below are intended to be exemplary only and in no way foreclose AGIS from citing or relying on additional documents, information, source code, and/or testimony at a later time. These contentions are preliminary in nature, and an analysis of HTC's products, internal documentation, source code, and/or testimony from relevant witnesses may more fully and accurately describe the infringing features of its accused products. Accordingly, AGIS reserves the right to supplement, correct, modify, and/or amend these contentions once such additional information is made available to AGIS. Furthermore, AGIS reserves the right to supplement, correct, modify, and/or amend these contentions as discovery in this case

**Exhibit A for US Patent No. 8,213,970 Against HTC Accused Products**

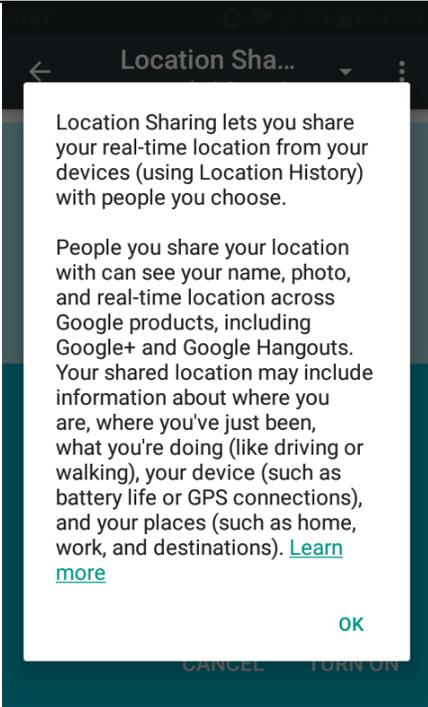
progresses; in view of the Court’s claim construction order(s); in view of any positions taken by HTC, including but not limited to positions on claim construction, invalidity, and/or non-infringement; and in connection with the preparation and exchange of expert reports.

Claim - 8,213,970	Accused Products
<p>1[P]. A communication system for transmitting, receiving, confirming receipt, and responding to an electronic message, comprising: a predetermined network of participants, wherein each participant has a similarly equipped PDA/cell phone that includes a CPU and a touch screen display a CPU and memory;</p>	<p>HTC infringes either directly or indirectly, induces others to infringe, and/or contributes to the infringement of this system as set forth below. HTC makes, uses, sells, and otherwise provides this system by making, using, selling, and importing Android devices, as well as by providing its servers for use with Android devices to enable features such as Find My Device in a communication system for transmitting, receiving, confirming receipt, and responding to an electronic message, comprising: a predetermined network of participants, wherein each participant has a similarly equipped PDA/cell phone that includes a CPU and a touch screen display a CPU and memory.</p> <p>HTC makes, uses, sells, and otherwise provides this communication system by making, using, selling, and importing Android OS devices such as Accused Products.</p> <p>The Accused Products meet the claim limitations by providing device-location tracking features such as those features described below. For example, the Accused Products meet the claim limitations because they are pre-installed with Android mobile operating systems containing code for providing device-location tracking features as provided in the claims limitations herein. For example, the Accused Products run applications and/or software that run within the Android mobile operating system and that use components of the Android mobile operating system to provide device-location tracking features. Upon information and belief, in addition to the components and features of the Android mobile operating system itself, the following applications and/or software run within the Android mobile operating system and use components of the Android mobile operating system to provide device-location tracking features: Android Device Manager, Find My Device.</p> <p>Android Device Manager is the predecessor to Find My Device and has been available as a standard, pre-installed feature since 2013 and downloadable as a software application. The</p>

**Exhibit A for US Patent No. 8,213,970 Against HTC Accused Products**

Claim - 8,213,970	Accused Products
	<p>current iteration, Find My Device, often called the “new and improved Android Device Manager” or “rebranded Android Device Manager” is now part of the standard Google Play Protect suite which is “built in and enabled on all devices,” i.e., the Accused Products running Android OS. AGIS sets forth the Find My Device feature of the Accused Products as representative of this method. AGIS reserves the right to supplement these contentions to the extent that HTC requires additional information in accordance with P.R. 3-1 and for any other reason for which it may deem necessary.</p> <p>See, e.g., <a href="https://www.androidcentral.com/find-my-device">https://www.androidcentral.com/find-my-device</a>; <a href="https://support.google.com/android/answer/6160491?hl=en">https://support.google.com/android/answer/6160491?hl=en</a>; <a href="https://android.googleblog.com/2013/08/find-your-lost-phone-with-android.html">https://android.googleblog.com/2013/08/find-your-lost-phone-with-android.html</a>; <a href="https://play.google.com/store/apps/details?id=com.google.android.apps.adm&amp;hl=en">https://play.google.com/store/apps/details?id=com.google.android.apps.adm&amp;hl=en</a>; <a href="https://www.blog.google/products/android/google-play-protect/">https://www.blog.google/products/android/google-play-protect/</a> Each “Google Account” is associated with a pre-determined number of devices, which include a CPU and a touchscreen, and which are registered when a customer acquires an Accused Product, such as a HTC Android OS based device.</p>

**Exhibit A for US Patent No. 8,213,970 Against HTC Accused Products**

Claim - 8,213,970	Accused Products
	 <p><b>Find your device using Android Device Manager</b></p> <p>If you've lost a device, you can use Android Device Manager to find its approximate location on a map and when it was last used. When Android Device Manager locates your device, that device will get a notification.</p> <p><b>Before you can use Android Device Manager to locate your device:</b> Your device's <a href="#">location access</a> need to be turned on <a href="#">🔗</a> and be signed in to your Google Account. Android Device Manager won't work for devices that are turned off or that don't have a mobile data or Wi-Fi connection.</p> <p><b>Tip:</b> If you've linked your phone to Google, you can locate or ring it by searching for <a href="#">find my phone</a> on <a href="#">google.com</a> <a href="#">🔗</a>.</p> <p><a href="https://support.google.com/pixelphone/answer/6160491">https://support.google.com/pixelphone/answer/6160491</a></p>

# Exhibit A for US Patent No. 8,213,970 Against HTC Accused Products

Claim - 8,213,970	Accused Products
	<p data-bbox="701 233 1094 264"><b>Link your phone to Google</b></p> <p data-bbox="701 280 1383 315">You can connect your Android phone to Google, which lets you send information from your computer to your phone. For example, you can send directions you searched for on your computer to Google Maps on your phone.</p> <p data-bbox="701 342 919 360"><b>Link your Android phone</b></p> <p data-bbox="701 386 919 404"><b>Step 1: Update the Google app</b></p> <ol data-bbox="701 409 1066 448" style="list-style-type: none"><li>1. On your phone, go to the <a href="#">Google app page on the Play Store</a>.</li><li>2. Tap <b>Update</b>.</li></ol> <p data-bbox="701 472 905 490"><b>Step 2: Turn on Google Now</b></p> <ol data-bbox="701 495 1031 578" style="list-style-type: none"><li>1. On your phone, open the Google app .</li><li>2. At the top left, tap Menu  &gt; <b>Settings</b> &gt; <b>Now cards</b>.</li><li>3. Turn on <b>Show cards</b>.</li><li>4. Turn on <b>Show notifications</b>.</li></ol> <p data-bbox="701 602 953 620"><b>Step 3: Turn on Web &amp; App Activity</b></p> <ol data-bbox="701 625 926 664" style="list-style-type: none"><li>1. Visit the <a href="#">Account History page</a>.</li><li>2. Make sure the switch is on (green).</li></ol> <p data-bbox="701 688 926 706"><b>Step 4: Sign in to your browser</b></p> <ol data-bbox="701 711 1157 837" style="list-style-type: none"><li>1. On your phone, open the Google app .</li><li>2. At the top left, tap the Menu .</li><li>3. At the top left, you'll see the email address you use for the Google app.</li><li>4. Visit <a href="http://www.google.com">www.google.com</a>  on your computer.</li><li>5. If you aren't signed in already, click <b>Sign in</b> in the top right corner of the page.</li><li>6. Sign in using the Google Account you use for the Google app.</li></ol> <p data-bbox="701 862 982 880"><b>Step 5: Send information to your phone</b></p> <ol data-bbox="701 885 1392 943" style="list-style-type: none"><li>1. Do one of the searches below, like <b>note to self</b>, or <b>send directions to my phone</b>.</li><li>2. If a box doesn't pop up with the option to send information to your phone, try refreshing the page. If you just turned on Google Now, it may take a few minutes for the box to show up.</li></ol> <p data-bbox="701 951 1398 979"><a href="https://support.google.com/websearch/answer/6128427">https://support.google.com/websearch/answer/6128427</a></p>

### Exhibit A for US Patent No. 8,213,970 Against HTC Accused Products

Claim - 8,213,970	Accused Products
	<p data-bbox="699 235 1192 259"><b>What you can do once your phone is linked</b></p> <hr/> <p data-bbox="720 293 848 313"><b>Find my phone</b> <span data-bbox="1560 293 1581 313">^</span></p> <p data-bbox="741 334 1222 354">You can get the current location of your phone if you can't find it.</p> <ol data-bbox="741 370 1570 467" style="list-style-type: none"><li>1. On your computer's browser, search on <a href="http://www.google.com">www.google.com</a> <span data-bbox="1192 370 1213 389">^</span> for <b>find my phone</b>.</li><li>2. If your phone is turned on and connected to the Internet, you'll see your phone's location.</li><li>3. If your phone's location is unavailable, you can still make it ring for 5 minutes on full volume by clicking <b>Ring</b>. You can stop the ringing from your phone when you find it.</li></ol> <p data-bbox="741 483 1570 526"><b>Tip:</b> You can also find your missing phone using the <a href="#">Android Device manager</a> <span data-bbox="1318 483 1339 503">^</span> which lets you find your device or remotely ring, lock, or erase it.</p> <hr/> <p data-bbox="720 586 966 605"><b>Send directions to my phone</b> <span data-bbox="1560 586 1581 605">^</span></p> <p data-bbox="741 626 1570 669">Once you've looked up directions on your computer, you can send them to your phone so you have them on your trip.</p> <ol data-bbox="741 685 1497 782" style="list-style-type: none"><li>1. On your computer's browser, search on <a href="http://www.google.com">www.google.com</a> <span data-bbox="1192 685 1213 704">^</span> for <b>send directions to my phone</b>.</li><li>2. Enter in your destination.</li><li>3. Click <b>Send directions to your phone</b>.</li><li>4. You'll get a notification on your phone. Tap to navigate to your destination using Google Maps.</li></ol> <hr/> <p data-bbox="720 846 934 865"><b>Send a note to my phone</b> <span data-bbox="1560 846 1581 865">^</span></p> <ol data-bbox="741 886 1549 984" style="list-style-type: none"><li>1. On your computer's browser, search on <a href="http://www.google.com">www.google.com</a> <span data-bbox="1192 886 1213 906">^</span> for <b>send a note to my phone</b>.</li><li>2. Type your note in the box.</li><li>3. Click <b>Send note to your phone</b>.</li><li>4. You'll get a notification on your phone with your note that you can either save to one of your apps or copy.</li></ol> <hr/> <p data-bbox="720 1047 831 1066"><b>Set an alarm</b> <span data-bbox="1560 1047 1581 1066">^</span></p> <ol data-bbox="741 1088 1360 1185" style="list-style-type: none"><li>1. On your computer's browser, search on <a href="http://www.google.com">www.google.com</a> <span data-bbox="1192 1088 1213 1107">^</span> for <b>set an alarm</b>.</li><li>2. Choose the time you want the alarm to go off.</li><li>3. Click <b>Set an alarm on your phone</b>.</li><li>4. An alarm will now be set on your phone's Clock app.</li></ol> <hr/> <p data-bbox="720 1248 848 1268"><b>Set a reminder</b> <span data-bbox="1560 1248 1581 1268">^</span></p> <ol data-bbox="741 1289 1518 1360" style="list-style-type: none"><li>1. On your computer's browser, search on <a href="http://www.google.com">www.google.com</a> <span data-bbox="1192 1289 1213 1308">^</span> for <b>set an reminder</b>.</li><li>2. Type what you want to be reminded about, and either when or where you want the reminder to go off.</li><li>3. Click <b>Remind me on my devices</b>.</li></ol> <hr/> <p data-bbox="699 1369 1402 1393"><a href="https://support.google.com/websearch/answer/6128427">https://support.google.com/websearch/answer/6128427</a></p>

**Exhibit A for US Patent No. 8,213,970 Against HTC Accused Products**

Claim - 8,213,970	Accused Products
<p>[1A] a data transmission means that facilitates the transmission of electronic files between said PDA/cell phones in different locations;</p>	<p>HTC infringes directly and/or indirectly by performing, inducing others to perform, and/or contributing to the performance of a first device programmed to perform operations comprising: the transmission of electronic files between said PDA/cell phones in different locations via a data transmission means that facilitates the transmission of electronic files between said PDA/cell phones in different locations. See, e.g., claim 1[P], which is incorporated herein by reference in its entirety.</p> <p>This claim term is governed by 35 U.S.C. 112(6).</p> <p>Function: facilitating the transmission of electronic files between said PDA/cell phones in different locations.</p> <p>Structure: Communications network server; Communication network interfaces '970 Patent at 1:39-43; 2:36-43; Figs. 2, 3A, 3B, and 4.</p> <p>The Accused Products meet this limitation. HTC provides access to one or more communication network servers via a modem interface such as a 3G or LTE modem</p>
<p>[1B] a sender PDA/cell phone and at least one recipient PDA/cell phone for each electronic message;</p>	<p>HTC infringes directly and/or indirectly by performing, inducing others to perform, and/or contributing to the performance of the participants of the predetermined network by providing a sender PDA/cell phone and at least one recipient PDA/cell phone for each electronic message. See, e.g., claim 1A and 1[P], which are incorporated herein by reference in their entirety.</p> <p>HTC makes, uses, imports, sells or otherwise provides Android devices, such as the Accused Products, to its customers. These devices include PDA/cell phones, i.e. smartphones, tablets, and other devices with cellular connections. Each claimed electronic message is sent from at least one Accused Product to another Accused Product, which share a common "Google Account."</p>



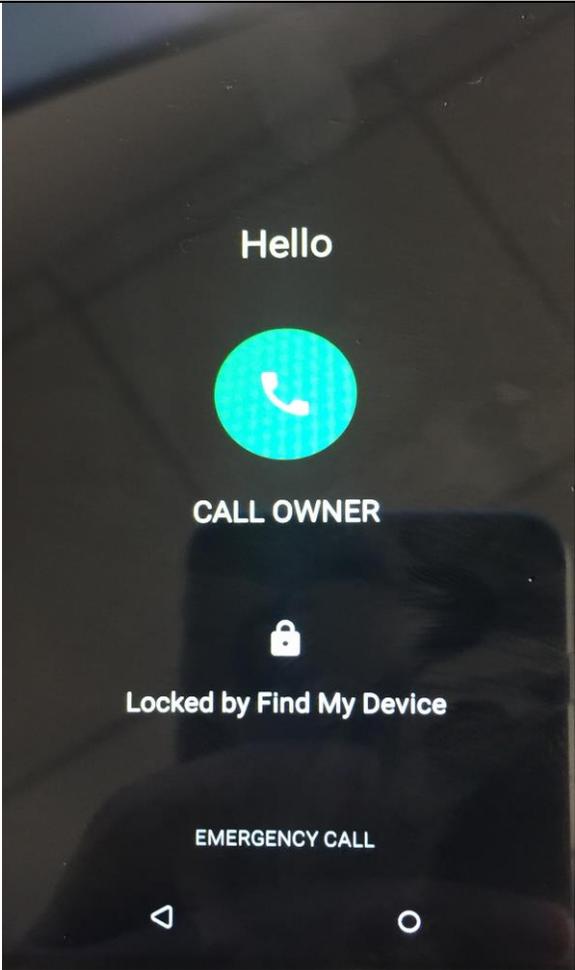
**Exhibit A for US Patent No. 8,213,970 Against HTC Accused Products**

<b>Claim - 8,213,970</b>	<b>Accused Products</b>
<p>[1C] a forced message alert software application program including a list of required possible responses to be selected by a participant recipient of a forced message response loaded on each participating PDA/cell phone;</p>	<p>HTC infringes directly and/or indirectly by performing, inducing others to perform, and/or contributing to the performance of one or more of the network participants programmed to perform operations comprising: including a list of required possible responses to be selected by a participant recipient of a forced message response loaded on each participating PDA/cell phone via, e.g., a forced message alert software application program. See, e.g., claim 1B, which is incorporated herein by reference in its entirety.</p> <p>For example, HTC’s Find My Device software application, which HTC provides as a pre-installed feature and a downloadable as a software application allows Android OS users to track other HTC devices, such as PDA/cell phones and tablets, linked to the same Google Account. Find My Devices provides a list of the status for each device within the same Google Account” that tracks location status, responses to location requests / time since last update, and actual locations.</p>

**Exhibit A for US Patent No. 8,213,970 Against HTC Accused Products**

Claim - 8,213,970	Accused Products
	<h2 data-bbox="722 250 1724 305">Find, lock, or erase a lost Android device</h2> <p data-bbox="722 329 1887 386">If you lose an Android phone, tablet, or Wear watch, you can find, lock, or erase it. If you've added a Google Account to your Android device, Find My Device is automatically turned on.</p> <p data-bbox="722 415 1157 440">To use Find My Device, your lost device must:</p> <ul data-bbox="722 469 1100 695" style="list-style-type: none"><li>• Be turned on</li><li>• Be signed in to a Google Account</li><li>• Be connected to mobile data or Wi-Fi</li><li>• Be visible on Google Play</li><li>• Have Location turned on</li><li>• Have Find My Device turned on</li></ul> <p data-bbox="722 724 1436 748">Lost Android Wear watches must also be running Android Wear 2.0 and up.</p> <p data-bbox="722 777 1367 802"><a href="#">Learn how to make sure that your device can be found if it gets lost.</a></p> <p data-bbox="722 831 1829 855"><b>Tip:</b> If you've <a href="#">linked your phone to Google</a>, you can find or ring it by searching for <b>find my phone</b> on <a href="#">google.com</a> <a href="#">🔗</a>.</p> <p data-bbox="695 917 1451 950"><a href="https://support.google.com/android/answer/6160491?hl=en">https://support.google.com/android/answer/6160491?hl=en</a></p> <p data-bbox="695 985 1906 1144">The forced message alert software includes with it a list of required possible responses to be selected by a particular recipient, such as “call owner,” “emergency call,” or entering the recipients password, pushing or swiping the unlock button, or fingerprint recognition to unlock the phone.</p>

**Exhibit A for US Patent No. 8,213,970 Against HTC Accused Products**

Claim - 8,213,970	Accused Products
	
<p>[1D] means for attaching a forced message alert software packet to a voice or text message creating a forced message alert that is transmitted by said sender PDA/cell</p>	<p>HTC infringes directly and/or indirectly by performing, inducing others to perform, and/or contributing to the performance of the participants of the predetermined network to perform operations comprising attaching a forced message alert software packet to a voice or text message creating a forced message alert that is transmitted by said sender PDA/cell phone to the recipient PDA/cell phone by, e.g., providing the system with means for attaching a forced</p>

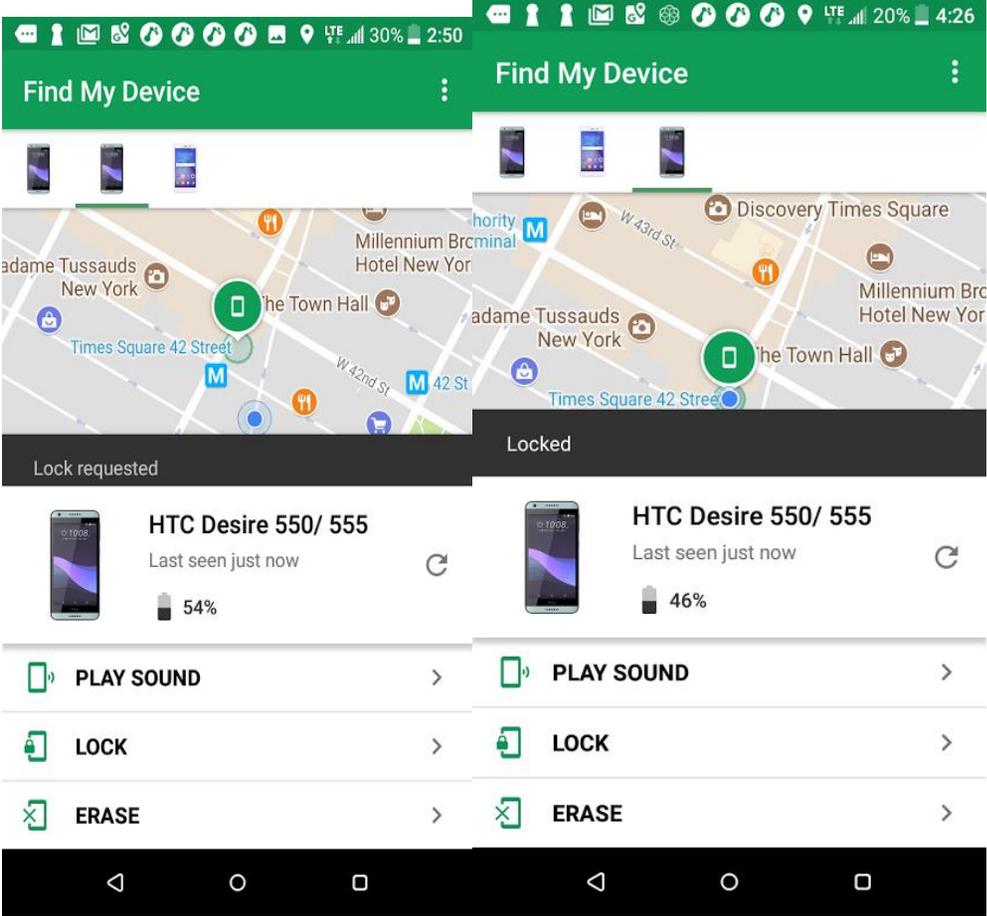
**Exhibit A for US Patent No. 8,213,970 Against HTC Accused Products**

Claim - 8,213,970	Accused Products
<p>phone to the recipient PDA/cell phone,</p>	<p>message alert software packet to a voice or text message creating a forced message alert that is transmitted by said sender PDA/cell phone to the recipient PDA/cell phone. See, e.g., claim 1C, which is incorporated herein by reference in its entirety.</p> <p>This claim term is governed by 35 U.S.C. 112(6).</p> <p>Function: attaching a forced message alert software packet to a voice or text message creating a forced message alert that is transmitted by said sender PDA/cell phone to the recipient PDA/cell phone.</p> <p>Structure: Algorithm set forth in Fig 2, 3A, 3B. 7:8-63.</p> <p>HTC’s Find My Device feature allows Android OS users to track other phones, tablets, and computers linked to the same Google Account. Find My Device provides a list of the status for each device that tracks location status, response to location requests / time since last update, and actual locations.</p> <p>The Sender electronically transmits the message to the Receiver through the use of the Accused Products. Because this is a security feature, the recipient phones have no control over the receipt of the message and thus the message is “forced.”</p> <p>The forced message alert causes automatic responses as shown in the response list in 1[C] above.</p> <p>This claim is literally met by one or more TCP/IP or other protocol packets (including HTC’s specific protocols via one or more communication network servers via a modem interface such as a 3G or LTE modem.. To the extent that the claims literally cover only a single packet, the limitation would still be met under the doctrine of equivalents. One of ordinary skill in the art would readily appreciate that packetized communications can be subdivided into multiple packet-sizes. These packets may be further subdivided in order to pass over different network layers. Thus, this limitation is equivalently met by packetized communication transmitting a forced message.</p>

**Exhibit A for US Patent No. 8,213,970 Against HTC Accused Products**

Claim - 8,213,970	Accused Products
<p>[1E]said forced message alert software packet containing a list of possible required responses and requiring the forced message alert software on said recipient PDA/cell phone to transmit an automatic acknowledgment to the sender PDA/cell phone as soon as said forced message alert is received by the recipient PDA/cell phone;</p>	<p>HTC infringes directly and/or indirectly by performing, inducing others to perform, and/or contributing to the performance of the participants of the predetermined network to perform operations comprising said forced message alert software packet containing a list of possible required responses comprising: requiring the forced message alert software on said recipient PDA/cell phone to transmit an automatic acknowledgment to the sender PDA/cell phone as soon as said forced message alert is received by the recipient PDA/cell phone. See, e.g., claim 1D, which is incorporated herein by reference in its entirety.</p> <p>Touching the “lock” button electronically transmits the message. Because this is a security feature, the recipient phones have no control over the receipt of the message and thus is “forced.” To enter lock mode, the sender attaches a phone number and message. Once the forced message is received by the recipient, an automatic acknowledgement is displayed by the sender cell phone (stating that the recipient’s phone is “locked.” At least the phone number constitutes a list of required responses, i.e. to call the sender.</p> <p>The sender device sends a request and in response, the receiver device sends an automatic acknowledgement which confirms that the device is in lock mode and provides the recipients location. This automatic acknowledgement is shown to the Sender when the Find My Device App states “lock requested” and then “locked.”</p>

**Exhibit A for US Patent No. 8,213,970 Against HTC Accused Products**

Claim - 8,213,970	Accused Products
	 <p>The forced message alert software packet from the sender device contains a list of possible required responses as shown above in claim 1[C]. To the extent this limitation is not literally met, each of these responses are at least the equivalent of a required response because they perform the same function (alerting the receiver device) in substantially the same way (locking the phone and making it otherwise unusable until one of the required manual actions is performed) to achieve substantially the same result (the lost message is conveyed to the</p>

**Exhibit A for US Patent No. 8,213,970 Against HTC Accused Products**

Claim - 8,213,970	Accused Products
<p>[1F] means for requiring a required manual response from the response list by the recipient in order to clear recipient's response list from recipient's cell phone display;</p>	<p>receiver phone and responsive action is taken.)</p> <p>HTC infringes directly and/or indirectly by performing, inducing others to perform, and/or contributing to the performance of the participants of the predetermined network to perform operations comprising requiring a required manual response from the response list by the recipient in order to clear recipient's response list from recipient's cell phone display by, e.g., providing the system with means for requiring a required manual response from the response list by the recipient in order to clear recipient's response list from recipient's cell phone display. See, e.g., claim 1E, which is incorporated herein by reference in its entirety.</p> <p>This claim is governed by 35 U.S.C. 112(6).</p> <p>Function: requiring a required manual response from the response list by the recipient in order to clear recipient's response list from recipient's cell phone display.</p> <p>Structure: Algorithm set forth in Figure 4 and 8:16-57; 11:1-21.</p> <p>This algorithm is implemented in software provided by HTC on the recipient phone device.</p> <p>Additionally, lock mode cannot be disabled on the recipient device without entering the password. Furthermore, a locked device may also be able to receive phone calls, but receiving a phone call does not affect this limitation and the phone remains otherwise locked.</p> <p>As set forth above in claim 1[C], required responses incorporated into HTC's software used in the Accused Products include "call owner," "emergency call," or the unlock button which requires entering the recipient's password, pushing or swiping the unlock button, or fingerprint recognition to unlock the phone. Each of these responses is a required response because the receiver device is otherwise inoperable unless these responses are selected. To the extent this limitation is not literally met, each of these responses are at least the equivalent of a required response because they perform the same function (alerting the receiver device) in substantially the same way (locking the phone and making it otherwise unusable until one of the required manual actions is performed) to achieve substantially the same result (the lost message is converted to the receiver phone and responsive action is taken</p>

**Exhibit A for US Patent No. 8,213,970 Against HTC Accused Products**

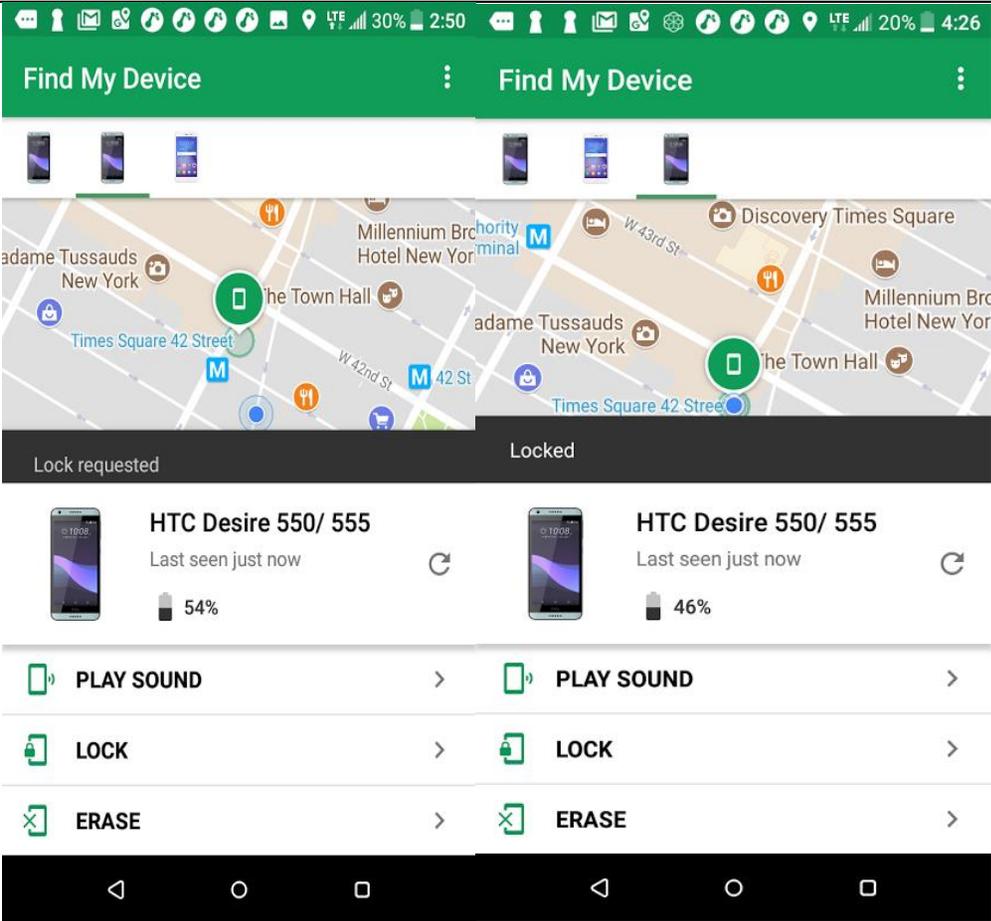
Claim - 8,213,970	Accused Products
<p>[1G] means for receiving and displaying a listing of which recipient PDA/cell phones have automatically acknowledged the forced message alert and which recipient PDA/cell phones have not automatically acknowledged the forced message alert;</p>	<p>HTC infringes directly and/or indirectly by performing, inducing others to perform, and/or contributing to the performance of the participants of the predetermined network to perform operations comprising receiving and displaying a listing of which recipient PDA/cell phones have automatically acknowledged the forced message alert and which recipient PDA/cell phones have not automatically acknowledged the forced message alert by, e.g., providing the system with means for receiving and displaying a listing of which recipient PDA/cell phones have automatically acknowledged the forced message alert and which recipient PDA/cell phones have not automatically acknowledged the forced message alert. See, e.g., claim 1F, which is incorporated herein by reference in its entirety.</p> <p>This claim is governed by 35 U.S.C. 112(6).</p> <p>Function: receiving and displaying a listing of which recipient PDA/cell phones have automatically acknowledged the forced message alert and which recipient PDA/cell phones have not automatically acknowledged the forced message alert.</p> <p>Structure: PDA/cell phone hardware including touch screen 16, and wireless transmitter or cellular modem. '970 Patent at col. 4:12-46.<sup>1</sup></p> <p>Each of the Accused Products includes a display that can display which devices have automatically acknowledged the forced message (e.g. entering lock mode and providing updates on receiver location, battery, and network connectivity.)</p>

---

<sup>1</sup> In the alternative, to the extent that HTC may allege that this implementation is software-implemented, the structure for such display software is set forth in the algorithms in Figures 2, 3A, 3B, 6:38-7:4; 7:17-8:15.



**Exhibit A for US Patent No. 8,213,970 Against HTC Accused Products**

Claim - 8,213,970	Accused Products
	
<p>[1H] means for periodically resending said forced message alert to said recipient PDA/cell phones that have not automatically acknowledged the forced message alert;</p>	<p>HTC infringes directly and/or indirectly by performing, inducing others to perform, and/or contributing to the performance of the participants of the predetermined network to perform operations comprising transmitting, receiving, confirming receipt, and responding to an electronic message by providing means for periodically resending said forced message alert to said recipient PDA/cell phones that have not automatically acknowledged the forced message alert. See, e.g., claim 1G, which is incorporated herein by reference in its entirety.</p>

**Exhibit A for US Patent No. 8,213,970 Against HTC Accused Products**

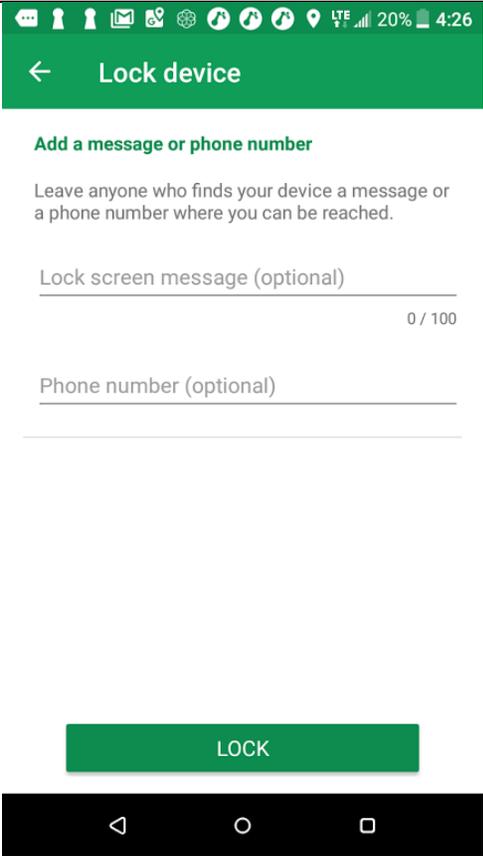
Claim - 8,213,970	Accused Products
	<p>Function: periodically resending said forced message alert to said recipient PDA/cell phones that have not automatically acknowledged the forced message alert.</p> <p>Structure: PDA/cell phone hardware including a wireless transmitter or cellular modem. '970 Patent at col. 4:12-46.</p> <p>This limitation is met by the cellular or wireless connectivity of each Accused Product, which periodically resend forced message alerts to the recipient phone. For example, the device will display the “Lock requested” message but will not display “locked” message until the second device has been locked. The periodic resending of the message is demonstrated when a lock is requested while the second device does not have a data connection. The “locked” message will only display after the second device’s data connection has been restored, demonstrating that the lock message is resent periodically to PDA/cell phones that have not automatically acknowledged the forced message alert.</p>
<p>[1I] and means for receiving and displaying a listing of which recipient PDA/cell phones have transmitted a manual response to said forced message alert and details the response from each recipient PDA/cell phone that responded.</p>	<p>HTC infringes directly and/or indirectly by performing, inducing others to perform, and/or contributing to the performance of the participants of the predetermined network to perform operations comprising receiving and displaying a listing of which recipient PDA/cell phones have transmitted a manual response to said forced message alert and details the response from each recipient PDA/cell phone that responded by, e.g., providing means for receiving and displaying a listing of which recipient PDA/cell phones have transmitted a manual response to said forced message alert and details the response from each recipient PDA/cell phone that responded. See, e.g., claim 1H, which is incorporated herein by reference in its entirety.</p> <p>This claim term is governed by 35 U.S.C. 112(6).</p> <p>Function: receiving and displaying a listing of which recipient PDA/cell phones have transmitted a manual response to said forced message alert and details the response from each recipient PDA/cell phone that has responded.</p> <p>Structure: PDA/cell phone hardware including touch screen 16, and wireless transmitter or</p>

**Exhibit A for US Patent No. 8,213,970 Against HTC Accused Products**

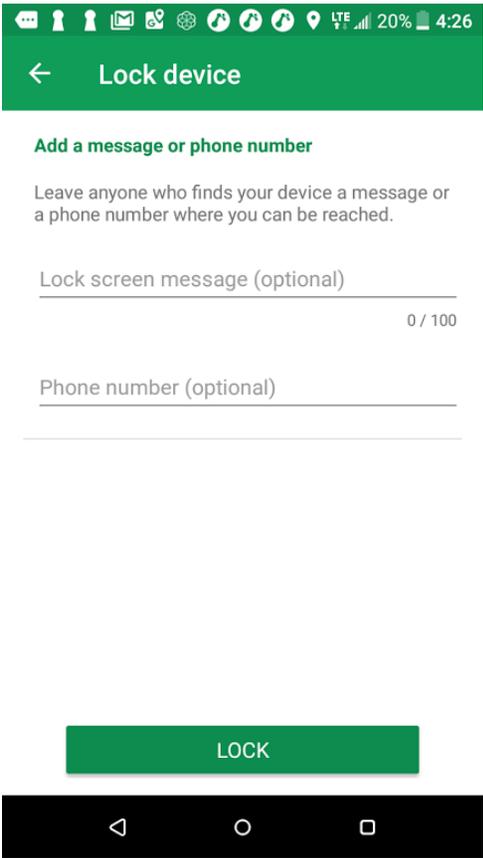
Claim - 8,213,970	Accused Products
	<p>cellular modem. '970 Patent at col. 4:12-46.<sup>2</sup></p> <p>Each of the Accused Products includes a display that can display which recipient devices have transmitted a manual response (e.g., calling back or sending a text message to the sender device.)</p>
<p>3. The system as in claim 1, wherein said data transmission means is TCP/IP or another communications protocol.</p>	<p>HTC infringes directly and/or indirectly by performing, inducing others to perform, and/or contributing to the performance of the participants of the predetermined network to perform operations comprising wherein said data transmission means is TCP/IP or another communications protocol. See, e.g., claim 1, which is incorporated herein by reference in its entirety.</p> <p>See Claim [1A] above.</p>
<p>4. The system as in claim 1, wherein the response list that is transmitted within the forced message alert software packet is a default response list that is embedded in the forced message alert software application program.</p>	<p>HTC infringes directly and/or indirectly by performing, inducing others to perform, and/or contributing to the performance of the participants of the predetermined network to perform operations comprising wherein the response list that is transmitted within the forced message alert software packet is a default response list that is embedded in the forced message alert software application program. See, e.g., claim 1, which is incorporated herein by reference in its entirety.</p> <p>See Claim [1F] above.</p> <p>Default options include “emergency call,” or pushing or swiping or swiping to unlock or fingerprint recognition to unlock the phone. As shown below, messages and call back number are “optional.”</p>

<sup>2</sup> In the alternative, to the extent that HTC may allege that this implementation is software-implemented, the structure for such display software is set forth in the algorithms in Figures 2, 3A, 3B, 6:38-7:4; 7:17-8:15.

**Exhibit A for US Patent No. 8,213,970 Against HTC Accused Products**

Claim - 8,213,970	Accused Products
	 <p>For example, the forced message initiating lock mode may be sent without entering optional information, and thus the default list will be sent.</p>
<p>5. The system as in claim 1, wherein the response list that is transmitted within the forced message alert software packet is a custom response list that is created at the time the specific forced message alert is</p>	<p>HTC infringes directly and/or indirectly by performing, inducing others to perform, and/or contributing to the performance of the participants of the predetermined network to perform operations comprising wherein the response list that is transmitted within the forced message alert software packet is a custom response list that is created at the time the specific forced message alert is created on the sender PDA/cell phone. See, e.g., claim 1, which is incorporated herein by reference in its entirety.</p>

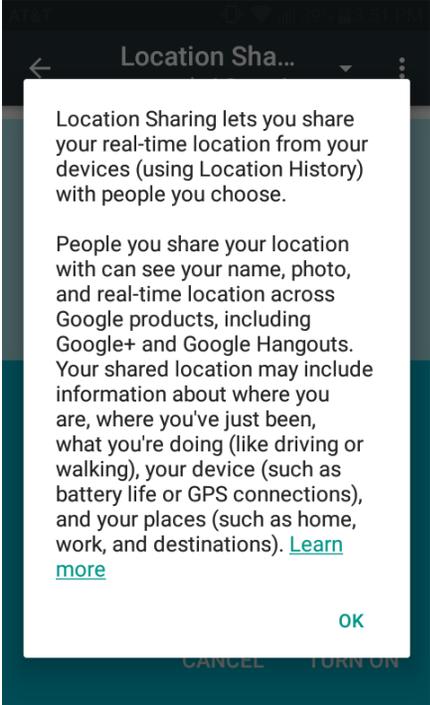
**Exhibit A for US Patent No. 8,213,970 Against HTC Accused Products**

Claim - 8,213,970	Accused Products
<p>created on the sender PDA/cell phone.</p>	<p>See Claim [1F] above.</p> <p>The forced message alert may be standard or customized. For example, forced message initiating lock mode may be sent with optional information, and thus the specific list will be sent at the time the forced message is created.</p> 
<p>6[A]. A method of sending a forced</p>	<p>HTC performs either directly or indirectly, induces others to perform, and/or contributes to the</p>

**Exhibit A for US Patent No. 8,213,970 Against HTC Accused Products**

Claim - 8,213,970	Accused Products
<p>message alert to one or more recipient PDA/cell phones within a predetermined communication network, wherein the receipt and response to said forced message alert by each intended recipient PDA/cell phone is tracked, said method comprising the steps of:</p>	<p>performance of each step of this method as set forth below. See, e.g., claim 1, which is incorporated herein by reference in its entirety.</p> <p>The Accused Products meet the claim limitations by providing device-location tracking features such as those features described below. For example, the Accused Products meet the claim limitations because they are pre-installed with Android mobile operating systems containing code for providing device-location tracking features as provided in the claims limitations herein. For example, the Accused Products run applications and/or software that run within the Android mobile operating system and that use components of the Android mobile operating system to provide device-location tracking features. Upon information and belief, in addition to the components and features of the Android mobile operating system itself, the following applications and/or software run within the Android mobile operating system and use components of the Android mobile operating system to provide device-location tracking features: Android Device Manager, Find My Device.</p> <p>Android Device Manager is the predecessor to Find My Device and has been available as a standard, pre-installed feature since 2013 and downloadable as a software application. The current iteration, Find My Device, often called the “new and improved Android Device Manager” or “rebranded Android Device Manager” is now part of the standard Google Play Protect suite which is “built in and enabled on all devices,” i.e., the Accused Products running Android OS. AGIS sets forth the Find My Device feature of the Accused Products as representative of this method. AGIS reserves the right to supplement these contentions to the extent that HTC requires additional information in accordance with P.R. 3-1 and for any other reason for which it may deem necessary.</p> <p>See, e.g., <a href="https://www.androidcentral.com/find-my-device">https://www.androidcentral.com/find-my-device</a>;  <a href="https://support.google.com/android/answer/6160491?hl=en">https://support.google.com/android/answer/6160491?hl=en</a>;  <a href="https://android.googleblog.com/2013/08/find-your-lost-phone-with-android.html">https://android.googleblog.com/2013/08/find-your-lost-phone-with-android.html</a>;  <a href="https://play.google.com/store/apps/details?id=com.google.android.apps.adm&amp;hl=en">https://play.google.com/store/apps/details?id=com.google.android.apps.adm&amp;hl=en</a>;  <a href="https://www.blog.google/products/android/google-play-protect/">https://www.blog.google/products/android/google-play-protect/</a></p> <p>Each “Google Account” is associated with a pre-determined number of devices, which include</p>

**Exhibit A for US Patent No. 8,213,970 Against HTC Accused Products**

Claim - 8,213,970	Accused Products
	<p>a CPU and a touchscreen, and which are registered when a customer acquires an Accused Product, such as a HTC Android OS based device.</p>  <p><b>Find your device using Android Device Manager</b></p> <p>If you've lost a device, you can use Android Device Manager to find its approximate location on a map and when it was last used. When Android Device Manager locates your device, that device will get a notification.</p> <p><b>Before you can use Android Device Manager to locate your device:</b> Your device's <a href="#">location access need to be turned on</a> and be signed in to your Google Account. Android Device Manager won't work for devices that are turned off or that don't have a mobile data or Wi-Fi connection.</p> <p><b>Tip:</b> If you've <a href="#">linked your phone to Google</a>, you can locate or ring it by searching for <a href="#">find my phone on google.com</a>.</p> <p><a href="https://support.google.com/pixelphone/answer/6160491">https://support.google.com/pixelphone/answer/6160491</a></p>

# Exhibit A for US Patent No. 8,213,970 Against HTC Accused Products

Claim - 8,213,970	Accused Products
	<p data-bbox="701 233 1094 264"><b>Link your phone to Google</b></p> <p data-bbox="701 280 1383 315">You can connect your Android phone to Google, which lets you send information from your computer to your phone. For example, you can send directions you searched for on your computer to Google Maps on your phone.</p> <p data-bbox="701 342 919 360"><b>Link your Android phone</b></p> <p data-bbox="701 386 919 404"><b>Step 1: Update the Google app</b></p> <ol data-bbox="701 409 1066 448" style="list-style-type: none"><li>1. On your phone, go to the <a href="#">Google app page on the Play Store</a>.</li><li>2. Tap <b>Update</b>.</li></ol> <p data-bbox="701 472 905 490"><b>Step 2: Turn on Google Now</b></p> <ol data-bbox="701 495 1031 578" style="list-style-type: none"><li>1. On your phone, open the Google app .</li><li>2. At the top left, tap Menu  &gt; <b>Settings</b> &gt; <b>Now cards</b>.</li><li>3. Turn on <b>Show cards</b>.</li><li>4. Turn on <b>Show notifications</b>.</li></ol> <p data-bbox="701 602 953 620"><b>Step 3: Turn on Web &amp; App Activity</b></p> <ol data-bbox="701 625 926 664" style="list-style-type: none"><li>1. Visit the <a href="#">Account History page</a>.</li><li>2. Make sure the switch is on (green).</li></ol> <p data-bbox="701 688 926 706"><b>Step 4: Sign in to your browser</b></p> <ol data-bbox="701 711 1161 839" style="list-style-type: none"><li>1. On your phone, open the Google app .</li><li>2. At the top left, tap the Menu .</li><li>3. At the top left, you'll see the email address you use for the Google app.</li><li>4. Visit <a href="http://www.google.com">www.google.com</a>  on your computer.</li><li>5. If you aren't signed in already, click <b>Sign in</b> in the top right corner of the page.</li><li>6. Sign in using the Google Account you use for the Google app.</li></ol> <p data-bbox="701 863 982 881"><b>Step 5: Send information to your phone</b></p> <ol data-bbox="701 886 1392 943" style="list-style-type: none"><li>1. Do one of the searches below, like <b>note to self</b>, or <b>send directions to my phone</b>.</li><li>2. If a box doesn't pop up with the option to send information to your phone, try refreshing the page. If you just turned on Google Now, it may take a few minutes for the box to show up.</li></ol> <p data-bbox="701 951 1398 979"><a href="https://support.google.com/websearch/answer/6128427">https://support.google.com/websearch/answer/6128427</a></p>



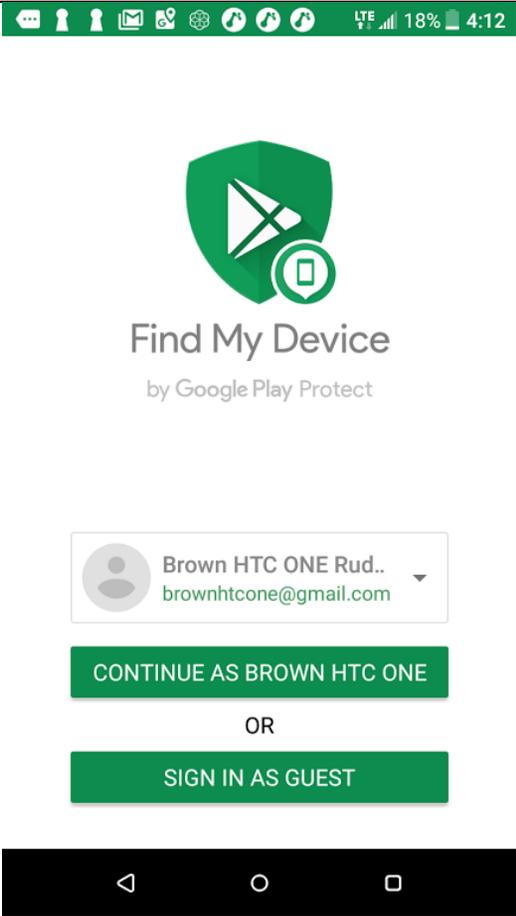
## Exhibit A for US Patent No. 8,213,970 Against HTC Accused Products

Claim - 8,213,970	Accused Products
	<p data-bbox="699 235 1192 261"><b>What you can do once your phone is linked</b></p> <hr/> <p data-bbox="718 293 848 315"><b>Find my phone</b> <span data-bbox="1556 298 1577 315">^</span></p> <p data-bbox="743 334 1220 355">You can get the current location of your phone if you can't find it.</p> <ol data-bbox="751 371 1570 469" style="list-style-type: none"><li data-bbox="751 371 1367 393">1. On your computer's browser, search on <a href="http://www.google.com">www.google.com</a> <span data-bbox="1192 375 1213 391">^</span> for <b>find my phone</b>.</li><li data-bbox="751 399 1423 420">2. If your phone is turned on and connected to the Internet, you'll see your phone's location.</li><li data-bbox="751 427 1570 469">3. If your phone's location is unavailable, you can still make it ring for 5 minutes on full volume by clicking <b>Ring</b>. You can stop the ringing from your phone when you find it.</li></ol> <p data-bbox="743 485 1570 527"><b>Tip:</b> You can also find your missing phone using the <a href="#">Android Device manager</a> <span data-bbox="1318 488 1339 505">^</span> which lets you find your device or remotely ring, lock, or erase it.</p> <hr/> <p data-bbox="718 586 968 607"><b>Send directions to my phone</b> <span data-bbox="1556 591 1577 607">^</span></p> <p data-bbox="743 626 1570 669">Once you've looked up directions on your computer, you can send them to your phone so you have them on your trip.</p> <ol data-bbox="751 685 1499 789" style="list-style-type: none"><li data-bbox="751 685 1499 706">1. On your computer's browser, search on <a href="http://www.google.com">www.google.com</a> <span data-bbox="1192 688 1213 704">^</span> for <b>send directions to my phone</b>.</li><li data-bbox="751 712 953 734">2. Enter in your destination.</li><li data-bbox="751 740 1037 761">3. Click <b>Send directions to your phone</b>.</li><li data-bbox="751 768 1465 789">4. You'll get a notification on your phone. Tap to navigate to your destination using Google Maps.</li></ol> <hr/> <p data-bbox="718 847 932 868"><b>Send a note to my phone</b> <span data-bbox="1556 852 1577 868">^</span></p> <ol data-bbox="751 888 1549 992" style="list-style-type: none"><li data-bbox="751 888 1457 909">1. On your computer's browser, search on <a href="http://www.google.com">www.google.com</a> <span data-bbox="1192 891 1213 907">^</span> for <b>send a note to my phone</b>.</li><li data-bbox="751 915 961 937">2. Type your note in the box.</li><li data-bbox="751 943 995 964">3. Click <b>Send note to your phone</b>.</li><li data-bbox="751 971 1549 992">4. You'll get a notification on your phone with your note that you can either save to one of your apps or copy.</li></ol> <hr/> <p data-bbox="718 1050 831 1071"><b>Set an alarm</b> <span data-bbox="1556 1055 1577 1071">^</span></p> <ol data-bbox="751 1091 1360 1195" style="list-style-type: none"><li data-bbox="751 1091 1360 1112">1. On your computer's browser, search on <a href="http://www.google.com">www.google.com</a> <span data-bbox="1192 1094 1213 1110">^</span> for <b>set an alarm</b>.</li><li data-bbox="751 1118 1108 1140">2. Choose the time you want the alarm to go off.</li><li data-bbox="751 1146 1016 1167">3. Click <b>Set an alarm on your phone</b>.</li><li data-bbox="751 1174 1150 1195">4. An alarm will now be set on your phone's Clock app.</li></ol> <hr/> <p data-bbox="718 1253 848 1274"><b>Set a reminder</b> <span data-bbox="1556 1258 1577 1274">^</span></p> <ol data-bbox="751 1294 1516 1365" style="list-style-type: none"><li data-bbox="751 1294 1388 1315">1. On your computer's browser, search on <a href="http://www.google.com">www.google.com</a> <span data-bbox="1192 1297 1213 1313">^</span> for <b>set an reminder</b>.</li><li data-bbox="751 1321 1516 1343">2. Type what you want to be reminded about, and either when or where you want the reminder to go off.</li><li data-bbox="751 1349 1010 1370">3. Click <b>Remind me on my devices</b>.</li></ol> <hr/> <p data-bbox="699 1372 1398 1398"><a href="https://support.google.com/websearch/answer/6128427">https://support.google.com/websearch/answer/6128427</a></p>

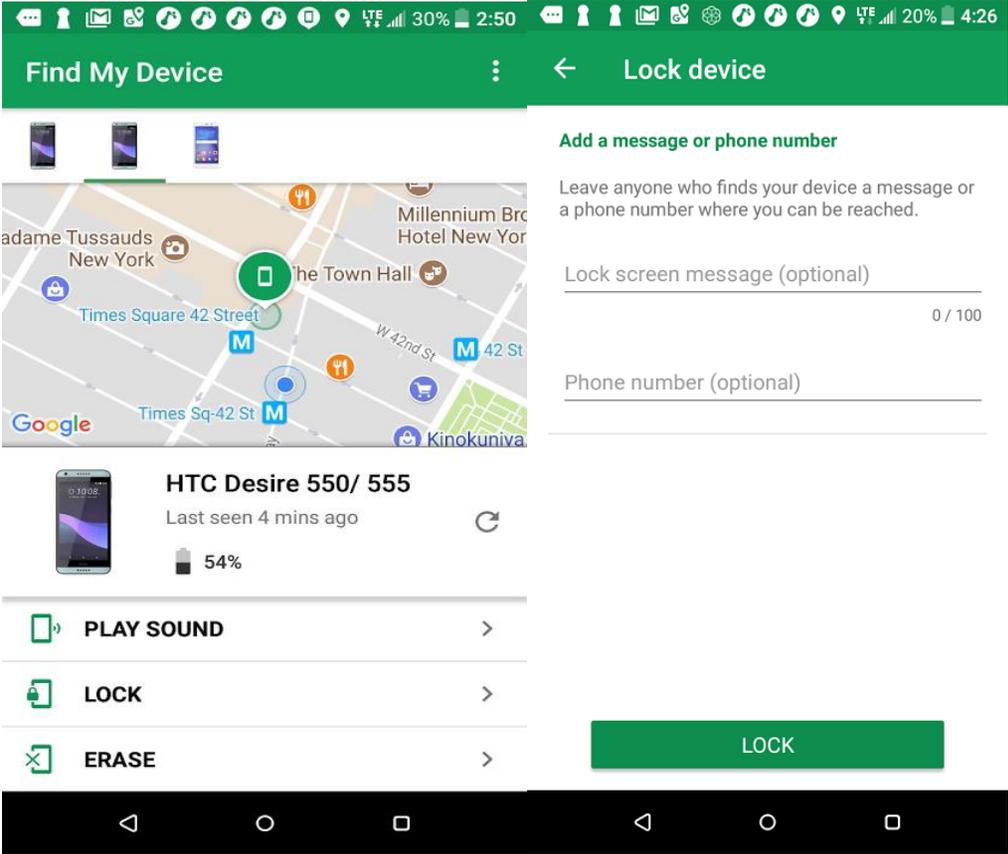
## Exhibit A for US Patent No. 8,213,970 Against HTC Accused Products

Claim - 8,213,970	Accused Products
6[B] accessing a forced message alert software application program on a sender PDA/cell phone;	<p>HTC infringes directly and/or indirectly by performing, inducing others to perform, and/or contributing to the performance of steps of the method comprising: accessing a forced message alert software application program on a sender PDA/cell phone. See, e.g., claims 1 and 6[A], which are incorporated herein by reference in their entirety.</p> <p>The user of the Accused Products (the sender) performs the step of “accessing a forced message alert software application program on a sender PDA/cell phone” by accessing the Find My Device app.</p>

**Exhibit A for US Patent No. 8,213,970 Against HTC Accused Products**

Claim - 8,213,970	Accused Products
	
<p>6[C] creating the forced message alert on said sender PDA/cell phone by attaching a voice or text message to a forced message alert application software packet to said voice or text message;</p>	<p>HTC infringes directly and/or indirectly by performing, inducing others to perform, and/or contributing to the performance of steps of the method comprising: creating the forced message alert on said sender PDA/cell phone by attaching a voice or text message to a forced message alert application software packet to said voice or text message. See, e.g., claims 1 and 6[B], which are incorporated herein by reference in their entirety.</p>

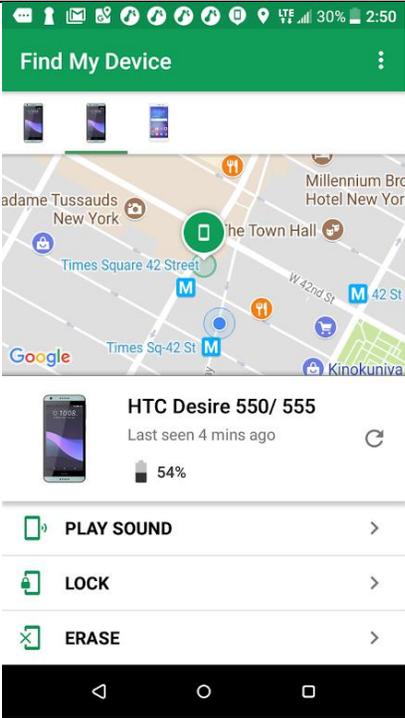
**Exhibit A for US Patent No. 8,213,970 Against HTC Accused Products**

Claim - 8,213,970	Accused Products
	<p>For example, this limitation is met when a user of the Accused Products chooses from the options play sound, lock mode, or erase phone and following any necessary prompts.</p>  <p>The screenshot shows the 'Find My Device' app interface. At the top, there are two tabs: 'Find My Device' and 'Lock device'. Below the tabs, there are three device icons. A map shows the location of the device in Times Square, New York. Below the map, the device is identified as 'HTC Desire 550/ 555' with a battery level of 54% and a last seen time of 4 minutes ago. There are three main options: 'PLAY SOUND', 'LOCK', and 'ERASE'. The 'LOCK' option is highlighted with a green button labeled 'LOCK'.</p> <p>To the extent this claim is construed to require both voice and text, play sound literally meets the limitation of voice. Furthermore, “play sound” is the equivalent of voice, because it performs substantially the same function (alerting users of a forced message alert) in substantially the same way (sound) with substantially the same result (the user alerted as to the forced message).</p>

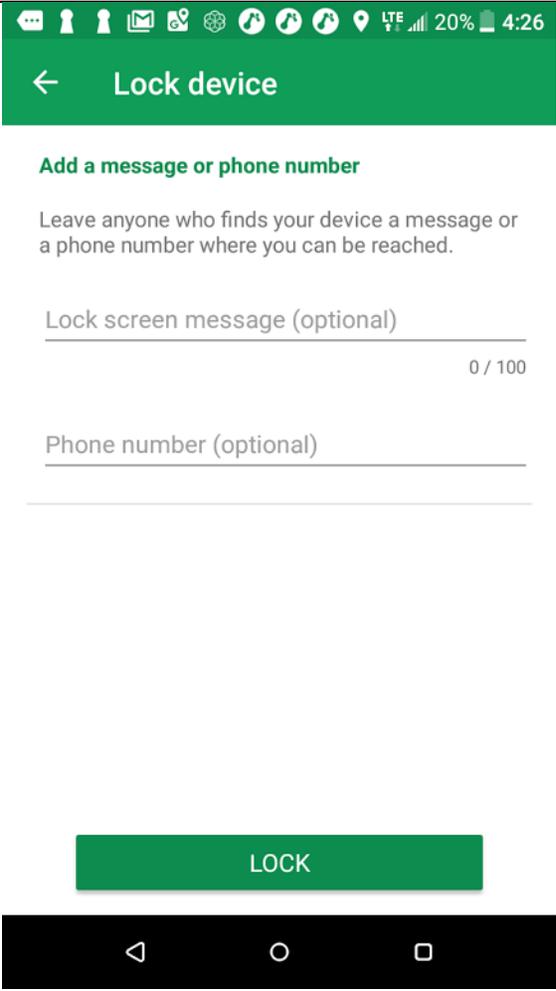
**Exhibit A for US Patent No. 8,213,970 Against HTC Accused Products**

Claim - 8,213,970	Accused Products
	<p>Entering lock mode creates a forced message alert packet including textual messages (e.g., the call-back number and a message). The play sound option also triggers the creation and sending of a forced message packet, because the sound is played on the recipient phone automatically without the recipient’s control. The “erase phone” option also triggers the creation and sending of a forced message packet that forces the data on the phone to be destroyed.</p> <p>Each of the communications is packetized over TCP/IP communication and thus satisfies the “packet” portion of the limitation whether a single or multiple packets are actually sent over the transport medium.</p>
<p>6[D] designating one or more recipient PDA/cell phones in the communication network;</p>	<p>HTC infringes directly and/or indirectly by performing, inducing others to perform, and/or contributing to the performance of steps of the method comprising: designating one or more recipient PDA/cell phones in the communication network. See, e.g., claims 1 and 6[C], which are incorporated herein by reference in their entirety.</p> <p>The user of the Accused Products (the sender) may designate one or more other phones either by touching on the phone name in a list of devices, or by touching on the location of the device on the map.</p>

**Exhibit A for US Patent No. 8,213,970 Against HTC Accused Products**

Claim - 8,213,970	Accused Products
	
<p>6[E] electronically transmitting the forced message alert to said recipient PDA/cell phones;</p>	<p>HTC infringes directly and/or indirectly by performing, inducing others to perform, and/or contributing to the performance of steps of the method comprising: electronically transmitting the forced message alert to said recipient PDA/cell phones. See, e.g., claims 1 and 6[D], which are incorporated herein by reference in their entirety.</p> <p>The user of the Accused Products (the sender) electronically transmits the forced message alert to said recipient PDA/cell phones by entering lock mode and following the menu prompts.</p> <p>The message is ultimately sent when the “lock” button is selected. Because this is a security feature, the recipient phones have no control over the receipt of the message and thus the message is forced.</p>

**Exhibit A for US Patent No. 8,213,970 Against HTC Accused Products**

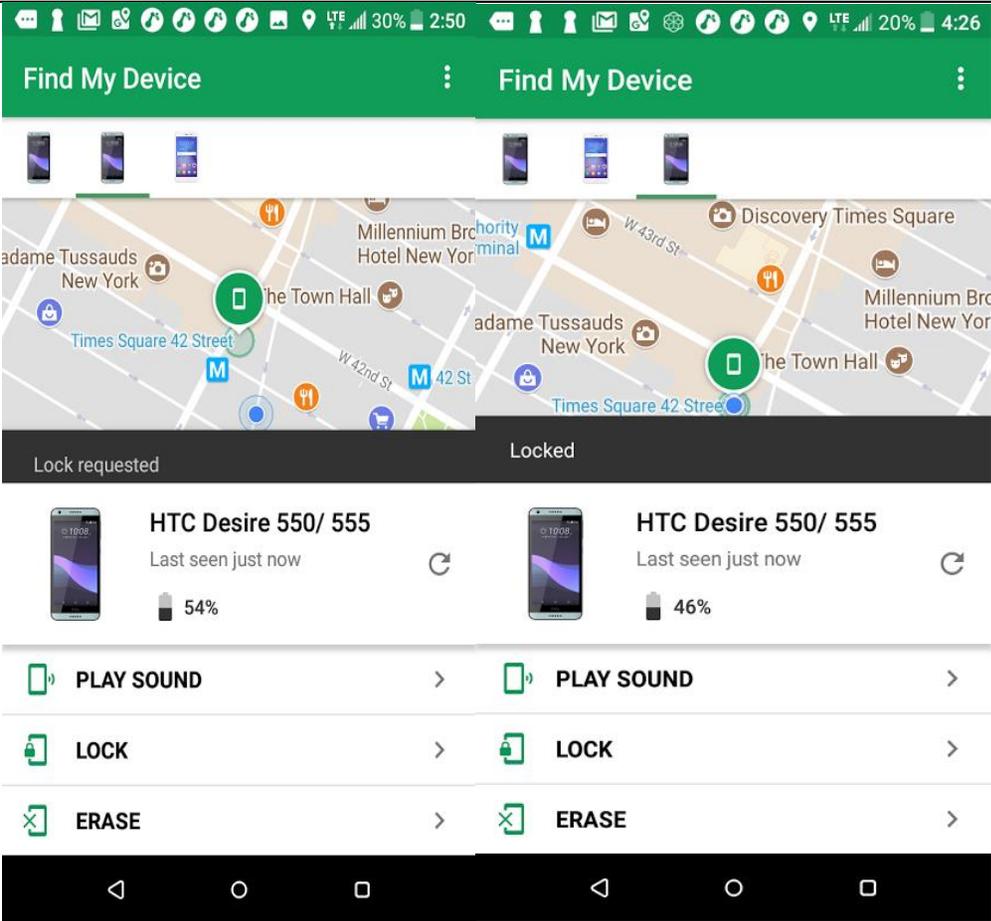
Claim - 8,213,970	Accused Products
	
<p>6[F] receiving automatic acknowledgements from the recipient PDA/cell phones that</p>	<p>HTC infringes directly and/or indirectly by performing, inducing others to perform, and/or contributing to the performance of steps of the method comprising: receiving automatic acknowledgements from the recipient PDA/cell phones that received the message and</p>

**Exhibit A for US Patent No. 8,213,970 Against HTC Accused Products**

Claim - 8,213,970	Accused Products
<p>received the message and displaying a listing of which recipient PDA/cell phones have acknowledged receipt of the forced message alert and which recipient PDA/cell phones have not acknowledged receipt of the forced message alert;</p>	<p>displaying a listing of which recipient PDA/cell phones have acknowledged receipt of the forced message alert and which recipient PDA/cell phones have not acknowledged receipt of the forced message alert. See, e.g., claims 1 and 6[E], which are incorporated herein by reference in their entirety.</p> <p>The user of the Accused Products (the sender) receives receiving automatic acknowledgements from the recipient PDA/cell phones that received the message and displaying a listing of which recipient PDA/cell phones have acknowledged receipt of the forced message alert and which recipient PDA/cell phones have not acknowledged receipt of the forced message alert.</p> <p>For example, the sender receives status updates as locations and times on the main screen of the Find My Device app. Once the lost phone (the “receiver”) is “located,” the receiver phone automatically sends a message to the “sender” indicating that it has entered “locked” mode.</p>



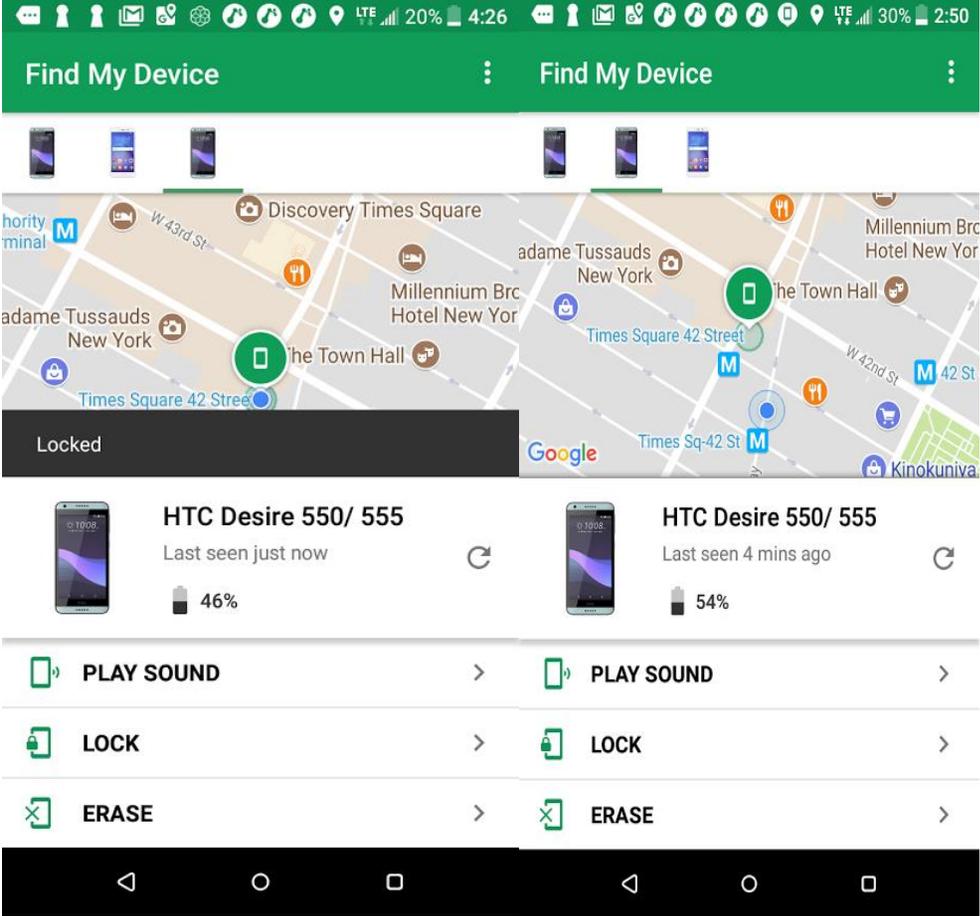
**Exhibit A for US Patent No. 8,213,970 Against HTC Accused Products**

Claim - 8,213,970	Accused Products
	 <p>The screenshot displays the 'Find My Device' application interface. At the top, there is a green header with the text 'Find My Device' and a menu icon. Below the header, a map shows the location of two devices. The first device is labeled 'Lock requested' and has a battery level of 54%. The second device is labeled 'Locked' and has a battery level of 46%. Both devices are identified as 'HTC Desire 550/ 555' and were 'Last seen just now'. Below the map, there are three rows of action buttons: 'PLAY SOUND', 'LOCK', and 'ERASE', each with a corresponding icon and a right-pointing arrow. The bottom of the screen shows the Android navigation bar with back, home, and recent apps buttons.</p> <p>The claim requires “a list” of phones having both receipt-acknowledged and receipt-unacknowledged states. To the extent this limitation requires two separate lists, the listing of phones in multiple states is the equivalent of two lists under the doctrine of equivalents because same the same function (showing a listing of acknowledged and unacknowledged recipients) is performed in substantially the same way (displaying said recipients on the display) to obtain substantially the same result (alerting the user as to which recipients</p>

**Exhibit A for US Patent No. 8,213,970 Against HTC Accused Products**

Claim - 8,213,970	Accused Products
	<p>received and acknowledged receipt of the forced message alert and which recipients did not).</p> <p>The sender device sends a request and in response, the receiver device sends an automatic response which confirms that the device is in locked mode.</p>
<p>6[G] periodically resending the forced message alert to the recipient PDA/cell phones that have not acknowledged receipt;</p>	<p>HTC infringes directly and/or indirectly by performing, inducing others to perform, and/or contributing to the performance of steps of the method comprising: periodically resending the forced message alert to the recipient PDA/cell phones that have not acknowledged receipt. See, e.g., claims 1 and 6[F], which are incorporated herein by reference in their entirety.</p> <p>The periodic resending of the message is demonstrated when a lock is requested while the second device does not have a data connection. The “locked” message will only display after the second device’s data connection has been restored, demonstrating that the lock message is resent periodically to PDA/cell phones that have not automatically acknowledged the forced message alert.</p>

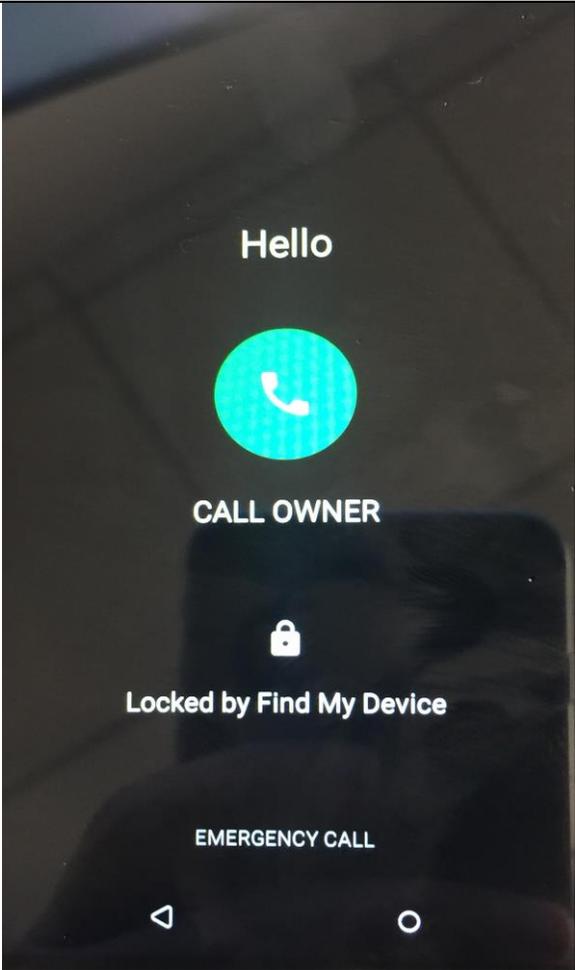
**Exhibit A for US Patent No. 8,213,970 Against HTC Accused Products**

Claim - 8,213,970	Accused Products
	
<p>6[H] receiving responses to the forced message alert from the recipient PDA/cell phones and displaying the response from each recipient PDA/cell phone;</p>	<p>HTC infringes directly and/or indirectly by performing, inducing others to perform, and/or contributing to the performance of steps of the method comprising: receiving responses to the forced message alert from the recipient PDA/cell phones and displaying the response from each recipient PDA/cell phone. See, e.g., claims 1 and 6[G], which are incorporated herein by reference in their entirety.</p>

**Exhibit A for US Patent No. 8,213,970 Against HTC Accused Products**

Claim - 8,213,970	Accused Products
	<p>The user of the Accused Products (the sender) performs the step of receiving responses to the forced message alert from the recipient PDA/cell phones and displaying the response from each recipient PDA/cell phone by continuing to use the Find My Device app in lock mode. The receipt of responses to the forced message alerts are depicted above in limitation 6[G].</p> <p>Each of the Accused Products includes a display that can display which recipient devices have transmitted a manual response (e.g., calling back or sending a text message to the sender device.)</p>
<p>6[I] providing a manual response list on the display of the recipient PDA/cell phone that can only be cleared by the recipient providing a required response from the list;</p>	<p>HTC infringes directly and/or indirectly by performing, inducing others to perform, and/or contributing to the performance of steps of the method comprising: providing a manual response list on the display of the recipient PDA/cell phone that can only be cleared by the recipient providing a required response from the list. See, e.g., claims 1 and 6[H], which are incorporated herein by reference in their entirety.</p> <p>The user of the Accused Products (the sender) performs the step of providing a manual response list on the display of the recipient PDA/cell phone that can only be cleared by the recipient providing a required response from the list.</p> <p>The manual response list is provided on the display of the recipient PDA/cell phone by the “sender” selecting the Locked mode option and following the prompts. If the sender enters a phone number and passcode, the following list will be provided to the recipient phone automatically.</p> <p>The Response list is either the single “call owner” option, or, depending on the construction of the term “list,” if the list must contain more than one entry, it is both the “call owner,” “emergency call,” or the push to unlock symbol which requires entering the recipients password, pushing or swiping the unlock button, or fingerprint recognition to unlock the phone.</p>

**Exhibit A for US Patent No. 8,213,970 Against HTC Accused Products**

Claim - 8,213,970	Accused Products
	 <p>Similarly, to the extent that list requires more than one entry, a list containing a single entry is the equivalent of a multiple-entry list under the doctrine of equivalents because the substantially same function of providing the user with required response is accomplished in the same way by displaying a single response when only a single response is required of the user to obtain substantially the same result guiding the user to a required course of action.</p>

**Exhibit A for US Patent No. 8,213,970 Against HTC Accused Products**

Claim - 8,213,970	Accused Products
	<p>Other responses include emergency and unlocking options such as swipe to unlock, push to unlock or fingerprint recognition. Each of these responses is a required response because the receiver device is otherwise inoperable unless these responses are selected. To the extent this limitation is not literally met, each of these responses are at least the equivalent of a required response because they perform the same function (alerting the receiver device) in substantially the same way (locking the phone and making it otherwise unusable until one of the required manual actions is performed) to achieve substantially the same result (the lost message is conveyed to the receiver phone and responsive action is taken).</p>
<p>6[J] clearing the recipient's display screen or causing the repeating voice alert to cease upon recipient selecting a response from the response list required that can only be cleared by manually selecting and transmitting a response to the manual response list.</p>	<p>HTC infringes directly and/or indirectly by performing, inducing others to perform, and/or contributing to the performance of steps of the method comprising: clearing the recipient's display screen or causing the repeating voice alert to cease upon recipient selecting a response from the response list required that can only be cleared by manually selecting and transmitting a response to the manual response list. See, e.g., claims 1 and 6[I], which are incorporated herein by reference in their entirety.</p> <p>The user of the Accused Products (the sender) performs the step of clearing the recipient's display screen or causing the repeating voice alert to cease upon recipient selecting a response from the response list required that can only be cleared by manually selecting and transmitting a response to the manual response list.</p> <p>The recipient's display screen can be cleared by the recipient device selecting "call owner," or entering the recipients password, pushing or swiping the unlock button, or fingerprint recognition to unlock the phone.</p> <p>The sender either performs this limitation him or herself (by entering his or her passcode), or directs or controls another person to follow the instructions set forth on the lock screen of the Accused Product in lock mode.</p> <p>To the extent a phone runs out of battery, or is tuned off, that phone would no longer be a part of the communications network and the limitation would still be met.</p>

**Exhibit A for US Patent No. 8,213,970 Against HTC Accused Products**

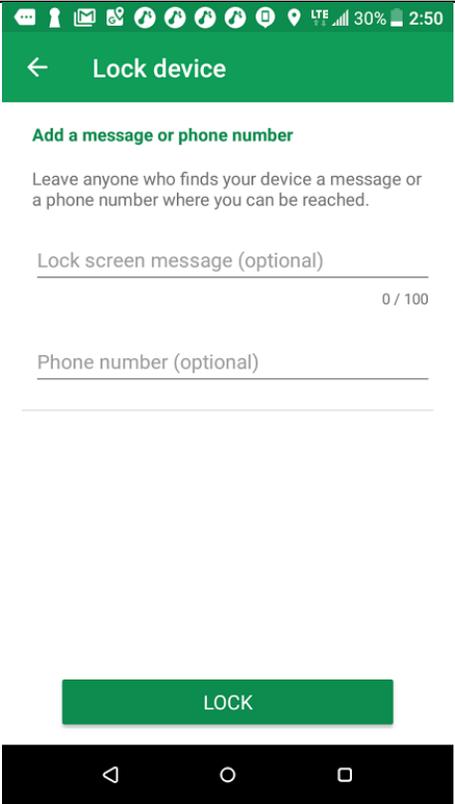
Claim - 8,213,970	Accused Products
	<p>Additionally, to the extent that Find My Device can be disabled remotely, this limitation would still be met literally when a manual method is utilized. Disabling the forced message remotely essentially rescinds the forced message, and such a use case is not within the scope of the claims and the existence of such a use case does not preclude infringement.</p>
<p>7. The method as in claim 6, wherein each PDA/cell phone within a predetermined communication network is similarly equipped and has the forced message alert software application program loaded on it.</p>	<p>HTC infringes directly and/or indirectly by performing, inducing others to perform, and/or contributing to the performance of steps of the method comprising: wherein each PDA/cell phone within a predetermined communication network is similarly equipped and has the forced message alert software application program loaded on it. See, e.g., claims 1 and 6, which are incorporated herein by reference in their entirety.</p> <p>Each of the Accused Products performs this additional limitation wherein each PDA/cell phone within a predetermined communication network is similarly equipped and has the forced message alert software application program loaded on it.</p> <p>For example, each Accused Product is currently pre-installed with Android mobile operating systems containing code for providing device-location tracking and forced message alert software. Android Device Manager is the predecessor to Find My Device and has been available as a standard, pre-installed feature since 2013 and downloadable as a software application. The current iteration, Find My Device, often called the “new and improved Android Device Manager” or “rebranded Android Device Manager” is now part of the standard Google Play Protect suite which is “built in and enabled on all devices,” i.e., the Accused Products running Android OS.</p> <p>See, e.g., <a href="https://www.androidcentral.com/find-my-device;">https://www.androidcentral.com/find-my-device;</a>  <a href="https://support.google.com/android/answer/6160491?hl=en;">https://support.google.com/android/answer/6160491?hl=en;</a>  <a href="https://android.googleblog.com/2013/08/find-your-lost-phone-with-android.html;">https://android.googleblog.com/2013/08/find-your-lost-phone-with-android.html;</a>  <a href="https://play.google.com/store/apps/details?id=com.google.android.apps.adm&amp;hl=en;">https://play.google.com/store/apps/details?id=com.google.android.apps.adm&amp;hl=en;</a>  <a href="https://www.blog.google/products/android/google-play-protect/">https://www.blog.google/products/android/google-play-protect/</a></p> <p>Additionally, the devices shown on the Find My Device app within the communication</p>

**Exhibit A for US Patent No. 8,213,970 Against HTC Accused Products**

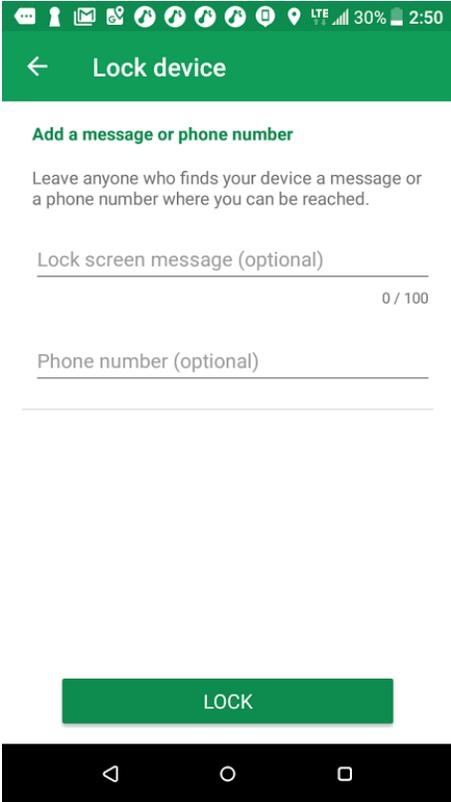
Claim - 8,213,970	Accused Products
	network all have Google Protect Services loaded.
<p>8. The method as in claim 6, wherein said forced message alert application software packet contains a response list, wherein said response list is a default list embedded in the forced message alert software application program.</p>	<p>HTC infringes directly and/or indirectly by performing, inducing others to perform, and/or contributing to the performance of steps of the method comprising: wherein said forced message alert application software packet contains a response list, wherein said response list is a default list embedded in the forced message alert software application program. See, e.g., claims 4 and 6, which are incorporated herein by reference in their entirety.</p> <p>Each of the Accused Products performs this additional limitation wherein said forced message alert application software packet contains a response list, wherein said response list is a default list embedded in the forced message alert software application program.</p> <p>For example, the forced message initiating lock mode may be sent without entering optional information, and thus the default list will be sent.</p>



**Exhibit A for US Patent No. 8,213,970 Against HTC Accused Products**

Claim - 8,213,970	Accused Products
	
<p>9. The method as in claim 6, wherein said forced message alert application software packet contains a response list, wherein said response list is a custom response list that is created at the time the specific forced message alert is created on the sender PDA/cell phone.</p>	<p>HTC infringes directly and/or indirectly by performing, inducing others to perform, and/or contributing to the performance of steps of the method comprising: wherein said forced message alert application software packet contains a response list, wherein said response list is a custom response list that is created at the time the specific forced message alert is created on the sender PDA/cell phone. See, e.g., claims 5 and 6, which are incorporated herein by reference in their entirety.</p> <p>Each of the Accused Products performs this additional limitation wherein said forced message alert application software packet contains a response list, wherein said response list is a</p>

**Exhibit A for US Patent No. 8,213,970 Against HTC Accused Products**

Claim - 8,213,970	Accused Products
	<p>custom response list that is created at the time the specific forced message alert is created on the sender PDA/cell phone.</p> <p>For example, the forced message initiating lock mode may be sent with optional information, and thus the specific list will be sent at the time the forced message is created.</p> 

## **Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

In these Infringement Contentions, AGIS Software Development LLC (“AGIS”) contends that at least the following claims of U.S. Patent No. 9,408,055 (the “’055 Patent”) identified below are infringed by the Accused Products (*e.g.*, phones and tablets) which are manufactured, sold, offered for sale, and/or used by HTC Corporation (“HTC”).

The Accused Products comprise HTC products running the Android mobile operating system and manufactured, used, or sold during and after 2011. For example, the Accused Products comprise the following Android-based phones and tablets: 10, 10 evo, 10 Lifestyle, 2125 / 2100 (Faraday), 3125 / Smartflip / 8500 (Star Trek), 5800 / Fusion / S720, 7 Mozart, 7 Pro, 7 Surround, 7 Trophy, 8125 / 8100 / MDA (USA) / K-JAM / P4300 (Wizard), 8XT, Amaze 4G, Aria, Arrive, Arrive / 7 Pro (CDMA), Bolt, Butterfly, Butterfly 2, Butterfly 3, Butterfly S, ChaCha, Dash / S620 / S621 (Excalibur), Dash 3G / Snap (GSM), Desire, Desire (CDMA), Desire / Desire 601 (CDMA), Desire 10 Compact, Desire 10 Lifestyle, Desire 10 Pro, Desire 200, Desire 210 dual sim, Desire 300, Desire 310, Desire 310 dual sim, Desire 320, Desire 326G dual sim, Desire 400 dual sim, Desire 500, Desire 501, Desire 501 dual sim, Desire 510, Desire 510 (CDMA), Desire 510 (GSM), Desire 516 dual sim, Desire 520, Desire 526, Desire 526 (CDMA), Desire 526G+ dual sim, Desire 530, Desire 555, Desire 600 dual sim, Desire 601, Desire 601 dual sim, Desire 610, Desire 610 (GSM), Desire 612, Desire 612 (CDMA), Desire 616 dual sim, Desire 620, Desire 620G dual sim, Desire 625, Desire 626, Desire 626 (CDMA), Desire 626 (GSM), Desire 626 (USA), Desire 626G+, Desire 626s, Desire 626s (CDMA), Desire 626s (GSM), Desire 628, Desire 630, Desire 650, Desire 700, Desire 700 dual sim, Desire 728 dual sim, Desire 728 Ultra, Desire 816, Desire 816 dual sim, Desire 816G dual sim, Desire 820, Desire 820 dual sim, Desire 820G+ dual sim, Desire 820q dual sim, Desire 820s dual sim, Desire 825, Desire 826 dual sim, Desire 828 dual sim, Desire 830, Desire C, Desire C (CDMA), Desire Eye, Desire HD, Desire L, Desire P, Desire Q, Desire S, Desire SV, Desire U, Desire V, Desire VC, Desire VT, Desire X, Desire XC, Desire Z, Dream, DROID DNA, DROID ERIS, Droid Incredible, DROID Incredible 2, DROID Incredible 4G LTE, EVO 3D, EVO 3D CDMA, Evo 4G, Evo 4G LTE, Evo 4G+, EVO Design 4G, EVO Design 4G / Hero S (CDMA), EVO Shift 4G, EVO V 4G / EVO 3D (CDMA), EVO View 4G, Explorer, First, Flyer, Flyer Wi-Fi, Freestyle, Fuze / Touch Pro (GSM), G1, G2, Glacier, Gratia, HD mini, HD2, HD7, HD7 / HD7S, HD7S, Hero, Hero (CDMA), Hero S, Imagio, Incredible S, Inspire 4G, J, JAMin / S200 (Prophet), Jetstream, Lead, Legend, Magic, MAX 4G, MDA Compact / xda II mini / JAM (Magician), Merge, Mogul / XV6800 / PPC6800 / P4000, myTouch 3G / Magic, myTouch 3G Slide, myTouch 4G, myTouch 4G Slide, One, One (E8), One (E8) CDMA, One (M7 / CDMA), One (M7 / GSM), One (M8 Eye), One (M8), One (M8) (CDMA), One (M8) (GSM), One (M8) CDMA, One (M8) dual sim, One A9, One A9s, One Dual Sim, One E9, One E9+, One M8s, One M9, One M9 (CDMA), One M9 (GSM), One M9 Prime Camera, One M9+, One M9+ Supreme Camera, One M9s, One Max, One max (CDMA), One ME, One mini, One mini 2, One mini 2 (GSM), One Remix, One Remix / One mini 2 (CDMA), One S, One S C2, One S9, One SC, One ST, One SV, One SV CDMA, One V, One VX, One X, One X

## **Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

AT&T, One X+, One X10, One X9, One XC, One XL, Ozone, Ozone XV6175, Panache, Paradise, Prime, Pure, Pure / Touch Diamond2, Radar, Raider 4G, Rezound, Rhyme, Rhyme CDMA, Rider, S710 (Vox), S730, S740, Salsa, Schubert, SDA (USA) / SP5m (Tornado), Sensation, Sensation 4G, Sensation XE, Sensation XL, Shadow, Shadow (2009), Smart, Snap, Snap S511 (CDMA), SP3i / SDA (Europe) (Feeler), SPV C550 (Hurricane), SPV E200 / XPhone (Voyager), Status, Surround, Tattoo, ThunderBolt, ThunderBolt 4G, Tilt 8925 / TyTN II, Tilt2, Titan, Titan II, Touch (CDMA) / XV6900, Touch 3G, Touch Cruise, Touch Cruise 09, Touch Diamond (CDMA), Touch Diamond2, Touch Diamond2 CDMA, Touch Dual, Touch HD, Touch HD T8285, Touch Pro, Touch Pro (CDMA), Touch Pro2, Touch Pro2 (CDMA), Touch Pro2 (GSM) / Tilt 2, Touch Viva, Touch2, Trophy, Trophy (CDMA), TyTN / 8525 / JasJam (Hermes), U Play, U Ultra, U Ultra , U11, U11, U11 Eyes, U11 Life, U11 Plus, U11+, Velocity 4G, Vivid, Wildfire, Wildfire (CDMA), Wildfire S, Wildfire S (CDMA), Wildfire S (GSM), Wing / P4350 (Herald), xda II / MDA II, FLYER, JETSTREAM, FLYER WI-FI, EVO View 4G, FLYER CDMA, and any variants thereof. AGIS reserves the right to amend this list of Accused Products as discovery progresses. For example, AGIS reviewed Android-based products from multiple Android-based handset manufacturers, including two HTC phones (serial numbers FA73J1500645, FA73E1500899) which are available for inspection at HTC's request. For example, the Accused Products comprise HTC products, including but not limited to the phones and tablets as described herein, running the following versions (and all intervening updates and sub-versions) of the Android mobile operating system: Android 2.3, 4.0, 4.1, 4.2, 4.3, 4.4, 5.0, 5.1, 6.0, 7.0, 7.1, 8.0, and 8.1. For example, the Accused Products comprise HTC products, including but not limited to the phones and tablets as described herein, running any versions of the following Android-based applications and/or software: Android Device Manager, Find My Phone, Find My Device, Google Latitude, Google Plus, Google Hangouts, Google Maps, Google Assistant, Google Search, Google Messages, Android Messenger, Google Allo, Google Duo, Gmail, and Google Chrome. For example, the Accused Products comprise HTC products, including but not limited to the phones and tablets described herein, participating in any networks and/or services related to the execution and/or use of the Android mobile operating system versions and Android-based applications and/or software described herein.

AGIS does not concede that any claims of the '055 Patent that are not listed below are not infringed by the identified products. Moreover, the citations to certain documents and other information below are intended to be exemplary only and in no way foreclose AGIS from citing or relying on additional documents, information, source code, and/or testimony at a later time. These contentions are preliminary in nature, and an analysis of HTC's products, internal documentation, source code, and/or testimony from relevant witnesses may more fully and accurately describe the infringing features of its accused products. Accordingly, AGIS reserves the right to supplement, correct, modify, and/or amend these contentions once such additional information is made available to AGIS. Furthermore, AGIS reserves the right to supplement, correct, modify, and/or amend these contentions as discovery in this case

**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

progresses; in view of the Court’s claim construction order(s); in view of any positions taken by HTC, including but not limited to positions on claim construction, invalidity, and/or non-infringement; and in connection with the preparation and exchange of expert reports.

US9408055B2	HTC
<p>1[P] A method comprising: performing by a first device:</p>	<p>HTC performs either directly or indirectly, induces others to perform, and/or contributes to the performance of each step of this method as set forth below.</p> <p>The Accused Products meet the claim limitations by providing device-location tracking features such as those features described below. For example, the Accused Products meet the claim limitations because they are pre-installed with Android mobile operating systems containing code for providing device-location tracking features as provided in the claims limitations herein. For example, the Accused Products run applications and/or software that run within the Android mobile operating system and that use components of the Android mobile operating system to provide device-location tracking features. Upon information and belief, in addition to the components and features of the Android mobile operating system itself, the following applications and/or software run within the Android mobile operating system and use components of the Android mobile operating system to provide device-location tracking features: Google Latitude, Google Plus, Google Hangouts (including Allo and Duo), Google Maps, Google Chrome, Google Messages, and Android Messenger.</p> <p><b><u>Google Maps Share Location</u></b></p> <p>Share Location is currently included as a standard feature on the Accused Devices operating as a feature of Google Maps. Google Maps is a pre-installed software application in Android OS. The Accused Devices have included the Share Location functionalities since 2009 as part of Google Latitude, which was an opt-in feature for Google Maps on Android OS-based mobile devices, such as the Accused Products. Share Location functionalities were briefly shifted from Latitude for Google Maps to Google Plus and Google Hangouts, until reappearing as a standard feature in Google Maps. Upon information and belief, the Share Location method also uses and/or works in conjunction with functionalities associated with Google Maps, Google Messages, Android Messenger, Location Access, and other features, which are pre-installed on the Accused Products. For the purposes of these contentions, AGIS sets forth Google Maps’ Share Location</p>

**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC
	<p>feature of the Accused Products as representative of this exemplary software. AGIS reserves the right to supplement these contentions to the extent that defendant requires additional information in accordance with P.R. 3-1 and for any other reason.</p> <p><i>See, e.g.,</i> <a href="https://techcrunch.com/2017/03/22/google-maps-now-lets-you-share-your-location-with-friends-and-family-for-a-specific-period-of-time/">https://techcrunch.com/2017/03/22/google-maps-now-lets-you-share-your-location-with-friends-and-family-for-a-specific-period-of-time/</a>; <a href="https://googleblog.blogspot.com/2009/02/see-where-your-friends-are-with-google.html">https://googleblog.blogspot.com/2009/02/see-where-your-friends-are-with-google.html</a>; <a href="https://googleblog.blogspot.jp/2013/07/a-new-google-maps-app-for-smartphones.html">https://googleblog.blogspot.jp/2013/07/a-new-google-maps-app-for-smartphones.html</a>; <a href="http://googleplusproject.blogspot.com/2013/05/google-for-android-42.html">http://googleplusproject.blogspot.com/2013/05/google-for-android-42.html</a>; <a href="https://googleblog.blogspot.com/2013/10/google-hangouts-and-photos-save-some.html">https://googleblog.blogspot.com/2013/10/google-hangouts-and-photos-save-some.html</a></p> <p><b>Control within reach, even when your device isn't</b></p> <p>One of the biggest security risks you're likely to face is simply losing your phone. To help in these times of need, we're launching <a href="#">Find My Device</a> as part of Google Play Protect. With Find My Device you can locate, ring, lock and erase your Android devices—phones, tablets, and even watches. This feature is built in and enabled on all devices; visit <a href="http://android.com/find">android.com/find</a> or check out <a href="#">the app</a>.</p> <p><i>See, e.g.,</i> <a href="https://www.blog.google/products/android/google-play-protect/">https://www.blog.google/products/android/google-play-protect/</a></p>

**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC
	<p data-bbox="520 233 911 264"><b>Link your phone to Google</b></p> <p data-bbox="520 280 1201 315">You can connect your Android phone to Google, which lets you send information from your computer to your phone. For example, you can send directions you searched for on your computer to Google Maps on your phone.</p> <p data-bbox="520 342 737 360"><b>Link your Android phone</b></p> <p data-bbox="520 386 737 404"><b>Step 1: Update the Google app</b></p> <ol data-bbox="520 410 884 448" style="list-style-type: none"><li>1. On your phone, go to the <a href="#">Google app page on the Play Store</a>.</li><li>2. Tap <b>Update</b>.</li></ol> <p data-bbox="520 472 720 490"><b>Step 2: Turn on Google Now</b></p> <ol data-bbox="520 496 848 578" style="list-style-type: none"><li>1. On your phone, open the Google app .</li><li>2. At the top left, tap Menu  &gt; <b>Settings</b> &gt; <b>Now cards</b>.</li><li>3. Turn on <b>Show cards</b>.</li><li>4. Turn on <b>Show notifications</b>.</li></ol> <p data-bbox="520 602 768 620"><b>Step 3: Turn on Web &amp; App Activity</b></p> <ol data-bbox="520 626 737 664" style="list-style-type: none"><li>1. Visit the <a href="#">Account History page</a>.</li><li>2. Make sure the switch is on (green).</li></ol> <p data-bbox="520 688 737 706"><b>Step 4: Sign in to your browser</b></p> <ol data-bbox="520 712 978 839" style="list-style-type: none"><li>1. On your phone, open the Google app .</li><li>2. At the top left, tap the Menu .</li><li>3. At the top left, you'll see the email address you use for the Google app.</li><li>4. Visit <a href="http://www.google.com">www.google.com</a>  on your computer.</li><li>5. If you aren't signed in already, click <b>Sign in</b> in the top right corner of the page.</li><li>6. Sign in using the Google Account you use for the Google app.</li></ol> <p data-bbox="520 863 800 881"><b>Step 5: Send information to your phone</b></p> <ol data-bbox="520 888 1209 941" style="list-style-type: none"><li>1. Do one of the searches below, like <b>note to self</b>, or <b>send directions to my phone</b>.</li><li>2. If a box doesn't pop up with the option to send information to your phone, try refreshing the page. If you just turned on Google Now, it may take a few minutes for the box to show up</li></ol> <p data-bbox="512 953 1218 984"><a href="https://support.google.com/websearch/answer/6128427">https://support.google.com/websearch/answer/6128427</a></p>

## Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products

US9408055B2	HTC
	<p><b>What you can do once your phone is linked</b></p> <hr/> <p><b>Find my phone</b> <span style="float: right;">^</span></p> <p>You can get the current location of your phone if you can't find it.</p> <ol style="list-style-type: none"> <li>1. On your computer's browser, search on <a href="http://www.google.com">www.google.com</a> for <b>find my phone</b>.</li> <li>2. If your phone is turned on and connected to the Internet, you'll see your phone's location.</li> <li>3. If your phone's location is unavailable, you can still make it ring for 5 minutes on full volume by clicking <b>Ring</b>. You can stop the ringing from your phone when you find it.</li> </ol> <p><b>Tip:</b> You can also find your missing phone using the <a href="#">Android Device manager</a> which lets you find your device or remotely ring, lock, or erase it.</p> <hr/> <p><b>Send directions to my phone</b> <span style="float: right;">^</span></p> <p>Once you've looked up directions on your computer, you can send them to your phone so you have them on your trip.</p> <ol style="list-style-type: none"> <li>1. On your computer's browser, search on <a href="http://www.google.com">www.google.com</a> for <b>send directions to my phone</b>.</li> <li>2. Enter in your destination.</li> <li>3. Click <b>Send directions to your phone</b>.</li> <li>4. You'll get a notification on your phone. Tap to navigate to your destination using Google Maps.</li> </ol> <hr/> <p><b>Send a note to my phone</b> <span style="float: right;">^</span></p> <ol style="list-style-type: none"> <li>1. On your computer's browser, search on <a href="http://www.google.com">www.google.com</a> for <b>send a note to my phone</b>.</li> <li>2. Type your note in the box.</li> <li>3. Click <b>Send note to your phone</b>.</li> <li>4. You'll get a notification on your phone with your note that you can either save to one of your apps or copy.</li> </ol> <hr/> <p><b>Set an alarm</b> <span style="float: right;">^</span></p> <ol style="list-style-type: none"> <li>1. On your computer's browser, search on <a href="http://www.google.com">www.google.com</a> for <b>set an alarm</b>.</li> <li>2. Choose the time you want the alarm to go off.</li> <li>3. Click <b>Set an alarm on your phone</b>.</li> <li>4. An alarm will now be set on your phone's Clock app.</li> </ol> <hr/> <p><b>Set a reminder</b> <span style="float: right;">^</span></p> <ol style="list-style-type: none"> <li>1. On your computer's browser, search on <a href="http://www.google.com">www.google.com</a> for <b>set an reminder</b>.</li> <li>2. Type what you want to be reminded about, and either when or where you want the reminder to go off.</li> <li>3. Click <b>Remind me on my devices</b>.</li> </ol>



**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC
	<p data-bbox="512 235 1220 264"><a href="https://support.google.com/websearch/answer/6128427">https://support.google.com/websearch/answer/6128427</a></p> <h2 data-bbox="548 298 1327 345">Share your location using Google Maps</h2> <p data-bbox="548 363 1465 410">You can't share your location in Google+ anymore. If you used to share your location in Google+ and want to keep sharing it, you'll need to share it again in Google Maps.</p> <p data-bbox="520 427 1635 456"><a href="https://support.google.com/plus/answer/3302509?hl=en&amp;co=GENIE.Platform%3DAndroid&amp;oco=1">https://support.google.com/plus/answer/3302509?hl=en&amp;co=GENIE.Platform%3DAndroid&amp;oco=1</a></p> <h3 data-bbox="533 505 630 529"><b>Location</b></h3> <p data-bbox="533 542 1411 634">Turn on location service, your phone determines your approximate location using Wi-Fi and mobile networks. When you select this option, you're asked whether you consent to allowing Google to use your location when providing these services.</p> <ul data-bbox="562 649 1369 862" style="list-style-type: none"><li>• <b>Mode</b> – Sets the how your current location information is determined.</li><li>• <b>Recent Location Request</b> – Displays applications and services that have recently requested your location information.</li><li>• <b>Camera</b> – Checkmark to tag photos or videos with their locations.</li><li>• <b>Google Location History</b> – Allows you to view and manage your Google location history.</li></ul> <h3 data-bbox="533 876 716 901"><b>Accounts &amp; sync</b></h3> <p data-bbox="533 915 1402 1044">Use the Accounts &amp; sync settings menu to add, remove, and manage your Google and other supported accounts. You also use these settings to control how and whether all applications send, receive, and sync data on their own schedules and whether all applications can synchronize user data automatically.</p> <p data-bbox="533 1057 1373 1149">Gmail™, Calendar, and other applications may also have their own settings to control how they synchronize data; see the sections on those applications for details. Touch <b>Add account</b> to add new account.</p>

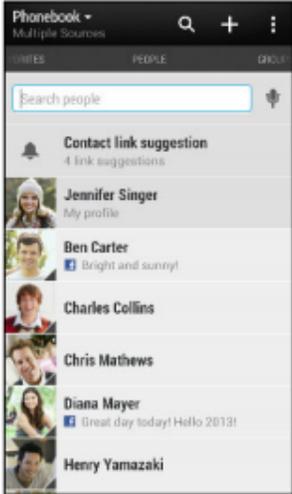
**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC
	 <p><a href="http://googlemobile.blogspot.com/2009/02/locate-your-friends-in-real-time-with.html">http://googlemobile.blogspot.com/2009/02/locate-your-friends-in-real-time-with.html</a></p> <p>Google’s location-sharing feature also appeared in Google+, Google Trust Contacts, and Google Hangouts services until its current integration in Google Maps.</p> <p>HTC makes, uses, sells, and otherwise provides this first device by making, using, selling, and importing Android devices such as HTC mobile devices, HTC tablets, and HTC Smartwatches as well as by providing its servers or using third party servers (e.g., Google servers) for use with Android devices to enable features such as Maps. Below are example HTC Android devices that perform each step of this method as set forth below.</p>
[1A] obtaining contact information of a plurality of second	HTC infringes directly and/or indirectly by performing, inducing others to perform, and/or contributing to the performance of: obtaining contact information of a plurality of second devices, wherein the contact information comprises respective telephone numbers of the second devices.

**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

<b>US9408055B2</b>	<b>HTC</b>
devices, wherein the contact information comprises respective telephone numbers of the second devices;	For example, the Accused products include a contacts app to access contact information for second users using respective second devices.

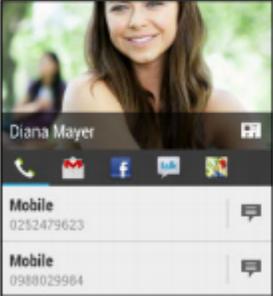
## Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products

US9408055B2	HTC
	<p data-bbox="562 233 936 272"><b>Your contacts list</b></p> <p data-bbox="562 310 1560 402">The Contacts app lists all contacts you've stored on HTC One and from online accounts you're logged in to. Use the Contacts app to easily manage communications with people that matter to you.</p> <ol data-bbox="590 431 909 456" style="list-style-type: none"><li data-bbox="590 431 909 456">1. Open the Contacts app.</li></ol>  <p data-bbox="590 997 982 1021">2. On your contacts list, you can:</p> <ul data-bbox="653 1049 1556 1289" style="list-style-type: none"><li data-bbox="653 1049 1289 1073">▪ View your profile and edit your contact information.</li><li data-bbox="653 1084 1087 1109">▪ Create, edit, find, or send contacts.</li><li data-bbox="653 1120 909 1144">▪ See status updates.</li><li data-bbox="653 1156 1499 1180">▪ Tap a contact photo to find ways to quickly connect with the contact.</li><li data-bbox="653 1192 1465 1216">▪ See a notification icon when a contact has sent you new messages.</li><li data-bbox="653 1227 1556 1289">▪ Check out who's online in Google Talk™. Online status icons are displayed if you're signed in to Google Talk.</li></ul> <p data-bbox="516 1317 1566 1390"> To sort your contacts by their first or last name, tap  &gt; Settings &gt; Sort contact list.</p>

## Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products

US9408055B2	HTC
	<p data-bbox="531 245 926 277"><b>Filtering your contacts list</b></p> <p data-bbox="531 302 1575 329">When your contacts list gets long, you can choose which contact accounts to show.</p> <ol data-bbox="562 358 1444 488" style="list-style-type: none"> <li data-bbox="562 358 957 386">1. On the Contacts tab, tap ▼.</li> <li data-bbox="562 407 1444 435">2. Choose the accounts that contain the contacts you want to display.</li> <li data-bbox="562 456 716 483">3. Press &lt;.</li> </ol> <p data-bbox="531 529 758 561"><b>Finding people</b></p> <p data-bbox="531 586 1518 646">Search for contacts stored on HTC One, your company directory if you have an Exchange ActiveSync account, or social networks you've signed into.</p> <ol data-bbox="562 675 1608 1097" style="list-style-type: none"> <li data-bbox="562 675 905 703">1. Open the Contacts app.</li> <li data-bbox="562 724 1608 1097">2. On the Contacts tab, you can: <ul data-bbox="625 781 1608 1097" style="list-style-type: none"> <li data-bbox="625 781 1608 841">▪ Find people in your contacts list. Tap the Search people box, and then enter the first few letters of the contact name.</li> <li data-bbox="625 854 1608 951">▪ Find people on your company directory. Tap the Search people box, enter the first few letters of the contact name, and then tap Search contacts in your Company Directory.</li> <li data-bbox="625 964 1608 1097">▪ Search for people you know on your social networks. Tap ☰ &gt; Settings &gt; Find people you know on, and then select the social networks you're signed in to. The Contacts app then uploads your contacts to the selected social networks to help you find friends.</li> </ul> </li> </ol> <p data-bbox="531 1138 1591 1235">) Aside from searching for a contact by name, you can search using a contact's email address or company name. On the Contacts tab, tap ☰ &gt; Settings &gt; Search contacts by, and then choose a search criteria.</p>

## Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products

US9408055B2	HTC
	<p data-bbox="569 240 1283 285"><b>Getting in touch with a contact</b></p> <ol data-bbox="600 329 1629 440" style="list-style-type: none"><li data-bbox="600 329 940 358">1. Open the Contacts app.</li><li data-bbox="600 378 1629 440">2. Tap a contact's photo (not the name), and then choose how you want to get in touch with that contact.</li></ol>  <p data-bbox="527 805 1646 867"> For more ways of getting in touch with your contact, tap an icon below the contact photo.</p>

**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC
	<p>In another example, the Accused products run Android Messages and Google Hangouts which both access contact information for second users using respective second devices.</p> <h2 data-bbox="527 347 1016 402">Contacts Provider</h2> <p data-bbox="527 436 1472 699">The Contacts Provider is a powerful and flexible Android component that manages the device's central repository of data about people. The Contacts Provider is the source of data you see in the device's contacts application, and you can also access its data in your own application and transfer data between the device and online services. The provider accommodates a wide range of data sources and tries to manage as much data as possible for each person, with the result that its organization is complex. Because of this, the provider's API includes an extensive set of contract classes and interfaces that facilitate both data retrieval and modification.</p> <p data-bbox="527 732 835 756">This guide describes the following:</p> <ul data-bbox="527 784 1373 959" style="list-style-type: none"><li data-bbox="527 784 806 808">• The basic provider structure.</li><li data-bbox="527 836 894 860">• How to retrieve data from the provider.</li><li data-bbox="527 888 863 912">• How to modify data in the provider.</li><li data-bbox="527 940 1373 959">• How to write a sync adapter for synchronizing data from your server to the Contacts Provider.</li></ul> <p data-bbox="512 974 1486 1003"><a href="https://developer.android.com/guide/topics/providers/contacts-provider.html">https://developer.android.com/guide/topics/providers/contacts-provider.html</a></p>

**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC
	<p data-bbox="533 240 657 266"><b>Overview</b></p> <p data-bbox="533 293 1608 313">ContactsContract defines an extensible database of contact-related information. Contact information is stored in a three-tier data model:</p> <ul data-bbox="533 339 1713 544" style="list-style-type: none"><li data-bbox="533 339 1713 391">• A row in the <code>ContactsContract.Data</code> table can store any kind of personal data, such as a phone number or email addresses. The set of data kinds that can be stored in this table is open-ended. There is a predefined set of common kinds, but any application can add its own data kinds.</li><li data-bbox="533 417 1713 469">• A row in the <code>ContactsContract.RawContacts</code> table represents a set of data describing a person and associated with a single account (for example, one of the user's Gmail accounts).</li><li data-bbox="533 495 1713 544">• A row in the <code>ContactsContract.Contacts</code> table represents an aggregate of one or more RawContacts presumably describing the same person. When data in or associated with the RawContacts table is changed, the affected aggregate contacts are updated as necessary.</li></ul> <p data-bbox="533 573 695 592">Other tables include:</p> <ul data-bbox="533 618 1713 898" style="list-style-type: none"><li data-bbox="533 618 1713 670">• <code>ContactsContract.Groups</code>, which contains information about raw contact groups such as Gmail contact groups. The current API does not support the notion of groups spanning multiple accounts.</li><li data-bbox="533 696 1325 716">• <code>ContactsContract.StatusUpdates</code>, which contains social status updates including IM availability.</li><li data-bbox="533 742 1493 761">• <code>ContactsContract.AggregationExceptions</code>, which is used for manual aggregation and disaggregation of raw contacts</li><li data-bbox="533 787 1325 807">• <code>ContactsContract.Settings</code>, which contains visibility and sync settings for accounts and groups.</li><li data-bbox="533 833 1346 852">• <code>ContactsContract.SyncState</code>, which contains free-form data maintained on behalf of sync adapters</li><li data-bbox="533 878 1129 898">• <code>ContactsContract.PhoneLookup</code>, which is used for quick caller-ID lookup</li></ul> <p data-bbox="512 924 1541 950"><a href="https://developer.android.com/reference/android/provider/ContactsContract.html">https://developer.android.com/reference/android/provider/ContactsContract.html</a></p> <p data-bbox="533 995 615 1021"><b>Data</b></p> <p data-bbox="533 1063 1749 1206">As noted previously, the data for a raw contact is stored in a <code>ContactsContract.Data</code> row that is linked to the raw contact's <code>_ID</code> value. This allows a single raw contact to have multiple instances of the same type of data such as email addresses or phone numbers. For example, if "Thomas Higginson" for <code>emilyd@gmail.com</code> (the raw contact row for Thomas Higginson associated with the Google account <code>emilyd@gmail.com</code>) has a home email address of <code>thigg@gmail.com</code> and a work email address of <code>thomas.higginson@gmail.com</code>, the Contacts Provider stores the two email address rows and links them both to the raw contact.</p> <p data-bbox="533 1232 1749 1346">Notice that different types of data are stored in this single table. Display name, phone number, email, postal address, photo, and website detail rows are all found in the <code>ContactsContract.Data</code> table. To help manage this, the <code>ContactsContract.Data</code> table has some columns with descriptive names, and others with generic names. The contents of a descriptive-name column have the same meaning regardless of the type of data in the row, while the contents of a generic-name column have different meanings depending on the type of data.</p>



**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC										
	<p data-bbox="512 235 1486 264"><a href="https://developer.android.com/guide/topics/providers/contacts-provider.html">https://developer.android.com/guide/topics/providers/contacts-provider.html</a></p> <table border="1" data-bbox="520 305 1745 976"><thead><tr><th data-bbox="520 305 619 344">Task</th><th data-bbox="619 305 856 344">Action</th><th data-bbox="856 305 1190 344">Data</th><th data-bbox="1190 305 1480 344">MIME type</th><th data-bbox="1480 305 1745 344">Notes</th></tr></thead><tbody><tr><td data-bbox="520 344 619 976">Pick a contact from a list</td><td data-bbox="619 344 856 976">ACTION_PICK</td><td data-bbox="856 344 1190 976">One of:<ul data-bbox="877 391 1176 784" style="list-style-type: none"><li data-bbox="877 391 1176 446">• <code>Contacts.CONTENT_URI</code>, which displays a list of contacts.</li><li data-bbox="877 472 1176 557">• <code>Phone.CONTENT_URI</code>, which displays a list of phone numbers for a raw contact.</li><li data-bbox="877 583 1176 667">• <code>StructuredPostal.CONTENT_URI</code>, which displays a list of postal addresses for a raw contact.</li><li data-bbox="877 693 1176 784">• <code>Email.CONTENT_URI</code>, which displays a list of email addresses for a raw contact.</li></ul></td><td data-bbox="1190 344 1480 976">Not used</td><td data-bbox="1480 344 1745 976">Displays a list of raw contacts or a list of data from a raw contact, depending on the content URI type you supply. Call <code>startActivityForResult()</code>, which returns the content URI of the selected row. The form of the URI is the table's content URI with the row's <code>LOOKUP_ID</code> appended to it. The device's contacts app delegates read and write permissions to this content URI for the life of your activity. See the <a href="#">Content Provider Basics</a> guide for more details.</td></tr></tbody></table> <p data-bbox="512 1015 1486 1044"><a href="https://developer.android.com/guide/topics/providers/contacts-provider.html">https://developer.android.com/guide/topics/providers/contacts-provider.html</a></p>	Task	Action	Data	MIME type	Notes	Pick a contact from a list	ACTION_PICK	One of: <ul data-bbox="877 391 1176 784" style="list-style-type: none"><li data-bbox="877 391 1176 446">• <code>Contacts.CONTENT_URI</code>, which displays a list of contacts.</li><li data-bbox="877 472 1176 557">• <code>Phone.CONTENT_URI</code>, which displays a list of phone numbers for a raw contact.</li><li data-bbox="877 583 1176 667">• <code>StructuredPostal.CONTENT_URI</code>, which displays a list of postal addresses for a raw contact.</li><li data-bbox="877 693 1176 784">• <code>Email.CONTENT_URI</code>, which displays a list of email addresses for a raw contact.</li></ul>	Not used	Displays a list of raw contacts or a list of data from a raw contact, depending on the content URI type you supply. Call <code>startActivityForResult()</code> , which returns the content URI of the selected row. The form of the URI is the table's content URI with the row's <code>LOOKUP_ID</code> appended to it. The device's contacts app delegates read and write permissions to this content URI for the life of your activity. See the <a href="#">Content Provider Basics</a> guide for more details.
Task	Action	Data	MIME type	Notes							
Pick a contact from a list	ACTION_PICK	One of: <ul data-bbox="877 391 1176 784" style="list-style-type: none"><li data-bbox="877 391 1176 446">• <code>Contacts.CONTENT_URI</code>, which displays a list of contacts.</li><li data-bbox="877 472 1176 557">• <code>Phone.CONTENT_URI</code>, which displays a list of phone numbers for a raw contact.</li><li data-bbox="877 583 1176 667">• <code>StructuredPostal.CONTENT_URI</code>, which displays a list of postal addresses for a raw contact.</li><li data-bbox="877 693 1176 784">• <code>Email.CONTENT_URI</code>, which displays a list of email addresses for a raw contact.</li></ul>	Not used	Displays a list of raw contacts or a list of data from a raw contact, depending on the content URI type you supply. Call <code>startActivityForResult()</code> , which returns the content URI of the selected row. The form of the URI is the table's content URI with the row's <code>LOOKUP_ID</code> appended to it. The device's contacts app delegates read and write permissions to this content URI for the life of your activity. See the <a href="#">Content Provider Basics</a> guide for more details.							

**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC
	<pre>25  * Parsed form of the intent sent to the Contacts application. 26  */ 27  public class ContactsRequest { 28 29      /** Default mode: browse contacts */ 30      public static final int ACTION_DEFAULT = 10; 31 32      /** Show all contacts */ 33      public static final int ACTION_ALL_CONTACTS = 15; 34 35      /** Show all contacts with phone numbers */ 36      public static final int ACTION_CONTACTS_WITH_PHONES = 17; 37 38      /** Show contents of a specific group */ 39      public static final int ACTION_GROUP = 20; 40 41      /** Show all starred contacts */ 42      public static final int ACTION_STARRED = 30; 43 44      /** Show frequently contacted contacts */ 45      public static final int ACTION_FREQUENT = 40; 46 47      /** Show starred and the frequent */ 48      public static final int ACTION_STREQUENT = 50; 49 50      /** Show all contacts and pick them when clicking */ 51      public static final int ACTION_PICK_CONTACT = 60; 52 53      /** Show all contacts as well as the option to create a new one */ 54      public static final int ACTION_PICK_OR_CREATE_CONTACT = 70; 55 56      /** Show all contacts and pick them for edit when clicking, and allow creating a new contact */ 57      public static final int ACTION_INSERT_OR_EDIT_CONTACT = 80;</pre> <p><a href="https://android.googlesource.com/platform/packages/apps/Contacts/+nougat-mr1-release/src/com/android/contacts/list/ContactsRequest.java">https://android.googlesource.com/platform/packages/apps/Contacts/+nougat-mr1-release/src/com/android/contacts/list/ContactsRequest.java</a></p>

**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC
	<pre> 59      /** Show all phone numbers and pick them when clicking */ 60      public static final int ACTION_PICK_PHONE = 90; 61 62      /** Show all postal addresses and pick them when clicking */ 63      public static final int ACTION_PICK_POSTAL = 100; 64 65      /** Show all postal addresses and pick them when clicking */ 66      public static final int ACTION_PICK_EMAIL = 105; 67 68      /** Show all contacts and create a shortcut for the picked contact */ 69      public static final int ACTION_CREATE_SHORTCUT_CONTACT = 110; 70 71      /** Show all phone numbers and create a call shortcut for the picked number */ 72      public static final int ACTION_CREATE_SHORTCUT_CALL = 120; 73 74      /** Show all phone numbers and create an SMS shortcut for the picked number */ 75      public static final int ACTION_CREATE_SHORTCUT_SMS = 130; 76 77      /** Show all contacts and activate the specified one */ 78      public static final int ACTION_VIEW_CONTACT = 140; 79 80      /** Show contacts recommended for joining with a specified target contact */ 81      public static final int ACTION_PICK_JOIN = 150; </pre> <p><a href="https://android.googlesource.com/platform/packages/apps/Contacts+/nougat-mr1-release/src/com/android/contacts/list/ContactsRequest.java">https://android.googlesource.com/platform/packages/apps/Contacts+/nougat-mr1-release/src/com/android/contacts/list/ContactsRequest.java</a></p>

**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC
	<pre> 104      * Displays a list to browse contacts. 105      */ 106      public class PeopleActivity extends ContactsActivity implements 107          View.OnCreateContextMenuListener, 108          View.OnClickListener, 109          ActionBarAdapter.Listener, 110          DialogManager.DialogShowingViewActivity, 111          ContactListFilterController.ContactListFilterListener, 112          ProviderStatusListener, 113          MultiContactDeleteListener, 114          JoinContactsListener { </pre> <p data-bbox="506 662 1566 727"><a href="https://android.googlesource.com/platform/packages/apps/Contacts/+/nougat-mr1-release/src/com/android/contacts/activities/PeopleActivity.java">https://android.googlesource.com/platform/packages/apps/Contacts/+/nougat-mr1-release/src/com/android/contacts/activities/PeopleActivity.java</a></p> <pre> 145      * Showing a list of Contacts. Also used for showing search results in search mode. 146      */ 147      private MultiSelectContactsListFragment mAllFragment; 148      private ContactTileListFragment mFavoritesFragment; </pre> <p data-bbox="506 873 1566 937"><a href="https://android.googlesource.com/platform/packages/apps/Contacts/+/nougat-mr1-release/src/com/android/contacts/activities/PeopleActivity.java">https://android.googlesource.com/platform/packages/apps/Contacts/+/nougat-mr1-release/src/com/android/contacts/activities/PeopleActivity.java</a></p>

**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC
	<pre>458     private void configureFragments(boolean fromRequest) { 459         if (fromRequest) { 460             ContactListFilter filter = null; 461             int actionCode = mRequest.getActionCode(); 462             boolean searchMode = mRequest.isSearchMode(); 463             final int tabToOpen; 464             switch (actionCode) { 465                 case ContactsRequest.ACTION_ALL_CONTACTS: 466                     filter = ContactListFilter.createFilterWithType( 467                         ContactListFilter.FILTER_TYPE_ALL_ACCOUNTS); 468                     tabToOpen = TabState.ALL; 469                     break; 470                 case ContactsRequest.ACTION_CONTACTS_WITH_PHONES: 471                     filter = ContactListFilter.createFilterWithType( 472                         ContactListFilter.FILTER_TYPE_WITH_PHONE_NUMBERS_ONLY); 473                     tabToOpen = TabState.ALL; 474                     break; 475 476                 case ContactsRequest.ACTION_FREQUENT: 477                 case ContactsRequest.ACTION_STREQUENT: 478                 case ContactsRequest.ACTION_STARRED: 479                     tabToOpen = TabState.FAVORITES; 480                     break; 481                 case ContactsRequest.ACTION_VIEW_CONTACT: 482                     tabToOpen = TabState.ALL; 483                     break; 484                 default: 485                     tabToOpen = -1; 486                     break; 487             } </pre> <p data-bbox="506 1321 1566 1390"><a href="https://android.googlesource.com/platform/packages/apps/Contacts+/nougat-mr1-release/src/com/android/contacts/activities/PeopleActivity.java">https://android.googlesource.com/platform/packages/apps/Contacts+/nougat-mr1-release/src/com/android/contacts/activities/PeopleActivity.java</a></p>

**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC
	<pre>488         if (tabToOpen != -1) { 489             mActionBarAdapter.setCurrentTab(tabToOpen); 490         } 491 492         if (filter != null) { 493             mContactListFilterController.setContactListFilter(filter, false); 494             searchMode = false; 495         } 496 497         if (mRequest.getContactUri() != null) { 498             searchMode = false; 499         } 500 501         mActionBarAdapter.setSearchMode(searchMode); 502         configureContactListFragmentForRequest(); 503     } 504 505     configureContactListFragment(); 506 507     invalidateOptionsMenuIfNeeded(); 508 }</pre> <p data-bbox="506 1019 1570 1084"><a href="https://android.googlesource.com/platform/packages/apps/Contacts+/nougat-mr1-release/src/com/android/contacts/activities/PeopleActivity.java">https://android.googlesource.com/platform/packages/apps/Contacts+/nougat-mr1-release/src/com/android/contacts/activities/PeopleActivity.java</a></p>

**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC
	<pre>35 public class ProfileAndContactsLoader extends CursorLoader { 36 37     private boolean mLoadProfile; 38 39     private String[] mProjection; 40 41     private Uri mExtraUri; 42     private String[] mExtraProjection; 43     private String mExtraSelection; 44     private String[] mExtraSelectionArgs; 45     private boolean mMergeExtraContactsAfterPrimary; 46 47     public ProfileAndContactsLoader(Context context) { 48         super(context); 49     }</pre> <p><a href="https://android.googlesource.com/platform/packages/apps/ContactsCommon/+/nougat-mr1-release/src/com/android/contacts/common/list/ProfileAndContactsLoader.java">https://android.googlesource.com/platform/packages/apps/ContactsCommon/+/nougat-mr1-release/src/com/android/contacts/common/list/ProfileAndContactsLoader.java</a></p>

**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC
	<pre>35 public final class GroupMemberLoader extends CursorLoader { 36 37     public static class GroupEditorQuery { 38         private static final String[] PROJECTION = new String[] { 39             Data.CONTACT_ID,           // 0 40             Data.RAW_CONTACT_ID,      // 1 41             Data.DISPLAY_NAME_PRIMARY, // 2 42             Data.PHOTO_URI,           // 3 43             Data.LOOKUP_KEY,          // 4 44         }; 45 46         public static final int CONTACT_ID           = 0; 47         public static final int RAW_CONTACT_ID       = 1; 48         public static final int CONTACT_DISPLAY_NAME_PRIMARY = 2; 49         public static final int CONTACT_PHOTO_URI    = 3; 50         public static final int CONTACT_LOOKUP_KEY   = 4; 51     } 52 53     public static class GroupDetailQuery { 54         private static final String[] PROJECTION = new String[] { 55             Data.CONTACT_ID,           // 0 56             Data.PHOTO_URI,           // 1 57             Data.LOOKUP_KEY,          // 2 58             Data.DISPLAY_NAME_PRIMARY, // 3 59             Data.CONTACT_PRESENCE,    // 4 60             Data.CONTACT_STATUS,      // 5 61         }; 62 63         public static final int CONTACT_ID           = 0; 64         public static final int CONTACT_PHOTO_URI   = 1; 65         public static final int CONTACT_LOOKUP_KEY   = 2; 66         public static final int CONTACT_DISPLAY_NAME_PRIMARY = 3; 67         public static final int CONTACT_PRESENCE_STATUS = 4; 68         public static final int CONTACT_STATUS      = 5; 69     } 70 71     private final long mGroupId;</pre> <p><a href="https://android.googlesource.com/platform/packages/apps/Contacts/+nougat-mr1-release/src/com/android/contacts/GroupMemberLoader.java">https://android.googlesource.com/platform/packages/apps/Contacts/+nougat-mr1-release/src/com/android/contacts/GroupMemberLoader.java</a></p>



**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC
	<pre> 24  * Group loader for the group list that includes details such as the number of contacts per group 25  * and number of groups per account. This list is sorted by account type, account name, where the 26  * group names are in alphabetical order. Note that the list excludes default, favorite, and deleted 27  * groups. 28  */ 29  public final class GroupListLoader extends CursorLoader { 30 31      private final static String[] COLUMNS = new String[] { 32          Groups.ACCOUNT_NAME, 33          Groups.ACCOUNT_TYPE, 34          Groups.DATA_SET, 35          Groups._ID, 36          Groups.TITLE, 37          Groups.SUMMARY_COUNT, 38      }; 39 40      public final static int ACCOUNT_NAME = 0; 41      public final static int ACCOUNT_TYPE = 1; 42      public final static int DATA_SET = 2; 43      public final static int GROUP_ID = 3; 44      public final static int TITLE = 4; 45      public final static int MEMBER_COUNT = 5; 46 47      private static final Uri GROUP_LIST_URI = Groups.CONTENT_SUMMARY_URI; 48 49      public GroupListLoader(Context context) { 50          super(context, GROUP_LIST_URI, COLUMNS, Groups.ACCOUNT_TYPE + " NOT NULL AND " 51              + Groups.ACCOUNT_NAME + " NOT NULL AND " + Groups.AUTO_ADD + "=0 AND " + 52              Groups.FAVORITES + "=0 AND " + Groups.DELETED + "=0", null, 53              Groups.ACCOUNT_TYPE + ", " + Groups.ACCOUNT_NAME + ", " + Groups.DATA_SET + ", " + 54              Groups.TITLE + " COLLATE LOCALIZED ASC"); 55      } 56  } </pre> <p><a href="https://android.googlesource.com/platform/packages/apps/Contacts+/nougat-mr1-release/src/com/android/contacts/GroupListLoader.java">https://android.googlesource.com/platform/packages/apps/Contacts+/nougat-mr1-release/src/com/android/contacts/GroupListLoader.java</a></p>

**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC
	<pre> 24  * Group meta-data loader. Loads all groups or just a single group from the 25  * database (if given a {@link Uri}). 26  */ 27  public final class GroupMetaDataLoader extends CursorLoader { 28 29      private final static String[] COLUMNS = new String[] { 30          Groups.ACCOUNT_NAME, 31          Groups.ACCOUNT_TYPE, 32          Groups.DATA_SET, 33          Groups._ID, 34          Groups.TITLE, 35          Groups.AUTO_ADD, 36          Groups.FAVORITES, 37          Groups.GROUP_IS_READ_ONLY, 38          Groups.DELETED, 39      }; 40 41      public final static int ACCOUNT_NAME = 0; 42      public final static int ACCOUNT_TYPE = 1; 43      public final static int DATA_SET = 2; 44      public final static int GROUP_ID = 3; 45      public final static int TITLE = 4; 46      public final static int AUTO_ADD = 5; 47      public final static int FAVORITES = 6; 48      public final static int IS_READ_ONLY = 7; 49      public final static int DELETED = 8; 50 51      public GroupMetaDataLoader(Context context, Uri groupUri) { 52          super(context, ensureIsGroupUri(groupUri), COLUMNS, Groups.ACCOUNT_TYPE + " NOT NULL AND " 53              + Groups.ACCOUNT_NAME + " NOT NULL", null, null); 54      } 55 56      /** 57       * Ensures that this is a valid group URI. If invalid, then an exception is 58       * thrown. Otherwise, the original URI is returned. 59       */ 60      private static Uri ensureIsGroupUri(final Uri groupUri) { 61          // TODO: Fix ContactsProvider2 getType method to resolve the group Uris 62          if (groupUri == null) { 63              throw new IllegalArgumentException("Uri must not be null"); 64          } 65          if (!groupUri.toString().startsWith(Groups.CONTENT_URI.toString())) { 66              throw new IllegalArgumentException("Invalid group Uri: " + groupUri); 67          } 68          return groupUri; 69      } 70  } </pre> <p><a href="https://android.googlesource.com/platform/packages/apps/Contacts/+nougat-mr1-release/src/com/android/contacts/GroupMetaDataLoader.java">https://android.googlesource.com/platform/packages/apps/Contacts/+nougat-mr1-release/src/com/android/contacts/GroupMetaDataLoader.java</a></p>

**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC
	<pre> 19  * Meta-data for a contact group. We load all groups associated with the contact's 20  * constituent accounts. 21  */ 22  public final class GroupMetaData { 23      private String mAccountName; 24      private String mAccountType; 25      private String mDataSet; 26      private long mGroupId; 27      private String mTitle; 28      private boolean mDefaultGroup; 29      private boolean mFavorites; 30 31      public GroupMetaData(String accountName, String accountType, String dataSet, long groupId, 32                          String title, boolean defaultGroup, boolean favorites) { 33          this.mAccountName = accountName; 34          this.mAccountType = accountType; 35          this.mDataSet = dataSet; 36          this.mGroupId = groupId; 37          this.mTitle = title; 38          this.mDefaultGroup = defaultGroup; 39          this.mFavorites = favorites; 40      } 41 42      public String getAccountName() { 43          return mAccountName; 44      } 45 46      public String getAccountType() { 47          return mAccountType; 48      } 49 50      public String getDataSet() { 51          return mDataSet; 52      } 53 54      public long getGroupId() { 55          return mGroupId; 56      } 57 58      public String getTitle() { 59          return mTitle; 60      } 61 62      public boolean isDefaultGroup() { 63          return mDefaultGroup; 64      } 65 66      public boolean isFavorites() { 67          return mFavorites; </pre> <p><a href="https://android.googlesource.com/platform/packages/apps/ContactsCommon/+/nougat-mr1-release/src/com/android/contacts/common/GroupMetaData.java">https://android.googlesource.com/platform/packages/apps/ContactsCommon/+/nougat-mr1-release/src/com/android/contacts/common/GroupMetaData.java</a></p>

## Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products

US9408055B2	HTC
	<p data-bbox="541 277 1297 370"><b>Send &amp; receive text messages in Android Messages</b></p> <p data-bbox="541 386 1178 407">You can send and receive text messages with friends and contacts on Android Messages.</p> <p data-bbox="531 423 846 456"><b>Start a conversation</b></p> <ol data-bbox="541 480 1528 634" style="list-style-type: none"><li>1. Open the Android Messages app .</li><li>2. Tap Compose .</li><li>3. In "To," enter the names, phone numbers, or email addresses that you'd like to message. You can also pick from your top contacts or your whole contact list.</li><li>4. Tap Next .</li></ol> <p data-bbox="516 659 1472 691"><a href="https://support.google.com/nexus/answer/6080324?hl=en&amp;ref_topic=6080329">https://support.google.com/nexus/answer/6080324?hl=en&amp;ref_topic=6080329</a></p> <p data-bbox="541 813 894 854"><b>See your contacts</b></p> <ol data-bbox="541 878 968 951" style="list-style-type: none"><li>1. Open your device's Contacts app .</li><li>2. Tap Menu .</li></ol> <ul data-bbox="541 984 1738 1179" style="list-style-type: none"><li>• <b>See contacts by label:</b> Choose a label from the list.</li><li>• <b>See contacts for another account:</b> Tap Down arrow  &gt; pick an account.</li><li>• <b>See the contacts for all your accounts:</b> Choose <b>All contacts</b>. <b>Tip:</b> If you have multiple contacts with the same information, the information will be grouped into one contact.</li><li>• <b>See your Google Account contacts on the web:</b> Go to <a href="#">Google Contacts</a> .</li></ul> <p data-bbox="516 1243 1535 1276"><a href="https://support.google.com/android/answer/6118731?hl=en&amp;ref_topic=6118711">https://support.google.com/android/answer/6118731?hl=en&amp;ref_topic=6118711</a></p>

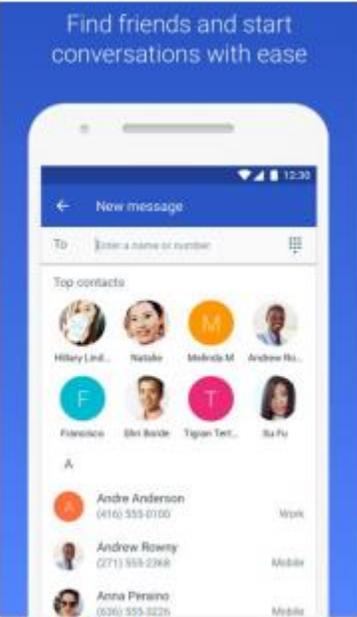
## Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products

US9408055B2	HTC
	<p data-bbox="541 240 894 277"><b>Label your contacts</b></p> <p data-bbox="541 305 982 326">You can group contacts together using labels.</p> <ol data-bbox="552 358 930 461" style="list-style-type: none"><li data-bbox="552 358 930 386">1. Open your device's Contacts app .</li><li data-bbox="552 399 863 420">2. Tap Menu  &gt; <b>Create label</b>.</li><li data-bbox="552 433 869 461">3. Enter a label name and tap <b>Ok</b>.</li></ol> <ul data-bbox="552 493 1717 553" style="list-style-type: none"><li data-bbox="552 493 1234 514">• <b>Add one contact to a label:</b> Tap Add contact  &gt; choose a contact.</li><li data-bbox="552 527 1717 553">• <b>Add multiple contacts to a label:</b> Tap Add contact  &gt; touch and hold a contact &gt; tap the other contacts &gt; tap <b>Add</b>.</li></ul> <p data-bbox="510 602 1535 630"><a href="https://support.google.com/android/answer/6118731?hl=en&amp;ref_topic=6118711">https://support.google.com/android/answer/6118731?hl=en&amp;ref_topic=6118711</a></p> <p data-bbox="552 751 940 789"><b>Share your contacts</b></p> <ol data-bbox="562 821 1045 979" style="list-style-type: none"><li data-bbox="562 821 978 849">1. Open your device's Contacts app .</li><li data-bbox="562 862 842 889">2. Tap a contact in the list.</li><li data-bbox="562 902 831 930">3. Tap More  &gt; <b>Share</b>.</li><li data-bbox="562 943 1045 979">4. Choose how you want to share the contact.</li></ol> <p data-bbox="510 1032 1535 1060"><a href="https://support.google.com/android/answer/6118731?hl=en&amp;ref_topic=6118711">https://support.google.com/android/answer/6118731?hl=en&amp;ref_topic=6118711</a></p>

**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC
	<div data-bbox="533 233 1041 272"><h3>Start a Hangout</h3></div> <div data-bbox="533 289 1041 308"><p>You can send and receive messages with one person or multiple people.</p></div> <div data-bbox="548 370 879 386"><p>COMPUTER <b>ANDROID</b> IPHONE &amp; IPAD</p></div> <div data-bbox="533 513 800 542"><h3>Start a conversation</h3></div> <div data-bbox="541 561 1121 698"> <ol style="list-style-type: none"> <li>1. On your Android phone or tablet, open the Hangouts app.</li> <li>2. At the bottom right, tap Add &gt; New Conversation.</li> <li>3. Type and select a person's name.</li> <li>4. Enter your message. You can also add emojis, photos, your location, or a sticker.</li> <li>5. Tap Send.</li> </ol> </div> <div data-bbox="518 711 1751 769"> <p><a href="https://support.google.com/hangouts/answer/3115553?hl=en&amp;ref_topic=6386410&amp;co=GENIE.Platform%3DAndroid&amp;oco=0">https://support.google.com/hangouts/answer/3115553?hl=en&amp;ref_topic=6386410&amp;co=GENIE.Platform%3DAndroid&amp;oco=0</a></p> </div> <div data-bbox="541 815 879 854"><h3>Contact someone</h3></div> <div data-bbox="541 883 1148 906"><p>You can call, email, or send text messages to your contacts.</p></div> <div data-bbox="548 938 953 1175"> <ol style="list-style-type: none"> <li>1. Open your device's Contacts app.</li> <li>2. Tap a contact in the list.</li> <li>3. Choose an option: <ul style="list-style-type: none"> <li>• Call</li> <li>• Email</li> <li>• New message</li> </ul> </li> </ol> </div> <div data-bbox="518 1195 1402 1224"> <p><a href="https://support.google.com/nexus/answer/6118731?hl=en&amp;ref_topic=6118711">https://support.google.com/nexus/answer/6118731?hl=en&amp;ref_topic=6118711</a></p> </div> <div data-bbox="1129 233 1398 711"> </div>

**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC
	<p data-bbox="527 688 762 712"><b>Start a conversation</b></p> <ol data-bbox="527 732 1272 862" style="list-style-type: none"><li>1. Open the Android Messages app</li><li>2. Tap Compose</li><li>3. In "To," enter the names, phone numbers, or email addresses that you'd like to message. You can also pick from your top contacts or your whole contact list.</li><li>4. Tap Next</li></ol> <p data-bbox="516 878 1612 943"><a href="https://support.google.com/nexus/answer/6080324?hl=en&amp;ref_topic=6080329">https://support.google.com/nexus/answer/6080324?hl=en&amp;ref_topic=6080329</a> <a href="https://play.google.com/store/apps/details?id=com.google.android.apps.messaging&amp;hl=en">https://play.google.com/store/apps/details?id=com.google.android.apps.messaging&amp;hl=en</a></p> 

**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

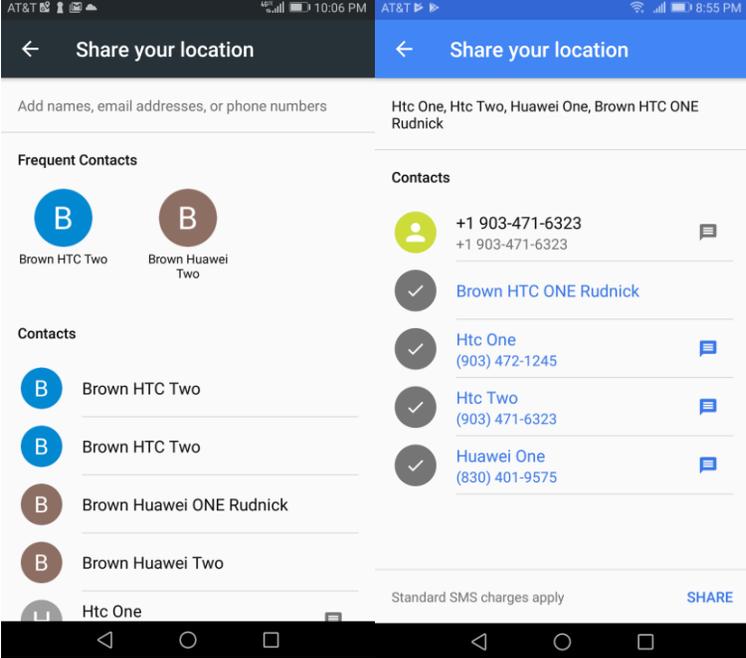
US9408055B2	HTC
	<div data-bbox="533 618 840 649" data-label="Section-Header"><p><b>Start a conversation</b></p></div> <div data-bbox="533 672 1218 829" data-label="List-Group"> <ol style="list-style-type: none"> <li>1. On your Android phone or tablet, open the Hangouts app .</li> <li>2. At the bottom right, tap Add  &gt; New Conversation .</li> <li>3. Type and select a person's name.</li> <li>4. Enter your message. You can also add emojis, photos, your location, or a sticker.</li> <li>5. Tap Send .</li> </ol> </div> <div data-bbox="520 862 1751 919" data-label="Text"><p><a href="https://support.google.com/hangouts/answer/3115553?hl=en&amp;ref_topic=6386410&amp;co=GENIE.Platform%3DAndroid&amp;oco=0">https://support.google.com/hangouts/answer/3115553?hl=en&amp;ref_topic=6386410&amp;co=GENIE.Platform%3DAndroid&amp;oco=0</a></p></div> <div data-bbox="533 979 1354 1136" data-label="List-Group"> <ol style="list-style-type: none"> <li>1. Open the Hangouts app .</li> <li>2. At the bottom, tap Add  &gt; <b>New conversation</b> &gt; <b>New group</b>.</li> <li>3. Enter and select the names, phone numbers, or email addresses of people in your group.</li> <li>4. Tap Done .</li> </ol> </div> <div data-bbox="520 1149 1751 1206" data-label="Text"><p><a href="https://support.google.com/hangouts/answer/3111943?hl=en&amp;ref_topic=6386410&amp;co=GENIE.Platform%3DAndroid&amp;oco=1">https://support.google.com/hangouts/answer/3111943?hl=en&amp;ref_topic=6386410&amp;co=GENIE.Platform%3DAndroid&amp;oco=1</a></p></div> <div data-bbox="1234 228 1591 857" data-label="Image"> </div>



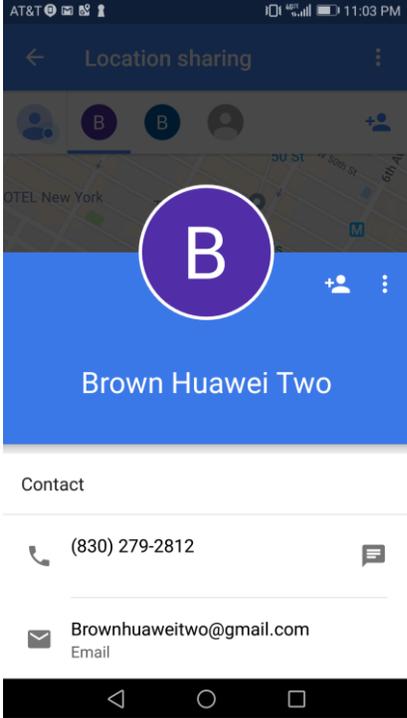
## Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products

US9408055B2	HTC
	<p data-bbox="541 233 905 272"><b>Contact someone</b></p> <p data-bbox="541 305 1192 332">You can call, email, or send text messages to your contacts.</p> <ol data-bbox="554 365 982 625" style="list-style-type: none"><li data-bbox="554 365 982 397">1. Open your device's Contacts app .</li><li data-bbox="554 409 842 441">2. Tap a contact in the list.</li><li data-bbox="554 456 781 488">3. Choose an option:<ul data-bbox="583 505 808 625" style="list-style-type: none"><li data-bbox="583 505 688 537">• Call </li><li data-bbox="583 548 709 581">• Email </li><li data-bbox="583 592 808 625">• New message </li></ul></li></ol> <p data-bbox="520 641 1472 673"><a href="https://support.google.com/nexus/answer/6118731?hl=en&amp;ref_topic=6118711">https://support.google.com/nexus/answer/6118731?hl=en&amp;ref_topic=6118711</a></p> <p data-bbox="512 711 1031 743"><b><u>Exemplary Google Maps Screenshots:</u></b></p>

**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC
	 <p>For example, the Accused Products include software that obtains contact information including the phone numbers . Furthermore, these phone calls can merge multiple parties into a conference call.</p>

**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC
	
<p>[1B] facilitating initiation of Internet Protocol (IP) based communication between the first device and the respective second devices by using respective telephone numbers to send, from the first device to the</p>	<p>HTC infringes directly and/or indirectly by performing, inducing others to perform, and/or contributing to the performance of: facilitating initiation of Internet Protocol (IP) based communication between the first device and the respective second devices by using respective telephone numbers to send, from the first device to the second devices, respective Short Message Service (SMS) messages including a telephone number of the first device and information usable by the respective second device to send IP-based communication to the first device. See also, e.g., 1[P]-[1A], which are incorporated herein by reference in their entirety.</p> <p>For example, the Accused Products utilize SMS-based messages to initiate IP communication between participants of Maps location sharing. For example, both Android Messages and Hangouts, in conjunction with Maps, utilize SMS messages, including group messages from one device to several devices, to send an</p>

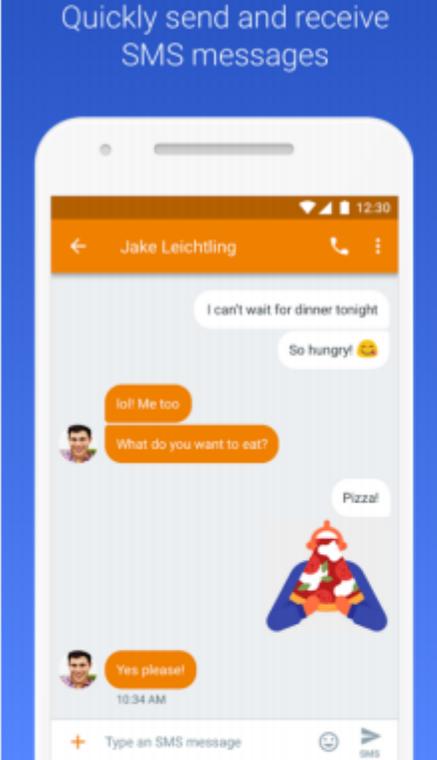
**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC
<p>second devices, respective Short Message Service (SMS) messages including a telephone number of the first device and information usable by the respective second device to send IP-based communication to the first device;</p>	<p>SMS message, with additional information, to a contact.  D2 Technologies Showcases its mCUE IP Communications Interface over WiMAX on HTC's EVO 4G Android Smartphone</p> <p><small>Amsterdam, Netherlands (WiMAX Forum Global Congress) and Santa Barbara, CA – June 14, 2010</small> –D2 Technologies, the market leader in embedded IP communications software platforms, today announced that it is holding private demonstrations of its mCUE® converged communications client for mobile devices and handsets on the HTC EVO™ 4G smartphone on Thursday, June 17 at the WiMAX Forum® Global Congress in Amsterdam. D2's mCUE on the HTC EVO, the first 4G phone to be introduced in the United States, was configured and installed in less than a week – clearly illustrating how OEMs and ODMs can more rapidly develop Android™-based devices by choosing to incorporate the converged presence-based communications user interface (CUI).</p> <p><a href="http://www.d2tech.com/press-releases-year.html?Y=2010">http://www.d2tech.com/press-releases-year.html?Y=2010</a></p>

**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC
	<p data-bbox="527 232 1289 302">Siemens Enterprise Communications and HTC Simplify Device Choice for Mobile UC   </p> <p data-bbox="527 329 1033 350">Reston, VA and Frankfurt, Germany, Feb 19, 2013</p> <p data-bbox="527 380 1451 464"><b>Enterasys' Mobile IAM and MDM connect™ BYOD solutions now support AirWatch to onboard and manage mobile devices and applications</b></p> <p data-bbox="527 480 1514 643">In an ongoing effort to support today's increasingly mobile workforce, Siemens Enterprise Communications and HTC Corporation today announced a strategic global partnership to make it easier for enterprises to embrace mobile unified communications (UC) on HTC enterprise-enabled devices. This partnership makes it even simpler for enterprises to embrace a BYOD strategy for mobile UC, since HTC's popular consumer Android smartphones now fully support Siemens Enterprise Communications' OpenScape Mobile and OpenScape Web Collaboration solutions.</p> <p data-bbox="527 693 625 714"><b>Key Facts</b></p> <ul data-bbox="527 730 1514 959" style="list-style-type: none"> <li>▪ Siemens Enterprise Communications OpenScape Mobile and OpenScape Web Collaboration solutions will be validated on select HTC devices to increase users' confidence that their chosen device will work seamlessly with their mobile communication tools</li> <li>▪ Siemens Enterprise Communications customers will have a simplified process to secure validated HTCPro devices supporting Siemens Enterprise Communications solutions</li> <li>▪ Siemens Enterprise Communications and HTC will collaborate to simplify deployment of mobile UC through joint marketing and fulfillment efforts</li> <li>▪ This collaboration will take place through HTCPro, a program that provides mobile solutions for companies and their employees and ensures that HTC's entire portfolio is business-ready</li> </ul> <p data-bbox="527 1000 1514 1068">Siemens Enterprise Communications OpenScape Mobile and OpenScape Web Collaboration solutions will be validated on select HTC devices to increase users' confidence that their chosen device will work seamlessly with their mobile communication tools.</p> <p data-bbox="512 1083 1713 1114"><a href="http://www.unify.com/us/news/E853B94A-F94F-4ADA-98DA-8C80BB965953/?isarchive=1">http://www.unify.com/us/news/E853B94A-F94F-4ADA-98DA-8C80BB965953/?isarchive=1</a></p> <p data-bbox="541 1144 1514 1245">Android Messages makes it easy to communicate with anyone by using SMS, MMS, and more. Stay in touch with friends and family, send group texts, and share your favorite pictures, videos, audio messages.</p> <p data-bbox="533 1274 1577 1343">• <b>Enhanced features:</b> On supported carriers, you can send messages over Wi-Fi or your data network, see when friends have read your message, and more.</p> <p data-bbox="527 1360 1619 1391"><a href="https://play.google.com/store/apps/details?id=com.google.android.apps.messaging&amp;hl=en">https://play.google.com/store/apps/details?id=com.google.android.apps.messaging&amp;hl=en</a></p>

## Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products

US9408055B2	HTC
	 <p data-bbox="520 998 1617 1031"><a href="https://play.google.com/store/apps/details?id=com.google.android.apps.messaging&amp;hl=en">https://play.google.com/store/apps/details?id=com.google.android.apps.messaging&amp;hl=en</a></p> <h3 data-bbox="541 1088 1087 1133">Get started with Hangouts</h3> <p data-bbox="541 1149 751 1172">You can use Hangouts to:</p> <ul data-bbox="541 1193 1134 1282" style="list-style-type: none"><li>• Start a chat conversation or video call.</li><li>• Make phone calls using Wi-Fi or data.</li><li>• Send text messages with your <a href="#">Google Voice</a> or <a href="#">Project Fi</a> phone number.</li></ul> <p data-bbox="541 1307 1465 1356">Hangouts sync automatically across devices. If you start a Hangout on your computer, you can continue your chat on another device, like your phone.</p> <p data-bbox="520 1364 1512 1396"><a href="https://support.google.com/hangouts/answer/2944865?hl=en&amp;ref_topic=6386410">https://support.google.com/hangouts/answer/2944865?hl=en&amp;ref_topic=6386410</a></p>

## Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products

US9408055B2	HTC
	<p data-bbox="541 266 835 305"><b>Start a Hangout</b></p> <p data-bbox="541 323 1054 341">You can send and receive messages with one person or multiple people.</p> <p data-bbox="554 404 888 420">COMPUTER <b>ANDROID</b> IPHONE &amp; IPAD</p> <hr/> <p data-bbox="541 553 806 581"><b>Start a conversation</b></p> <ol data-bbox="548 602 1136 737" style="list-style-type: none"><li>1. On your Android phone or tablet, open the Hangouts app .</li><li>2. At the bottom right, tap Add  &gt; New Conversation .</li><li>3. Type and select a person's name.</li><li>4. Enter your message. You can also add emojis, photos, your location, or a sticker.</li><li>5. Tap Send .</li></ol> <p data-bbox="525 753 1745 810"><a href="https://support.google.com/hangouts/answer/3115553?hl=en&amp;ref_topic=6386410&amp;co=GENIE.Platform%3DAndroid&amp;oco=0">https://support.google.com/hangouts/answer/3115553?hl=en&amp;ref_topic=6386410&amp;co=GENIE.Platform%3DAndroid&amp;oco=0</a></p> <ul data-bbox="560 862 1562 1013" style="list-style-type: none"><li>• Connect your Google Voice account for phone calling, SMS texting, and voicemail integration.</li><li>• Keep in touch with contacts across Android, iOS, and the web, and sync chats across all your devices.</li><li>• Message contacts anytime, even if they're offline.</li></ul> <p data-bbox="525 1045 1398 1073"><a href="https://play.google.com/store/apps/details?id=com.google.android.talk&amp;hl=en">https://play.google.com/store/apps/details?id=com.google.android.talk&amp;hl=en</a></p> <ol data-bbox="548 1084 1493 1344" style="list-style-type: none"><li>1. Open the Hangouts app .</li><li>2. At the bottom right, tap Add .</li><li>3. Choose <b>New SMS</b>.</li><li>4. Type the name or phone number. If you're traveling, use the "+" sign and country code when texting.</li><li>5. Tap the number or contact.</li><li>6. Tap Continue .</li><li>7. Type your message and tap Send .</li></ol> <p data-bbox="525 1360 1257 1388"><a href="https://support.google.com/hangouts/answer/3441321?hl=en">https://support.google.com/hangouts/answer/3441321?hl=en</a></p>

**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC
	<p data-bbox="512 235 911 264"><b><u>Google Maps Share Location</u></b></p> <p data-bbox="512 306 1902 760">Google Latitude, Google Plus, Google Hangouts, Google Messages, and Android Messenger, Google Duo, Google Allo, and Google Chrome, the Accused Products require a user to join the corresponding network by signing-in to the device with an identifier (e.g., Google Account). The sign-in process takes place within the Google Maps software on the Accused Product or by navigating to maps.google.com within the Google Chrome browser on the Accused Product. Alternatively, the sign-in process may partially or completely take place using credentials already provided when the user associates a Google Account with the Accused Product, e.g., during initial setup of the Accused Product. Subject to discovery, one or more additional or substitute identifiers may correspond to the group. The sign-in process involves a user entering its Google Account and additional authentication data on the interface of the Accused Product and sending a message containing the Google Account and additional authentication data over a network to members of a group. The group comprises the multiple identifiers, individuals, profiles, and/or devices associated with the group.</p> <p data-bbox="512 797 1902 1081">Further regarding Google Maps, Google Latitude, Google Plus, Google Hangouts, Google Messages, and Android Messenger, Google Duo, Google Allo, and Google Chrome, the Accused Products alternatively require a user to send a request containing the identifier (e.g., Google Account). Subject to discovery, additional identifiers may be assigned or used to correspond to the group. The request may be an invitation or message that associates a Google Account with one or more Google Accounts for the purposes of sharing locations within the group. The group comprises the multiple identifiers, individuals, profiles, and/or devices associated with the group</p> <p data-bbox="512 1118 1031 1148"><b><u>Exemplary Support for Google Maps:</u></b></p>



## Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products

US9408055B2	HTC
	<p data-bbox="541 240 903 259">COMPUTER <b>ANDROID</b> IPHONE &amp; IPAD</p> <hr data-bbox="541 279 1417 282"/> <h3 data-bbox="531 329 949 362">If they have a Google Account</h3> <ol data-bbox="541 380 1287 630" style="list-style-type: none"><li>1. If you haven't already, add their Gmail address to your <a href="#">Google Contacts</a>.</li><li>2. On your Android phone or tablet, open the Google Maps app  and sign in. <a href="#">Learn how to sign in.</a></li><li>3. Tap Menu ≡ &gt; <b>Location sharing</b> &gt; Add People .</li><li>4. Choose how long you want to share your location.</li><li>5. Tap <b>Select People</b>.<ul data-bbox="562 537 1047 557" style="list-style-type: none"><li>• If you're asked about your contacts, give Google Maps access.</li></ul></li><li>6. Choose who you want to share with.</li><li>7. Tap <b>Share</b>.</li></ol> <h3 data-bbox="531 680 1024 712">If they don't have a Google Account</h3> <ol data-bbox="541 730 1409 837" style="list-style-type: none"><li>1. On your Android phone or tablet, open the Google Maps app  and sign in. <a href="#">Learn how to sign in.</a></li><li>2. Tap Menu ≡ &gt; <b>Location sharing</b> &gt; Add People .</li><li>3. Tap More ... &gt; <b>Copy to clipboard</b>. People with this link can see your location for as long as you choose, up to 72 hours.</li></ol> <h3 data-bbox="531 878 819 911">Share using another app</h3> <p data-bbox="531 924 1104 943">You can also share through messaging apps. Tap More ... &gt; select an app.</p> <h3 data-bbox="531 993 709 1026">Stop sharing</h3> <ol data-bbox="541 1044 1104 1128" style="list-style-type: none"><li>1. Open the Google Maps app .</li><li>2. Tap Menu ≡ &gt; <b>Location sharing</b>.</li><li>3. Next to the person with whom you want to stop sharing, tap Remove .</li></ol> <p data-bbox="512 1175 1703 1208"><a href="https://support.google.com/maps/answer/7326816?co=GENIE.Platform%3DAndroid&amp;oco=1">https://support.google.com/maps/answer/7326816?co=GENIE.Platform%3DAndroid&amp;oco=1</a></p>

## Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products

US9408055B2	HTC
	<h3>Share your E.T.A</h3> <p>After you start your drive, you can share your destination, estimated arrival time, and where you are on the route.</p> <ol style="list-style-type: none"><li>1. Open the Google Maps app .</li><li>2. Set a driving destination. <a href="#">Learn how to navigate to a place.</a></li><li>3. After you start navigation, tap More ^ &gt; <b>Share trip progress.</b></li><li>4. Choose a person from the list.</li><li>5. Tap <b>Share.</b></li><li>6. Location Sharing will stop when you reach your destination or stop navigating.</li></ol> <ul style="list-style-type: none"><li>• To stop sharing before you arrive, tap More ^ &gt; <b>Stop sharing.</b></li></ul> <h3>See where someone is</h3> <p>If someone shares their location with you, you can see them on the map.</p> <ol style="list-style-type: none"><li>1. Open the Google Maps app .</li><li>2. Tap Menu ≡ &gt; <b>Location sharing.</b></li><li>3. Choose someone.</li></ol> <ul style="list-style-type: none"><li>• To see an updated location, tap on a friend's icon &gt; More ⋮ &gt; <b>Refresh.</b></li></ul> <h3>Stop seeing someone's location</h3> <ol style="list-style-type: none"><li>1. Open the Google Maps app .</li><li>2. On the map, tap their icon.</li><li>3. At the bottom, tap More ^ .</li><li>4. To temporarily hide someone, tap <b>Hide from map.</b> You can stop hiding them at any time.</li></ol> <p><b>Note:</b> You can stop someone's location from ever appearing on your map. <a href="#">Learn how to block another person's account.</a></p> <p><a href="https://support.google.com/maps/answer/7326816?co=GENIE.Platform%3DAndroid&amp;oco=1">https://support.google.com/maps/answer/7326816?co=GENIE.Platform%3DAndroid&amp;oco=1</a></p>

## Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products

US9408055B2	HTC
	<p data-bbox="520 240 997 289"><b>Create a list of places</b></p> <p data-bbox="520 305 1339 329">In Google Maps, you can create a list of places, like your favorite places or places you want to visit.</p> <p data-bbox="535 402 930 423">COMPUTER <b>ANDROID</b> IPHONE &amp; IPAD</p> <hr data-bbox="520 446 1369 451"/> <p data-bbox="520 500 766 532"><b>Make a new list</b></p> <ol data-bbox="531 557 1083 716" style="list-style-type: none"><li>1. On your Android phone or tablet, open the Google Maps app .</li><li>2. Tap Menu  &gt; <b>Your places</b> &gt; <b>Saved</b>.</li><li>3. In the bottom right, tap Add .</li><li>4. Enter a name and description.</li><li>5. Tap <b>Save</b>.</li></ol> <p data-bbox="520 773 842 805"><b>Save a place to a list</b></p> <ol data-bbox="531 829 978 989" style="list-style-type: none"><li>1. Open the Google Maps app .</li><li>2. Search for a place or tap it on the map.</li><li>3. At the bottom, tap the place's name or address.</li><li>4. Tap <b>Save</b>.</li><li>5. Choose a list. To create a new list, tap <b>New list</b> .</li></ol> <p data-bbox="520 1045 730 1078"><b>See your lists</b></p> <ol data-bbox="531 1102 879 1154" style="list-style-type: none"><li>1. Open the Google Maps app .</li><li>2. Tap Menu  &gt; <b>Your places</b> &gt; <b>Saved</b>.</li></ol> <p data-bbox="510 1198 1898 1268"><a href="https://support.google.com/maps/answer/7280933?hl=en&amp;ref_topic=7301134&amp;co=GENIE.Platform%3DAndroid&amp;oco=1">https://support.google.com/maps/answer/7280933?hl=en&amp;ref_topic=7301134&amp;co=GENIE.Platform%3DAndroid&amp;oco=1</a></p>

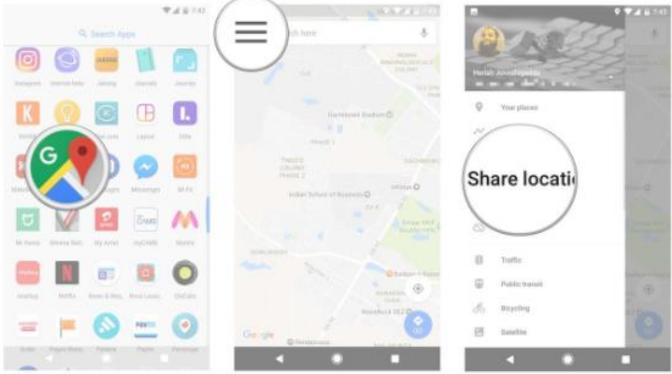
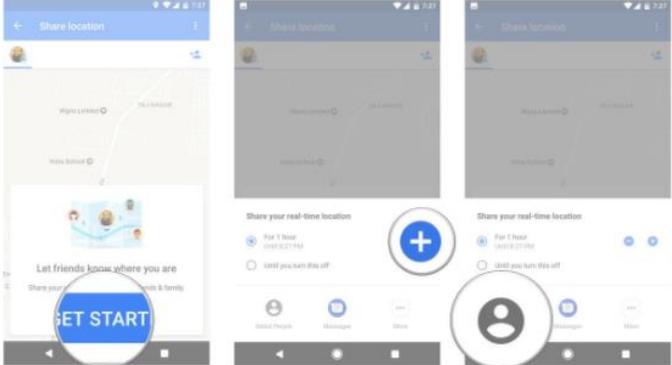
## Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products

US9408055B2	HTC
	<h3 data-bbox="541 245 877 280">Hide or share lists</h3> <p data-bbox="541 310 909 334"><b>Note:</b> You can't share starred places.</p> <ol data-bbox="554 367 1251 472" style="list-style-type: none"><li data-bbox="554 367 890 391">1. Open the Google Maps app .</li><li data-bbox="554 407 968 431">2. Tap Menu  &gt; <b>Your places</b> &gt; <b>Saved</b>.</li><li data-bbox="554 448 1251 472">3. Next to the list you want to share, tap More  &gt; choose an option:</li></ol> <ul data-bbox="583 488 1682 630" style="list-style-type: none"><li data-bbox="583 488 1440 513">• <b>Hide/Show on your map:</b> Display or hide your saved places when looking at the map.</li><li data-bbox="583 529 1058 553">• <b>Share list:</b> Allow others to see your saved list.</li><li data-bbox="583 570 1682 630">• <b>Sharing options:</b> You can make your list public, private, or shared. To let anyone with the link see your list, tap <b>Shared</b>. To let anyone find and follow your list, tap <b>Public</b>.</li></ul> <h3 data-bbox="541 699 768 735">Follow a list</h3> <p data-bbox="541 764 1728 821">If you follow a list made by someone else, their saved places will show up in Your Places. The places will also appear as suggested locations in Google Maps.</p> <h3 data-bbox="541 878 915 914">Follow a list using a link</h3> <ol data-bbox="554 935 1356 1040" style="list-style-type: none"><li data-bbox="554 935 957 959">1. Tap on the link you received to open it.</li><li data-bbox="554 976 1272 1000">2. Tap <b>Follow</b>. This list will now be added to the group of lists you follow.</li><li data-bbox="554 1016 1356 1040">3. <b>Optional:</b> To unfollow a list someone shared with you, tap the list &gt; <b>Following</b>.</li></ol> <h3 data-bbox="541 1097 926 1133">See lists made by others</h3> <p data-bbox="541 1154 1335 1179">If a user has any Google Maps lists that were made public, you can follow them.</p> <ol data-bbox="554 1211 1136 1317" style="list-style-type: none"><li data-bbox="554 1211 1136 1235">1. Tap on the name of a user whose list you want to follow.</li><li data-bbox="554 1252 674 1276">2. Tap <b>Lists</b>.</li><li data-bbox="554 1292 1136 1317">3. Tap on the list you want to follow &gt; More  &gt; <b>Follow</b>.</li></ol> <p data-bbox="512 1365 1902 1398"><a href="https://support.google.com/maps/answer/7280933?hl=en&amp;ref_topic=7301134&amp;co=GENIE.Platform%3DAn">https://support.google.com/maps/answer/7280933?hl=en&amp;ref_topic=7301134&amp;co=GENIE.Platform%3DAn</a></p>

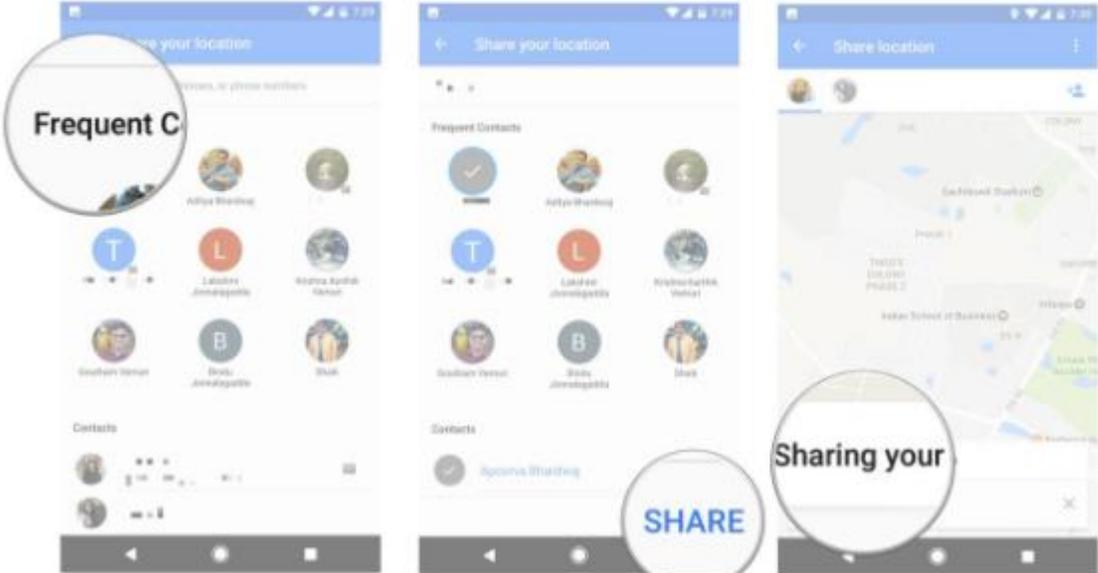
**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

<b>US9408055B2</b>	<b>HTC</b>
	<a href="#">droid&amp;oco=1</a>

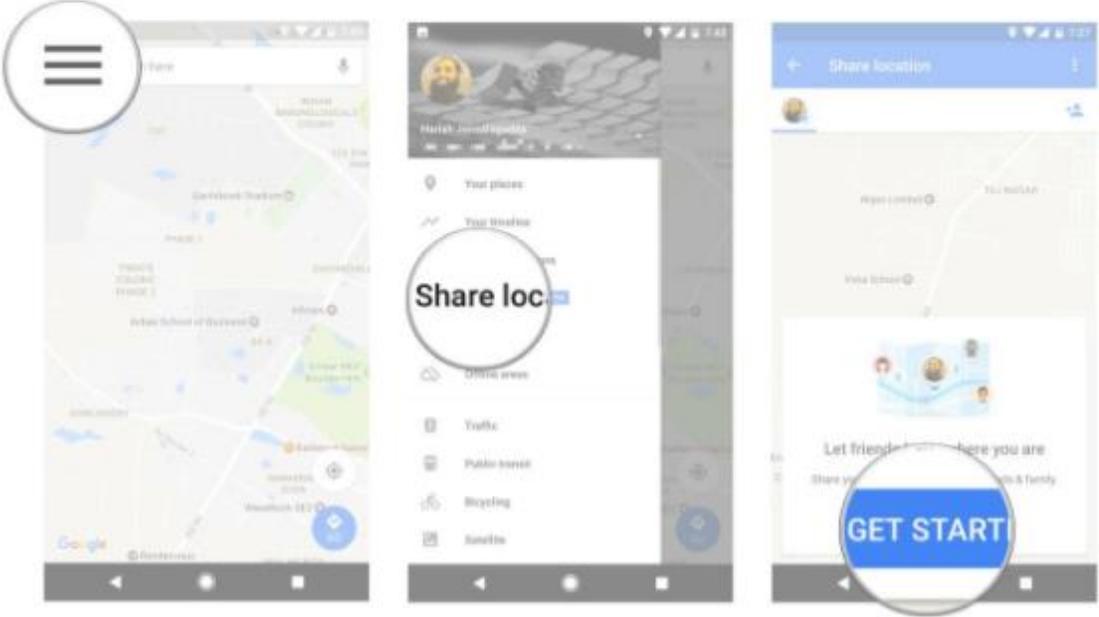
## Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products

US9408055B2	HTC
	<h3 data-bbox="520 240 1150 272">How to share your location in Google Maps</h3> <ol data-bbox="520 305 1150 389" style="list-style-type: none"><li>1. Open Google Maps from the app drawer or the home screen.</li><li>2. Tap the hamburger menu (the three horizontal lines) on the top left corner of the screen.</li><li>3. Select Share location.</li></ol>  <ol data-bbox="520 828 1150 933" style="list-style-type: none"><li>4. Tap Get Started.</li><li>5. Use the + icon to select a time period or select the Until you turn this off setting to share your location indefinitely.</li><li>6. Tap Select People.</li></ol> 

### Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products

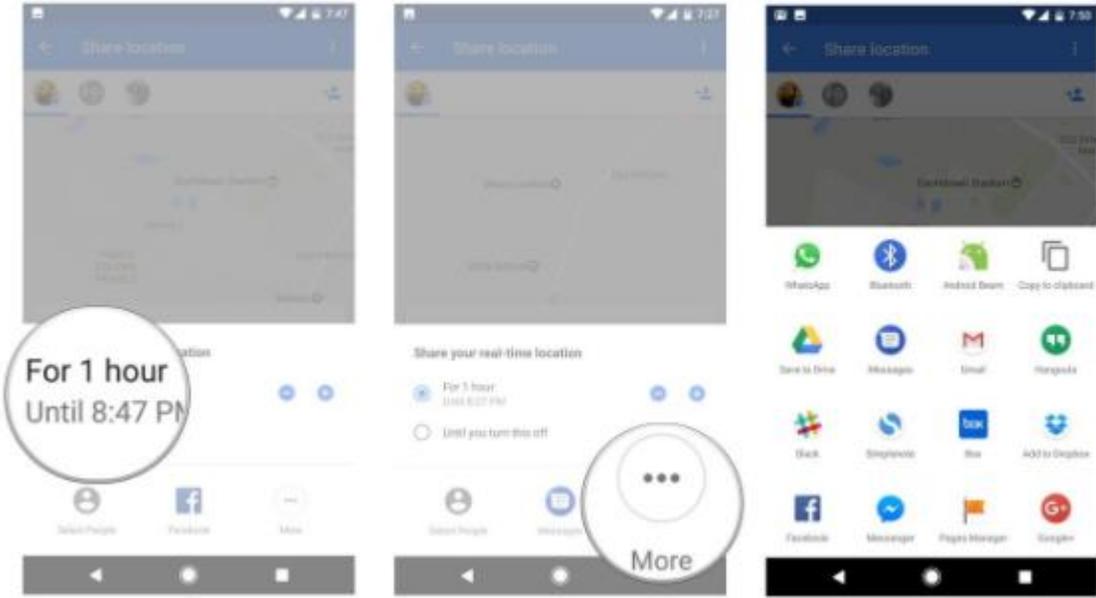
US9408055B2	HTC
	<p data-bbox="510 233 1356 264"><a href="https://www.androidcentral.com/how-share-location-google-maps">https://www.androidcentral.com/how-share-location-google-maps</a></p> <ol data-bbox="527 321 1577 493" style="list-style-type: none"><li>7. You'll see a list of your frequent contacts at the top, along with a full list of contacts. Pick the contacts by tapping their name.</li><li>8. Once you've selected the contacts you want to share your location to, tap <b>Share</b>.</li><li>9. You'll see a message saying that the selected contact can view your location.</li></ol>  <p data-bbox="510 1166 1356 1196"><a href="https://www.androidcentral.com/how-share-location-google-maps">https://www.androidcentral.com/how-share-location-google-maps</a></p>

## Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products

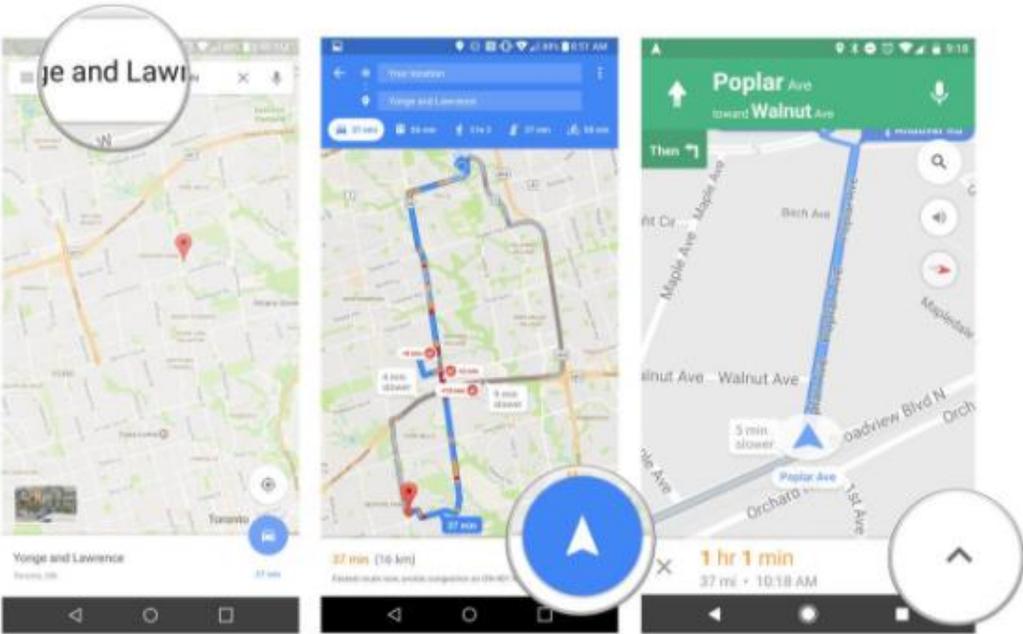
US9408055B2	HTC
	<h3 data-bbox="520 245 1255 293">How to create a shareable link</h3> <p data-bbox="520 334 1461 362">You can also create a link and use it to share your location easily. Here's how to do it:</p> <ol data-bbox="520 412 1234 553" style="list-style-type: none"><li data-bbox="520 412 1234 440">1. Tap the hamburger menu on the top left corner of the screen.</li><li data-bbox="520 467 800 495">2. Select Share location.</li><li data-bbox="520 522 737 550">3. Tap Get Started.</li></ol>  <p data-bbox="506 1263 1356 1295"><a href="https://www.androidcentral.com/how-share-location-google-maps">https://www.androidcentral.com/how-share-location-google-maps</a></p>



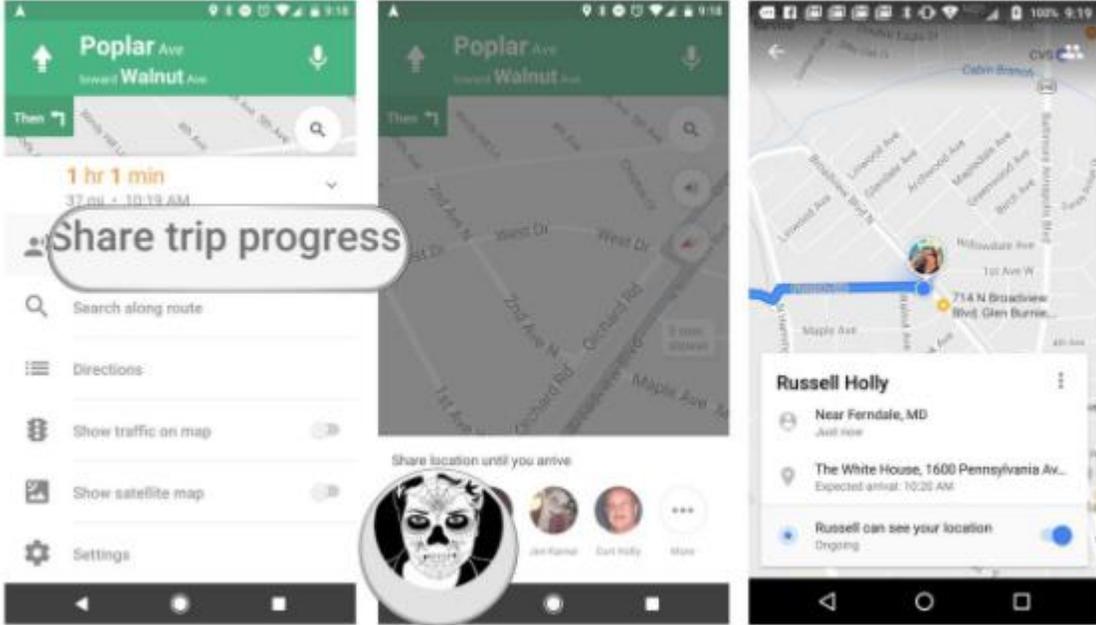
## Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products

US9408055B2	HTC
	<p data-bbox="520 245 1213 272">4. Select the <b>amount of time</b> you want to share your location.</p> <p data-bbox="520 302 680 329">5. Tap <b>More</b>.</p> <p data-bbox="520 358 1633 423">6. Select <b>your app of choice</b> to create and send a <b>unique URL</b> that broadcasts your current location. You can email it, send the link via Messenger, or even tweet it to the intended recipient.</p> <div data-bbox="569 459 1665 1057"></div> <p data-bbox="506 1117 1360 1149"><a href="https://www.androidcentral.com/how-share-location-google-maps">https://www.androidcentral.com/how-share-location-google-maps</a></p>

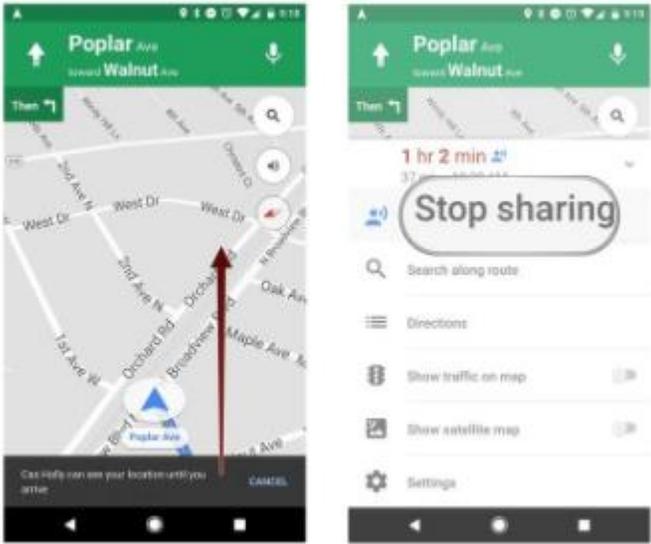
**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC
	<p data-bbox="527 240 1428 337"><b>How to share your navigation directions while you walk, drive or transit</b></p> <p data-bbox="527 375 1556 464">One of the best ways to share your location with someone is during a drive, walk or transit. If you're meeting a friend or family member somewhere, or navigating towards their home, Google Maps lets you share your location with them for the duration of the trip. It's magic!</p> <ol data-bbox="527 513 1396 643" style="list-style-type: none"> <li data-bbox="527 513 976 537">1. In the <b>search bar</b> enter your destination.</li> <li data-bbox="527 561 1396 586">2. Pick your navigation type (drive, transit, walk) and press the blue <b>navigate</b> button.</li> <li data-bbox="527 610 1396 634">3. Tap the arrow next to the <b>time-to-destination</b> number at the bottom of the screen.</li> </ol>  <p data-bbox="512 1360 1356 1393"><a href="https://www.androidcentral.com/how-share-location-google-maps">https://www.androidcentral.com/how-share-location-google-maps</a></p>

## Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products

US9408055B2	HTC
	<p data-bbox="527 378 835 406">4. Tap Share trip progress.</p> <p data-bbox="527 435 1150 462">5. Choose one or more contacts to share trip progress.</p>  <p data-bbox="537 1166 1339 1193">You can also stop sharing your location with someone before a trip ends.</p> <p data-bbox="512 1235 1356 1263"><a href="https://www.androidcentral.com/how-share-location-google-maps">https://www.androidcentral.com/how-share-location-google-maps</a></p>

## Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products

US9408055B2	HTC
	<p data-bbox="533 310 1470 396"><b>1.</b> Tap the arrow next to the <b>time-to-destination</b> number at the bottom of the screen. <b>2.</b> Tap <b>Stop sharing</b>.</p> <div data-bbox="747 448 1398 992"></div> <p data-bbox="541 1044 638 1068">That's it!</p> <p data-bbox="541 1112 1612 1136">Are you excited that location sharing is back in Google Maps? How often do you use the feature?</p> <p data-bbox="512 1187 1356 1211"><a href="https://www.androidcentral.com/how-share-location-google-maps">https://www.androidcentral.com/how-share-location-google-maps</a></p>

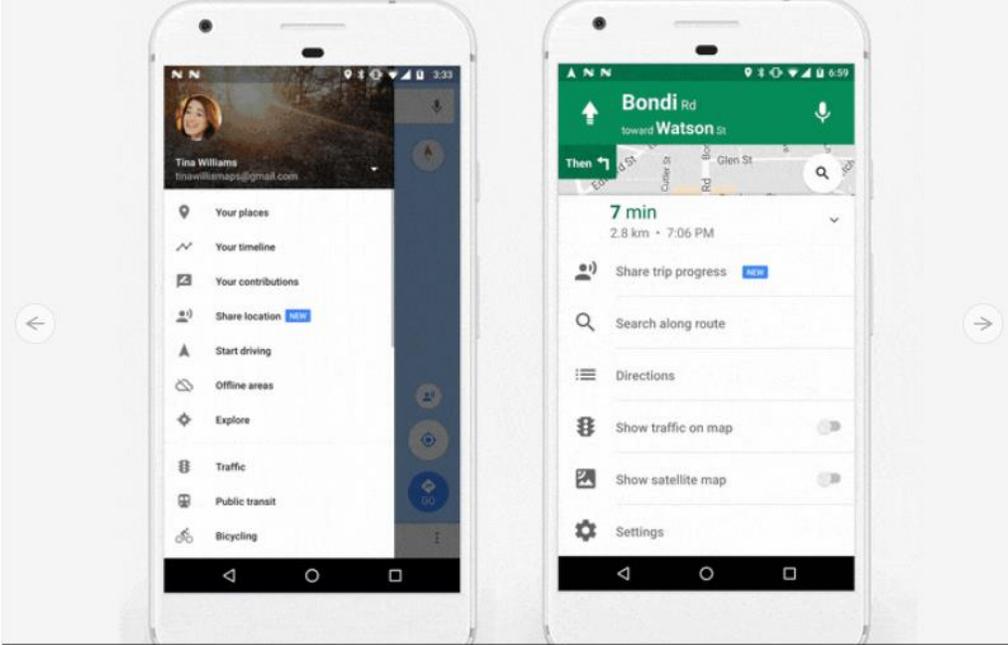
## Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products

US9408055B2	HTC
	<p>As shown below, a group may also be defined within Google Contacts.</p> <h3>See your contacts</h3> <ol style="list-style-type: none"><li>1. Open your device's Contacts app .</li><li>2. Tap Menu .</li></ol> <ul style="list-style-type: none"><li>• <b>See contacts by label:</b> Choose a label from the list.</li><li>• <b>See contacts for another account:</b> Tap Down arrow  &gt; pick an account.</li><li>• <b>See the contacts for all your accounts:</b> Choose <b>All contacts</b>.</li></ul> <p><b>Tip:</b> If you have multiple contacts with the same information, the information will be grouped into one contact.</p> <ul style="list-style-type: none"><li>• <b>See your Google Account contacts on the web:</b> Go to <a href="#">Google Contacts</a> .</li></ul> <p><a href="https://support.google.com/android/answer/6118731?hl=en&amp;ref_topic=6118711">https://support.google.com/android/answer/6118731?hl=en&amp;ref_topic=6118711</a></p> <h3>Label your contacts</h3> <p>You can group contacts together using labels.</p> <ol style="list-style-type: none"><li>1. Open your device's Contacts app .</li><li>2. Tap Menu  &gt; <b>Create label</b>.</li><li>3. Enter a label name and tap <b>Ok</b>.</li></ol> <ul style="list-style-type: none"><li>• <b>Add one contact to a label:</b> Tap Add contact  &gt; choose a contact.</li><li>• <b>Add multiple contacts to a label:</b> Tap Add contact  &gt; touch and hold a contact &gt; tap the other contacts &gt; tap <b>Add</b>.</li></ul> <p><a href="https://support.google.com/android/answer/6118731?hl=en&amp;ref_topic=6118711">https://support.google.com/android/answer/6118731?hl=en&amp;ref_topic=6118711</a></p>

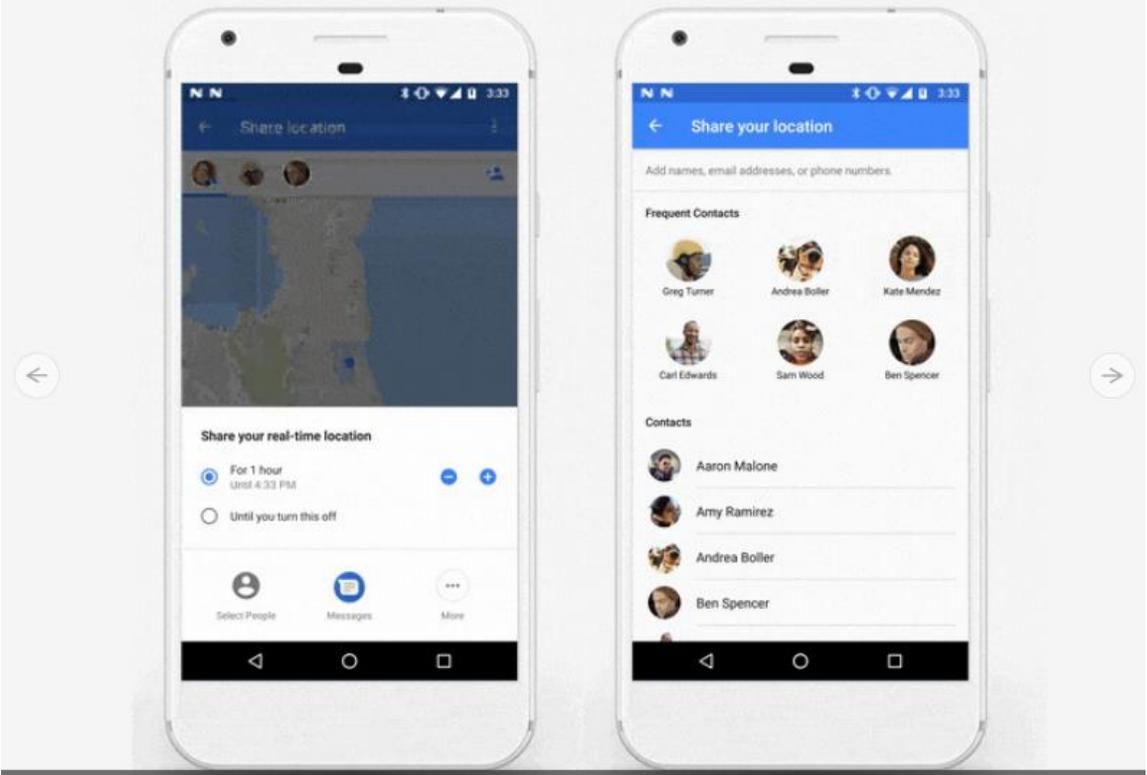
**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC
	<p data-bbox="548 315 945 358"><b>Share your contacts</b></p> <ol data-bbox="562 386 1045 545" style="list-style-type: none"><li data-bbox="562 386 976 415">1. Open your device's Contacts app .</li><li data-bbox="562 431 840 461">2. Tap a contact in the list.</li><li data-bbox="562 477 829 506">3. Tap More  &gt; <b>Share</b>.</li><li data-bbox="562 522 1045 552">4. Choose how you want to share the contact.</li></ol> <p data-bbox="512 597 1535 626"><a href="https://support.google.com/android/answer/6118731?hl=en&amp;ref_topic=6118711">https://support.google.com/android/answer/6118731?hl=en&amp;ref_topic=6118711</a></p>

## Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products

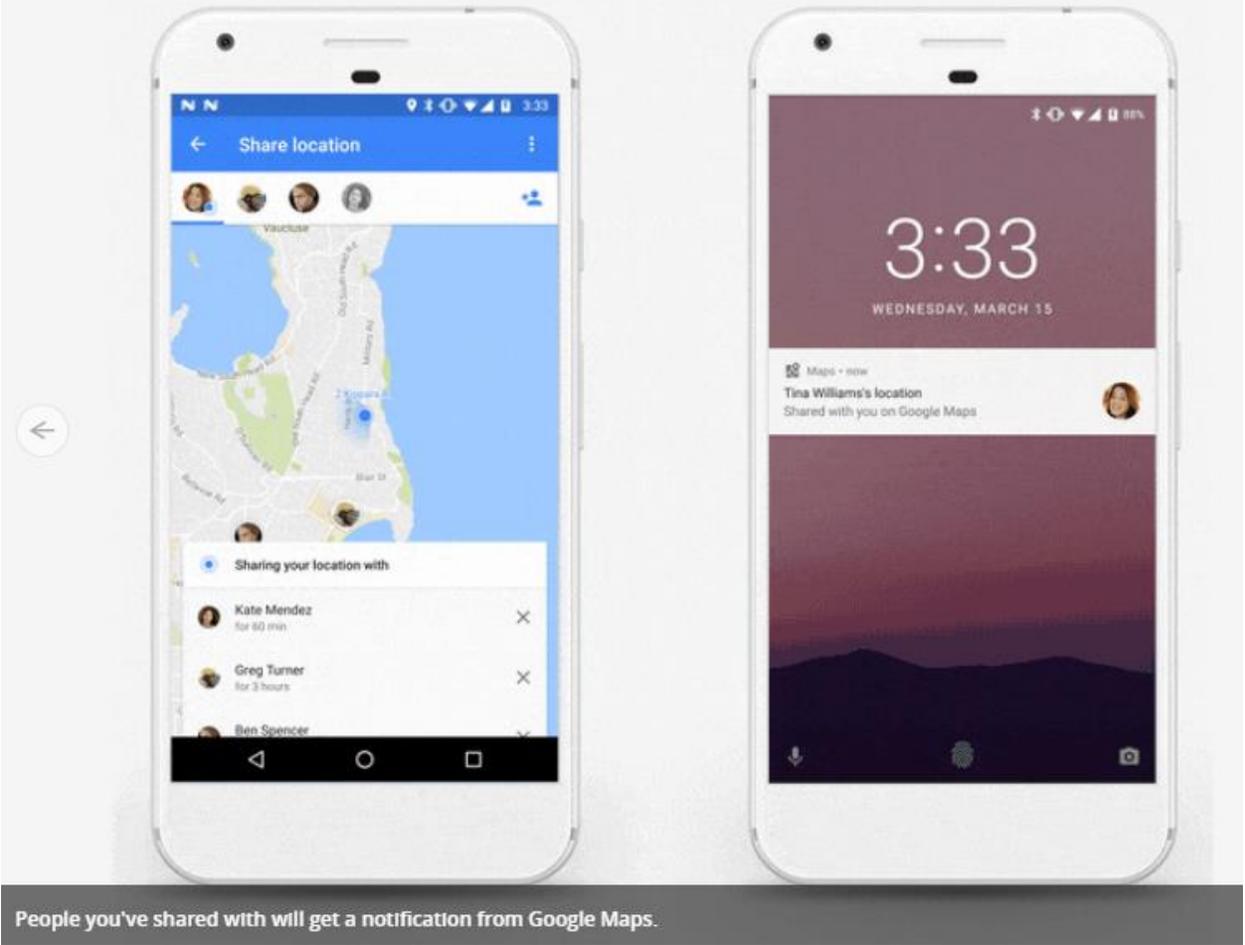
US9408055B2	HTC
	 <p data-bbox="514 893 1522 950">Ways to start location sharing: (left) from the app navigation drawer, and (right) from the menu, you can drag up while navigating somewhere.</p> <p data-bbox="514 982 1659 1031"><a href="https://arstechnica.com/gadgets/2017/03/location-sharing-finally-returns-to-google-maps/">https://arstechnica.com/gadgets/2017/03/location-sharing-finally-returns-to-google-maps/</a></p>

## Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products

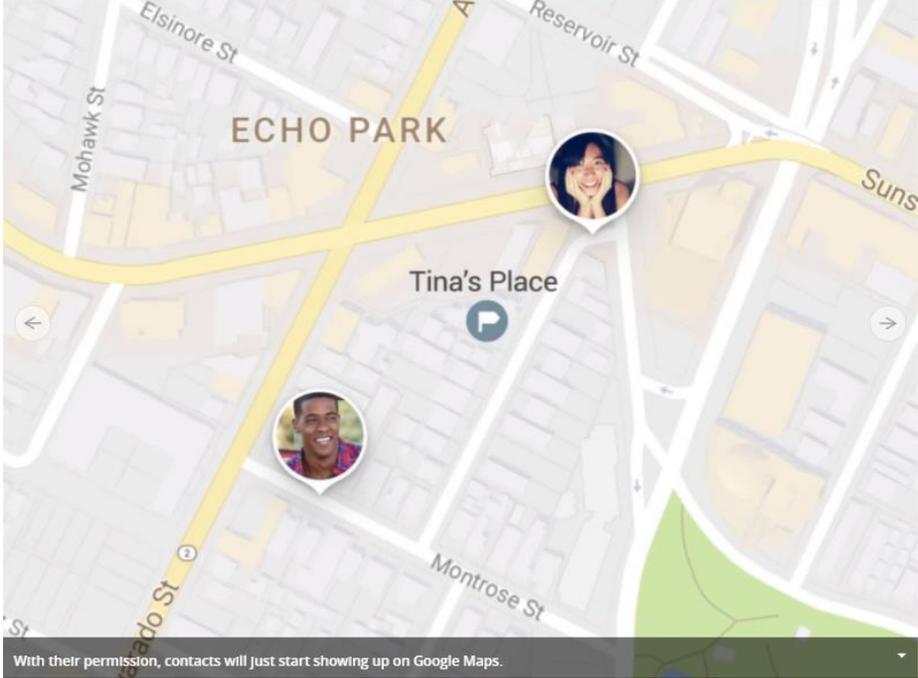
US9408055B2	HTC
	 <p data-bbox="514 1015 1661 1063">You can pick how long you want to share your location for, and with whom you want to share it.</p> <p data-bbox="514 1096 1661 1136"><a href="https://arstechnica.com/gadgets/2017/03/location-sharing-finally-returns-to-google-maps/">https://arstechnica.com/gadgets/2017/03/location-sharing-finally-returns-to-google-maps/</a></p>



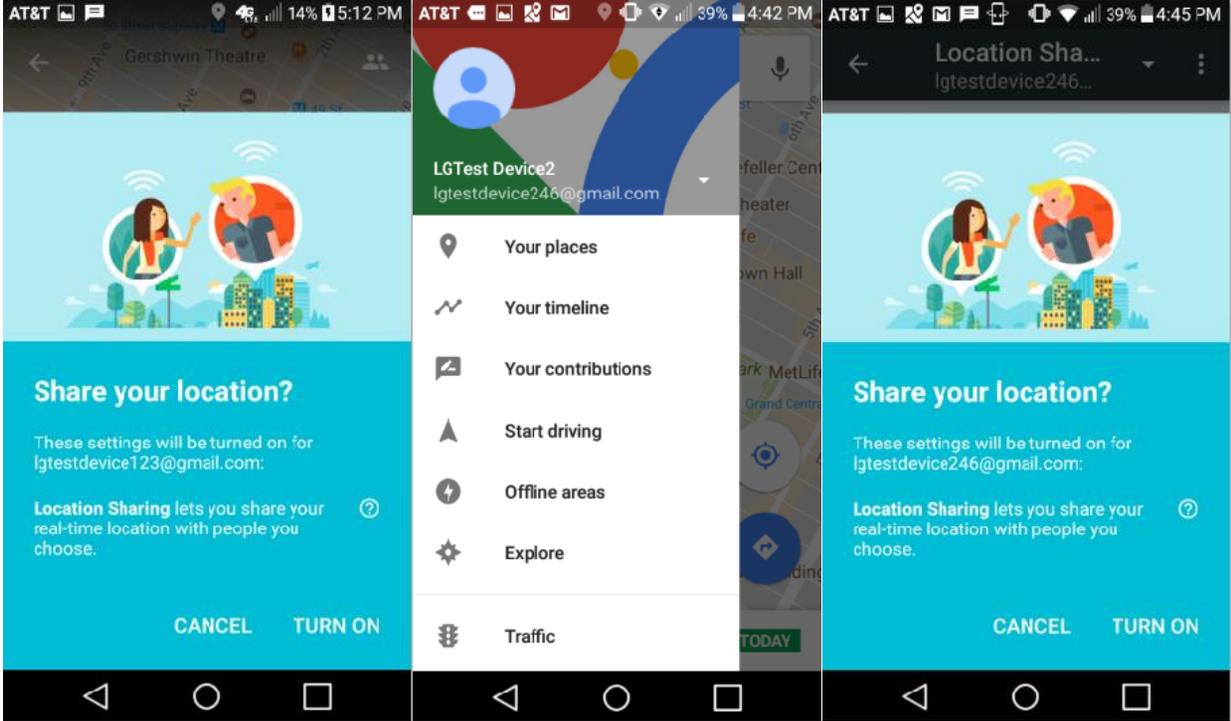
**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC
	 <p data-bbox="514 1144 1176 1177">People you've shared with will get a notification from Google Maps.</p> <p data-bbox="514 1218 1659 1258"><a href="https://arstechnica.com/gadgets/2017/03/location-sharing-finally-returns-to-google-maps/">https://arstechnica.com/gadgets/2017/03/location-sharing-finally-returns-to-google-maps/</a></p>

**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC
	 <p data-bbox="512 946 1656 979"><a href="https://arstechnica.com/gadgets/2017/03/location-sharing-finally-returns-to-google-maps/">https://arstechnica.com/gadgets/2017/03/location-sharing-finally-returns-to-google-maps/</a></p> <p data-bbox="512 1084 1020 1117"><b><u>Exemplary Google Maps Screenshots</u></b></p>

**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC
	 <p>Exemplary Source Code:</p> <p>The above functionality is performed at least in part by the following publicly available source code and/or source code that invokes or is invoked by the following source code (or a substantially similar copy compiled and loaded onto the Accused Products by HTC): AGIS reserves the right to amend these contentions to include additional source code as discovery progresses and as additional source code is made available.</p>

**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC
	<h2 data-bbox="527 240 1016 293">Contacts Provider</h2> <p data-bbox="527 329 1472 589">The Contacts Provider is a powerful and flexible Android component that manages the device's central repository of data about people. The Contacts Provider is the source of data you see in the device's contacts application, and you can also access its data in your own application and transfer data between the device and online services. The provider accommodates a wide range of data sources and tries to manage as much data as possible for each person, with the result that its organization is complex. Because of this, the provider's API includes an extensive set of contract classes and interfaces that facilitate both data retrieval and modification.</p> <p data-bbox="527 625 835 646">This guide describes the following:</p> <ul data-bbox="527 675 1373 850" style="list-style-type: none"><li data-bbox="527 675 806 696">• The basic provider structure.</li><li data-bbox="527 725 894 747">• How to retrieve data from the provider.</li><li data-bbox="527 776 863 797">• How to modify data in the provider.</li><li data-bbox="527 826 1373 847">• How to write a sync adapter for synchronizing data from your server to the Contacts Provider.</li></ul> <p data-bbox="512 894 1486 922"><a href="https://developer.android.com/guide/topics/providers/contacts-provider.html">https://developer.android.com/guide/topics/providers/contacts-provider.html</a></p>

**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC
	<p data-bbox="533 240 659 266"><b>Overview</b></p> <p data-bbox="533 297 1608 315">ContactsContract defines an extensible database of contact-related information. Contact information is stored in a three-tier data model:</p> <ul data-bbox="533 342 1713 545" style="list-style-type: none"> <li data-bbox="533 342 1713 391">• A row in the <code>ContactsContract.Data</code> table can store any kind of personal data, such as a phone number or email addresses. The set of data kinds that can be stored in this table is open-ended. There is a predefined set of common kinds, but any application can add its own data kinds.</li> <li data-bbox="533 418 1713 467">• A row in the <code>ContactsContract.RawContacts</code> table represents a set of data describing a person and associated with a single account (for example, one of the user's Gmail accounts).</li> <li data-bbox="533 495 1713 545">• A row in the <code>ContactsContract.Contacts</code> table represents an aggregate of one or more RawContacts presumably describing the same person. When data in or associated with the RawContacts table is changed, the affected aggregate contacts are updated as necessary.</li> </ul> <p data-bbox="533 573 695 591">Other tables include:</p> <ul data-bbox="533 618 1713 902" style="list-style-type: none"> <li data-bbox="533 618 1713 667">• <code>ContactsContract.Groups</code>, which contains information about raw contact groups such as Gmail contact groups. The current API does not support the notion of groups spanning multiple accounts.</li> <li data-bbox="533 695 1325 712">• <code>ContactsContract.StatusUpdates</code>, which contains social status updates including IM availability.</li> <li data-bbox="533 740 1493 758">• <code>ContactsContract.AggregationExceptions</code>, which is used for manual aggregation and disaggregation of raw contacts</li> <li data-bbox="533 786 1325 803">• <code>ContactsContract.Settings</code>, which contains visibility and sync settings for accounts and groups.</li> <li data-bbox="533 831 1346 849">• <code>ContactsContract.SyncState</code>, which contains free-form data maintained on behalf of sync adapters</li> <li data-bbox="533 876 1125 894">• <code>ContactsContract.PhoneLookup</code>, which is used for quick caller-ID lookup</li> </ul> <p data-bbox="512 954 1541 980"><a href="https://developer.android.com/reference/android/provider/ContactsContract.html">https://developer.android.com/reference/android/provider/ContactsContract.html</a></p> <p data-bbox="533 1024 617 1050"><b>Data</b></p> <p data-bbox="533 1094 1745 1235">As noted previously, the data for a raw contact is stored in a <code>ContactsContract.Data</code> row that is linked to the raw contact's <code>_id</code> value. This allows a single raw contact to have multiple instances of the same type of data such as email addresses or phone numbers. For example, if "Thomas Higginson" for <code>emilyd@gmail.com</code> (the raw contact row for Thomas Higginson associated with the Google account <code>emilyd@gmail.com</code>) has a home email address of <code>thigg@gmail.com</code> and a work email address of <code>thomas.higginson@gmail.com</code>, the Contacts Provider stores the two email address rows and links them both to the raw contact.</p> <p data-bbox="533 1263 1745 1373">Notice that different types of data are stored in this single table. Display name, phone number, email, postal address, photo, and website detail rows are all found in the <code>ContactsContract.Data</code> table. To help manage this, the <code>ContactsContract.Data</code> table has some columns with descriptive names, and others with generic names. The contents of a descriptive-name column have the same meaning regardless of the type of data in the row, while the contents of a generic-name column have different meanings depending on the type of data.</p>

**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC										
	<p data-bbox="512 238 1486 266"><a href="https://developer.android.com/guide/topics/providers/contacts-provider.html">https://developer.android.com/guide/topics/providers/contacts-provider.html</a></p> <table border="1" data-bbox="520 375 1745 1045"> <thead> <tr> <th data-bbox="527 380 617 417">Task</th> <th data-bbox="623 380 854 417">Action</th> <th data-bbox="861 380 1188 417">Data</th> <th data-bbox="1194 380 1486 417">MIME type</th> <th data-bbox="1493 380 1738 417">Notes</th> </tr> </thead> <tbody> <tr> <td data-bbox="527 422 617 548">Pick a contact from a list</td> <td data-bbox="623 422 854 548">ACTION_PICK</td> <td data-bbox="861 422 1188 857">                     One of:                     <ul style="list-style-type: none"> <li data-bbox="867 459 1182 516">• <code>Contacts.CONTENT_URI</code>, which displays a list of contacts.</li> <li data-bbox="867 540 1182 630">• <code>Phone.CONTENT_URI</code>, which displays a list of phone numbers for a raw contact.</li> <li data-bbox="867 654 1182 743">• <code>StructuredPostal.CONTENT_URI</code>, which displays a list of postal addresses for a raw contact.</li> <li data-bbox="867 768 1182 857">• <code>Email.CONTENT_URI</code>, which displays a list of email addresses for a raw contact.</li> </ul> </td> <td data-bbox="1194 422 1486 548">Not used</td> <td data-bbox="1493 422 1738 1036">                     Displays a list of raw contacts or a list of data from a raw contact, depending on the content URI type you supply. Call <code>startActivityForResult()</code>, which returns the content URI of the selected row. The form of the URI is the table's content URI with the row's <code>LOOKUP_ID</code> appended to it. The device's contacts app delegates read and write permissions to this content URI for the life of your activity. See the <a href="#">Content Provider Basics</a> guide for more details.                 </td> </tr> </tbody> </table> <p data-bbox="512 1084 1486 1112"><a href="https://developer.android.com/guide/topics/providers/contacts-provider.html">https://developer.android.com/guide/topics/providers/contacts-provider.html</a></p>	Task	Action	Data	MIME type	Notes	Pick a contact from a list	ACTION_PICK	One of: <ul style="list-style-type: none"> <li data-bbox="867 459 1182 516">• <code>Contacts.CONTENT_URI</code>, which displays a list of contacts.</li> <li data-bbox="867 540 1182 630">• <code>Phone.CONTENT_URI</code>, which displays a list of phone numbers for a raw contact.</li> <li data-bbox="867 654 1182 743">• <code>StructuredPostal.CONTENT_URI</code>, which displays a list of postal addresses for a raw contact.</li> <li data-bbox="867 768 1182 857">• <code>Email.CONTENT_URI</code>, which displays a list of email addresses for a raw contact.</li> </ul>	Not used	Displays a list of raw contacts or a list of data from a raw contact, depending on the content URI type you supply. Call <code>startActivityForResult()</code> , which returns the content URI of the selected row. The form of the URI is the table's content URI with the row's <code>LOOKUP_ID</code> appended to it. The device's contacts app delegates read and write permissions to this content URI for the life of your activity. See the <a href="#">Content Provider Basics</a> guide for more details.
Task	Action	Data	MIME type	Notes							
Pick a contact from a list	ACTION_PICK	One of: <ul style="list-style-type: none"> <li data-bbox="867 459 1182 516">• <code>Contacts.CONTENT_URI</code>, which displays a list of contacts.</li> <li data-bbox="867 540 1182 630">• <code>Phone.CONTENT_URI</code>, which displays a list of phone numbers for a raw contact.</li> <li data-bbox="867 654 1182 743">• <code>StructuredPostal.CONTENT_URI</code>, which displays a list of postal addresses for a raw contact.</li> <li data-bbox="867 768 1182 857">• <code>Email.CONTENT_URI</code>, which displays a list of email addresses for a raw contact.</li> </ul>	Not used	Displays a list of raw contacts or a list of data from a raw contact, depending on the content URI type you supply. Call <code>startActivityForResult()</code> , which returns the content URI of the selected row. The form of the URI is the table's content URI with the row's <code>LOOKUP_ID</code> appended to it. The device's contacts app delegates read and write permissions to this content URI for the life of your activity. See the <a href="#">Content Provider Basics</a> guide for more details.							

**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC
	<pre>25  * Parsed form of the intent sent to the Contacts application. 26  */ 27  public class ContactsRequest { 28 29      /** Default mode: browse contacts */ 30      public static final int ACTION_DEFAULT = 10; 31 32      /** Show all contacts */ 33      public static final int ACTION_ALL_CONTACTS = 15; 34 35      /** Show all contacts with phone numbers */ 36      public static final int ACTION_CONTACTS_WITH_PHONES = 17; 37 38      /** Show contents of a specific group */ 39      public static final int ACTION_GROUP = 20; 40 41      /** Show all starred contacts */ 42      public static final int ACTION_STARRED = 30; 43 44      /** Show frequently contacted contacts */ 45      public static final int ACTION_FREQUENT = 40; 46 47      /** Show starred and the frequent */ 48      public static final int ACTION_STREQUENT = 50; 49 50      /** Show all contacts and pick them when clicking */ 51      public static final int ACTION_PICK_CONTACT = 60; 52 53      /** Show all contacts as well as the option to create a new one */ 54      public static final int ACTION_PICK_OR_CREATE_CONTACT = 70; 55 56      /** Show all contacts and pick them for edit when clicking, and allow creating a new contact */ 57      public static final int ACTION_INSERT_OR_EDIT_CONTACT = 80;</pre> <p><a href="https://android.googlesource.com/platform/packages/apps/Contacts+/nougat-mr1-release/src/com/android/contacts/list/ContactsRequest.java">https://android.googlesource.com/platform/packages/apps/Contacts+/nougat-mr1-release/src/com/android/contacts/list/ContactsRequest.java</a></p>

**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC
	<pre>59  /** Show all phone numbers and pick them when clicking */ 60  public static final int ACTION_PICK_PHONE = 90; 61 62  /** Show all postal addresses and pick them when clicking */ 63  public static final int ACTION_PICK_POSTAL = 100; 64 65  /** Show all postal addresses and pick them when clicking */ 66  public static final int ACTION_PICK_EMAIL = 105; 67 68  /** Show all contacts and create a shortcut for the picked contact */ 69  public static final int ACTION_CREATE_SHORTCUT_CONTACT = 110; 70 71  /** Show all phone numbers and create a call shortcut for the picked number */ 72  public static final int ACTION_CREATE_SHORTCUT_CALL = 120; 73 74  /** Show all phone numbers and create an SMS shortcut for the picked number */ 75  public static final int ACTION_CREATE_SHORTCUT_SMS = 130; 76 77  /** Show all contacts and activate the specified one */ 78  public static final int ACTION_VIEW_CONTACT = 140; 79 80  /** Show contacts recommended for joining with a specified target contact */ 81  public static final int ACTION_PICK_JOIN = 150;</pre> <p><a href="https://android.googlesource.com/platform/packages/apps/Contacts/+nougat-mr1-release/src/com/android/contacts/list/ContactsRequest.java">https://android.googlesource.com/platform/packages/apps/Contacts/+nougat-mr1-release/src/com/android/contacts/list/ContactsRequest.java</a></p>



**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC
	<pre> 104  * Displays a list to browse contacts. 105  */ 106  public class PeopleActivity extends ContactsActivity implements 107      View.OnCreateContextMenuListener, 108      View.OnClickListener, 109      ActionBarAdapter.Listener, 110      DialogManager.DialogShowingViewActivity, 111      ContactListFilterController.ContactListFilterListener, 112      ProviderStatusListener, 113      MultiContactDeleteListener, 114      JoinContactsListener {  https://android.googleusercontent.com/platform/packages/apps/Contacts+/nougat-mr1- release/src/com/android/contacts/activities/PeopleActivity.java  145      * Showing a list of Contacts. Also used for showing search results in search mode. 146      */ 147      private MultiSelectContactsListFragment mAllFragment; 148      private ContactTileListFragment mFavoritesFragment;  https://android.googleusercontent.com/platform/packages/apps/Contacts+/nougat-mr1- release/src/com/android/contacts/activities/PeopleActivity.java </pre>

**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC
	<pre>458     private void configureFragments(boolean fromRequest) { 459         if (fromRequest) { 460             ContactListFilter filter = null; 461             int actionCode = mRequest.getActionCode(); 462             boolean searchMode = mRequest.isSearchMode(); 463             final int tabToOpen; 464             switch (actionCode) { 465                 case ContactsRequest.ACTION_ALL_CONTACTS: 466                     filter = ContactListFilter.createFilterWithType( 467                         ContactListFilter.FILTER_TYPE_ALL_ACCOUNTS); 468                     tabToOpen = TabState.ALL; 469                     break; 470                 case ContactsRequest.ACTION_CONTACTS_WITH_PHONES: 471                     filter = ContactListFilter.createFilterWithType( 472                         ContactListFilter.FILTER_TYPE_WITH_PHONE_NUMBERS_ONLY); 473                     tabToOpen = TabState.ALL; 474                     break; 475 476                 case ContactsRequest.ACTION_FREQUENT: 477                 case ContactsRequest.ACTION_STREQUENT: 478                 case ContactsRequest.ACTION_STARRED: 479                     tabToOpen = TabState.FAVORITES; 480                     break; 481                 case ContactsRequest.ACTION_VIEW_CONTACT: 482                     tabToOpen = TabState.ALL; 483                     break; 484                 default: 485                     tabToOpen = -1; 486                     break; 487             } </pre> <p><a href="https://android.googlesource.com/platform/packages/apps/Contacts+/nougat-mr1-">https://android.googlesource.com/platform/packages/apps/Contacts+/nougat-mr1-</a></p>

**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC
	<pre>release/src/com/android/contacts/activities/PeopleActivity.java  488         if (tabToOpen != -1) { 489             mActionBarAdapter.setCurrentTab(tabToOpen); 490         } 491 492         if (filter != null) { 493             mContactListFilterController.setContactListFilter(filter, false); 494             searchMode = false; 495         } 496 497         if (mRequest.getContactUri() != null) { 498             searchMode = false; 499         } 500 501         mActionBarAdapter.setSearchMode(searchMode); 502         configureContactListFragmentForRequest(); 503     } 504 505     configureContactListFragment(); 506 507     invalidateOptionsMenuIfNeeded(); 508 }</pre> <p><a href="https://android.googlesource.com/platform/packages/apps/Contacts/+nougat-mr1-release/src/com/android/contacts/activities/PeopleActivity.java">https://android.googlesource.com/platform/packages/apps/Contacts/+nougat-mr1-release/src/com/android/contacts/activities/PeopleActivity.java</a></p>

**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC
	<pre>35 public class ProfileAndContactsLoader extends CursorLoader { 36 37     private boolean mLoadProfile; 38 39     private String[] mProjection; 40 41     private Uri mExtraUri; 42     private String[] mExtraProjection; 43     private String mExtraSelection; 44     private String[] mExtraSelectionArgs; 45     private boolean mMergeExtraContactsAfterPrimary; 46 47     public ProfileAndContactsLoader(Context context) { 48         super(context); 49     }</pre> <p><a href="https://android.googlesource.com/platform/packages/apps/ContactsCommon/+/nougat-mr1-release/src/com/android/contacts/common/list/ProfileAndContactsLoader.java">https://android.googlesource.com/platform/packages/apps/ContactsCommon/+/nougat-mr1-release/src/com/android/contacts/common/list/ProfileAndContactsLoader.java</a></p>

**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC
	<pre>35 public final class GroupMemberLoader extends CursorLoader { 36 37     public static class GroupEditorQuery { 38         private static final String[] PROJECTION = new String[] { 39             Data.CONTACT_ID,           // 0 40             Data.RAW_CONTACT_ID,      // 1 41             Data.DISPLAY_NAME_PRIMARY, // 2 42             Data.PHOTO_URI,           // 3 43             Data.LOOKUP_KEY,          // 4 44         }; 45 46         public static final int CONTACT_ID           = 0; 47         public static final int RAW_CONTACT_ID      = 1; 48         public static final int CONTACT_DISPLAY_NAME_PRIMARY = 2; 49         public static final int CONTACT_PHOTO_URI   = 3; 50         public static final int CONTACT_LOOKUP_KEY  = 4; 51     } 52 53     public static class GroupDetailQuery { 54         private static final String[] PROJECTION = new String[] { 55             Data.CONTACT_ID,           // 0 56             Data.PHOTO_URI,            // 1 57             Data.LOOKUP_KEY,           // 2 58             Data.DISPLAY_NAME_PRIMARY, // 3 59             Data.CONTACT_PRESENCE,     // 4 60             Data.CONTACT_STATUS,      // 5 61         }; 62 63         public static final int CONTACT_ID           = 0; 64         public static final int CONTACT_PHOTO_URI   = 1; 65         public static final int CONTACT_LOOKUP_KEY  = 2; 66         public static final int CONTACT_DISPLAY_NAME_PRIMARY = 3; 67         public static final int CONTACT_PRESENCE_STATUS = 4; 68         public static final int CONTACT_STATUS     = 5; 69     } 70 71     private final long mGroupId;</pre> <p><a href="https://android.googlesource.com/platform/packages/apps/Contacts/+nougat-mr1-release/src/com/android/contacts/GroupMemberLoader.java">https://android.googlesource.com/platform/packages/apps/Contacts/+nougat-mr1-release/src/com/android/contacts/GroupMemberLoader.java</a></p>

**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC
	<pre> 24  * Group loader for the group list that includes details such as the number of contacts per group 25  * and number of groups per account. This list is sorted by account type, account name, where the 26  * group names are in alphabetical order. Note that the list excludes default, favorite, and deleted 27  * groups. 28  */ 29  public final class GroupListLoader extends CursorLoader { 30 31      private final static String[] COLUMNS = new String[] { 32          Groups.ACCOUNT_NAME, 33          Groups.ACCOUNT_TYPE, 34          Groups.DATA_SET, 35          Groups._ID, 36          Groups.TITLE, 37          Groups.SUMMARY_COUNT, 38      }; 39 40      public final static int ACCOUNT_NAME = 0; 41      public final static int ACCOUNT_TYPE = 1; 42      public final static int DATA_SET = 2; 43      public final static int GROUP_ID = 3; 44      public final static int TITLE = 4; 45      public final static int MEMBER_COUNT = 5; 46 47      private static final Uri GROUP_LIST_URI = Groups.CONTENT_SUMMARY_URI; 48 49      public GroupListLoader(Context context) { 50          super(context, GROUP_LIST_URI, COLUMNS, Groups.ACCOUNT_TYPE + " NOT NULL AND " 51              + Groups.ACCOUNT_NAME + " NOT NULL AND " + Groups.AUTO_ADD + "=0 AND " + 52              Groups.FAVORITES + "=0 AND " + Groups.DELETED + "=0", null, 53              Groups.ACCOUNT_TYPE + ", " + Groups.ACCOUNT_NAME + ", " + Groups.DATA_SET + ", " + 54              Groups.TITLE + " COLLATE LOCALIZED ASC"); 55      } 56  } </pre> <p><a href="https://android.googlesource.com/platform/packages/apps/Contacts+/nougat-mr1-release/src/com/android/contacts/GroupListLoader.java">https://android.googlesource.com/platform/packages/apps/Contacts+/nougat-mr1-release/src/com/android/contacts/GroupListLoader.java</a></p>

**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC
	<pre> 24  * Group meta-data loader. Loads all groups or just a single group from the 25  * database (if given a {@link Uri}). 26  */ 27  public final class GroupMetaLoader extends CursorLoader { 28 29      private final static String[] COLUMNS = new String[] { 30          Groups.ACCOUNT_NAME, 31          Groups.ACCOUNT_TYPE, 32          Groups.DATA_SET, 33          Groups._ID, 34          Groups.TITLE, 35          Groups.AUTO_ADD, 36          Groups.FAVORITES, 37          Groups.GROUP_IS_READ_ONLY, 38          Groups.DELETED, 39      }; 40 41      public final static int ACCOUNT_NAME = 0; 42      public final static int ACCOUNT_TYPE = 1; 43      public final static int DATA_SET = 2; 44      public final static int GROUP_ID = 3; 45      public final static int TITLE = 4; 46      public final static int AUTO_ADD = 5; 47      public final static int FAVORITES = 6; 48      public final static int IS_READ_ONLY = 7; 49      public final static int DELETED = 8; 50 51      public GroupMetaLoader(Context context, Uri groupUri) { 52          super(context, ensureIsGroupUri(groupUri), COLUMNS, Groups.ACCOUNT_TYPE + " NOT NULL AND " 53              + Groups.ACCOUNT_NAME + " NOT NULL", null, null); 54      } 55 56      /** 57       * Ensures that this is a valid group URI. If invalid, then an exception is 58       * thrown. Otherwise, the original URI is returned. 59       */ 60      private static Uri ensureIsGroupUri(final Uri groupUri) { 61          // TODO: Fix ContactsProvider2 getType method to resolve the group Uris 62          if (groupUri == null) { 63              throw new IllegalArgumentException("Uri must not be null"); 64          } 65          if (!groupUri.toString().startsWith(Groups.CONTENT_URI.toString())) { 66              throw new IllegalArgumentException("Invalid group Uri: " + groupUri); 67          } 68          return groupUri; 69      } 70  } </pre> <p><a href="https://android.googlesource.com/platform/packages/apps/Contacts/+nougat-mr1-release/src/com/android/contacts/GroupMetaLoader.java">https://android.googlesource.com/platform/packages/apps/Contacts/+nougat-mr1-release/src/com/android/contacts/GroupMetaLoader.java</a></p>

**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC
	<pre>19  * Meta-data for a contact group. We load all groups associated with the contact's 20  * constituent accounts. 21  */ 22  public final class GroupMetaData { 23      private String mAccountName; 24      private String mAccountType; 25      private String mDataSet; 26      private long mGroupId; 27      private String mTitle; 28      private boolean mDefaultGroup; 29      private boolean mFavorites; 30 31      public GroupMetaData(String accountName, String accountType, String dataSet, long groupId, 32                          String title, boolean defaultGroup, boolean favorites) { 33          this.mAccountName = accountName; 34          this.mAccountType = accountType; 35          this.mDataSet = dataSet; 36          this.mGroupId = groupId; 37          this.mTitle = title; 38          this.mDefaultGroup = defaultGroup; 39          this.mFavorites = favorites; 40      } 41 42      public String getAccountName() { 43          return mAccountName; 44      } 45 46      public String getAccountType() { 47          return mAccountType; 48      } 49 50      public String getDataSet() { 51          return mDataSet; 52      } 53 54      public long getGroupId() { 55          return mGroupId; 56      } 57 58      public String getTitle() { 59          return mTitle; 60      } 61 62      public boolean isDefaultGroup() { 63          return mDefaultGroup; 64      } 65 66      public boolean isFavorites() { 67          return mFavorites; 68      } 69  }</pre> <p><a href="https://android.googlesource.com/platform/packages/apps/ContactsCommon/+nougat-mr1-">https://android.googlesource.com/platform/packages/apps/ContactsCommon/+nougat-mr1-</a></p>



**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC
	<p data-bbox="514 233 1333 261"><a href="https://android.googlesource.com/platform/packages/apps/Messaging/+/nougat-mr1-release/src/com/android/contacts/common/GroupMetaData.java">release/src/com/android/contacts/common/GroupMetaData.java</a></p> <pre data-bbox="535 511 1606 1323">44 * Class that sends chat message via SMS. 45 * 46 * The interface emulates a blocking sending similar to making an HTTP request. 47 * It calls the SmsManager to send a (potentially multipart) message and waits 48 * on the sent status on each part. The waiting has a timeout so it won't wait 49 * forever. Once the sent status of all parts received, the call returns. 50 * A successful sending requires success status for all parts. Otherwise, we 51 * pick the highest level of failure as the error for the whole message, which 52 * is used to determine if we need to retry the sending. 53 */ 54 public class SmsSender { 55     private static final String TAG = LogUtil.BUGLE_TAG; 56 57     public static final String EXTRA_PART_ID = "part_id"; 58 59     /* 60     * A map for pending sms messages. The key is the random request UUID. 61     */ 62     private static ConcurrentHashMap&lt;Uri, SendResult&gt; sPendingMessageMap = 63         new ConcurrentHashMap&lt;Uri, SendResult&gt;(); 64 65     private static final Random RANDOM = new Random(); 66 67     // Whether we should send multipart SMS as separate messages 68     private static Boolean sSendMultipartSmsAsSeparateMessages = null; 69</pre> <p data-bbox="514 1372 1596 1404"><a href="https://android.googlesource.com/platform/packages/apps/Messaging/+/nougat-mr1-">https://android.googlesource.com/platform/packages/apps/Messaging/+/nougat-mr1-</a></p>

**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC
	<p data-bbox="514 233 1234 261"><a href="#">release/src/com/android/messaging/sms/SmsSender.java</a></p> <pre data-bbox="514 305 1745 1307"> 253     // Actually sending the message using SmsManager 254     private static void sendInternal(final Context context, final int subId, String dest, 255         final ArrayList&lt;String&gt; messages, final String serviceCenter, 256         final boolean requireDeliveryReport, final Uri messageUri) throws SmsException { 257         Assert.notNull(context); 258         final SmsManager smsManager = PhoneUtils.get(subId).getSmsManager(); 259         final int messageCount = messages.size(); 260         final ArrayList&lt;PendingIntent&gt; deliveryIntents = new ArrayList&lt;PendingIntent&gt;(messageCount); 261         final ArrayList&lt;PendingIntent&gt; sentIntents = new ArrayList&lt;PendingIntent&gt;(messageCount); 262         for (int i = 0; i &lt; messageCount; i++) { 263             // Make pending intents different for each message part 264             final int partId = (messageCount &lt;= 1 ? 0 : i + 1); 265             if (requireDeliveryReport &amp;&amp; (i == (messageCount - 1))) { 266                 // TODO we only care about the delivery status of the last part 267                 // Shall we have better tracking of delivery status of all parts? 268                 deliveryIntents.add(PendingIntent.getBroadcast( 269                     context, 270                     partId, 271                     getSendStatusIntent(context, SendStatusReceiver.MESSAGE_DELIVERED_ACTION, 272                         messageUri, partId, subId), 273                     0/*flag*/)); 274             } else { 275                 deliveryIntents.add(null); 276             } 277             sentIntents.add(PendingIntent.getBroadcast( 278                 context, 279                 partId, 280                 getSendStatusIntent(context, SendStatusReceiver.MESSAGE_SENT_ACTION, 281                     messageUri, partId, subId), 282                 0/*flag*/)); 283         } 284         if (sSendMultipartSmsAsSeparateMessages == null) { 285             sSendMultipartSmsAsSeparateMessages = MmsConfig.get(subId) 286                 .getSendMultipartSmsAsSeparateMessages(); 287         } </pre> <p data-bbox="514 1349 1593 1377"><a href="https://android.googlesource.com/platform/packages/apps/Messaging/+nougat-mr1-">https://android.googlesource.com/platform/packages/apps/Messaging/+nougat-mr1-</a></p>

**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC
	<p data-bbox="514 240 1234 264"><a href="https://android.googlesource.com/platform/packages/apps/Messaging/+/nougat-mr1-release/src/com/android/messaging/sms/SmsSender.java">release/src/com/android/messaging/sms/SmsSender.java</a></p> <pre data-bbox="514 305 1680 868">288     try { 289         if (sSendMultipartSmsAsSeparateMessages) { 290             // If multipart sms is not supported, send them as separate messages 291             for (int i = 0; i &lt; messageCount; i++) { 292                 smsManager.sendTextMessage(dest, 293                     serviceCenter, 294                     messages.get(i), 295                     sentIntents.get(i), 296                     deliveryIntents.get(i)); 297             } 298         } else { 299             smsManager.sendMultipartTextMessage( 300                 dest, serviceCenter, messages, sentIntents, deliveryIntents); 301         } 302     } catch (final Exception e) { 303         throw new SmsException("SmsSender: caught exception in sending " + e); 304     } 305 }</pre> <p data-bbox="514 917 1596 987"><a href="https://android.googlesource.com/platform/packages/apps/Messaging/+/nougat-mr1-release/src/com/android/messaging/sms/SmsSender.java">https://android.googlesource.com/platform/packages/apps/Messaging/+/nougat-mr1-release/src/com/android/messaging/sms/SmsSender.java</a></p>

**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC
	<pre>56 * Class that receives incoming SMS messages through android.provider.Telephony.SMS_RECEIVED 57 * 58 * This class serves two purposes: 59 * - Process phone verification SMS messages 60 * - Handle SMS messages when the user has enabled us to be the default SMS app (Pre-KLP) 61 */ 62 public final class SmsReceiver extends BroadcastReceiver { 63     private static final String TAG = LogUtil.BUGLE_TAG; 64 65     private static ArrayList&lt;Pattern&gt; sIgnoreSmsPatterns; 66</pre> <p><a href="https://android.googlesource.com/platform/packages/apps/Messaging/+/nougat-mr1-release/src/com/android/messaging/receiver/SmsReceiver.java">https://android.googlesource.com/platform/packages/apps/Messaging/+/nougat-mr1-release/src/com/android/messaging/receiver/SmsReceiver.java</a></p>

**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC
	<pre> 200     public static void deliverSmsMessages(final Context context, final int subId, 201         final int errorCode, final android.telephony.SmsMessage[] messages) { 202         final ContentValues messageValues = 203             MmsUtils.parseReceivedSmsMessage(context, messages, errorCode); 204 205         LogUtil.v(TAG, "SmsReceiver.deliverSmsMessages"); 206 207         final long nowInMillis = System.currentTimeMillis(); 208         final long receivedTimestampMs = MmsUtils.getMessageDate(messages[0], nowInMillis); 209 210         messageValues.put(Sms.Inbox.DATE, receivedTimestampMs); 211         // Default to unread and unseen for us but ReceiveSmsMessageAction will override 212         // seen for the telephony db. 213         messageValues.put(Sms.Inbox.READ, 0); 214         messageValues.put(Sms.Inbox.SEEN, 0); 215         if (OsUtil.isAtLeastL_MR1()) { 216             messageValues.put(Sms.SUBSCRIPTION_ID, subId); 217         } 218 219         if (messages[0].getMessageClass() == android.telephony.SmsMessage.MessageClass.CLASS_0    220             DebugUtils.debugClassZeroSmsEnabled()) { 221             Factory.get().getUIIntents().launchClassZeroActivity(context, messageValues); 222         } else { 223             final ReceiveSmsMessageAction action = new ReceiveSmsMessageAction(messageValues); 224             action.start(); 225         } 226     } </pre> <p><a href="https://android.googlesource.com/platform/packages/apps/Messaging/+/nougat-mr1-release/src/com/android/messaging/receiver/SmsReceiver.java">https://android.googlesource.com/platform/packages/apps/Messaging/+/nougat-mr1-release/src/com/android/messaging/receiver/SmsReceiver.java</a></p>

**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC
	<pre>228     @Override 229     public void onReceive(final Context context, final Intent intent) { 230         LogUtil.v(TAG, "SmsReceiver.onReceive " + intent); 231         // On KLP+ we only take delivery of SMS messages in SmsDeliverReceiver. 232         if (PhoneUtils.getDefault().isSmsEnabled()) { 233             final String action = intent.getAction(); 234             if (OsUtil.isSecondaryUser() &amp;&amp; 235                 (Telephony.Sms.Intents.SMS_RECEIVED_ACTION.equals(action)    236                  // TODO: update this with the actual constant from Telephony 237                  "android.provider.Telephony.MMS_DOWNLOADED".equals(action))) { 238                 postNewMessageSecondaryUserNotification(); 239             } else if (!OsUtil.isAtLeastKLP()) { 240                 deliverSmsIntent(context, intent); 241             } 242         } 243     }</pre> <p data-bbox="512 760 1596 831"><a href="https://android.googlesource.com/platform/packages/apps/Messaging/+nougat-mr1-release/src/com/android/messaging/receiver/SmsReceiver.java">https://android.googlesource.com/platform/packages/apps/Messaging/+nougat-mr1-release/src/com/android/messaging/receiver/SmsReceiver.java</a></p>

**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC
	<pre>52  * Class that sends chat message via MMS. 53  * 54  * The interface emulates a blocking send similar to making an HTTP request. 55  */ 56  public class MmsSender { 57      private static final String TAG = LogUtil.BUGLE_TAG; 58 59      /** 60       * Send an MMS message. 61       * 62       * @param context Context 63       * @param messageUri The unique URI of the message for identifying it during sending 64       * @param sendReq The SendReq PDU of the message 65       * @throws MmsFailureException 66       */ 67      public static void sendMms(final Context context, final int subId, final Uri messageUri, 68                               final SendReq sendReq, final Bundle sentIntentExtras) throws MmsFailureException { 69          sendMms(context, 70                subId, 71                messageUri, 72                null /* locationUri */, 73                sendReq, 74                true /* responseImportant */, 75                sentIntentExtras); 76      }</pre> <p><a href="https://android.googlesource.com/platform/packages/apps/Messaging/+/nougat-mr1-release/src/com/android/messaging/sms/MmsSender.java">https://android.googlesource.com/platform/packages/apps/Messaging/+/nougat-mr1-release/src/com/android/messaging/sms/MmsSender.java</a></p>

**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC
	<pre> 240      * Download an MMS message. 241      * 242      * @param context Context 243      * @param contentLocation The url of the MMS message 244      * @throws MmsFailureException 245      * @throws InvalidHeaderValueException 246      */ 247      public static void downloadMms(final Context context, final int subId, 248          final String contentLocation, Bundle extras) throws MmsFailureException, 249          InvalidHeaderValueException { 250          final Uri requestUri = Uri.parse(contentLocation); 251          final Uri contentUri = MmsFileProvider.buildRawMmsUri(); 252 253          final Intent downloadedIntent = new Intent(SendStatusReceiver.MMS_DOWNLOADED_ACTION, 254              requestUri, 255              context, 256              SendStatusReceiver.class); 257          downloadedIntent.putExtra(SendMessageAction.EXTRA_CONTENT_URI, contentUri); 258          if (extras != null) { 259              downloadedIntent.putExtras(extras); 260          } 261          final PendingIntent downloadedPendingIntent = PendingIntent.getBroadcast( 262              context, 263              0 /*request code*/, 264              downloadedIntent, 265              PendingIntent.FLAG_UPDATE_CURRENT); 266 267          MmsManager.downloadMultimediaMessage(subId, context, contentLocation, contentUri, 268              downloadedPendingIntent); 269      } </pre> <p><a href="https://android.googlesource.com/platform/packages/apps/Messaging/+nougat-mr1-release/src/com/android/messaging/sms/MmsSender.java">https://android.googlesource.com/platform/packages/apps/Messaging/+nougat-mr1-release/src/com/android/messaging/sms/MmsSender.java</a></p>



**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC
	<pre> 97      * Execute an MMS HTTP request, either a POST (sending) or a GET (downloading) 98      * 99      * @param urlString The request URL, for sending it is usually the MMSC, and for downloading 100     *           it is the message URL 101     * @param pdu For POST (sending) only, the PDU to send 102     * @param method HTTP method, POST for sending and GET for downloading 103     * @param isProxySet Is there a proxy for the MMSC 104     * @param proxyHost The proxy host 105     * @param proxyPort The proxy port 106     * @param mmsConfig The MMS config to use 107     * @param userAgent The user agent header value 108     * @param uaProfUrl The UA Prof URL header value 109     * @return The HTTP response body 110     * @throws MmsHttpException For any failures 111     */ 112     public byte[] execute(String urlString, byte[] pdu, String method, boolean isProxySet, 113                          String proxyHost, int proxyPort, Bundle mmsConfig, String userAgent, String uaProfUrl) 114         throws MmsHttpException { 115         Log.d(MmsService.TAG, "HTTP: " + method + " " + Utils.redactUrlForNonVerbose(urlString) 116             + (isProxySet ? (" proxy=" + proxyHost + ":" + proxyPort) : "")) 117             + ", PDU size=" + (pdu != null ? pdu.length : 0)); 118         checkMethod(method); 119         HttpURLConnection connection = null; 120         try { 121             Proxy proxy = Proxy.NO_PROXY; 122             if (isProxySet) { 123                 proxy = new Proxy(Proxy.Type.HTTP, new InetSocketAddress(proxyHost, proxyPort)); 124             } 125             final URL url = new URL(urlString); 126             // Now get the connection 127             connection = (HttpURLConnection) url.openConnection(proxy); 128             connection.setDoInput(true); 129             connection.setConnectTimeout( 130                 mmsConfig.getInt(CarrierConfigValuesLoader.CONFIG_HTTP_SOCKET_TIMEOUT, 131                     CarrierConfigValuesLoader.CONFIG_HTTP_SOCKET_TIMEOUT_DEFAULT)); </pre> <p data-bbox="506 1252 1596 1328"> <a href="https://android.googlesource.com/platform/packages/apps/Messaging/+nougat-mr1-release/src/android/support/v7/mms/MmsHttpClient.java">https://android.googlesource.com/platform/packages/apps/Messaging/+nougat-mr1-release/src/android/support/v7/mms/MmsHttpClient.java</a> </p>

**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC
	<pre> 132 // ----- COMMON HEADERS ----- 133 // Header: Accept 134 connection.setRequestProperty(HEADER_ACCEPT, HEADER_VALUE_ACCEPT); 135 // Header: Accept-Language 136 connection.setRequestProperty( 137     HEADER_ACCEPT_LANGUAGE, getCurrentAcceptLanguage(Locale.getDefault())); 138 // Header: User-Agent 139 Log.i(MmsService.TAG, "HTTP: User-Agent=" + userAgent); 140 connection.setRequestProperty(HEADER_USER_AGENT, userAgent); 141 // Header: x-wap-profile 142 final String uaProfUrlTagName = mmsConfig.getString( 143     CarrierConfigValuesLoader.CONFIG_UA_PROF_TAG_NAME, 144     CarrierConfigValuesLoader.CONFIG_UA_PROF_TAG_NAME_DEFAULT); 145 if (uaProfUrl != null) { 146     Log.i(MmsService.TAG, "HTTP: UaProfUrl=" + uaProfUrl); 147     connection.setRequestProperty(uaProfUrlTagName, uaProfUrl); 148 } 149 // Add extra headers specified by mms_config.xml's httpparams 150 addExtraHeaders(connection, mmsConfig); 151 // Different stuff for GET and POST 152 if (METHOD_POST.equals(method)) { 153     if (pdu == null    pdu.length &lt; 1) { 154         Log.e(MmsService.TAG, "HTTP: empty pdu"); 155         throw new MmsHttpException(0/*statusCode*/, "Sending empty PDU"); 156     } 157     connection.setDoOutput(true); 158     connection.setRequestMethod(METHOD_POST); 159     if (mmsConfig.getBoolean( 160         CarrierConfigValuesLoader.CONFIG_SUPPORT_HTTP_CHARSET_HEADER, 161         CarrierConfigValuesLoader.CONFIG_SUPPORT_HTTP_CHARSET_HEADER_DEFAULT)) { 162         connection.setRequestProperty(HEADER_CONTENT_TYPE, 163             HEADER_VALUE_CONTENT_TYPE_WITH_CHARSET); 164     } else { 165         connection.setRequestProperty(HEADER_CONTENT_TYPE, 166             HEADER_VALUE_CONTENT_TYPE_WITHOUT_CHARSET); </pre> <p data-bbox="506 1289 1596 1360"> <a href="https://android.googlesource.com/platform/packages/apps/Messaging/+nougat-mr1-release/src/android/support/v7/mms/MmsHttpClient.java">https://android.googlesource.com/platform/packages/apps/Messaging/+nougat-mr1-release/src/android/support/v7/mms/MmsHttpClient.java</a> </p>

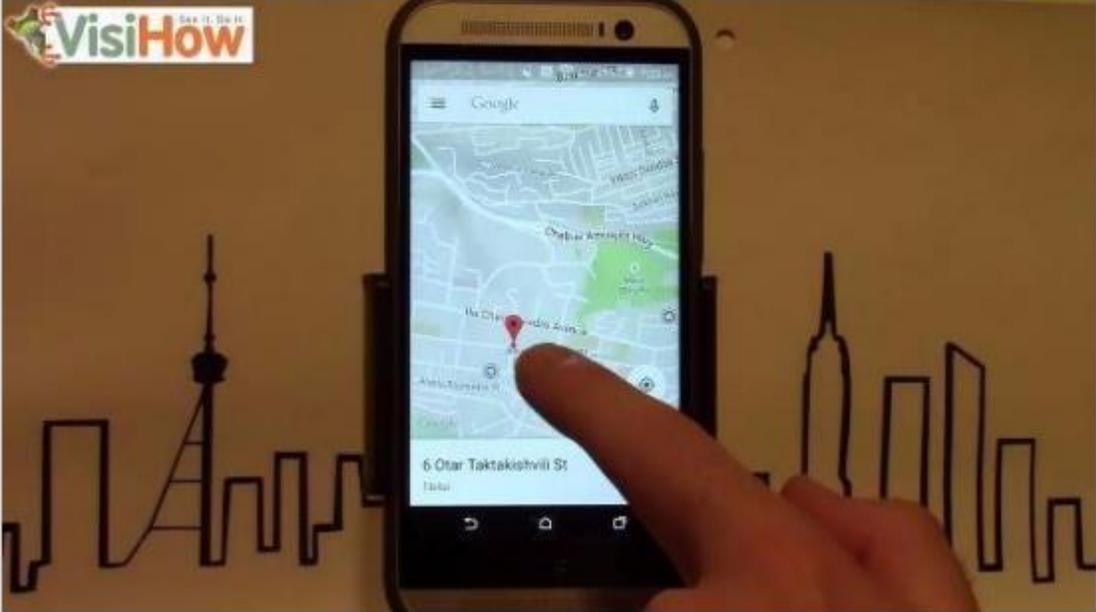
**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC
	<pre> 167     } 168     if (Log.isLoggable(MmsService.TAG, Log.VERBOSE)) { 169         logHttpHeaders(connection.getRequestProperties()); 170     } 171     connection.setFixedLengthStreamingMode(pdu.length); 172     // Sending request body 173     final OutputStream out = 174         new BufferedOutputStream(connection.getOutputStream()); 175     out.write(pdu); 176     out.flush(); 177     out.close(); 178 } else if (METHOD_GET.equals(method)) { 179     if (Log.isLoggable(MmsService.TAG, Log.VERBOSE)) { 180         logHttpHeaders(connection.getRequestProperties()); 181     } 182     connection.setRequestMethod(METHOD_GET); 183 } 184 // Get response 185 final int responseCode = connection.getResponseCode(); 186 final String responseMessage = connection.getResponseMessage(); 187 Log.d(MmsService.TAG, "HTTP: " + responseCode + " " + responseMessage); 188 if (Log.isLoggable(MmsService.TAG, Log.VERBOSE)) { 189     logHttpHeaders(connection.getHeaderFields()); 190 } 191 if (responseCode / 100 != 2) { 192     throw new MmsHttpException(responseCode, responseMessage); 193 } 194 final InputStream in = new BufferedInputStream(connection.getInputStream()); 195 final ByteArrayOutputStream byteOut = new ByteArrayOutputStream(); 196 final byte[] buf = new byte[4096]; 197 int count = 0; 198 while ((count = in.read(buf)) &gt; 0) { 199     byteOut.write(buf, 0, count); 200 } 201 in.close(); 202 final byte[] responseBody = byteOut.toByteArray(); 203 Log.d(MmsService.TAG, "HTTP: response size=" 204     + (responseBody != null ? responseBody.length : 0)); 205 return responseBody; </pre> <p><a href="https://android.googlesource.com/platform/packages/apps/Messaging/+nougat-mr1-">https://android.googlesource.com/platform/packages/apps/Messaging/+nougat-mr1-</a></p>

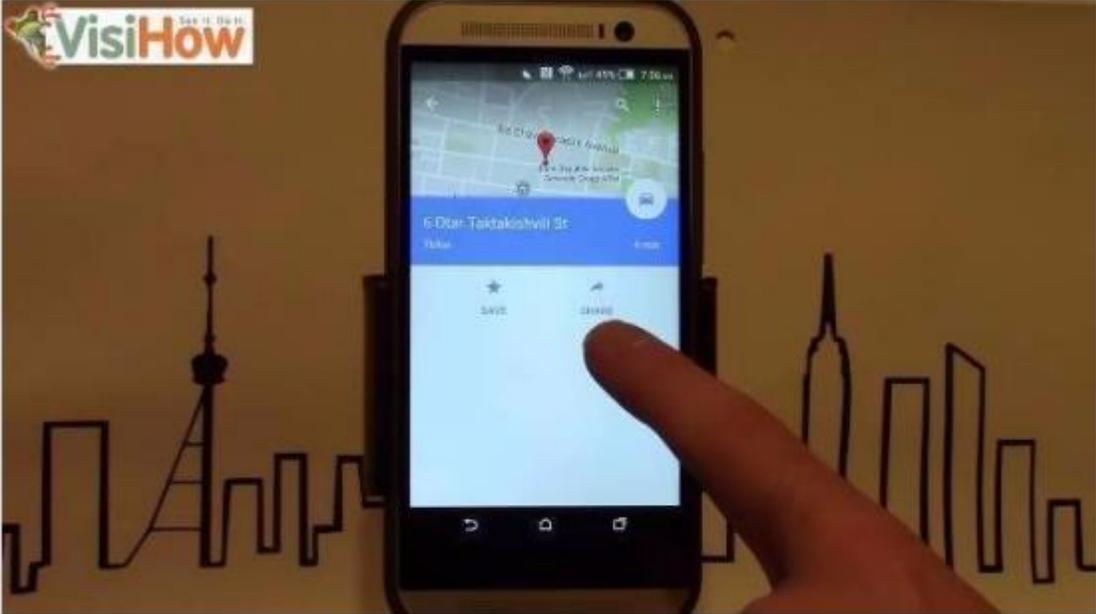
**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC
	<a href="https://source.android.com/support/v7/mms/MmsHttpClient.java">release/src/android/support/v7/mms/MmsHttpClient.java</a>
<p>[1C] receiving respective IP-based responses to the SMS messages, wherein the IP-based responses to the SMS messages include location information of the respective second devices;</p>	<p>HTC infringes directly and/or indirectly by performing, inducing others to perform, and/or contributing to the performance of: receiving respective IP-based responses to the SMS messages, wherein the IP-based responses to the SMS messages include location information of the respective second devices.</p> <p>For example, the HTC accused devices running Maps are configured to receive IP-based communications from the respective second devices that include location information of the second devices.</p> <p><b>Share Locations from Google Maps on HTC One M8</b></p>

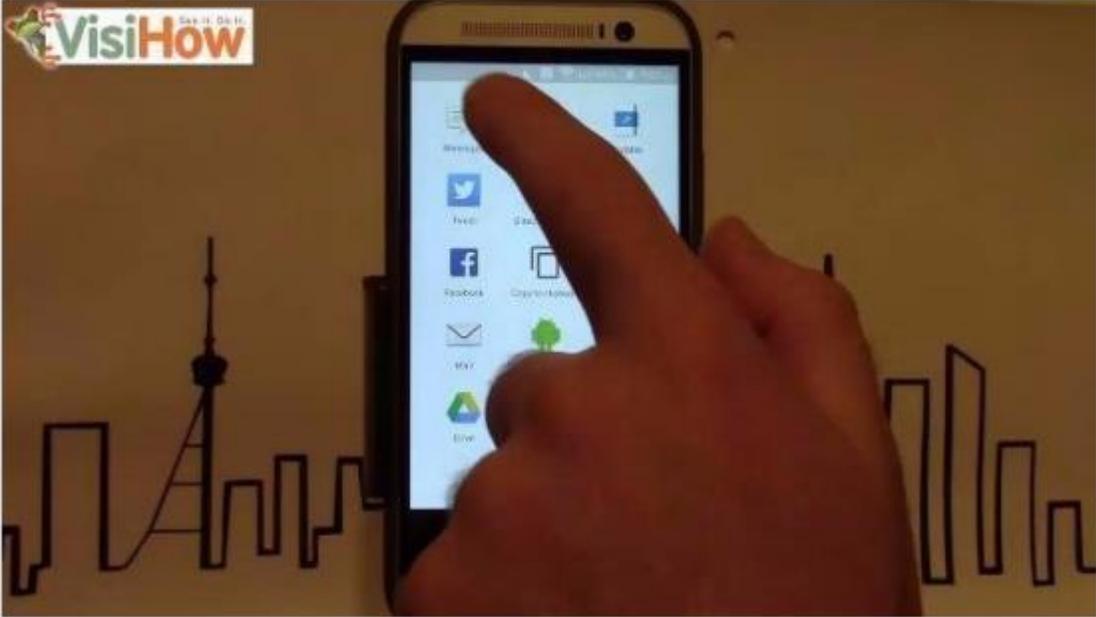
## Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products

US9408055B2	HTC
	<p data-bbox="531 233 829 256"><b>Place a pin on the map.</b></p> <p data-bbox="531 272 1640 380">We can do this anywhere in the entire world, anywhere, on any map that we find in the Google Maps application. We are going to choose a location, press down on the screen, and hold down until we can see that a red pin has been dropped.</p> 

## Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products

US9408055B2	HTC
	<p data-bbox="527 240 1207 310"><b>We have two options:</b> "SAVE" and "SHARE". Press down on the word "SHARE".</p>  <p data-bbox="527 1057 1633 1208"><b>A menu appears with all of the different ways that we can share this location.</b> We can share this location via messages, Bluetooth, Twitter, Facebook, mail, Gmail and Viber. If you have other applications, such as WhatsApp, or any other chat or email applications, you will be able to use them as well and will see them here.</p>

### Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products

US9408055B2	HTC
	<p data-bbox="527 237 856 264"><b>Let's choose "Messages".</b></p> <p data-bbox="527 277 1633 347">This option is located on the top left-hand corner of the screen. All options will have the same basic processes. They will ask us to send the location. Press down on the "Messages" icon.</p> 

## Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products

US9408055B2	HTC
	<p data-bbox="520 233 1260 256"><b>Press the box next to the contact who will be the recipient.</b></p> <p data-bbox="520 269 1549 292">A checkmark will appear in the box. We can select multiple contacts as well if we desire.</p>  <p data-bbox="520 1045 911 1068"><b>Press on the word "DONE (1)".</b></p> <p data-bbox="520 1081 1629 1153">This icon is located in the bottom right-hand corner of the screen. The bracketed number indicates the number of contacts we selected.</p> <p data-bbox="520 1166 1407 1188"><a href="http://visihow.com/Share_Locations_from_Google_Maps_on_HTC_One_M8">http://visihow.com/Share_Locations_from_Google_Maps_on_HTC_One_M8</a></p>



## Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products

US9408055B2	HTC
	<p data-bbox="533 240 884 282"><b>Send your location</b></p> <ol data-bbox="533 305 947 505" style="list-style-type: none"><li>1. Open the Android Messages app .</li><li>2. Open or start a conversation.</li><li>3. Tap Attach + .</li><li>4. Tap Location on .</li><li>5. To send your location, tap Send .</li></ol> <p data-bbox="520 526 1535 553"><a href="https://support.google.com/pixelphone/answer/6159880?hl=en&amp;ref_topic=6211804">https://support.google.com/pixelphone/answer/6159880?hl=en&amp;ref_topic=6211804</a></p> <div data-bbox="520 594 1255 1321"></div> <p data-bbox="520 1328 1514 1356"><a href="https://blog.google/topics/rcs/delivering-rcs-messaging-android-users-worldwide/">https://blog.google/topics/rcs/delivering-rcs-messaging-android-users-worldwide/</a></p>

**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC
	<div data-bbox="533 337 917 375" data-label="Section-Header"> <p>Share a location or place</p> </div> <div data-bbox="554 418 743 441" data-label="Section-Header"> <p>Share your location</p> </div> <div data-bbox="588 470 1104 594" data-label="List-Group"> <ol style="list-style-type: none"> <li>1. On your Android phone or tablet, open the Hangouts app</li> <li>2. Open a conversation.</li> <li>3. Tap Location</li> <li>4. Tap <b>Select this location</b> &gt; <b>Select</b>.</li> </ol> </div> <div data-bbox="554 672 690 696" data-label="Section-Header"> <p>Share a place</p> </div> <div data-bbox="588 724 1104 883" data-label="List-Group"> <ol style="list-style-type: none"> <li>1. On your Android phone or tablet, open the Hangouts app</li> <li>2. Open a conversation.</li> <li>3. Tap Location &gt; Search</li> <li>4. Type in a location or address.</li> <li>5. Tap <b>Select</b>.</li> </ol> </div> <div data-bbox="514 915 1581 1015" data-label="Text"> <p><a href="https://support.google.com/hangouts/answer/3115410?visit_id=1-636271867303650973-2491837168&amp;rd=1&amp;co=GENIE.Platform%3DAndroid&amp;oco=1">https://support.google.com/hangouts/answer/3115410?visit_id=1-636271867303650973-2491837168&amp;rd=1&amp;co=GENIE.Platform%3DAndroid&amp;oco=1</a>  <a href="https://play.google.com/store/apps/details?id=com.google.android.talk&amp;hl=en">https://play.google.com/store/apps/details?id=com.google.android.talk&amp;hl=en</a></p> </div> <div data-bbox="506 1052 1906 1339" data-label="Text"> <p><b>Regarding Google Maps</b>, Google Latitude, Google Plus, Google Hangouts, Google Messages, Google Allo, Google Duo, Google Chrome, and Android Messenger, the Accused Products require a user to join the corresponding network by signing-in to the device with an identifier (e.g., Google Account). When the signed-in user enables Google Maps Share Location and Location Services on the first device, the user shares its location and the first device’s location is sent to a server. When the user enables sharing to one or more contacts (of respective devices) and the one or more contacts enable sharing their location to the user of the first device, the user of the first device receives the locations of the one or more contacts.</p> </div> <div data-bbox="1346 224 1734 914" data-label="Image"> <p>The image is a screenshot of a mobile messaging application interface. At the top, the contact name is 'Mandy Jiang' with the status 'Active now'. Below the name is a question 'Where are you?'. The response is a location share card for 'Google Kirkland' with the address '747 8th St S, Kirkland, WA 980...'. The card includes a map snippet showing the location and a cartoon character on a skateboard. At the bottom of the screen, there is a 'Write a message' input field and a row of icons for attachments, camera, voice, emojis, and a green send button.</p> </div>

**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC
	<p>The first device’s participation in the group is based on receiving the message from the second device, i.e. a message indicating that the second device is sharing its location.</p> <p>By participating in the Maps location sharing functionality, the device sends location information to a server (e.g., a network server provided by an ISP such as AT&amp;T and/or a server running Google’s services). The device also receives location information from the server indicating the location of other devices that are sharing location information via Maps.</p> <p><b><u>Further regarding Google Maps</u></b>, Google Latitude, Google Plus, Google Hangouts, Google Messages, Google Allo, Google Duo, Google Chrome, and Android Messenger, the Accused Products alternatively require a user to send a request containing the identifier (e.g., Google Account). When the signed-in user enables Google Maps Share Location and Location Services on the first device, the user shares its location and the first device’s location is sent to a server. When the user sends a message to another contact through Google Maps, Google Messages, and/or another means from within the Google Maps application, the message including location information are sent to a server before transmission to the intended contact. When one or more contacts enable sharing their location to the user of the first device, or alternatively send a message containing location information, or alternatively accept a request to share their location with the first user, the user of the first device receives the locations of the one or more contacts.</p> <p><b><u>Exemplary Support for Google Maps:</u></b></p>

## Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products

US9408055B2	HTC
	<p data-bbox="548 240 968 261">COMPUTER <b>ANDROID</b> IPHONE &amp; IPAD</p> <hr data-bbox="533 289 1566 293"/> <h3 data-bbox="533 347 1024 383">If they have a Google Account</h3> <ol data-bbox="533 402 1419 699" style="list-style-type: none"><li data-bbox="533 402 1220 423">1. If you haven't already, add their Gmail address to your <a href="#">Google Contacts</a>.</li><li data-bbox="533 440 1419 461">2. On your Android phone or tablet, open the Google Maps app  and sign in. <a href="#">Learn how to sign in.</a></li><li data-bbox="533 477 1031 498">3. Tap Menu  &gt; <b>Location sharing</b> &gt; Add People .</li><li data-bbox="533 514 1003 535">4. Choose how long you want to share your location.</li><li data-bbox="533 552 1140 610">5. Tap <b>Select People</b>.<ul data-bbox="569 586 1140 610" style="list-style-type: none"><li data-bbox="569 586 1140 610">• If you're asked about your contacts, give Google Maps access.</li></ul></li><li data-bbox="533 634 884 656">6. Choose who you want to share with.</li><li data-bbox="533 672 663 693">7. Tap <b>Share</b>.</li></ol> <h3 data-bbox="533 753 1108 789">If they don't have a Google Account</h3> <ol data-bbox="533 813 1560 938" style="list-style-type: none"><li data-bbox="533 813 1419 834">1. On your Android phone or tablet, open the Google Maps app  and sign in. <a href="#">Learn how to sign in.</a></li><li data-bbox="533 850 1031 872">2. Tap Menu  &gt; <b>Location sharing</b> &gt; Add People .</li><li data-bbox="533 888 1560 938">3. Tap More  &gt; <b>Copy to clipboard</b>. People with this link can see your location for as long as you choose, up to 72 hours.</li></ol> <h3 data-bbox="533 987 869 1023">Share using another app</h3> <p data-bbox="533 1040 1205 1062">You can also share through messaging apps. Tap More  &gt; select an app.</p> <h3 data-bbox="533 1122 743 1157">Stop sharing</h3> <ol data-bbox="533 1182 1205 1279" style="list-style-type: none"><li data-bbox="533 1182 842 1203">1. Open the Google Maps app .</li><li data-bbox="533 1219 869 1240">2. Tap Menu  &gt; <b>Location sharing</b>.</li><li data-bbox="533 1256 1205 1279">3. Next to the person with whom you want to stop sharing, tap Remove .</li></ol> <p data-bbox="512 1328 1703 1360"><a href="https://support.google.com/maps/answer/7326816?co=GENIE.Platform%3DAndroid&amp;oco=1">https://support.google.com/maps/answer/7326816?co=GENIE.Platform%3DAndroid&amp;oco=1</a></p>

## Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products

US9408055B2	HTC
	<h3>Share your E.T.A</h3> <p>After you start your drive, you can share your destination, estimated arrival time, and where you are on the route.</p> <ol style="list-style-type: none"><li>1. Open the Google Maps app .</li><li>2. Set a driving destination. <a href="#">Learn how to navigate to a place.</a></li><li>3. After you start navigation, tap More  &gt; <b>Share trip progress.</b></li><li>4. Choose a person from the list.</li><li>5. Tap <b>Share.</b></li><li>6. Location Sharing will stop when you reach your destination or stop navigating.</li></ol> <ul style="list-style-type: none"><li>• To stop sharing before you arrive, tap More  &gt; <b>Stop sharing.</b></li></ul> <h3>See where someone is</h3> <p>If someone shares their location with you, you can see them on the map.</p> <ol style="list-style-type: none"><li>1. Open the Google Maps app .</li><li>2. Tap Menu  &gt; <b>Location sharing.</b></li><li>3. Choose someone.</li></ol> <ul style="list-style-type: none"><li>• To see an updated location, tap on a friend's icon &gt; More  &gt; <b>Refresh.</b></li></ul> <h3>Stop seeing someone's location</h3> <ol style="list-style-type: none"><li>1. Open the Google Maps app .</li><li>2. On the map, tap their icon.</li><li>3. At the bottom, tap More .</li><li>4. To temporarily hide someone, tap <b>Hide from map.</b> You can stop hiding them at any time.</li></ol> <p><b>Note:</b> You can stop someone's location from ever appearing on your map. <a href="#">Learn how to block another person's account.</a></p>

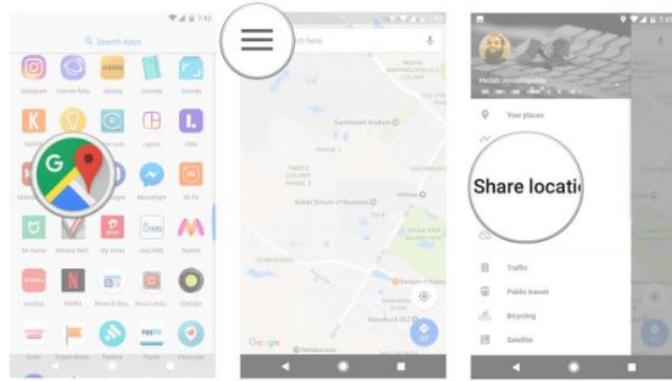
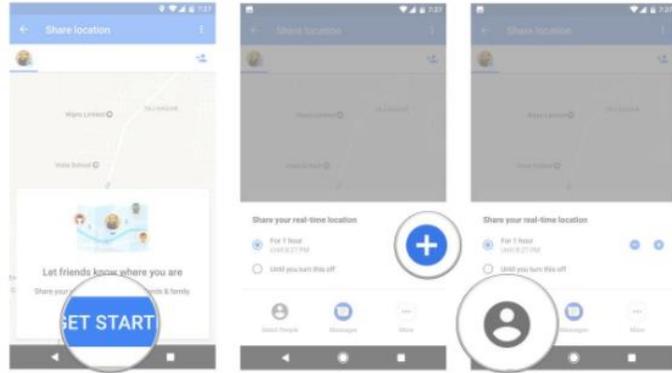
## Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products

US9408055B2	HTC
	<p data-bbox="510 237 1703 266"><a href="https://support.google.com/maps/answer/7326816?co=GENIE.Platform%3DAndroid&amp;oco=1">https://support.google.com/maps/answer/7326816?co=GENIE.Platform%3DAndroid&amp;oco=1</a></p> <h3 data-bbox="510 310 999 358">Create a list of places</h3> <p data-bbox="510 375 1346 399">In Google Maps, you can create a list of places, like your favorite places or places you want to visit.</p> <p data-bbox="537 472 932 493">COMPUTER <b>ANDROID</b> IPHONE &amp; IPAD</p> <hr data-bbox="510 516 1371 521"/> <h3 data-bbox="510 570 768 602">Make a new list</h3> <ol data-bbox="510 626 1087 786" style="list-style-type: none"><li>1. On your Android phone or tablet, open the Google Maps app .</li><li>2. Tap Menu  &gt; <b>Your places</b> &gt; <b>Saved</b>.</li><li>3. In the bottom right, tap Add .</li><li>4. Enter a name and description.</li><li>5. Tap <b>Save</b>.</li></ol> <h3 data-bbox="510 837 842 870">Save a place to a list</h3> <ol data-bbox="510 894 978 1053" style="list-style-type: none"><li>1. Open the Google Maps app .</li><li>2. Search for a place or tap it on the map.</li><li>3. At the bottom, tap the place's name or address.</li><li>4. Tap <b>Save</b>.</li><li>5. Choose a list. To create a new list, tap <b>New list</b> .</li></ol> <h3 data-bbox="510 1105 732 1138">See your lists</h3> <ol data-bbox="510 1162 879 1224" style="list-style-type: none"><li>1. Open the Google Maps app .</li><li>2. Tap Menu  &gt; <b>Your places</b> &gt; <b>Saved</b>.</li></ol> <p data-bbox="510 1260 1902 1338"><a href="https://support.google.com/maps/answer/7280933?hl=en&amp;ref_topic=7301134&amp;co=GENIE.Platform%3DAndroid&amp;oco=1">https://support.google.com/maps/answer/7280933?hl=en&amp;ref_topic=7301134&amp;co=GENIE.Platform%3DAndroid&amp;oco=1</a></p>

## Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products

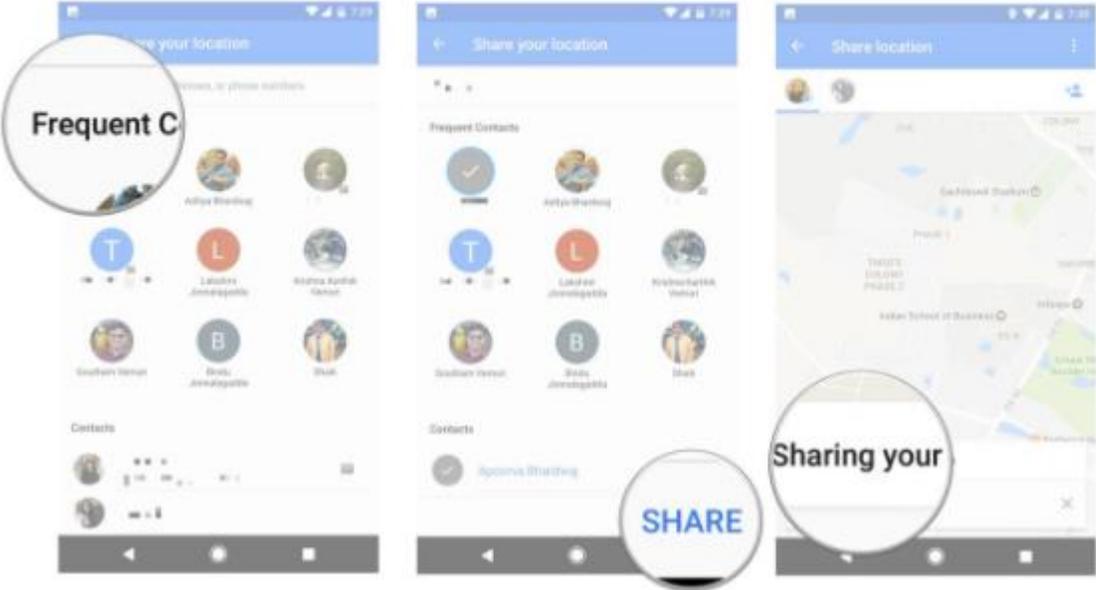
US9408055B2	HTC
	<h3 data-bbox="541 240 877 280">Hide or share lists</h3> <p data-bbox="541 310 909 334"><b>Note:</b> You can't share starred places.</p> <ol data-bbox="554 363 1682 630" style="list-style-type: none"><li data-bbox="554 363 890 388">1. Open the Google Maps app .</li><li data-bbox="554 407 968 431">2. Tap Menu  &gt; <b>Your places</b> &gt; <b>Saved</b>.</li><li data-bbox="554 451 1682 630">3. Next to the list you want to share, tap More  &gt; choose an option:<ul data-bbox="583 488 1682 630" style="list-style-type: none"><li data-bbox="583 488 1444 513">• <b>Hide/Show on your map:</b> Display or hide your saved places when looking at the map.</li><li data-bbox="583 532 1058 557">• <b>Share list:</b> Allow others to see your saved list.</li><li data-bbox="583 576 1682 630">• <b>Sharing options:</b> You can make your list public, private, or shared. To let anyone with the link see your list, tap <b>Shared</b>. To let anyone find and follow your list, tap <b>Public</b>.</li></ul></li></ol> <h3 data-bbox="541 699 768 740">Follow a list</h3> <p data-bbox="541 764 1728 821">If you follow a list made by someone else, their saved places will show up in Your Places. The places will also appear as suggested locations in Google Maps.</p> <h3 data-bbox="541 878 915 919">Follow a list using a link</h3> <ol data-bbox="554 938 1356 1044" style="list-style-type: none"><li data-bbox="554 938 961 963">1. Tap on the link you received to open it.</li><li data-bbox="554 982 1272 1006">2. Tap <b>Follow</b>. This list will now be added to the group of lists you follow.</li><li data-bbox="554 1026 1356 1044">3. <b>Optional:</b> To unfollow a list someone shared with you, tap the list &gt; <b>Following</b>.</li></ol> <h3 data-bbox="541 1101 926 1141">See lists made by others</h3> <p data-bbox="541 1157 1335 1182">If a user has any Google Maps lists that were made public, you can follow them.</p> <ol data-bbox="554 1211 1136 1317" style="list-style-type: none"><li data-bbox="554 1211 1136 1235">1. Tap on the name of a user whose list you want to follow.</li><li data-bbox="554 1255 680 1279">2. Tap <b>Lists</b>.</li><li data-bbox="554 1299 1136 1317">3. Tap on the list you want to follow &gt; More  &gt; <b>Follow</b>.</li></ol> <p data-bbox="512 1369 1900 1393"><a href="https://support.google.com/maps/answer/7280933?hl=en&amp;ref_topic=7301134&amp;co=GENIE.Platform%3DAn">https://support.google.com/maps/answer/7280933?hl=en&amp;ref_topic=7301134&amp;co=GENIE.Platform%3DAn</a></p>

## Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products

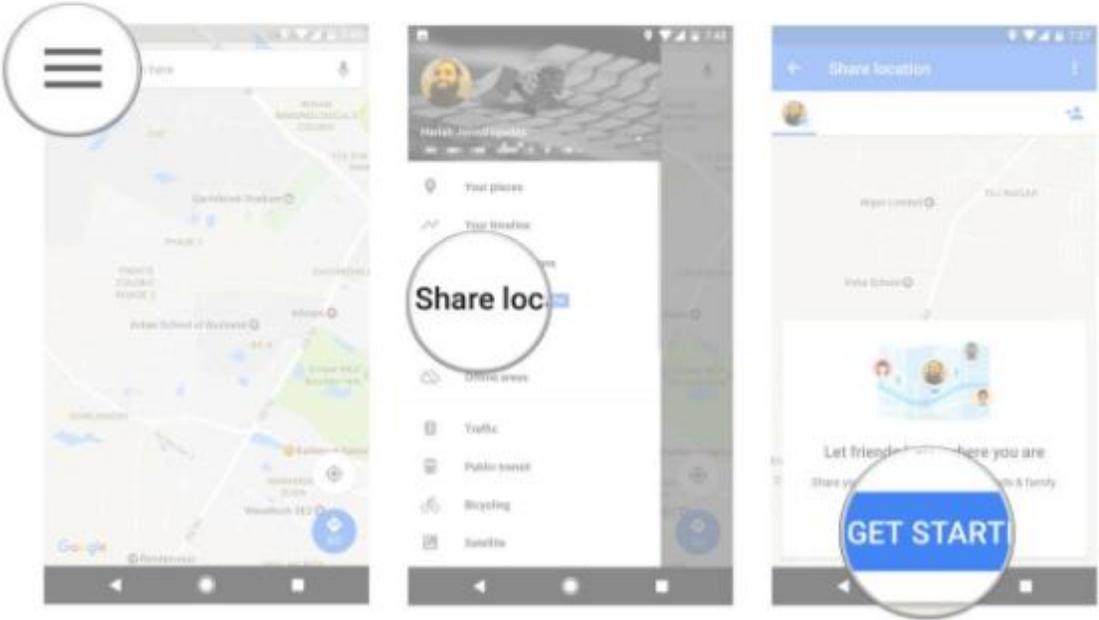
US9408055B2	HTC
	<p data-bbox="514 235 682 267"><a href="#">droid&amp;oco=1</a></p> <h3 data-bbox="514 308 1144 341">How to share your location in Google Maps</h3> <ol data-bbox="514 365 1134 462" style="list-style-type: none"><li>1. Open Google Maps from the app drawer or the home screen.</li><li>2. Tap the hamburger menu (the three horizontal lines) on the top left corner of the screen.</li><li>3. Select Share location.</li></ol>  <ol data-bbox="514 893 1134 966" style="list-style-type: none"><li>4. Tap Get Started.</li><li>5. Use the + Icon to select a time period or select the Until you turn this off setting to share your location indefinitely.</li><li>6. Tap Select People.</li></ol> 



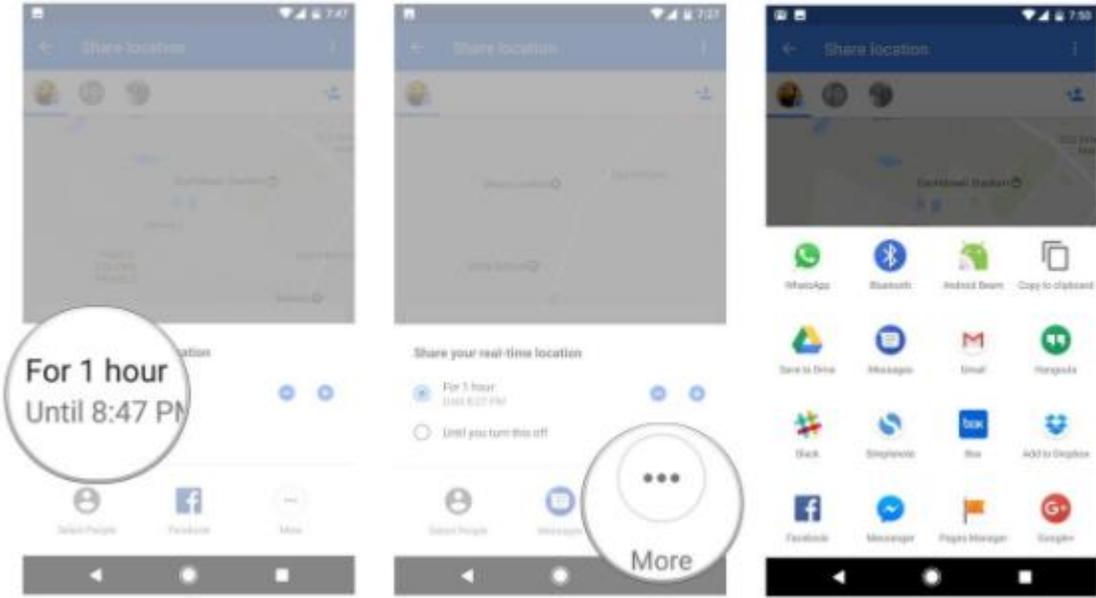
## Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products

US9408055B2	HTC
	<p data-bbox="512 237 1356 266"><a href="https://www.androidcentral.com/how-share-location-google-maps">https://www.androidcentral.com/how-share-location-google-maps</a></p> <ol data-bbox="527 326 1577 496" style="list-style-type: none"><li data-bbox="527 326 1577 383">7. You'll see a list of your frequent contacts at the top, along with a full list of contacts. Pick the <b>contacts</b> by tapping their name.</li><li data-bbox="527 410 1457 440">8. Once you've selected the contacts you want to share your location to, tap <b>Share</b>.</li><li data-bbox="527 467 1419 496">9. You'll see a message saying that the selected contact can view your location.</li></ol>  <p data-bbox="512 1235 1356 1265"><a href="https://www.androidcentral.com/how-share-location-google-maps">https://www.androidcentral.com/how-share-location-google-maps</a></p>

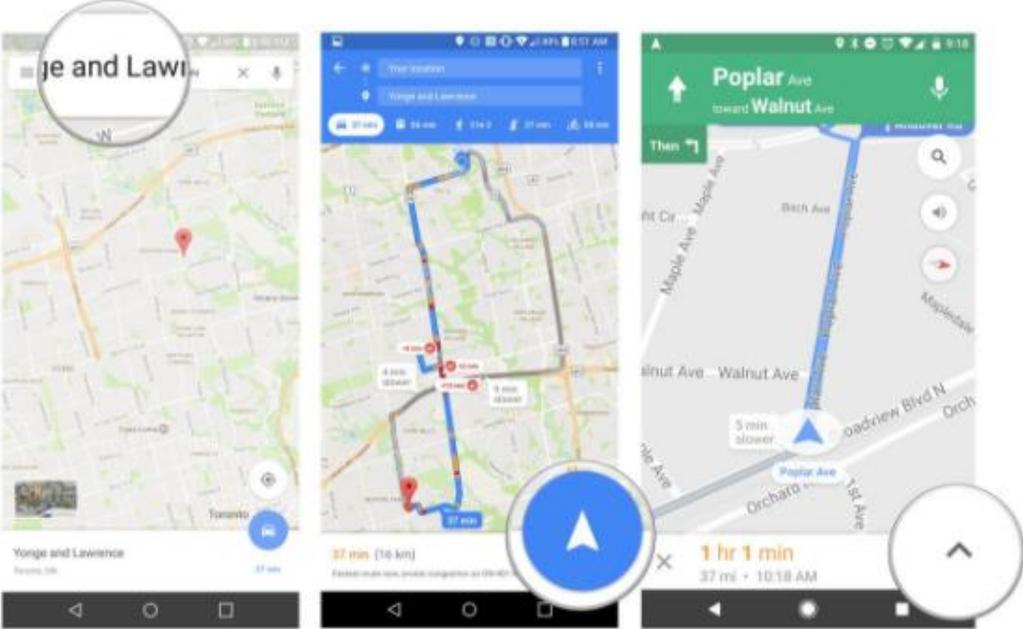
## Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products

US9408055B2	HTC
	<h3 data-bbox="527 245 1255 289">How to create a shareable link</h3> <p data-bbox="527 334 1461 358">You can also create a link and use it to share your location easily. Here's how to do it:</p> <ol data-bbox="527 412 1234 548" style="list-style-type: none"><li data-bbox="527 412 1234 440">1. Tap the hamburger menu on the top left corner of the screen.</li><li data-bbox="527 467 800 495">2. Select Share location.</li><li data-bbox="527 522 737 550">3. Tap Get Started.</li></ol>  <p data-bbox="512 1263 1356 1295"><a href="https://www.androidcentral.com/how-share-location-google-maps">https://www.androidcentral.com/how-share-location-google-maps</a></p>

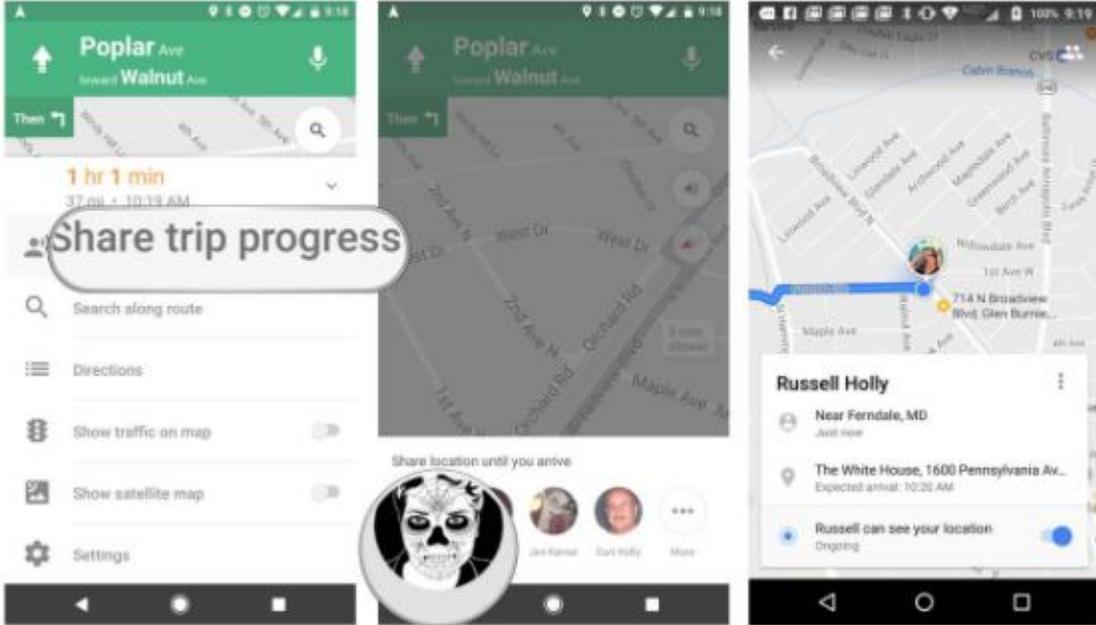
## Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products

US9408055B2	HTC
	<p data-bbox="520 245 1213 272">4. Select the <b>amount of time</b> you want to share your location.</p> <p data-bbox="520 302 680 329">5. Tap <b>More</b>.</p> <p data-bbox="520 358 1633 418">6. Select <b>your app of choice</b> to create and send a <b>unique URL</b> that broadcasts your current location. You can email it, send the link via Messenger, or even tweet it to the intended recipient.</p> <div data-bbox="569 459 1665 1057"></div> <p data-bbox="506 1117 1360 1149"><a href="https://www.androidcentral.com/how-share-location-google-maps">https://www.androidcentral.com/how-share-location-google-maps</a></p>

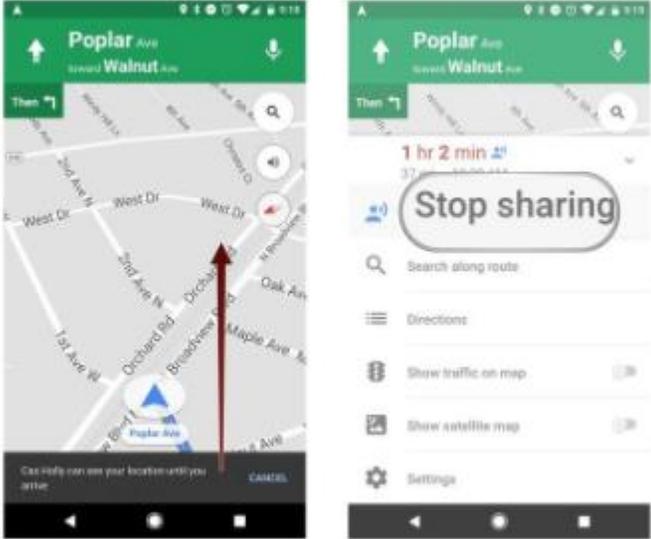
**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC
	<p data-bbox="527 240 1428 337"><b>How to share your navigation directions while you walk, drive or transit</b></p> <p data-bbox="527 375 1556 464">One of the best ways to share your location with someone is during a drive, walk or transit. If you're meeting a friend or family member somewhere, or navigating towards their home, Google Maps lets you share your location with them for the duration of the trip. It's magic!</p> <ol data-bbox="527 513 1396 643" style="list-style-type: none"> <li data-bbox="527 513 976 537">1. In the <b>search bar</b> enter your destination.</li> <li data-bbox="527 561 1396 586">2. Pick your navigation type (drive, transit, walk) and press the blue <b>navigate</b> button.</li> <li data-bbox="527 610 1396 634">3. Tap the arrow next to the <b>time-to-destination</b> number at the bottom of the screen.</li> </ol>  <p data-bbox="512 1360 1356 1393"><a href="https://www.androidcentral.com/how-share-location-google-maps">https://www.androidcentral.com/how-share-location-google-maps</a></p>

## Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products

US9408055B2	HTC
	<p data-bbox="527 240 835 267">4. Tap Share trip progress.</p> <p data-bbox="527 297 1150 324">5. Choose one or more contacts to share trip progress.</p>  <p data-bbox="537 1027 1339 1055">You can also stop sharing your location with someone before a trip ends.</p> <p data-bbox="512 1097 1356 1125"><a href="https://www.androidcentral.com/how-share-location-google-maps">https://www.androidcentral.com/how-share-location-google-maps</a></p>

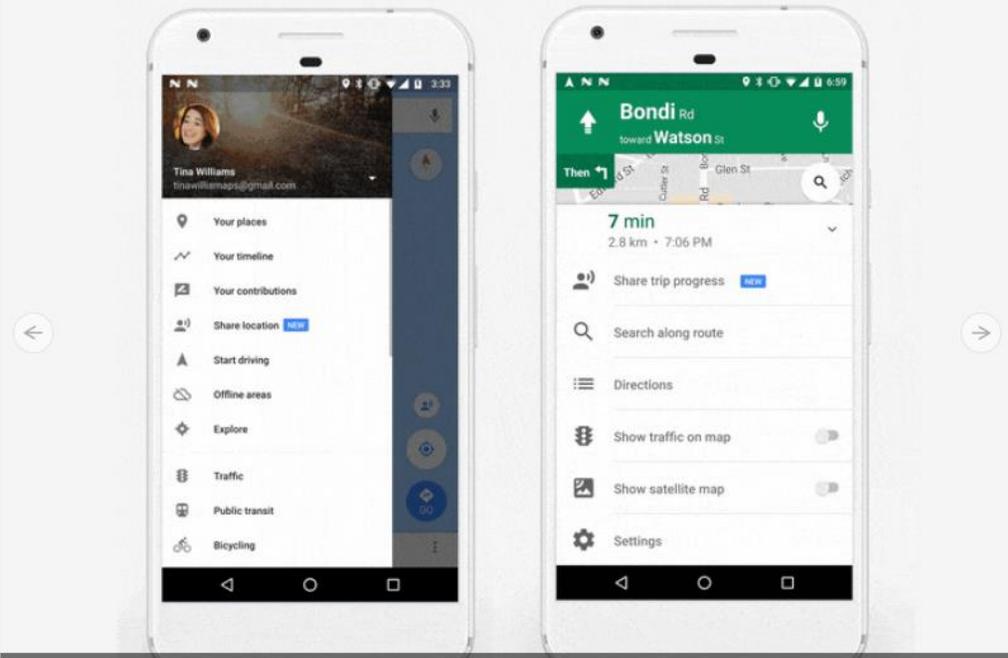
## Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products

US9408055B2	HTC
	<ol style="list-style-type: none"><li data-bbox="533 245 1465 272">1. Tap the arrow next to the <b>time-to-destination</b> number at the bottom of the screen.</li><li data-bbox="533 302 768 329">2. Tap <b>Stop sharing</b>.</li></ol> <div data-bbox="747 383 1398 922"></div> <p data-bbox="541 976 638 1003">That's it!</p> <p data-bbox="541 1045 1608 1073">Are you excited that location sharing is back in Google Maps? How often do you use the feature?</p> <p data-bbox="512 1117 1356 1149"><a href="https://www.androidcentral.com/how-share-location-google-maps">https://www.androidcentral.com/how-share-location-google-maps</a></p> <p data-bbox="512 1187 1419 1219">As shown below, a group may also be defined within Google Contacts.</p>

## Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products

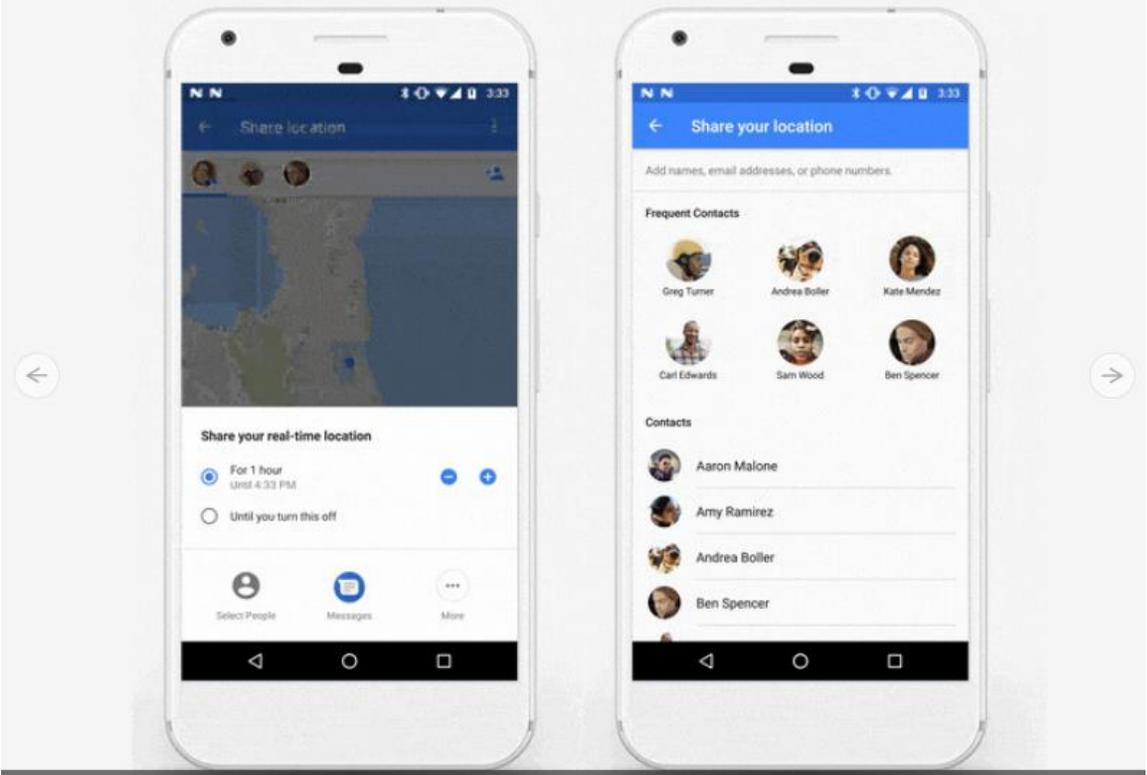
US9408055B2	HTC
	<p data-bbox="541 237 894 277"><b>See your contacts</b></p> <ol data-bbox="552 305 968 378" style="list-style-type: none"><li data-bbox="552 305 968 337">1. Open your device's Contacts app .</li><li data-bbox="552 350 730 378">2. Tap Menu .</li></ol> <ul data-bbox="546 410 1738 605" style="list-style-type: none"><li data-bbox="546 410 1115 438">• <b>See contacts by label:</b> Choose a label from the list.</li><li data-bbox="546 454 1367 482">• <b>See contacts for another account:</b> Tap Down arrow  &gt; pick an account.</li><li data-bbox="546 498 1213 526">• <b>See the contacts for all your accounts:</b> Choose <b>All contacts</b>.</li></ul> <p data-bbox="569 537 1738 565"><b>Tip:</b> If you have multiple contacts with the same information, the information will be grouped into one contact.</p> <ul data-bbox="546 581 1360 605" style="list-style-type: none"><li data-bbox="546 581 1360 605">• <b>See your Google Account contacts on the web:</b> Go to <a href="#">Google Contacts</a> .</li></ul> <p data-bbox="510 667 1535 699"><a href="https://support.google.com/android/answer/6118731?hl=en&amp;ref_topic=6118711">https://support.google.com/android/answer/6118731?hl=en&amp;ref_topic=6118711</a></p> <p data-bbox="541 740 894 781"><b>Label your contacts</b></p> <p data-bbox="541 805 982 833">You can group contacts together using labels.</p> <ol data-bbox="552 862 930 967" style="list-style-type: none"><li data-bbox="552 862 930 894">1. Open your device's Contacts app .</li><li data-bbox="552 902 863 930">2. Tap Menu  &gt; <b>Create label</b>.</li><li data-bbox="552 938 873 967">3. Enter a label name and tap <b>Ok</b>.</li></ol> <ul data-bbox="546 992 1717 1065" style="list-style-type: none"><li data-bbox="546 992 1234 1019">• <b>Add one contact to a label:</b> Tap Add contact  &gt; choose a contact.</li><li data-bbox="546 1036 1717 1065">• <b>Add multiple contacts to a label:</b> Tap Add contact  &gt; touch and hold a contact &gt; tap the other contacts &gt; tap <b>Add</b>.</li></ul> <p data-bbox="510 1105 1535 1138"><a href="https://support.google.com/android/answer/6118731?hl=en&amp;ref_topic=6118711">https://support.google.com/android/answer/6118731?hl=en&amp;ref_topic=6118711</a></p>

## Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products

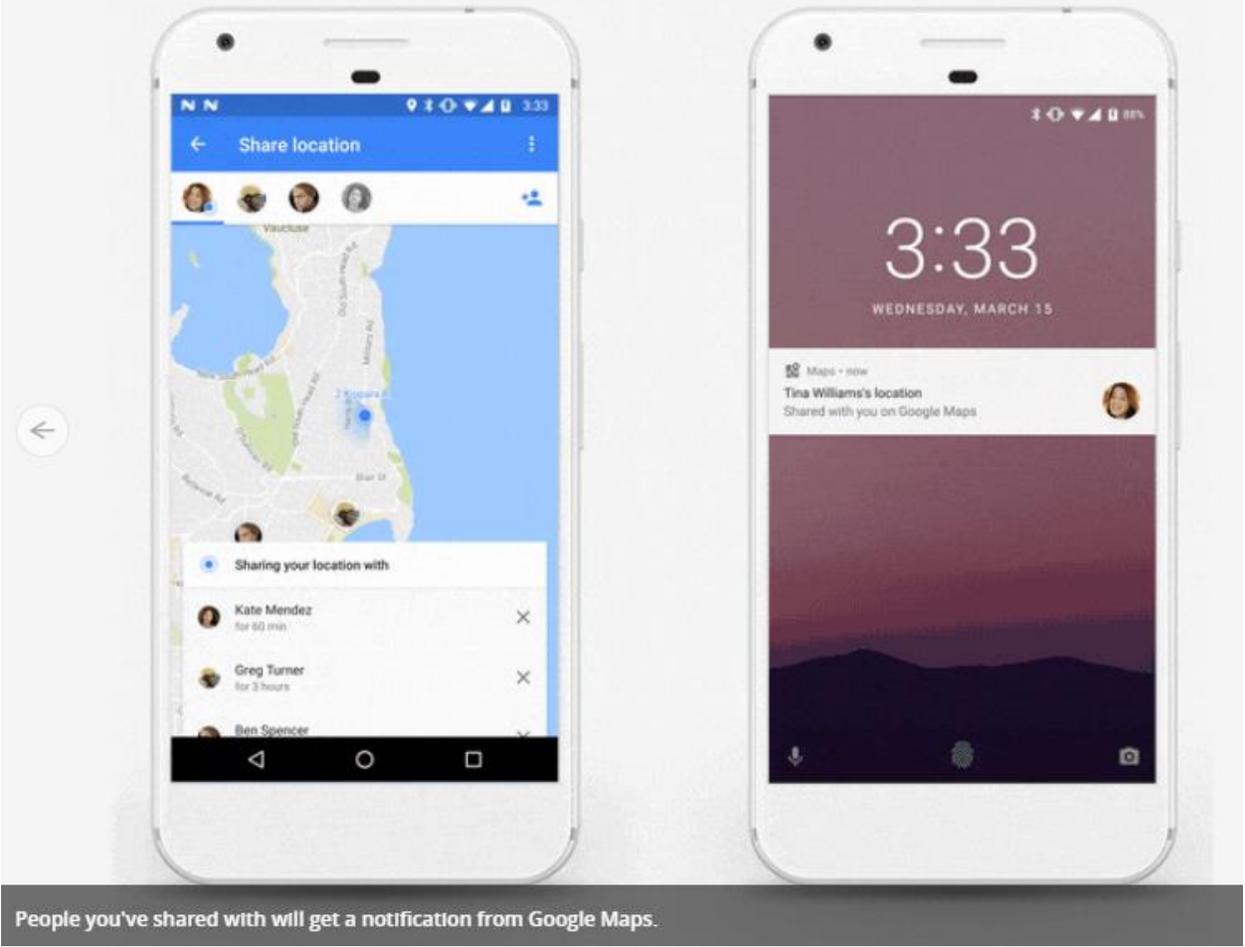
US9408055B2	HTC
	<p data-bbox="548 245 945 289"><b>Share your contacts</b></p> <ol data-bbox="562 318 1045 477" style="list-style-type: none"><li>1. Open your device's Contacts app .</li><li>2. Tap a contact in the list.</li><li>3. Tap More  &gt; <b>Share</b>.</li><li>4. Choose how you want to share the contact.</li></ol> <p data-bbox="512 529 1535 561"><a href="https://support.google.com/android/answer/6118731?hl=en&amp;ref_topic=6118711">https://support.google.com/android/answer/6118731?hl=en&amp;ref_topic=6118711</a></p>  <p data-bbox="522 1263 1518 1312">Ways to start location sharing: (left) from the app navigation drawer, and (right) from the menu, you can drag up while navigating somewhere.</p> <p data-bbox="512 1357 1656 1390"><a href="https://arstechnica.com/gadgets/2017/03/location-sharing-finally-returns-to-google-maps/">https://arstechnica.com/gadgets/2017/03/location-sharing-finally-returns-to-google-maps/</a></p>



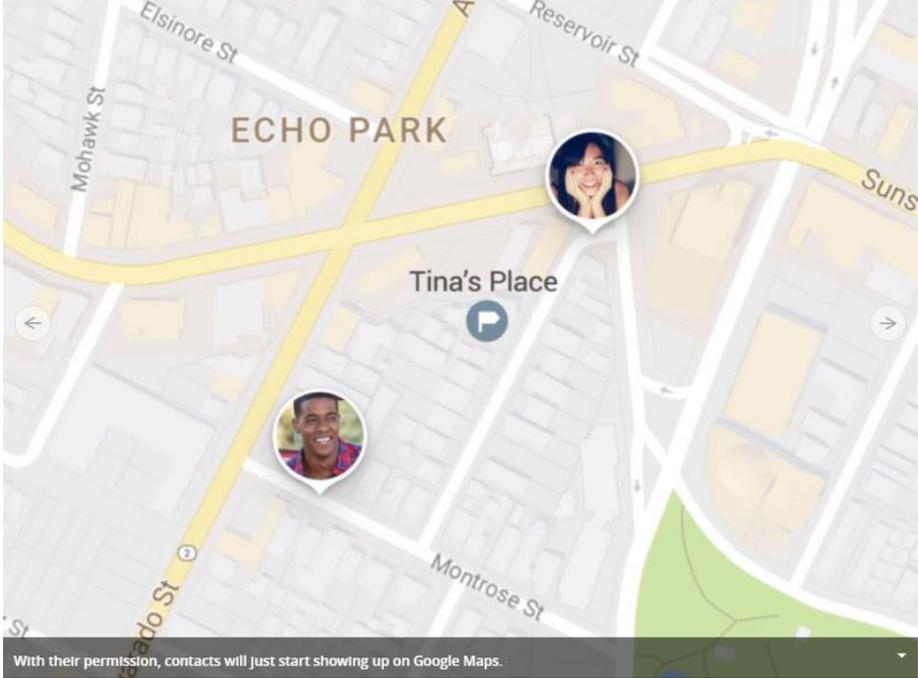
### Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products

US9408055B2	HTC
	 <p data-bbox="514 1019 1661 1063">You can pick how long you want to share your location for, and with whom you want to share it.</p> <p data-bbox="514 1096 1661 1136"><a href="https://arstechnica.com/gadgets/2017/03/location-sharing-finally-returns-to-google-maps/">https://arstechnica.com/gadgets/2017/03/location-sharing-finally-returns-to-google-maps/</a></p>

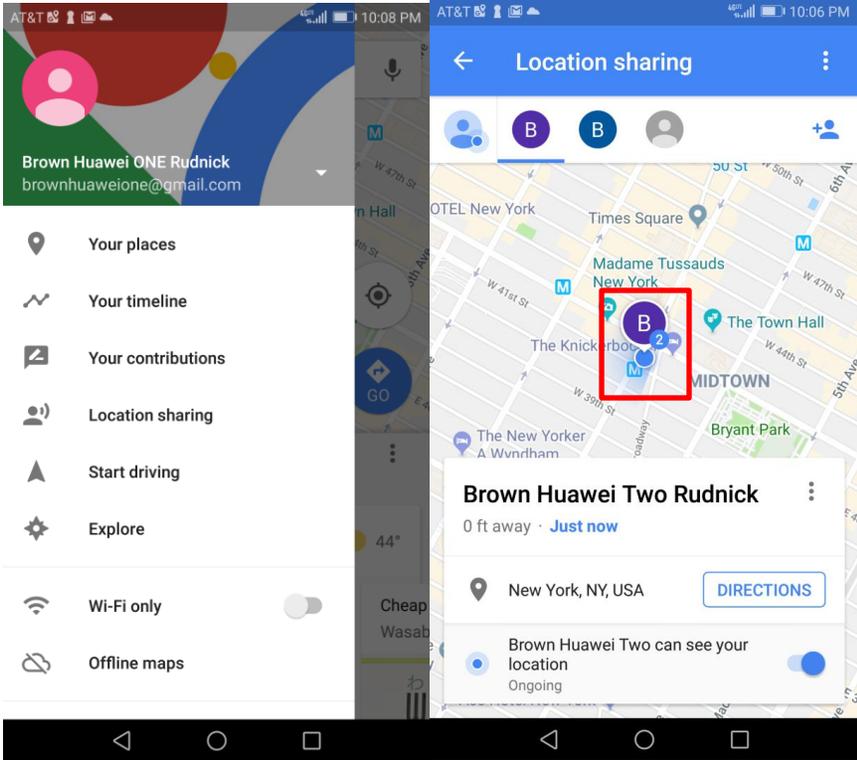
**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC
	 <p data-bbox="514 1144 1176 1177">People you've shared with will get a notification from Google Maps.</p> <p data-bbox="514 1218 1659 1258"><a href="https://arstechnica.com/gadgets/2017/03/location-sharing-finally-returns-to-google-maps/">https://arstechnica.com/gadgets/2017/03/location-sharing-finally-returns-to-google-maps/</a></p>

**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC
	 <p data-bbox="512 946 1656 980"><a href="https://arstechnica.com/gadgets/2017/03/location-sharing-finally-returns-to-google-maps/">https://arstechnica.com/gadgets/2017/03/location-sharing-finally-returns-to-google-maps/</a></p> <p data-bbox="512 1015 1022 1049"><b><u>Exemplary Google Maps Screenshots</u></b></p>

### Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products

US9408055B2	HTC
	 <p>The screenshot shows an HTC mobile phone interface. On the left, a menu is open with options: 'Your places', 'Your timeline', 'Your contributions', 'Location sharing', 'Start driving', 'Explore', 'Wi-Fi only', and 'Offline maps'. The 'Location sharing' option is selected. On the right, a map of New York City is displayed with a location pin for 'Brown Huawei Two Rudnick' highlighted by a red box. Below the map, a card shows the name 'Brown Huawei Two Rudnick', the distance '0 ft away · Just now', the location 'New York, NY, USA', and a 'DIRECTIONS' button. At the bottom of the card, it says 'Brown Huawei Two can see your location' with a toggle switch set to 'Ongoing'.</p> <p>Location information is shared via IP-based communication resulting in map that displays location information</p> <p><b><u>Exemplary Source Code:</u></b></p> <p>The above functionality is performed at least in part by the following publicly available source code and/or source code that invokes or is invoked by the following source code (or a substantially similar copy compiled and loaded onto the Accused Products by HTC). AGIS reserves the right to amend these contentions to include additional source code as discovery progresses and as additional source code is made available.</p>

## Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products

US9408055B2	HTC
	<h3 data-bbox="527 240 1016 293">Contacts Provider</h3> <p data-bbox="527 329 1472 591">The Contacts Provider is a powerful and flexible Android component that manages the device's central repository of data about people. The Contacts Provider is the source of data you see in the device's contacts application, and you can also access its data in your own application and transfer data between the device and online services. The provider accommodates a wide range of data sources and tries to manage as much data as possible for each person, with the result that its organization is complex. Because of this, the provider's API includes an extensive set of contract classes and interfaces that facilitate both data retrieval and modification.</p> <p data-bbox="527 626 835 646">This guide describes the following:</p> <ul data-bbox="527 675 1373 850" style="list-style-type: none"><li data-bbox="527 675 806 695">• The basic provider structure.</li><li data-bbox="527 724 894 743">• How to retrieve data from the provider.</li><li data-bbox="527 773 863 792">• How to modify data in the provider.</li><li data-bbox="527 821 1373 841">• How to write a sync adapter for synchronizing data from your server to the Contacts Provider.</li></ul> <p data-bbox="512 894 1486 924"><a href="https://developer.android.com/guide/topics/providers/contacts-provider.html">https://developer.android.com/guide/topics/providers/contacts-provider.html</a></p>

## Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products

US9408055B2	HTC
	<p><b>Overview</b></p> <p>ContactsContract defines an extensible database of contact-related information. Contact information is stored in a three-tier data model:</p> <ul style="list-style-type: none"> <li>• A row in the <code>ContactsContract.Data</code> table can store any kind of personal data, such as a phone number or email addresses. The set of data kinds that can be stored in this table is open-ended. There is a predefined set of common kinds, but any application can add its own data kinds.</li> <li>• A row in the <code>ContactsContract.RawContacts</code> table represents a set of data describing a person and associated with a single account (for example, one of the user's Gmail accounts).</li> <li>• A row in the <code>ContactsContract.Contacts</code> table represents an aggregate of one or more RawContacts presumably describing the same person. When data in or associated with the RawContacts table is changed, the affected aggregate contacts are updated as necessary.</li> </ul> <p>Other tables include:</p> <ul style="list-style-type: none"> <li>• <code>ContactsContract.Groups</code>, which contains information about raw contact groups such as Gmail contact groups. The current API does not support the notion of groups spanning multiple accounts.</li> <li>• <code>ContactsContract.StatusUpdates</code>, which contains social status updates including IM availability.</li> <li>• <code>ContactsContract.AggregationExceptions</code>, which is used for manual aggregation and disaggregation of raw contacts</li> <li>• <code>ContactsContract.Settings</code>, which contains visibility and sync settings for accounts and groups.</li> <li>• <code>ContactsContract.SyncState</code>, which contains free-form data maintained on behalf of sync adapters</li> <li>• <code>ContactsContract.PhoneLookup</code>, which is used for quick caller-ID lookup</li> </ul> <p><a href="https://developer.android.com/reference/android/provider/ContactsContract.html">https://developer.android.com/reference/android/provider/ContactsContract.html</a></p> <p><b>Data</b></p> <p>As noted previously, the data for a raw contact is stored in a <code>ContactsContract.Data</code> row that is linked to the raw contact's <code>_id</code> value. This allows a single raw contact to have multiple instances of the same type of data such as email addresses or phone numbers. For example, if "Thomas Higginson" for <code>emilyd@gmail.com</code> (the raw contact row for Thomas Higginson associated with the Google account <code>emilyd@gmail.com</code>) has a home email address of <code>thigg@gmail.com</code> and a work email address of <code>thomas.higginson@gmail.com</code>, the Contacts Provider stores the two email address rows and links them both to the raw contact.</p> <p>Notice that different types of data are stored in this single table. Display name, phone number, email, postal address, photo, and website detail rows are all found in the <code>ContactsContract.Data</code> table. To help manage this, the <code>ContactsContract.Data</code> table has some columns with descriptive names, and others with generic names. The contents of a descriptive-name column have the same meaning regardless of the type of data in the row, while the contents of a generic-name column have different meanings depending on the type of data.</p>

**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC														
	<p data-bbox="512 238 1486 266"><a href="https://developer.android.com/guide/topics/providers/contacts-provider.html">https://developer.android.com/guide/topics/providers/contacts-provider.html</a></p> <table border="1" data-bbox="520 375 1745 1045"> <thead> <tr> <th data-bbox="527 380 617 417">Task</th> <th data-bbox="623 380 854 417">Action</th> <th data-bbox="861 380 1188 417">Data</th> <th data-bbox="1194 380 1486 417">MIME type</th> <th data-bbox="1493 380 1738 417">Notes</th> </tr> </thead> <tbody> <tr> <td data-bbox="527 422 617 548">Pick a contact from a list</td> <td data-bbox="623 422 854 548">ACTION_PICK</td> <td data-bbox="861 422 1188 857">                     One of:                     <ul style="list-style-type: none"> <li data-bbox="867 459 1171 516">• <code>Contacts.CONTENT_URI</code>, which displays a list of contacts.</li> <li data-bbox="867 540 1171 630">• <code>Phone.CONTENT_URI</code>, which displays a list of phone numbers for a raw contact.</li> <li data-bbox="867 654 1171 743">• <code>StructuredPostal.CONTENT_URI</code>, which displays a list of postal addresses for a raw contact.</li> <li data-bbox="867 768 1171 857">• <code>Email.CONTENT_URI</code>, which displays a list of email addresses for a raw contact.</li> </ul> </td> <td data-bbox="1194 422 1486 548">Not used</td> <td data-bbox="1493 422 1738 1036">                     Displays a list of raw contacts or a list of data from a raw contact, depending on the content URI type you supply. Call <code>startActivityForResult()</code>, which returns the content URI of the selected row. The form of the URI is the table's content URI with the row's <code>LOOKUP_ID</code> appended to it. The device's contacts app delegates read and write permissions to this content URI for the life of your activity. See the <a href="#">Content Provider Basics</a> guide for more details.                 </td> </tr> </tbody> </table> <p data-bbox="512 1084 1486 1112"><a href="https://developer.android.com/guide/topics/providers/contacts-provider.html">https://developer.android.com/guide/topics/providers/contacts-provider.html</a></p>					Task	Action	Data	MIME type	Notes	Pick a contact from a list	ACTION_PICK	One of: <ul style="list-style-type: none"> <li data-bbox="867 459 1171 516">• <code>Contacts.CONTENT_URI</code>, which displays a list of contacts.</li> <li data-bbox="867 540 1171 630">• <code>Phone.CONTENT_URI</code>, which displays a list of phone numbers for a raw contact.</li> <li data-bbox="867 654 1171 743">• <code>StructuredPostal.CONTENT_URI</code>, which displays a list of postal addresses for a raw contact.</li> <li data-bbox="867 768 1171 857">• <code>Email.CONTENT_URI</code>, which displays a list of email addresses for a raw contact.</li> </ul>	Not used	Displays a list of raw contacts or a list of data from a raw contact, depending on the content URI type you supply. Call <code>startActivityForResult()</code> , which returns the content URI of the selected row. The form of the URI is the table's content URI with the row's <code>LOOKUP_ID</code> appended to it. The device's contacts app delegates read and write permissions to this content URI for the life of your activity. See the <a href="#">Content Provider Basics</a> guide for more details.
Task	Action	Data	MIME type	Notes											
Pick a contact from a list	ACTION_PICK	One of: <ul style="list-style-type: none"> <li data-bbox="867 459 1171 516">• <code>Contacts.CONTENT_URI</code>, which displays a list of contacts.</li> <li data-bbox="867 540 1171 630">• <code>Phone.CONTENT_URI</code>, which displays a list of phone numbers for a raw contact.</li> <li data-bbox="867 654 1171 743">• <code>StructuredPostal.CONTENT_URI</code>, which displays a list of postal addresses for a raw contact.</li> <li data-bbox="867 768 1171 857">• <code>Email.CONTENT_URI</code>, which displays a list of email addresses for a raw contact.</li> </ul>	Not used	Displays a list of raw contacts or a list of data from a raw contact, depending on the content URI type you supply. Call <code>startActivityForResult()</code> , which returns the content URI of the selected row. The form of the URI is the table's content URI with the row's <code>LOOKUP_ID</code> appended to it. The device's contacts app delegates read and write permissions to this content URI for the life of your activity. See the <a href="#">Content Provider Basics</a> guide for more details.											

**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC
	<pre>25  * Parsed form of the intent sent to the Contacts application. 26  */ 27  public class ContactsRequest { 28 29      /** Default mode: browse contacts */ 30      public static final int ACTION_DEFAULT = 10; 31 32      /** Show all contacts */ 33      public static final int ACTION_ALL_CONTACTS = 15; 34 35      /** Show all contacts with phone numbers */ 36      public static final int ACTION_CONTACTS_WITH_PHONES = 17; 37 38      /** Show contents of a specific group */ 39      public static final int ACTION_GROUP = 20; 40 41      /** Show all starred contacts */ 42      public static final int ACTION_STARRED = 30; 43 44      /** Show frequently contacted contacts */ 45      public static final int ACTION_FREQUENT = 40; 46 47      /** Show starred and the frequent */ 48      public static final int ACTION_STREQUENT = 50; 49 50      /** Show all contacts and pick them when clicking */ 51      public static final int ACTION_PICK_CONTACT = 60; 52 53      /** Show all contacts as well as the option to create a new one */ 54      public static final int ACTION_PICK_OR_CREATE_CONTACT = 70; 55 56      /** Show all contacts and pick them for edit when clicking, and allow creating a new contact */ 57      public static final int ACTION_INSERT_OR_EDIT_CONTACT = 80;</pre> <p><a href="https://android.googlesource.com/platform/packages/apps/Contacts+/nougat-mr1-release/src/com/android/contacts/list/ContactsRequest.java">https://android.googlesource.com/platform/packages/apps/Contacts+/nougat-mr1-release/src/com/android/contacts/list/ContactsRequest.java</a></p>



**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC
	<pre>59  /** Show all phone numbers and pick them when clicking */ 60  public static final int ACTION_PICK_PHONE = 90; 61 62  /** Show all postal addresses and pick them when clicking */ 63  public static final int ACTION_PICK_POSTAL = 100; 64 65  /** Show all postal addresses and pick them when clicking */ 66  public static final int ACTION_PICK_EMAIL = 105; 67 68  /** Show all contacts and create a shortcut for the picked contact */ 69  public static final int ACTION_CREATE_SHORTCUT_CONTACT = 110; 70 71  /** Show all phone numbers and create a call shortcut for the picked number */ 72  public static final int ACTION_CREATE_SHORTCUT_CALL = 120; 73 74  /** Show all phone numbers and create an SMS shortcut for the picked number */ 75  public static final int ACTION_CREATE_SHORTCUT_SMS = 130; 76 77  /** Show all contacts and activate the specified one */ 78  public static final int ACTION_VIEW_CONTACT = 140; 79 80  /** Show contacts recommended for joining with a specified target contact */ 81  public static final int ACTION_PICK_JOIN = 150;</pre> <p><a href="https://android.googlesource.com/platform/packages/apps/Contacts/+nougat-mr1-release/src/com/android/contacts/list/ContactsRequest.java">https://android.googlesource.com/platform/packages/apps/Contacts/+nougat-mr1-release/src/com/android/contacts/list/ContactsRequest.java</a></p>

**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC
	<pre> 104      * Displays a list to browse contacts. 105      */ 106      public class PeopleActivity extends ContactsActivity implements 107          View.OnCreateContextMenuListener, 108          View.OnClickListener, 109          ActionBarAdapter.Listener, 110          DialogManager.DialogShowingViewActivity, 111          ContactListFilterController.ContactListFilterListener, 112          ProviderStatusListener, 113          MultiContactDeleteListener, 114          JoinContactsListener {  https://android.googlesource.com/platform/packages/apps/Contacts+/nougat-mr1- release/src/com/android/contacts/activities/PeopleActivity.java  145          * Showing a list of Contacts. Also used for showing search results in search mode. 146          */ 147          private MultiSelectContactsListFragment mAllFragment; 148          private ContactTileListFragment mFavoritesFragment;  https://android.googlesource.com/platform/packages/apps/Contacts+/nougat-mr1- release/src/com/android/contacts/activities/PeopleActivity.java </pre>

**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC
	<pre>458     private void configureFragments(boolean fromRequest) { 459         if (fromRequest) { 460             ContactListFilter filter = null; 461             int actionCode = mRequest.getActionCode(); 462             boolean searchMode = mRequest.isSearchMode(); 463             final int tabToOpen; 464             switch (actionCode) { 465                 case ContactsRequest.ACTION_ALL_CONTACTS: 466                     filter = ContactListFilter.createFilterWithType( 467                         ContactListFilter.FILTER_TYPE_ALL_ACCOUNTS); 468                     tabToOpen = TabState.ALL; 469                     break; 470                 case ContactsRequest.ACTION_CONTACTS_WITH_PHONES: 471                     filter = ContactListFilter.createFilterWithType( 472                         ContactListFilter.FILTER_TYPE_WITH_PHONE_NUMBERS_ONLY); 473                     tabToOpen = TabState.ALL; 474                     break; 475 476                 case ContactsRequest.ACTION_FREQUENT: 477                 case ContactsRequest.ACTION_STREQUENT: 478                 case ContactsRequest.ACTION_STARRED: 479                     tabToOpen = TabState.FAVORITES; 480                     break; 481                 case ContactsRequest.ACTION_VIEW_CONTACT: 482                     tabToOpen = TabState.ALL; 483                     break; 484                 default: 485                     tabToOpen = -1; 486                     break; 487             } </pre> <p data-bbox="506 1352 1566 1422"><a href="https://android.googlesource.com/platform/packages/apps/Contacts+/nougat-mr1-release/src/com/android/contacts/activities/PeopleActivity.java">https://android.googlesource.com/platform/packages/apps/Contacts+/nougat-mr1-release/src/com/android/contacts/activities/PeopleActivity.java</a></p>

**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC
	<pre>488         if (tabToOpen != -1) { 489             mActionBarAdapter.setCurrentTab(tabToOpen); 490         } 491 492         if (filter != null) { 493             mContactListFilterController.setContactListFilter(filter, false); 494             searchMode = false; 495         } 496 497         if (mRequest.getContactUri() != null) { 498             searchMode = false; 499         } 500 501         mActionBarAdapter.setSearchMode(searchMode); 502         configureContactListFragmentForRequest(); 503     } 504 505     configureContactListFragment(); 506 507     invalidateOptionsMenuIfNeeded(); 508 }</pre> <p><a href="https://android.googlesource.com/platform/packages/apps/Contacts/+/nougat-mr1-release/src/com/android/contacts/activities/PeopleActivity.java">https://android.googlesource.com/platform/packages/apps/Contacts/+/nougat-mr1-release/src/com/android/contacts/activities/PeopleActivity.java</a></p>

**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC
	<pre>35 public class ProfileAndContactsLoader extends CursorLoader { 36 37     private boolean mLoadProfile; 38 39     private String[] mProjection; 40 41     private Uri mExtraUri; 42     private String[] mExtraProjection; 43     private String mExtraSelection; 44     private String[] mExtraSelectionArgs; 45     private boolean mMergeExtraContactsAfterPrimary; 46 47     public ProfileAndContactsLoader(Context context) { 48         super(context); 49     }</pre> <p><a href="https://android.googlesource.com/platform/packages/apps/ContactsCommon/+nougat-mr1-release/src/com/android/contacts/common/list/ProfileAndContactsLoader.java">https://android.googlesource.com/platform/packages/apps/ContactsCommon/+nougat-mr1-release/src/com/android/contacts/common/list/ProfileAndContactsLoader.java</a></p>

**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC
	<pre> 35 public final class GroupMemberLoader extends CursorLoader { 36 37     public static class GroupEditorQuery { 38         private static final String[] PROJECTION = new String[] { 39             Data.CONTACT_ID,           // 0 40             Data.RAW_CONTACT_ID,       // 1 41             Data.DISPLAY_NAME_PRIMARY,  // 2 42             Data.PHOTO_URI,            // 3 43             Data.LOOKUP_KEY,           // 4 44         }; 45 46         public static final int CONTACT_ID           = 0; 47         public static final int RAW_CONTACT_ID       = 1; 48         public static final int CONTACT_DISPLAY_NAME_PRIMARY = 2; 49         public static final int CONTACT_PHOTO_URI    = 3; 50         public static final int CONTACT_LOOKUP_KEY   = 4; 51     } 52 53     public static class GroupDetailQuery { 54         private static final String[] PROJECTION = new String[] { 55             Data.CONTACT_ID,           // 0 56             Data.PHOTO_URI,            // 1 57             Data.LOOKUP_KEY,           // 2 58             Data.DISPLAY_NAME_PRIMARY,  // 3 59             Data.CONTACT_PRESENCE,     // 4 60             Data.CONTACT_STATUS,       // 5 61         }; 62 63         public static final int CONTACT_ID           = 0; 64         public static final int CONTACT_PHOTO_URI    = 1; 65         public static final int CONTACT_LOOKUP_KEY   = 2; 66         public static final int CONTACT_DISPLAY_NAME_PRIMARY = 3; 67         public static final int CONTACT_PRESENCE_STATUS = 4; 68         public static final int CONTACT_STATUS       = 5; 69     } 70 71     private final long mGroupId; </pre> <p><a href="https://android.googlesource.com/platform/packages/apps/Contacts+/nougat-mr1-release/src/com/android/contacts/GroupMemberLoader.java">https://android.googlesource.com/platform/packages/apps/Contacts+/nougat-mr1-release/src/com/android/contacts/GroupMemberLoader.java</a></p>

**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC
	<pre> 24  * Group loader for the group list that includes details such as the number of contacts per group 25  * and number of groups per account. This list is sorted by account type, account name, where the 26  * group names are in alphabetical order. Note that the list excludes default, favorite, and deleted 27  * groups. 28  */ 29  public final class GroupListLoader extends CursorLoader { 30 31      private final static String[] COLUMNS = new String[] { 32          Groups.ACCOUNT_NAME, 33          Groups.ACCOUNT_TYPE, 34          Groups.DATA_SET, 35          Groups._ID, 36          Groups.TITLE, 37          Groups.SUMMARY_COUNT, 38      }; 39 40      public final static int ACCOUNT_NAME = 0; 41      public final static int ACCOUNT_TYPE = 1; 42      public final static int DATA_SET = 2; 43      public final static int GROUP_ID = 3; 44      public final static int TITLE = 4; 45      public final static int MEMBER_COUNT = 5; 46 47      private static final Uri GROUP_LIST_URI = Groups.CONTENT_SUMMARY_URI; 48 49      public GroupListLoader(Context context) { 50          super(context, GROUP_LIST_URI, COLUMNS, Groups.ACCOUNT_TYPE + " NOT NULL AND " 51              + Groups.ACCOUNT_NAME + " NOT NULL AND " + Groups.AUTO_ADD + "=0 AND " + 52              Groups.FAVORITES + "=0 AND " + Groups.DELETED + "=0", null, 53              Groups.ACCOUNT_TYPE + ", " + Groups.ACCOUNT_NAME + ", " + Groups.DATA_SET + ", " + 54              Groups.TITLE + " COLLATE LOCALIZED ASC"); 55      } 56  } </pre> <p><a href="https://android.googlesource.com/platform/packages/apps/Contacts+/nougat-mr1-release/src/com/android/contacts/GroupListLoader.java">https://android.googlesource.com/platform/packages/apps/Contacts+/nougat-mr1-release/src/com/android/contacts/GroupListLoader.java</a></p>

**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC
	<pre> 24  * Group meta-data loader. Loads all groups or just a single group from the 25  * database (if given a {@link Uri}). 26  */ 27  public final class GroupMetaDataLoader extends CursorLoader { 28 29      private final static String[] COLUMNS = new String[] { 30          Groups.ACCOUNT_NAME, 31          Groups.ACCOUNT_TYPE, 32          Groups.DATA_SET, 33          Groups._ID, 34          Groups.TITLE, 35          Groups.AUTO_ADD, 36          Groups.FAVORITES, 37          Groups.GROUP_IS_READ_ONLY, 38          Groups.DELETED, 39      }; 40 41      public final static int ACCOUNT_NAME = 0; 42      public final static int ACCOUNT_TYPE = 1; 43      public final static int DATA_SET = 2; 44      public final static int GROUP_ID = 3; 45      public final static int TITLE = 4; 46      public final static int AUTO_ADD = 5; 47      public final static int FAVORITES = 6; 48      public final static int IS_READ_ONLY = 7; 49      public final static int DELETED = 8; 50 51      public GroupMetaDataLoader(Context context, Uri groupUri) { 52          super(context, ensureIsGroupUri(groupUri), COLUMNS, Groups.ACCOUNT_TYPE + " NOT NULL AND " 53              + Groups.ACCOUNT_NAME + " NOT NULL", null, null); 54      } 55 56      /** 57       * Ensures that this is a valid group URI. If invalid, then an exception is 58       * thrown. Otherwise, the original URI is returned. 59       */ 60      private static Uri ensureIsGroupUri(final Uri groupUri) { 61          // TODO: Fix ContactsProvider2 getType method to resolve the group Uris 62          if (groupUri == null) { 63              throw new IllegalArgumentException("Uri must not be null"); 64          } 65          if (!groupUri.toString().startsWith(Groups.CONTENT_URI.toString())) { 66              throw new IllegalArgumentException("Invalid group Uri: " + groupUri); 67          } 68          return groupUri; 69      } 70  } </pre> <p><a href="https://android.googlesource.com/platform/packages/apps/Contacts/+nougat-mr1-release/src/com/android/contacts/GroupMetaDataLoader.java">https://android.googlesource.com/platform/packages/apps/Contacts/+nougat-mr1-release/src/com/android/contacts/GroupMetaDataLoader.java</a></p>



**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC
	<pre>19  * Meta-data for a contact group. We load all groups associated with the contact's 20  * constituent accounts. 21  */ 22  public final class GroupMetaData { 23      private String mAccountName; 24      private String mAccountType; 25      private String mDataSet; 26      private long mGroupId; 27      private String mTitle; 28      private boolean mDefaultGroup; 29      private boolean mFavorites; 30 31      public GroupMetaData(String accountName, String accountType, String dataSet, long groupId, 32                          String title, boolean defaultGroup, boolean favorites) { 33          this.mAccountName = accountName; 34          this.mAccountType = accountType; 35          this.mDataSet = dataSet; 36          this.mGroupId = groupId; 37          this.mTitle = title; 38          this.mDefaultGroup = defaultGroup; 39          this.mFavorites = favorites; 40      } 41 42      public String getAccountName() { 43          return mAccountName; 44      } 45 46      public String getAccountType() { 47          return mAccountType; 48      } 49 50      public String getDataSet() { 51          return mDataSet; 52      } 53 54      public long getGroupId() { 55          return mGroupId; 56      } 57 58      public String getTitle() { 59          return mTitle; 60      } 61 62      public boolean isDefaultGroup() { 63          return mDefaultGroup; 64      } 65 66      public boolean isFavorites() { 67          return mFavorites;</pre>

**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC
	<p data-bbox="512 233 1686 305"><a href="https://android.googlesource.com/platform/packages/apps/ContactsCommon/+nougat-mr1-release/src/com/android/contacts/common/GroupMetaData.java">https://android.googlesource.com/platform/packages/apps/ContactsCommon/+nougat-mr1-release/src/com/android/contacts/common/GroupMetaData.java</a></p> <pre data-bbox="533 347 1602 1166"> 44 * Class that sends chat message via SMS. 45 * 46 * The interface emulates a blocking sending similar to making an HTTP request. 47 * It calls the SmsManager to send a (potentially multipart) message and waits 48 * on the sent status on each part. The waiting has a timeout so it won't wait 49 * forever. Once the sent status of all parts received, the call returns. 50 * A successful sending requires success status for all parts. Otherwise, we 51 * pick the highest level of failure as the error for the whole message, which 52 * is used to determine if we need to retry the sending. 53 */ 54 public class SmsSender { 55     private static final String TAG = LogUtil.BUGLE_TAG; 56 57     public static final String EXTRA_PART_ID = "part_id"; 58 59     /* 60     * A map for pending sms messages. The key is the random request UUID. 61     */ 62     private static ConcurrentHashMap&lt;Uri, SendResult&gt; sPendingMessageMap = 63         new ConcurrentHashMap&lt;Uri, SendResult&gt;(); 64 65     private static final Random RANDOM = new Random(); 66 67     // Whether we should send multipart SMS as separate messages 68     private static Boolean sSendMultipartSmsAsSeparateMessages = null; 69 </pre> <p data-bbox="512 1211 1596 1282"><a href="https://android.googlesource.com/platform/packages/apps/Messaging/+nougat-mr1-release/src/com/android/messaging/sms/SmsSender.java">https://android.googlesource.com/platform/packages/apps/Messaging/+nougat-mr1-release/src/com/android/messaging/sms/SmsSender.java</a></p>

**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC
	<pre> 253 // Actually sending the message using SmsManager 254 private static void sendInternal(final Context context, final int subId, String dest, 255     final ArrayList&lt;String&gt; messages, final String serviceCenter, 256     final boolean requireDeliveryReport, final Uri messageUri) throws SmsException { 257     Assert.notNull(context); 258     final SmsManager smsManager = PhoneUtils.get(subId).getSmsManager(); 259     final int messageCount = messages.size(); 260     final ArrayList&lt;PendingIntent&gt; deliveryIntents = new ArrayList&lt;PendingIntent&gt;(messageCount); 261     final ArrayList&lt;PendingIntent&gt; sentIntents = new ArrayList&lt;PendingIntent&gt;(messageCount); 262     for (int i = 0; i &lt; messageCount; i++) { 263         // Make pending intents different for each message part 264         final int partId = (messageCount &lt;= 1 ? 0 : i + 1); 265         if (requireDeliveryReport &amp;&amp; (i == (messageCount - 1))) { 266             // TODO we only care about the delivery status of the last part 267             // Shall we have better tracking of delivery status of all parts? 268             deliveryIntents.add(PendingIntent.getBroadcast( 269                 context, 270                 partId, 271                 getSendStatusIntent(context, SendStatusReceiver.MESSAGE_DELIVERED_ACTION, 272                     messageUri, partId, subId), 273                 0/*flag*/)); 274         } else { 275             deliveryIntents.add(null); 276         } 277         sentIntents.add(PendingIntent.getBroadcast( 278             context, 279             partId, 280             getSendStatusIntent(context, SendStatusReceiver.MESSAGE_SENT_ACTION, 281                 messageUri, partId, subId), 282             0/*flag*/)); 283     } 284     if (sSendMultipartSmsAsSeparateMessages == null) { 285         sSendMultipartSmsAsSeparateMessages = MmsConfig.get(subId) 286             .getSendMultipartSmsAsSeparateMessages(); 287     } </pre> <p><a href="https://android.googlesource.com/platform/packages/apps/Messaging/+nougat-mr1-release/src/com/android/messaging/sms/SmsSender.java">https://android.googlesource.com/platform/packages/apps/Messaging/+nougat-mr1-release/src/com/android/messaging/sms/SmsSender.java</a></p>

**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC
	<pre> 288     try { 289         if (sSendMultipartSmsAsSeparateMessages) { 290             // If multipart sms is not supported, send them as separate messages 291             for (int i = 0; i &lt; messageCount; i++) { 292                 smsManager.sendTextMessage(dest, 293                     serviceCenter, 294                     messages.get(i), 295                     sentIntents.get(i), 296                     deliveryIntents.get(i)); 297             } 298         } else { 299             smsManager.sendMultipartTextMessage( 300                 dest, serviceCenter, messages, sentIntents, deliveryIntents); 301         } 302     } catch (final Exception e) { 303         throw new SmsException("SmsSender: caught exception in sending " + e); 304     } 305 } </pre> <p><a href="https://android.googlesource.com/platform/packages/apps/Messaging/+/nougat-mr1-release/src/com/android/messaging/sms/SmsSender.java">https://android.googlesource.com/platform/packages/apps/Messaging/+/nougat-mr1-release/src/com/android/messaging/sms/SmsSender.java</a></p> <pre> 56     * Class that receives incoming SMS messages through android.provider.Telephony.SMS_RECEIVED 57     * 58     * This class serves two purposes: 59     * - Process phone verification SMS messages 60     * - Handle SMS messages when the user has enabled us to be the default SMS app (Pre-KLP) 61     */ 62     public final class SmsReceiver extends BroadcastReceiver { 63         private static final String TAG = LogUtil.BUGLE_TAG; 64 65         private static ArrayList&lt;Pattern&gt; sIgnoreSmsPatterns; 66 </pre> <p><a href="https://android.googlesource.com/platform/packages/apps/Messaging/+/nougat-mr1-release/src/com/android/messaging/sms/SmsReceiver.java">https://android.googlesource.com/platform/packages/apps/Messaging/+/nougat-mr1-release/src/com/android/messaging/sms/SmsReceiver.java</a></p>

**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC
	<p data-bbox="514 233 1312 261"><a href="#">release/src/com/android/messaging/receiver/SmsReceiver.java</a></p> <pre data-bbox="527 302 1732 1084"> 200     public static void deliverSmsMessages(final Context context, final int subId, 201         final int errorCode, final android.telephony.SmsMessage[] messages) { 202         final ContentValues messageValues = 203             MmsUtils.parseReceivedSmsMessage(context, messages, errorCode); 204 205         LogUtil.v(TAG, "SmsReceiver.deliverSmsMessages"); 206 207         final long nowInMillis = System.currentTimeMillis(); 208         final long receivedTimestampMs = MmsUtils.getMessageDate(messages[0], nowInMillis); 209 210         messageValues.put(Sms.Inbox.DATE, receivedTimestampMs); 211         // Default to unread and unseen for us but ReceiveSmsMessageAction will override 212         // seen for the telephony db. 213         messageValues.put(Sms.Inbox.READ, 0); 214         messageValues.put(Sms.Inbox.SEEN, 0); 215         if (OsUtil.isAtLeastLMR1()) { 216             messageValues.put(Sms.SUBSCRIPTION_ID, subId); 217         } 218 219         if (messages[0].getMessageClass() == android.telephony.SmsMessage.MessageClass.CLASS_0    220             DebugUtils.debugClassZeroSmsEnabled()) { 221             Factory.get().getUIIntents().launchClassZeroActivity(context, messageValues); 222         } else { 223             final ReceiveSmsMessageAction action = new ReceiveSmsMessageAction(messageValues); 224             action.start(); 225         } 226     } </pre> <p data-bbox="514 1130 1596 1203"><a href="https://android.googlesource.com/platform/packages/apps/Messaging/+nougat-mr1-release/src/com/android/messaging/receiver/SmsReceiver.java">https://android.googlesource.com/platform/packages/apps/Messaging/+nougat-mr1-release/src/com/android/messaging/receiver/SmsReceiver.java</a></p>

**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC
	<pre>228     @Override 229     public void onReceive(final Context context, final Intent intent) { 230         LogUtil.v(TAG, "SmsReceiver.onReceive " + intent); 231         // On KLP+ we only take delivery of SMS messages in SmsDeliverReceiver. 232         if (PhoneUtils.getDefault().isSmsEnabled()) { 233             final String action = intent.getAction(); 234             if (OsUtil.isSecondaryUser() &amp;&amp; 235                 (Telephony.Sms.Intents.SMS_RECEIVED_ACTION.equals(action)    236                  // TODO: update this with the actual constant from Telephony 237                  "android.provider.Telephony.MMS_DOWNLOADED".equals(action))) { 238                 postNewMessageSecondaryUserNotification(); 239             } else if (!OsUtil.isAtLeastKLP()) { 240                 deliverSmsIntent(context, intent); 241             } 242         } 243     }</pre> <p data-bbox="512 760 1596 831"><a href="https://android.googlesource.com/platform/packages/apps/Messaging/+nougat-mr1-release/src/com/android/messaging/receiver/SmsReceiver.java">https://android.googlesource.com/platform/packages/apps/Messaging/+nougat-mr1-release/src/com/android/messaging/receiver/SmsReceiver.java</a></p>

**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC
	<pre>52  * Class that sends chat message via MMS. 53  * 54  * The interface emulates a blocking send similar to making an HTTP request. 55  */ 56  public class MmsSender { 57      private static final String TAG = LogUtil.BUGLE_TAG; 58 59      /** 60       * Send an MMS message. 61       * 62       * @param context Context 63       * @param messageUri The unique URI of the message for identifying it during sending 64       * @param sendReq The SendReq PDU of the message 65       * @throws MmsFailureException 66       */ 67      public static void sendMms(final Context context, final int subId, final Uri messageUri, 68                               final SendReq sendReq, final Bundle sentIntentExtras) throws MmsFailureException { 69          sendMms(context, 70                subId, 71                messageUri, 72                null /* locationUri */, 73                sendReq, 74                true /* responseImportant */, 75                sentIntentExtras); 76      }</pre> <p><a href="https://android.googlesource.com/platform/packages/apps/Messaging/+/nougat-mr1-release/src/com/android/messaging/sms/MmsSender.java">https://android.googlesource.com/platform/packages/apps/Messaging/+/nougat-mr1-release/src/com/android/messaging/sms/MmsSender.java</a></p>

**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC
	<pre> 240      * Download an MMS message. 241      * 242      * @param context Context 243      * @param contentLocation The url of the MMS message 244      * @throws MmsFailureException 245      * @throws InvalidHeaderValueException 246      */ 247      public static void downloadMms(final Context context, final int subId, 248          final String contentLocation, Bundle extras) throws MmsFailureException, 249          InvalidHeaderValueException { 250          final Uri requestUri = Uri.parse(contentLocation); 251          final Uri contentUri = MmsFileProvider.buildRawMmsUri(); 252 253          final Intent downloadedIntent = new Intent(SendStatusReceiver.MMS_DOWNLOADED_ACTION, 254              requestUri, 255              context, 256              SendStatusReceiver.class); 257          downloadedIntent.putExtra(SendMessageAction.EXTRA_CONTENT_URI, contentUri); 258          if (extras != null) { 259              downloadedIntent.putExtras(extras); 260          } 261          final PendingIntent downloadedPendingIntent = PendingIntent.getBroadcast( 262              context, 263              0 /*request code*/, 264              downloadedIntent, 265              PendingIntent.FLAG_UPDATE_CURRENT); 266 267          MmsManager.downloadMultimediaMessage(subId, context, contentLocation, contentUri, 268              downloadedPendingIntent); 269      } </pre> <p><a href="https://android.googlesource.com/platform/packages/apps/Messaging/+nougat-mr1-release/src/com/android/messaging/sms/MmsSender.java">https://android.googlesource.com/platform/packages/apps/Messaging/+nougat-mr1-release/src/com/android/messaging/sms/MmsSender.java</a></p>



**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC
	<pre> 97      * Execute an MMS HTTP request, either a POST (sending) or a GET (downloading) 98      * 99      * @param urlString The request URL, for sending it is usually the MMSC, and for downloading 100     *           it is the message URL 101     * @param pdu For POST (sending) only, the PDU to send 102     * @param method HTTP method, POST for sending and GET for downloading 103     * @param isProxySet Is there a proxy for the MMSC 104     * @param proxyHost The proxy host 105     * @param proxyPort The proxy port 106     * @param mmsConfig The MMS config to use 107     * @param userAgent The user agent header value 108     * @param uaProfUrl The UA Prof URL header value 109     * @return The HTTP response body 110     * @throws MmsHttpException For any failures 111     */ 112     public byte[] execute(String urlString, byte[] pdu, String method, boolean isProxySet, 113         String proxyHost, int proxyPort, Bundle mmsConfig, String userAgent, String uaProfUrl) 114         throws MmsHttpException { 115         Log.d(MmsService.TAG, "HTTP: " + method + " " + Utils.redactUrlForNonVerbose(urlString) 116             + (isProxySet ? (" proxy=" + proxyHost + ":" + proxyPort) : "") 117             + ", PDU size=" + (pdu != null ? pdu.length : 0)); 118         checkMethod(method); 119         HttpURLConnection connection = null; 120         try { 121             Proxy proxy = Proxy.NO_PROXY; 122             if (isProxySet) { 123                 proxy = new Proxy(Proxy.Type.HTTP, new InetSocketAddress(proxyHost, proxyPort)); 124             } 125             final URL url = new URL(urlString); 126             // Now get the connection 127             connection = (HttpURLConnection) url.openConnection(proxy); 128             connection.setDoInput(true); 129             connection.setConnectTimeout( 130                 mmsConfig.getInt(CarrierConfigValuesLoader.CONFIG_HTTP_SOCKET_TIMEOUT, 131                     CarrierConfigValuesLoader.CONFIG_HTTP_SOCKET_TIMEOUT_DEFAULT)); </pre> <p data-bbox="506 1252 1596 1328"> <a href="https://android.googlesource.com/platform/packages/apps/Messaging/+nougat-mr1-release/src/android/support/v7/mms/MmsHttpClient.java">https://android.googlesource.com/platform/packages/apps/Messaging/+nougat-mr1-release/src/android/support/v7/mms/MmsHttpClient.java</a> </p>

**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC
	<pre> 132 // ----- COMMON HEADERS ----- 133 // Header: Accept 134 connection.setRequestProperty(HEADER_ACCEPT, HEADER_VALUE_ACCEPT); 135 // Header: Accept-Language 136 connection.setRequestProperty( 137     HEADER_ACCEPT_LANGUAGE, getCurrentAcceptLanguage(Locale.getDefault())); 138 // Header: User-Agent 139 Log.i(MmsService.TAG, "HTTP: User-Agent=" + userAgent); 140 connection.setRequestProperty(HEADER_USER_AGENT, userAgent); 141 // Header: x-wap-profile 142 final String uaProfUrlTagName = mmsConfig.getString( 143     CarrierConfigValuesLoader.CONFIG_UA_PROF_TAG_NAME, 144     CarrierConfigValuesLoader.CONFIG_UA_PROF_TAG_NAME_DEFAULT); 145 if (uaProfUrl != null) { 146     Log.i(MmsService.TAG, "HTTP: UaProfUrl=" + uaProfUrl); 147     connection.setRequestProperty(uaProfUrlTagName, uaProfUrl); 148 } 149 // Add extra headers specified by mms_config.xml's httpparams 150 addExtraHeaders(connection, mmsConfig); 151 // Different stuff for GET and POST 152 if (METHOD_POST.equals(method)) { 153     if (pdu == null    pdu.length &lt; 1) { 154         Log.e(MmsService.TAG, "HTTP: empty pdu"); 155         throw new MmsHttpException(0/*statusCode*/, "Sending empty PDU"); 156     } 157     connection.setDoOutput(true); 158     connection.setRequestMethod(METHOD_POST); 159     if (mmsConfig.getBoolean( 160         CarrierConfigValuesLoader.CONFIG_SUPPORT_HTTP_CHARSET_HEADER, 161         CarrierConfigValuesLoader.CONFIG_SUPPORT_HTTP_CHARSET_HEADER_DEFAULT)) { 162         connection.setRequestProperty(HEADER_CONTENT_TYPE, 163             HEADER_VALUE_CONTENT_TYPE_WITH_CHARSET); 164     } else { 165         connection.setRequestProperty(HEADER_CONTENT_TYPE, 166             HEADER_VALUE_CONTENT_TYPE_WITHOUT_CHARSET); </pre> <p><a href="https://android.googlesource.com/platform/packages/apps/Messaging/+nougat-mr1-release/src/android/support/v7/mms/MmsHttpClient.java">https://android.googlesource.com/platform/packages/apps/Messaging/+nougat-mr1-release/src/android/support/v7/mms/MmsHttpClient.java</a></p>

**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC
	<pre> 167         } 168         if (Log.isLoggable(MmsService.TAG, Log.VERBOSE)) { 169             logHttpHeaders(connection.getRequestProperties()); 170         } 171         connection.setFixedLengthStreamingMode(pdu.length); 172         // Sending request body 173         final OutputStream out = 174             new BufferedOutputStream(connection.getOutputStream()); 175         out.write(pdu); 176         out.flush(); 177         out.close(); 178     } else if (METHOD_GET.equals(method)) { 179         if (Log.isLoggable(MmsService.TAG, Log.VERBOSE)) { 180             logHttpHeaders(connection.getRequestProperties()); 181         } 182         connection.setRequestMethod(METHOD_GET); 183     } 184     // Get response 185     final int responseCode = connection.getResponseCode(); 186     final String responseMessage = connection.getResponseMessage(); 187     Log.d(MmsService.TAG, "HTTP: " + responseCode + " " + responseMessage); 188     if (Log.isLoggable(MmsService.TAG, Log.VERBOSE)) { 189         logHttpHeaders(connection.getHeaderFields()); 190     } 191     if (responseCode / 100 != 2) { 192         throw new MmsHttpException(responseCode, responseMessage); 193     } 194     final InputStream in = new BufferedInputStream(connection.getInputStream()); 195     final ByteArrayOutputStream byteOut = new ByteArrayOutputStream(); 196     final byte[] buf = new byte[4096]; 197     int count = 0; 198     while ((count = in.read(buf)) &gt; 0) { 199         byteOut.write(buf, 0, count); 200     } 201     in.close(); 202     final byte[] responseBody = byteOut.toByteArray(); 203     Log.d(MmsService.TAG, "HTTP: response size=" 204         + (responseBody != null ? responseBody.length : 0)); 205     return responseBody; </pre> <p><a href="https://android.googlesource.com/platform/packages/apps/Messaging/+nougat-mr1-release/src/android/support/v7/mms/MmsHttpClient.java">https://android.googlesource.com/platform/packages/apps/Messaging/+nougat-mr1-release/src/android/support/v7/mms/MmsHttpClient.java</a></p>

**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC
	<pre> 38  * Request to send an MMS 39  */ 40  class SendRequest extends MmsRequest { 41      // Max send response PDU size in bytes (exceeding this may cause problem with 42      // system intent delivery). 43      private static final int MAX_SEND_RESPONSE_SIZE = 1000 * 1024; 44 45      private byte[] mPduData; 46 47      SendRequest(final String locationUri, final Uri pduUri, final PendingIntent sentIntent) { 48          super(locationUri, pduUri, sentIntent); 49      } 50 51      @Override 52      protected boolean loadRequest(final Context context, final Bundle mmsConfig) { 53          mPduData = readPduFromContentUri( 54              context, 55              mPduUri, 56              mmsConfig.getInt( 57                  CarrierConfigValuesLoader.CONFIG_MAX_MESSAGE_SIZE, 58                  CarrierConfigValuesLoader.CONFIG_MAX_MESSAGE_SIZE_DEFAULT)); 59          return (mPduData != null); 60      } 61 62      @Override 63      protected boolean transferResponse(final Context context, final Intent fillIn, 64          final byte[] response) { 65          // SendConf pdus are always small and can be included in the intent 66          if (response != null &amp;&amp; fillIn != null) { 67              if (response.length &gt; MAX_SEND_RESPONSE_SIZE) { 68                  // If the response PDU is too large, it won't be able to fit in 69                  // the PendingIntent to be transferred via system IPC. 70                  return false; 71              } 72              fillIn.putExtra(SmsManager.EXTRA_MMS_DATA, response); 73          } 74          return true; 75      } </pre>

## Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products

US9408055B2	HTC
	<p data-bbox="512 235 1596 308"><a href="https://android.googlesource.com/platform/packages/apps/Messaging/+/nougat-mr1-release/src/android/support/v7/mms/SendRequest.java">https://android.googlesource.com/platform/packages/apps/Messaging/+/nougat-mr1-release/src/android/support/v7/mms/SendRequest.java</a></p> <pre data-bbox="533 430 976 462">public static LocationRequest create ()</pre> <p data-bbox="522 495 1029 519">Create a location request with default parameters.</p> <p data-bbox="522 552 1638 609">Default parameters are for a block accuracy, slowly updated location. It can then be adjusted as required by the applications before passing to the <a href="#">FusedLocationProviderApi</a>.</p> <p data-bbox="546 633 630 657"><b>Returns</b></p> <ul data-bbox="556 682 808 706" style="list-style-type: none"><li>• a new location request</li></ul> <p data-bbox="512 755 1795 787"><a href="https://developers.google.com/android/reference/com/google/android/gms/location/LocationRequest">https://developers.google.com/android/reference/com/google/android/gms/location/LocationRequest</a></p>

## Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products

US9408055B2	HTC
	<p data-bbox="533 250 1745 282"><code>public static final int PRIORITY_BALANCED_POWER_ACCURACY</code></p> <p data-bbox="533 315 1178 342">Used with <code>setPriority(int)</code> to request "block" level accuracy.</p> <p data-bbox="533 375 1644 431">Block level accuracy is considered to be about 100 meter accuracy. Using a coarse accuracy such as this often consumes less power.</p> <p data-bbox="533 456 730 483">Constant Value: 102</p> <p data-bbox="533 537 1745 570"><code>public static final int PRIORITY_HIGH_ACCURACY</code></p> <p data-bbox="533 602 1335 630">Used with <code>setPriority(int)</code> to request the most accurate locations available.</p> <p data-bbox="533 662 961 690">This will return the finest location available.</p> <p data-bbox="533 714 730 742">Constant Value: 100</p> <p data-bbox="533 795 1745 828"><code>public static final int PRIORITY_LOW_POWER</code></p> <p data-bbox="533 860 1157 888">Used with <code>setPriority(int)</code> to request "city" level accuracy.</p> <p data-bbox="533 920 1738 977">City level accuracy is considered to be about 10km accuracy. Using a coarse accuracy such as this often consumes less power.</p> <p data-bbox="533 1002 730 1029">Constant Value: 104</p> <p data-bbox="512 1078 1797 1110"><a href="https://developers.google.com/android/reference/com/google/android/gms/location/LocationRequest">https://developers.google.com/android/reference/com/google/android/gms/location/LocationRequest</a></p>

**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC
	<p><code>public Task&lt;Location&gt; getLastLocation ()</code></p> <p>Returns the best most recent location currently available.</p> <p>If a location is not available, which should happen very rarely, null will be returned. The best accuracy available while respecting the location permissions will be returned.</p> <p>This method provides a simplified way to get location. It is particularly well suited for applications that do not require an accurate location and that do not want to maintain extra logic for location updates.</p> <p><code>public Task&lt;LocationAvailability&gt; getLocationAvailability ()</code></p> <p>Returns the availability of location data. When <code>isLocationAvailable()</code> returns true, then the location returned by <code>getLastLocation()</code> will be reasonably up to date within the hints specified by the active <code>LocationRequest</code>s.</p> <p>If the client isn't connected to Google Play services and the request times out, null is returned.</p> <p>Note it's always possible for <code>getLastLocation()</code> to return null even when this method returns true (e.g. location settings were disabled between calls).</p> <p><a href="https://developers.google.com/android/reference/com/google/android/gms/location/FusedLocationProviderClient">https://developers.google.com/android/reference/com/google/android/gms/location/FusedLocationProviderClient</a></p>

## Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products

US9408055B2	HTC						
	<pre>public Task&lt;Void&gt; requestLocationUpdates (LocationRequest request,     LocationCallback callback, Looper looper)</pre> <p>Requests location updates with a callback on the specified Looper thread.</p> <p>This method is suited for the foreground use cases. For background use cases, the <code>PendingIntent</code> version of the method is recommended, see <code>requestLocationUpdates(LocationRequest, PendingIntent)</code>.</p> <p>Any previous LocationRequests registered on this LocationListener will be replaced.</p> <p>This call will keep the Google Play services connection active, so make sure to call <code>removeLocationUpdates(LocationCallback)</code> when you no longer need it, otherwise you lose the benefits of the automatic connection management.</p> <p>Callbacks for <code>LocationCallback</code> will be made on the specified thread, which must already be a prepared looper thread.</p> <p><b>Parameters</b></p> <table border="1"> <tr> <td><b>request</b></td> <td>The location request for the updates.</td> </tr> <tr> <td><b>callback</b></td> <td>The callback for the location updates.</td> </tr> <tr> <td><b>looper</b></td> <td>The Looper object whose message queue will be used to implement the callback mechanism, or null to make callbacks on the calling thread.</td> </tr> </table> <p><a href="https://developers.google.com/android/reference/com/google/android/gms/location/FusedLocationProviderClient">https://developers.google.com/android/reference/com/google/android/gms/location/FusedLocationProviderClient</a></p>	<b>request</b>	The location request for the updates.	<b>callback</b>	The callback for the location updates.	<b>looper</b>	The Looper object whose message queue will be used to implement the callback mechanism, or null to make callbacks on the calling thread.
<b>request</b>	The location request for the updates.						
<b>callback</b>	The callback for the location updates.						
<b>looper</b>	The Looper object whose message queue will be used to implement the callback mechanism, or null to make callbacks on the calling thread.						



## Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products

US9408055B2	HTC				
	<div style="background-color: #f0f0f0; padding: 5px; margin-bottom: 10px;"> <pre>public Task&lt;Void&gt; requestLocationUpdates (LocationRequest request, PendingIntent callbackIntent)</pre> </div> <p>Requests location updates with a callback on the specified PendingIntent.</p> <p>This method is suited for the background use cases, more specifically for receiving location updates, even when the app has been killed by the system. In order to do so, use a <code>PendingIntent</code> for a started service. For foreground use cases, the <code>LocationCallback</code> version of the method is recommended, see <code>requestLocationUpdates(LocationRequest, LocationCallback, Looper)</code>.</p> <p>Any previously registered requests that have the same PendingIntent (as defined by <code>equals(Object)</code>) will be replaced by this request.</p> <p>Both <code>LocationResult</code> and <code>LocationAvailability</code> are sent to the given PendingIntent. You can extract data from an Intent using <code>hasResult(Intent)</code>, <code>extractResult(Intent)</code>, <code>hasLocationAvailability(Intent)</code>, and <code>extractLocationAvailability(Intent)</code>.</p> <p><b>Parameters</b></p> <table border="1" style="width: 100%; border-collapse: collapse; margin-bottom: 10px;"> <tr> <td style="background-color: #cccccc; padding: 5px;"><code>request</code></td> <td style="padding: 5px;">The location request for the updates.</td> </tr> <tr> <td style="background-color: #cccccc; padding: 5px;"><code>callbackIntent</code></td> <td style="padding: 5px;">A pending intent to be sent for each location update.</td> </tr> </table> <p><b>Returns</b></p> <ul style="list-style-type: none"> <li>a Task for the call, check <code>isSuccessful()</code> to determine if it was successful.</li> </ul> <p><a href="https://developers.google.com/android/reference/com/google/android/gms/location/FusedLocationProviderClient">https://developers.google.com/android/reference/com/google/android/gms/location/FusedLocationProviderClient</a></p>	<code>request</code>	The location request for the updates.	<code>callbackIntent</code>	A pending intent to be sent for each location update.
<code>request</code>	The location request for the updates.				
<code>callbackIntent</code>	A pending intent to be sent for each location update.				

**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC				
	<p><code>public void onLocationAvailability (LocationAvailability locationAvailability)</code></p> <p>Called when there is a change in the availability of location data.</p> <p>When <code>isLocationAvailable()</code> returns <code>false</code> you can assume that location will not be returned in <code>onLocationResult(LocationResult)</code> until something changes in the device's settings or environment. Even when <code>isLocationAvailable()</code> returns <code>true</code> the <code>onLocationResult(LocationResult)</code> may not always be called regularly, however the device location is known and both the most recently delivered location and <code>getLastLocation(GoogleApiClient)</code> will be reasonably up to date given the hints specified by the active <code>LocationRequest</code>s.</p> <p><b>Parameters</b></p> <table border="1" data-bbox="527 639 1738 708"> <tr> <td><code>locationAvailability</code></td> <td>The current status of location availability.</td> </tr> </table> <p><code>public void onLocationResult (LocationResult result)</code></p> <p>Called when device location information is available.</p> <p>The most recent location returned by <code>getLastLocation()</code> is not guaranteed to be immediately fresh, but will be reasonably up to date given the hints specified by the active <code>LocationRequest</code>s.</p> <p><b>Parameters</b></p> <table border="1" data-bbox="527 1019 1738 1088"> <tr> <td><code>result</code></td> <td>The latest location result available.</td> </tr> </table> <p><a href="https://developers.google.com/android/reference/com/google/android/gms/location/LocationCallback">https://developers.google.com/android/reference/com/google/android/gms/location/LocationCallback</a></p>	<code>locationAvailability</code>	The current status of location availability.	<code>result</code>	The latest location result available.
<code>locationAvailability</code>	The current status of location availability.				
<code>result</code>	The latest location result available.				

**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC		
	<p>public abstract void <b>onLocationChanged</b> (<a href="#">Location</a> location)</p> <p>Called when the location has changed.</p> <p><b>Parameters</b></p> <table border="1" data-bbox="520 410 1745 480"> <tr> <td data-bbox="520 410 926 480"><b>location</b></td> <td data-bbox="932 410 1745 480">The updated location.</td> </tr> </table> <p><a href="https://developers.google.com/android/reference/com/google/android/gms/location/LocationListener">https://developers.google.com/android/reference/com/google/android/gms/location/LocationListener</a></p> <p>Public Constructors</p> <hr/> <p>public <b>MapView</b> (<a href="#">Context</a> context)</p> <p>public <b>MapView</b> (<a href="#">Context</a> context, <a href="#">AttributeSet</a> attrs)</p> <p>public <b>MapView</b> (<a href="#">Context</a> context, <a href="#">AttributeSet</a> attrs, int defStyle)</p> <p>public <b>MapView</b> (<a href="#">Context</a> context, <a href="#">GoogleMapOptions</a> options)</p> <p><a href="https://developers.google.com/android/reference/com/google/android/gms/maps/MapView">https://developers.google.com/android/reference/com/google/android/gms/maps/MapView</a></p>	<b>location</b>	The updated location.
<b>location</b>	The updated location.		

**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC		
	<p><code>public void <b>getMapAsync</b> (<a href="#">OnMapReadyCallback</a> callback)</code></p> <p>Returns a non-null instance of the <a href="#">GoogleMap</a>, ready to be used.</p> <p>Note that:</p> <ul style="list-style-type: none"> <li>• This method must be called from the main thread.</li> <li>• The callback will be executed in the main thread.</li> <li>• In the case where Google Play services is not installed on the user's device, the callback will not be triggered until the user installs it.</li> <li>• The <a href="#">GoogleMap</a> object provided by the callback is non-null.</li> </ul> <p><b>Parameters</b></p> <table border="1" data-bbox="527 688 1738 753"> <tr> <td><code>callback</code></td> <td>The callback object that will be triggered when the map is ready to be used.</td> </tr> </table> <p><code>public final void <b>onCreate</b> (<a href="#">Bundle</a> savedInstanceState)</code></p> <p>You must call this method from the parent Activity/Fragment's corresponding method.</p> <p><a href="https://developers.google.com/android/reference/com/google/android/gms/maps/MapView">https://developers.google.com/android/reference/com/google/android/gms/maps/MapView</a></p>	<code>callback</code>	The callback object that will be triggered when the map is ready to be used.
<code>callback</code>	The callback object that will be triggered when the map is ready to be used.		
<p>[1D] transmitting IP-based messages including a location of the first device to the respective second devices;</p>	<p>HTC infringes directly and/or indirectly by performing, inducing others to perform, and/or contributing to the performance of transmitting IP-based messages including a location of the first device to the respective second devices.</p> <p>For example, users send their location to a server and receive the location of other devices with whom the location is being shared. To send a location to the network, a user enables location service which enables the device to determine and send its location. If location service is already enabled, the device sends its location to the server as needed by the application (e.g. Google Maps). If location service is not enabled, the application will ask the user to enable location service in order to continue with full functionality, which includes using the device's location. Google Maps applications receive the location of other devices when those devices have location service enabled while using the same respective application. Android Device Manager and Google Maps use the received locations to display those locations on the map, indicating the</p>		

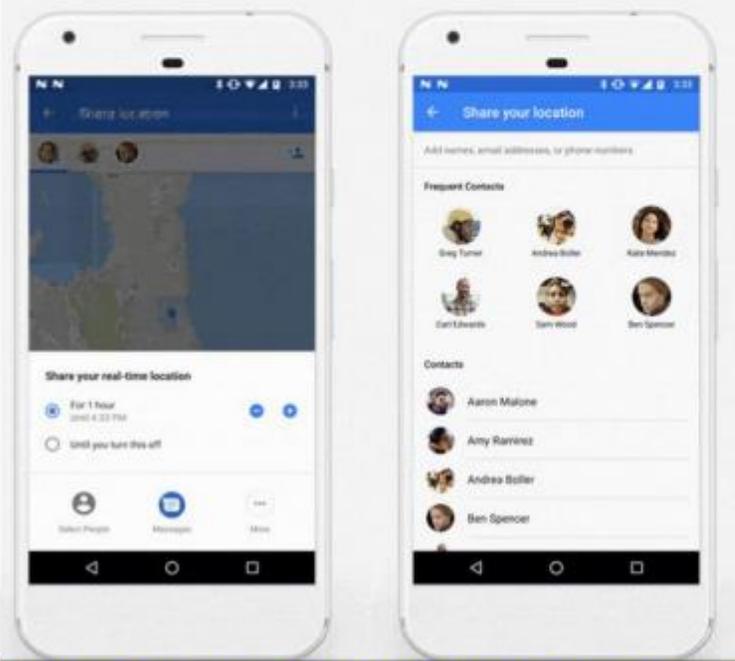
**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

<b>US9408055B2</b>	<b>HTC</b>
	<p>locations of other devices.</p> <p>See, e.g., location sharing including corresponding code described above with regard to limitation [1C].</p> <p>Using Google Maps, a user enables location services to send its location the network, but the user can also choose to share its location, as shown below. Again, each device that participates is able to see the location of the other device using Google Maps' share your location feature. For example:</p>

**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC
	<p data-bbox="527 233 1304 277"><b>HTC One V™ – Google Location Service &amp; GPS</b></p> <p data-bbox="527 321 1717 428">Google Maps lets you track your current location, view real-time traffic situations, and receive detailed directions to your destination. It also provides a search tool where you can locate a place of interest or an address on a vector or aerial map, or view locations in street level.</p> <p data-bbox="527 472 898 500"><b>Turning on Location Services</b></p> <div data-bbox="527 516 1717 954"> </div> <ol data-bbox="527 980 1717 1154" style="list-style-type: none"> <li>1. From the Home Screen, slide the <b>Notifications</b> panel open.</li> <li>2. In the top right corner, tap <b>Settings</b>.</li> <li>3. Tap <b>Location</b>.</li> <li>4. Make your selection by tapping <b>Google's location service</b>, <b>Use GPS satellites</b>, or both. <b>Note:</b> You will need to accept the location consent terms and conditions.</li> </ol>

## Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products

US9408055B2	HTC
	<ol style="list-style-type: none"><li>1. If you haven't already, add their Gmail address to your <a href="#">Google Contacts</a>.</li><li>2. On your Android phone or tablet, open the Google Maps app  and sign in. <a href="#">Learn how to sign in.</a></li><li>3. Tap the Menu  &gt; <b>Share location</b> &gt; Add People .</li><li>4. Choose how long you want to share your location.</li><li>5. Tap <b>Select People</b>.<ul style="list-style-type: none"><li>• If you're asked about your contacts, give Google Maps access.</li></ul></li><li>6. Choose who you want to share with.</li><li>7. Tap <b>Share</b>.</li></ol> <p><a href="https://support.google.com/plus/answer/3302509?co=GENIE.Platform%3DAndroid&amp;hl=en">https://support.google.com/plus/answer/3302509?co=GENIE.Platform%3DAndroid&amp;hl=en</a></p>  <p><a href="https://arstechnica.com/gadgets/2017/03/location-sharing-finally-returns-to-google-maps/">https://arstechnica.com/gadgets/2017/03/location-sharing-finally-returns-to-google-maps/</a></p>

**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC
	 <p><a href="https://arstechnica.com/gadgets/2017/03/location-sharing-finally-returns-to-google-maps/">https://arstechnica.com/gadgets/2017/03/location-sharing-finally-returns-to-google-maps/</a></p> <p>Below are exemplary methods used by Google applications to obtain, send, and receive locations.</p>



**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC
	<p data-bbox="527 233 982 261">The Google Play services Location API</p> <hr/> <p data-bbox="527 298 1356 347">The Google Play services <a href="#">Location API</a> is the preferred method for adding location awareness to your Android application. It includes functionality that lets you:</p> <ul data-bbox="552 371 1184 505" style="list-style-type: none"> <li>• Determine the device location.</li> <li>• Listen for location changes.</li> <li>• Determine the mode of transportation, if the device is moving.</li> <li>• Create and monitor predefined geographical regions, known as geofences.</li> </ul> <p data-bbox="527 531 1360 662">The location APIs make it easy for you to build power efficient, location-aware applications. Like the Google Maps Android API, the Location API is distributed as part of the Google Play services SDK. For more information on the Location API, please refer to the Android training class <a href="#">Making Your App Location Aware</a> or the <a href="#">Location API Reference</a>. Code examples are included as part of the Google Play services SDK.</p> <p data-bbox="527 667 1331 695"><a href="https://developers.google.com/maps/documentation/android-api/location">https://developers.google.com/maps/documentation/android-api/location</a></p> <div data-bbox="527 776 1339 1177"> <p>The diagram illustrates the architecture of Google Play services on a device. On the left, a box labeled 'Device' contains 'Your app' (represented by an Android robot icon) and the 'Google Play services library'. Inside the library is the 'Google API Client'. On the right, a dashed box labeled 'Google Play services' contains three service components: 'Games service', 'Drive service', and 'Other services'. Bidirectional arrows connect 'Your app' to the 'Google API Client', and the 'Google API Client' to each of the three service components.</p> </div> <p data-bbox="527 1193 1331 1268">Figure 1: An illustration showing how the Google API Client provides an interface for connecting and making calls to any of the available Google Play services such as Google Play Games and Google Drive.</p> <p data-bbox="527 1273 1247 1300"><a href="https://developers.google.com/android/guides/api-client#Starting">https://developers.google.com/android/guides/api-client#Starting</a></p>

**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC
	<h3 data-bbox="527 233 919 261">Get the Last Known Location</h3> <p data-bbox="527 289 1545 386">Once you have connected to Google Play services and the location services API, you can get the last known location of a user's device. When your app is connected to these you can use the fused location provider's <code>getLastLocation()</code> method to retrieve the device location. The precision of the location returned by this call is determined by the permission setting you put in your app manifest, as described in the <a href="#">Specify App Permissions</a> section of this document.</p> <p data-bbox="527 407 1545 480">To request the last known location, call the <code>getLastLocation()</code> method, passing it your instance of the <code>GoogleApiClient</code> object. Do this in the <code>onConnected()</code> callback provided by Google API Client, which is called when the client is ready. The following code snippet illustrates the request and a simple handling of the response:</p> <pre data-bbox="527 493 1545 781">public class MainActivity extends ActionBarActivity implements     ConnectionCallbacks, OnConnectionFailedListener {     ...     @Override     public void onConnected(Bundle connectionHint) {         mLstLocation = LocationServices.FusedLocationApi.getLastLocation(             mGoogleApiClient);         if (mLstLocation != null) {             mLatitudeText.setText(String.valueOf(mLstLocation.getLatitude()));             MLongitudeText.setText(String.valueOf(mLstLocation.getLongitude()));         }     } }</pre> <p data-bbox="527 797 1545 842">The <code>getLastLocation()</code> method returns a <code>Location</code> object from which you can retrieve the latitude and longitude coordinates of a geographic location. The location object returned may be null in rare cases when the location is not available.</p> <p data-bbox="527 849 1289 878"><a href="https://developer.android.com/training/location/retrieve-current.html">https://developer.android.com/training/location/retrieve-current.html</a></p>

**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC
	<p>Determining the user's current location</p> <hr/> <p>The Geolocation API offers a simple, "one-shot" method to obtain the user's location: <code>getCurrentPosition()</code>. A call to this method asynchronously reports on the user's current location.</p> <pre> window.onload = function() {   var startPos;   var geoSuccess = function(position) {     startPos = position;     document.getElementById('startLat').innerHTML = startPos.coords.latitude;     document.getElementById('startLon').innerHTML = startPos.coords.longitude;   };   navigator.geolocation.getCurrentPosition(geoSuccess); };     </pre> <p>If this is the first time that an application on this domain has requested permissions, the browser typically checks for user consent. Depending on the browser, there may also be preferences to always allow—or disallow—permission lookups, in which case the confirmation process is bypassed.</p> <p>Depending on the location device your browser is using, the position object might actually contain a lot more than just latitude and longitude; for example, it might include an altitude or a direction. You can't tell what extra information that location system uses until it actually returns the data.</p> <p><a href="https://developers.google.com/web/fundamentals/native-hardware/user-location/">https://developers.google.com/web/fundamentals/native-hardware/user-location/</a></p>
<p>[1E] presenting, via an interactive display of the first device, an interactive map and a plurality of user selectable symbols corresponding to the plurality of second devices, wherein the symbols are positioned on the map at respective positions</p>	<p>HTC infringes directly and/or indirectly by performing, inducing others to perform, and/or contributing to the performance of presenting, via an interactive display of the first device, an interactive map and a plurality of user selectable symbols corresponding to the plurality of second devices, wherein the symbols are positioned on the map at respective positions corresponding to the respective locations of the second devices.</p> <p>For example, the Accused Products use Android Device Manager, and Google Maps to display an interface with a map and symbols representing devices.</p> <p>Using Android Device Manager, the user is presented with a map that appears to be based on or imported from Google Maps. The map is interactive because the user can pan, zoom, and make selections, among other interactivities. Depending on the number of devices linked to the Google Account, Android Device</p>

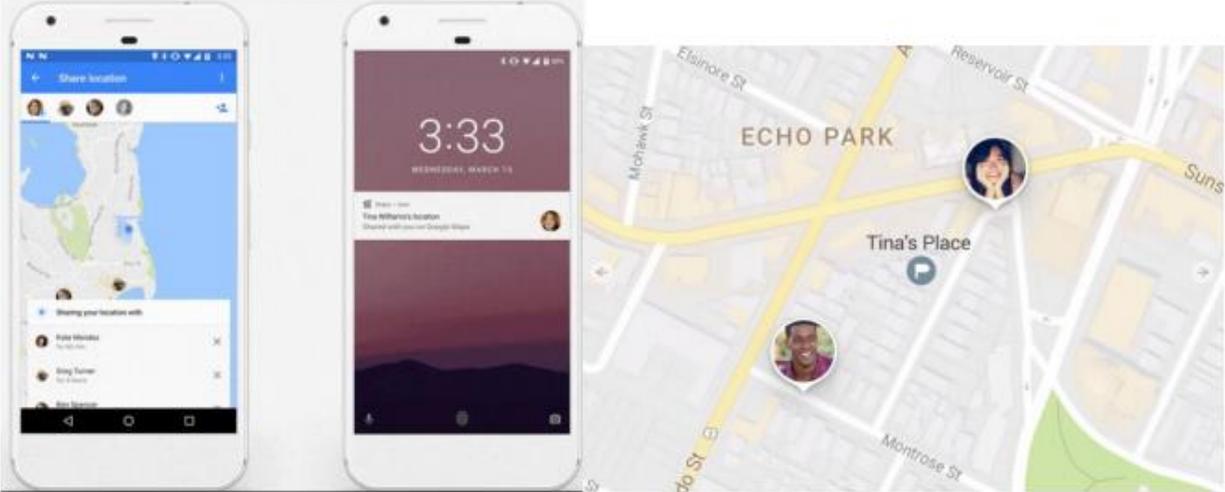
**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC
<p>corresponding to the respective locations of the second devices;</p>	<p>Manager places symbols on the map and in a list, where those symbols represent the linked devices and their positions. The symbols are selectable by touching the display. The map is georeferenced, at least because its contents, including the symbols, are placed and spaced according to geographical coordinates.</p> <p><b><u>Regarding Google Maps</u></b>, Google Latitude, Google Plus, Google Hangouts, Google Messages, Google Allo, Google Duo, Google Chrome, and Android Messenger, the Accused Products display, to the user on the display of the first device, a map with one or more symbols corresponding to one or more second users (or second devices corresponding to the second users). The map is interactive because the user may control the display of the map, e.g., pan, zoom, and/or effect change to the map in an otherwise interactive manner. The map is georeferenced for at least the reason that one or more symbols are associated with spatial locations, i.e., coordinates. The symbols are user-selectable because a user may touch the display to select the user or device associated with the symbol. For example, the user may input a touch selection directly on or near the portion of the display corresponding to the symbol’s coordinates on the map to effect a selection of the user or device.</p> <p><b><u>Exemplary Support for Google Maps:</u></b> Using Google Maps and its location sharing feature, the user is presented with a map that is interactive because the user can pan, zoom, and make selections, among other interactivities. Depending on how many other devices or Google Accounts are sharing their locations, Google Maps places symbols on the map and in a list, where those symbols represent the linked devices and their positions. The symbols are selectable by touching the display. The map is georeferenced, at least because its contents, including the symbols, are placed and spaced according to geographical coordinates.</p>

## Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products

US9408055B2	HTC
	<p data-bbox="548 240 886 284"><b>Location settings</b></p> <hr/> <p data-bbox="548 329 995 357"><b>Turning location services on or off</b></p> <hr/> <p data-bbox="548 386 1398 410">In order to find your location on HTC One, you need to enable location sources.</p> <ol data-bbox="579 444 1566 639" style="list-style-type: none"><li data-bbox="579 444 1031 469">1. Go to Settings, and then tap <b>Location</b>.</li><li data-bbox="579 493 1247 518">2. Tap the <b>On/Off</b> switch to turn location services on and off.</li><li data-bbox="579 542 1566 639">3. Under Location sources, select the location mode you want. For example, for a better estimate of your location, choose <b>High accuracy</b>. To save battery power, choose <b>Battery saving</b>.</li></ol> <p data-bbox="548 672 1581 802">Turning off a location source (for example GPS) means no applications on HTC One will collect your location data through that location source. However, third party applications may collect — and HTC One may continue to provide — location data through other sources, including through Wi-Fi and signal triangulation.</p> <p data-bbox="548 867 802 911"><b>Google Maps</b></p> <hr/> <p data-bbox="548 956 802 984"><b>About Google Maps</b></p> <hr/> <p data-bbox="548 1008 1572 1073">Google Maps™ lets you track your current location, view real-time traffic situations, and receive detailed directions to your destination.</p> <p data-bbox="548 1097 1608 1154">It also provides a search tool where you can locate places of interest or an address on a map, or view locations at street level.</p>

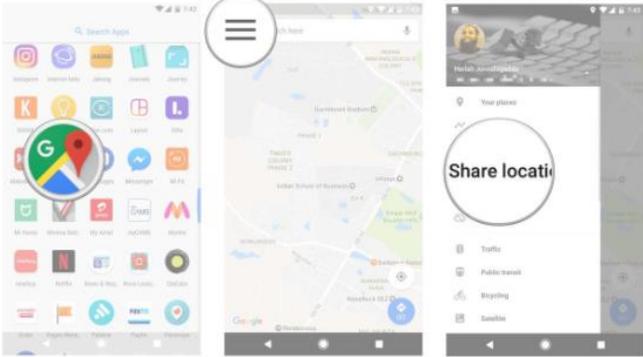
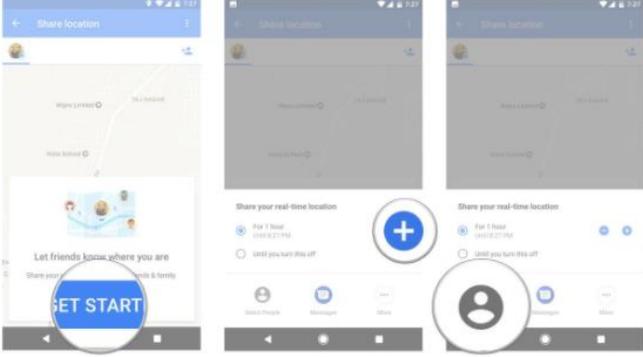
## Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products

US9408055B2	HTC
	 <p><a href="https://arstechnica.com/gadgets/2017/03/location-sharing-finally-returns-to-google-maps/">https://arstechnica.com/gadgets/2017/03/location-sharing-finally-returns-to-google-maps/</a></p> <h3>See where someone is</h3> <p>If someone shares their location with you, you can see them on the map.</p> <ol style="list-style-type: none"><li>1. Open the Google Maps app</li><li>2. Tap the Menu ≡ &gt; <b>Share location</b>.</li><li>3. Choose someone.</li></ol> <p>• To see an updated location, tap on a friend's icon &gt; More ≡ &gt; <b>Refresh</b>.</p> <h3>Stop seeing someone's location</h3> <ol style="list-style-type: none"><li>1. Open the Google Maps app</li><li>2. On the map, tap their icon.</li><li>3. At the bottom, tap More ≡</li><li>4. To temporarily hide someone, tap <b>Hide from map</b>. You can stop hiding them at any time.</li></ol> <p><b>Note:</b> You can stop someone's location from ever appearing on your map. <a href="#">Learn how to block another person's account.</a></p> <p><a href="https://support.google.com/maps/answer/7326816?hl=en&amp;ref_topic=3092425&amp;co=GENIE.Platform%3DAndroid&amp;oco=1">https://support.google.com/maps/answer/7326816?hl=en&amp;ref_topic=3092425&amp;co=GENIE.Platform%3DAndroid&amp;oco=1</a></p>

## Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products

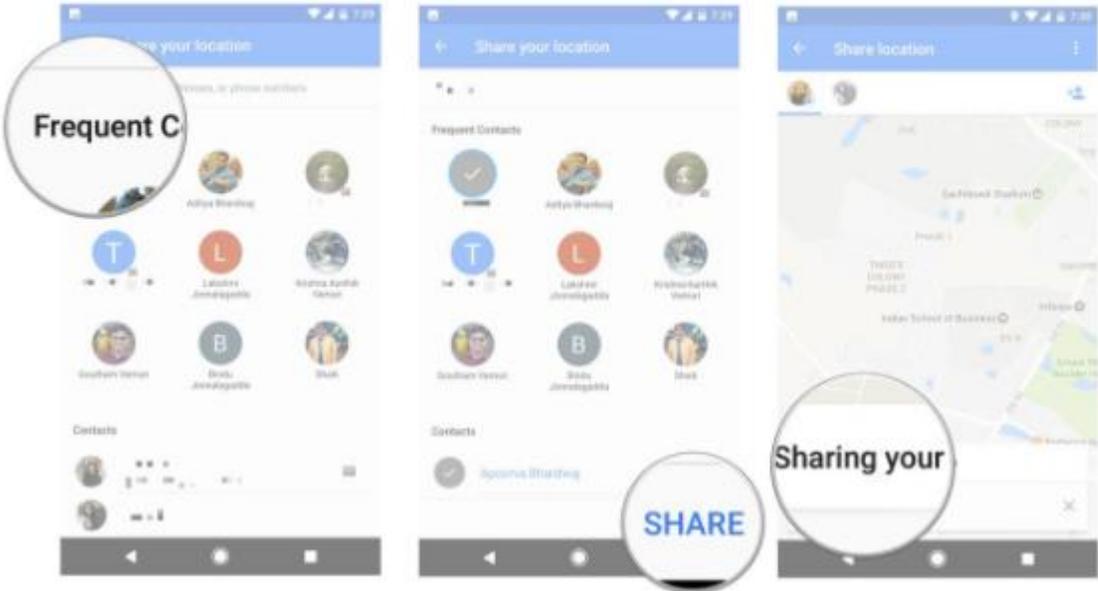
US9408055B2	HTC
	<h3>Share your E.T.A</h3> <p>After you start your drive, you can share your destination, estimated arrival time, and where you are on the route.</p> <ol style="list-style-type: none"><li>1. Open the Google Maps app .</li><li>2. Set a driving destination. <a href="#">Learn how to navigate to a place.</a></li><li>3. After you start navigation, tap More ^ &gt; <b>Share trip progress.</b></li><li>4. Choose a person from the list.</li><li>5. Tap <b>Share.</b></li><li>6. Location Sharing will stop when you reach your destination or stop navigating.</li></ol> <ul style="list-style-type: none"><li>• To stop sharing before you arrive, tap More ^ &gt; <b>Stop sharing.</b></li></ul> <h3>See where someone is</h3> <p>If someone shares their location with you, you can see them on the map.</p> <ol style="list-style-type: none"><li>1. Open the Google Maps app .</li><li>2. Tap Menu ≡ &gt; <b>Location sharing.</b></li><li>3. Choose someone.</li></ol> <ul style="list-style-type: none"><li>• To see an updated location, tap on a friend's icon &gt; More ⋮ &gt; <b>Refresh.</b></li></ul> <h3>Stop seeing someone's location</h3> <ol style="list-style-type: none"><li>1. Open the Google Maps app .</li><li>2. On the map, tap their icon.</li><li>3. At the bottom, tap More ^ .</li><li>4. To temporarily hide someone, tap <b>Hide from map.</b> You can stop hiding them at any time.</li></ol> <p><b>Note:</b> You can stop someone's location from ever appearing on your map. <a href="#">Learn how to block another person's account.</a></p> <p><a href="https://support.google.com/maps/answer/7326816?co=GENIE.Platform%3DAndroid&amp;oco=1">https://support.google.com/maps/answer/7326816?co=GENIE.Platform%3DAndroid&amp;oco=1</a></p>

## Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products

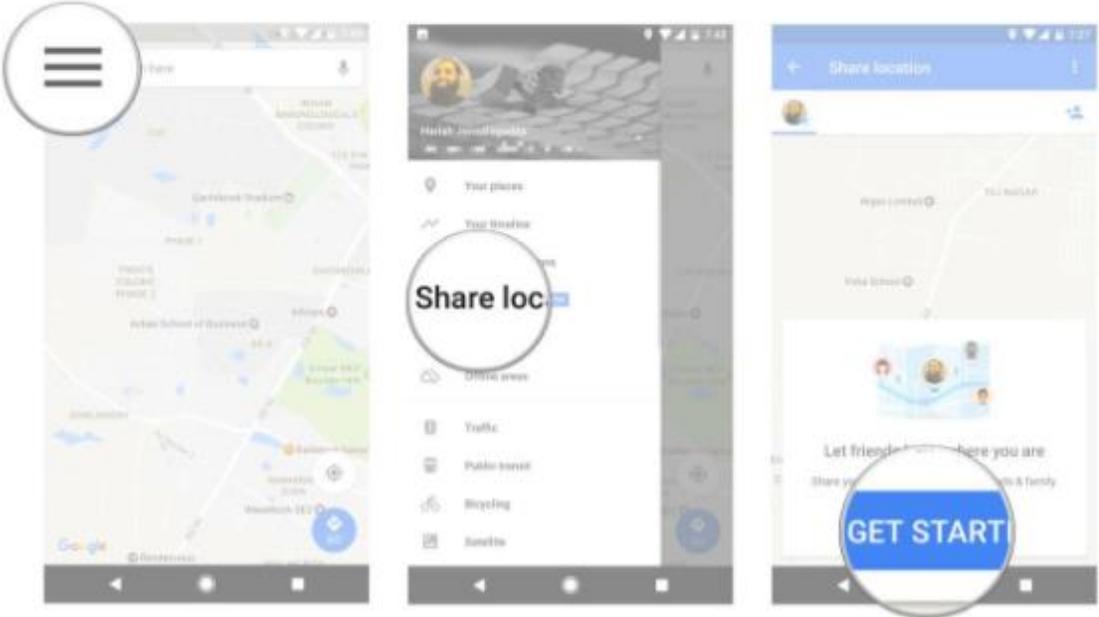
US9408055B2	HTC
	<p data-bbox="520 240 1129 272"><b>How to share your location in Google Maps</b></p> <ol data-bbox="520 298 1129 386" style="list-style-type: none"><li>1. Open Google Maps from the app drawer or the home screen.</li><li>2. Tap the <b>hamburger menu</b> (the three horizontal lines) on the top left corner of the screen.</li><li>3. Select <b>Share location</b>.</li></ol>  <ol data-bbox="520 808 1129 906" style="list-style-type: none"><li>4. Tap <b>Get Started</b>.</li><li>5. Use the <b>+</b> icon to select a time period or select the <b>Until you turn this off</b> setting to share your location indefinitely.</li><li>6. Tap <b>Select People</b>.</li></ol>  <p data-bbox="520 1331 1360 1364"><a href="https://www.androidcentral.com/how-share-location-google-maps">https://www.androidcentral.com/how-share-location-google-maps</a></p>



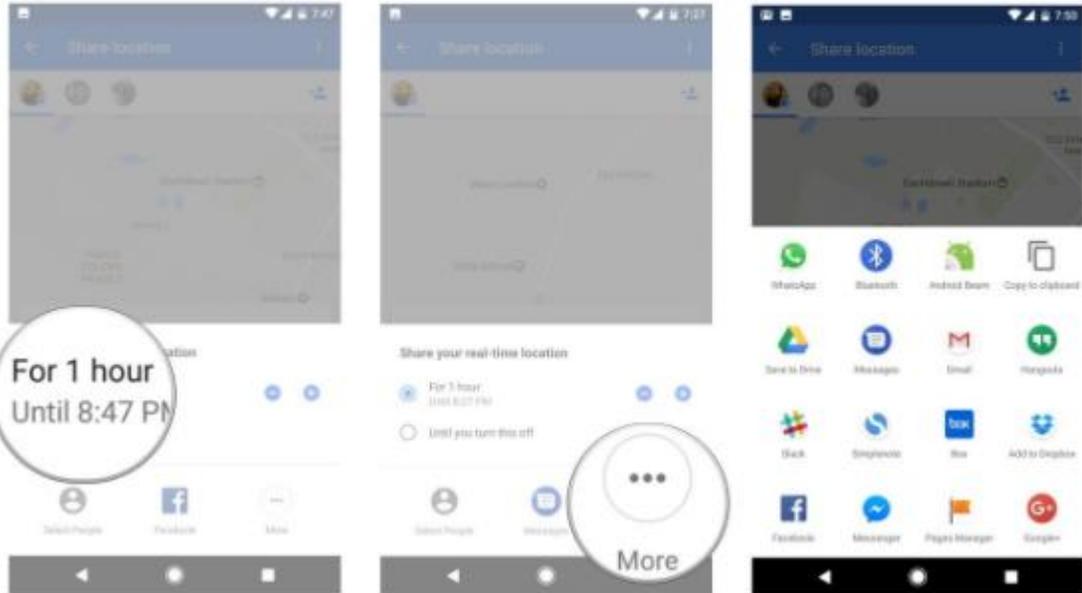
### Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products

US9408055B2	HTC
	<p data-bbox="527 253 1577 310">7. You'll see a list of your frequent contacts at the top, along with a full list of contacts. Pick the contacts by tapping their name.</p> <p data-bbox="527 337 1457 367">8. Once you've selected the contacts you want to share your location to, tap <b>Share</b>.</p> <p data-bbox="527 394 1419 423">9. You'll see a message saying that the selected contact can view your location.</p>  <p data-bbox="506 1166 1356 1195"><a href="https://www.androidcentral.com/how-share-location-google-maps">https://www.androidcentral.com/how-share-location-google-maps</a></p>

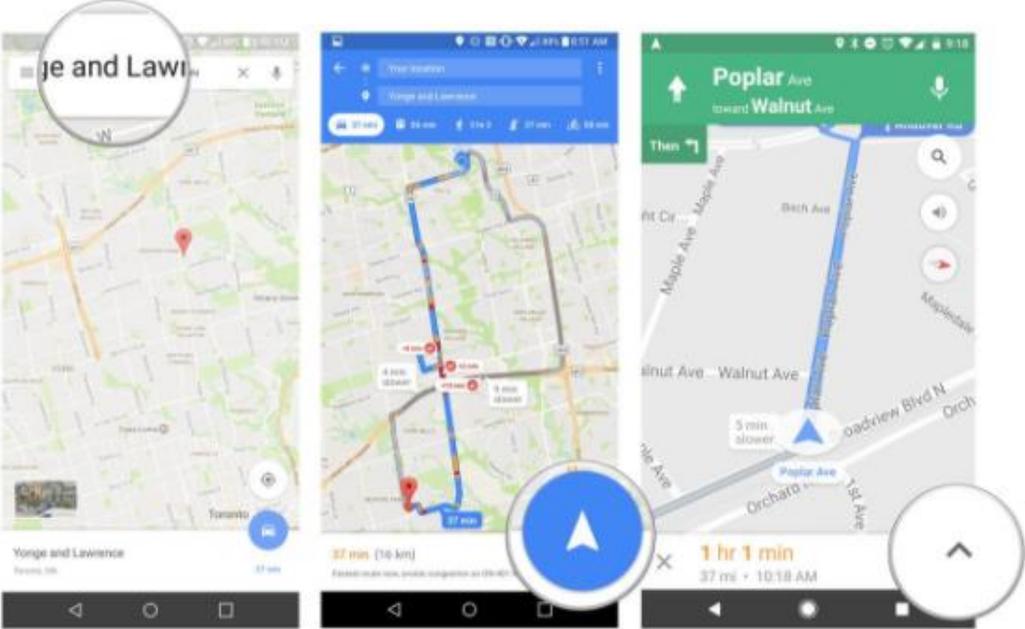
## Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products

US9408055B2	HTC
	<h3 data-bbox="520 245 1255 289">How to create a shareable link</h3> <p data-bbox="520 331 1461 358">You can also create a link and use it to share your location easily. Here's how to do it:</p> <ol data-bbox="520 410 1234 548" style="list-style-type: none"><li data-bbox="520 410 1234 438">1. Tap the hamburger menu on the top left corner of the screen.</li><li data-bbox="520 467 800 495">2. Select Share location.</li><li data-bbox="520 524 737 552">3. Tap Get Started.</li></ol>  <p data-bbox="506 1263 1356 1295"><a href="https://www.androidcentral.com/how-share-location-google-maps">https://www.androidcentral.com/how-share-location-google-maps</a></p>

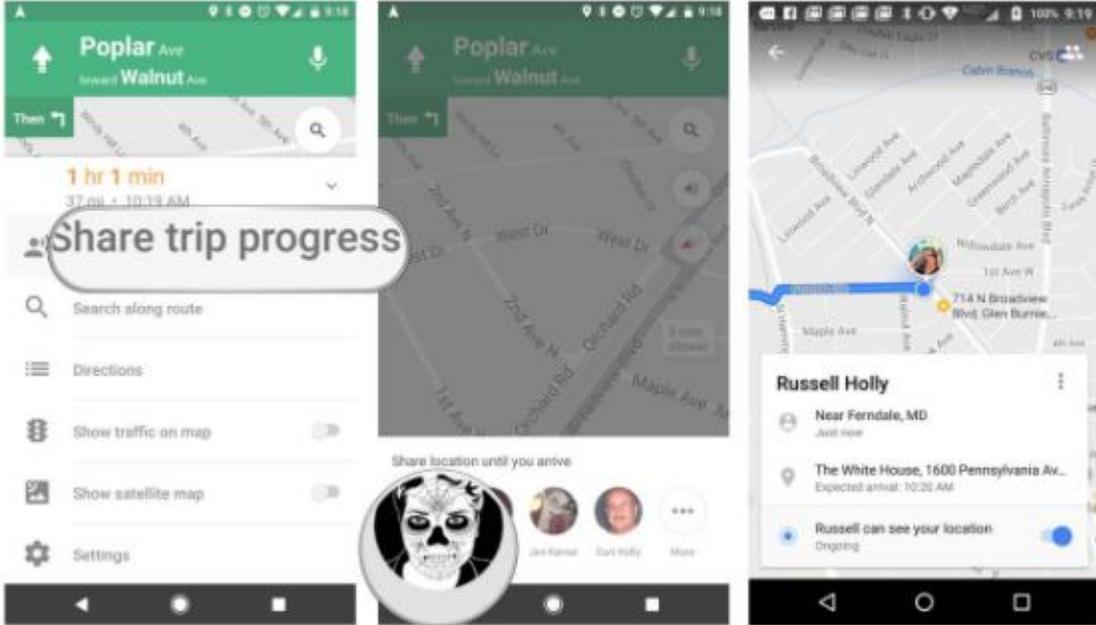
### Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products

US9408055B2	HTC
	<p data-bbox="520 245 1213 272">4. Select the <b>amount of time</b> you want to share your location.</p> <p data-bbox="520 302 680 329">5. Tap <b>More</b>.</p> <p data-bbox="520 358 1633 418">6. Select <b>your app of choice</b> to create and send a <b>unique URL</b> that broadcasts your current location. You can email it, send the link via Messenger, or even tweet it to the intended recipient.</p>  <p data-bbox="510 1117 1356 1149"><a href="https://www.androidcentral.com/how-share-location-google-maps">https://www.androidcentral.com/how-share-location-google-maps</a></p>

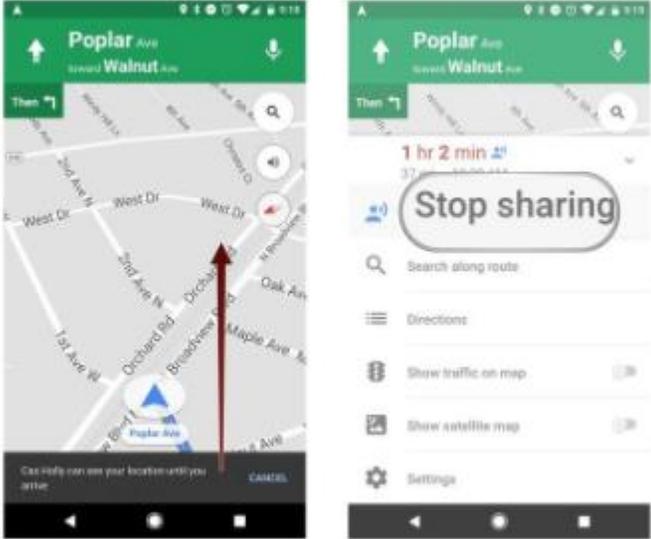
**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC
	<p data-bbox="527 240 1430 337"><b>How to share your navigation directions while you walk, drive or transit</b></p> <p data-bbox="527 375 1556 464">One of the best ways to share your location with someone is during a drive, walk or transit. If you're meeting a friend or family member somewhere, or navigating towards their home, Google Maps lets you share your location with them for the duration of the trip. It's magic!</p> <ol data-bbox="527 513 1398 643" style="list-style-type: none"> <li>1. In the <b>search bar</b> enter your destination.</li> <li>2. Pick your navigation type (drive, transit, walk) and press the blue <b>navigate</b> button.</li> <li>3. Tap the arrow next to the <b>time-to-destination</b> number at the bottom of the screen.</li> </ol>  <p data-bbox="512 1360 1356 1393"><a href="https://www.androidcentral.com/how-share-location-google-maps">https://www.androidcentral.com/how-share-location-google-maps</a></p>

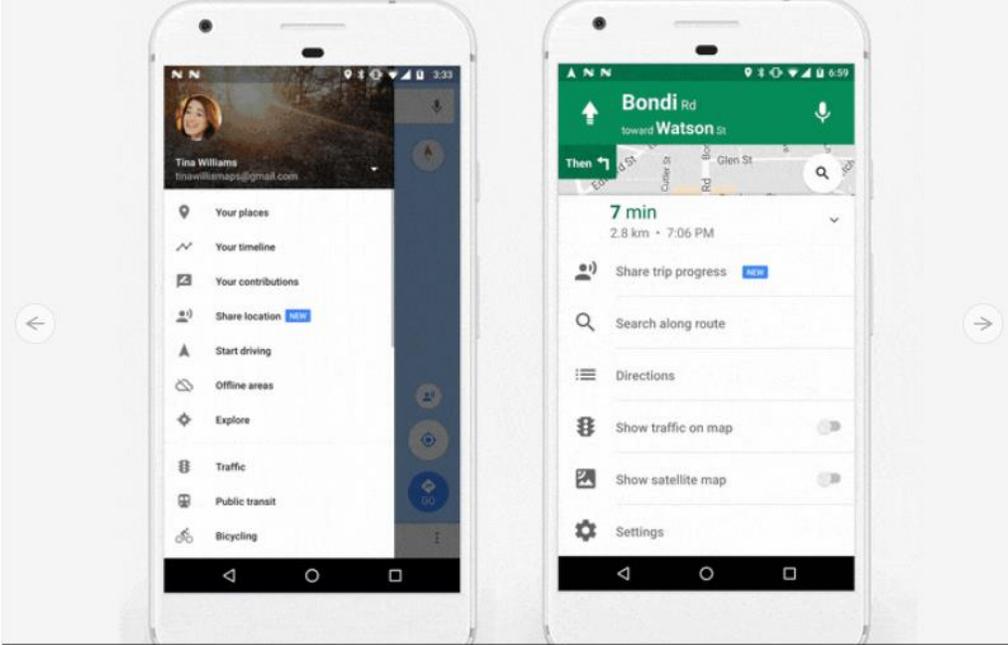
### Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products

US9408055B2	HTC
	<p data-bbox="527 240 835 266">4. Tap Share trip progress.</p> <p data-bbox="527 297 1150 323">5. Choose one or more contacts to share trip progress.</p>  <p data-bbox="537 1027 1339 1053">You can also stop sharing your location with someone before a trip ends.</p> <p data-bbox="512 1097 1356 1123"><a href="https://www.androidcentral.com/how-share-location-google-maps">https://www.androidcentral.com/how-share-location-google-maps</a></p>

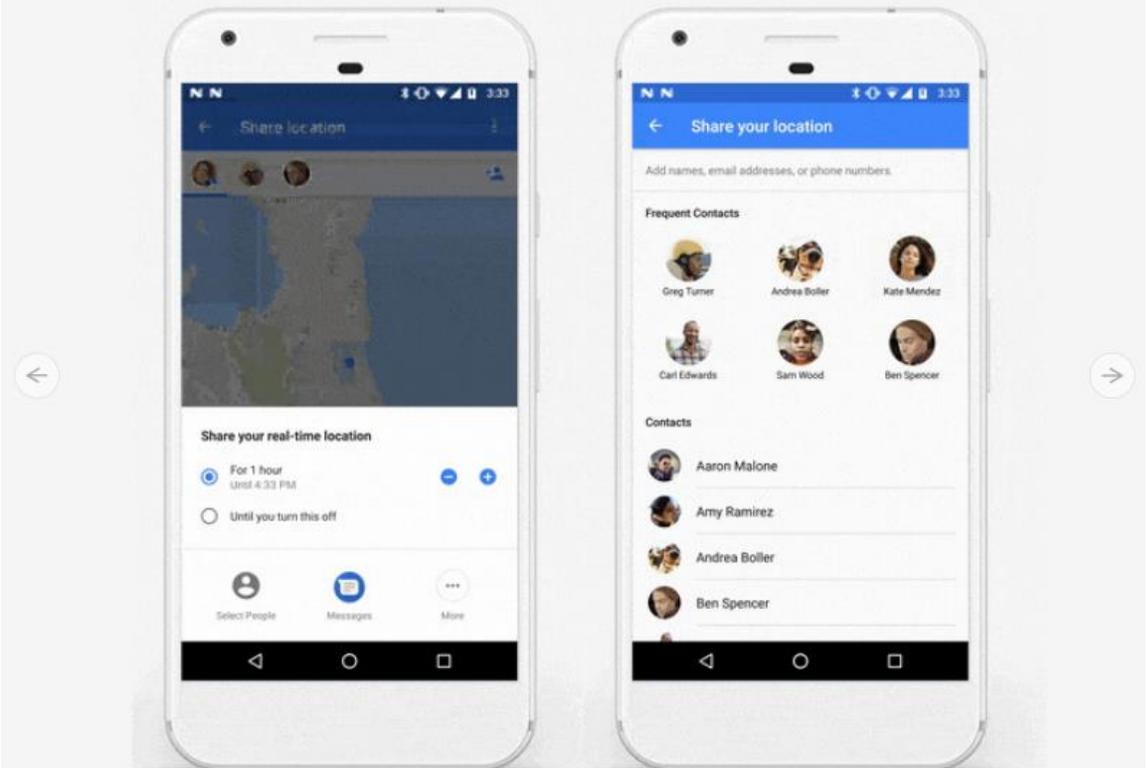
## Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products

US9408055B2	HTC
	<ol style="list-style-type: none"><li data-bbox="533 245 1465 272">1. Tap the arrow next to the <b>time-to-destination</b> number at the bottom of the screen.</li><li data-bbox="533 302 768 329">2. Tap Stop sharing.</li></ol> <div data-bbox="747 383 1398 922"></div> <p data-bbox="541 976 638 1003">That's it!</p> <p data-bbox="541 1045 1608 1073">Are you excited that location sharing is back in Google Maps? How often do you use the feature?</p> <p data-bbox="512 1117 1356 1149"><a href="https://www.androidcentral.com/how-share-location-google-maps">https://www.androidcentral.com/how-share-location-google-maps</a></p>

## Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products

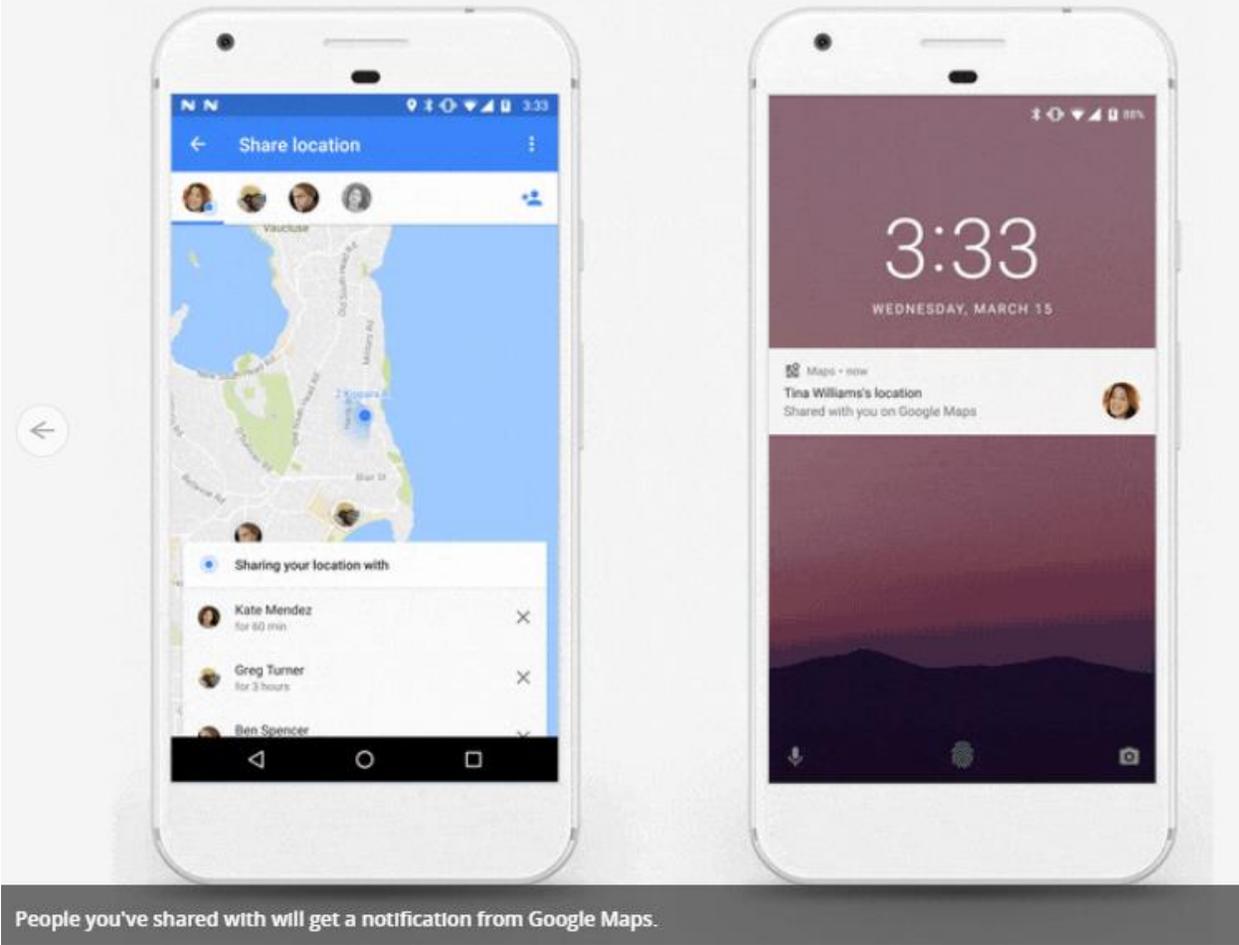
US9408055B2	HTC
	 <p data-bbox="514 893 1522 950">Ways to start location sharing: (left) from the app navigation drawer, and (right) from the menu, you can drag up while navigating somewhere.</p> <p data-bbox="514 982 1659 1031"><a href="https://arstechnica.com/gadgets/2017/03/location-sharing-finally-returns-to-google-maps/">https://arstechnica.com/gadgets/2017/03/location-sharing-finally-returns-to-google-maps/</a></p>

### Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products

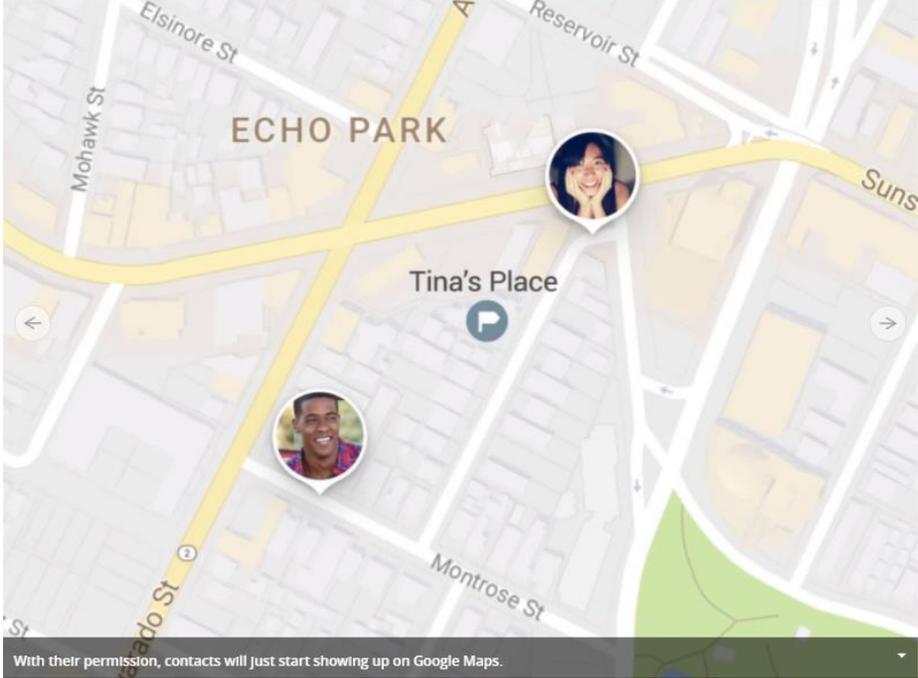
US9408055B2	HTC
	 <p data-bbox="514 1015 1661 1063">You can pick how long you want to share your location for, and with whom you want to share it.</p> <p data-bbox="514 1096 1661 1128"><a href="https://arstechnica.com/gadgets/2017/03/location-sharing-finally-returns-to-google-maps/">https://arstechnica.com/gadgets/2017/03/location-sharing-finally-returns-to-google-maps/</a></p>



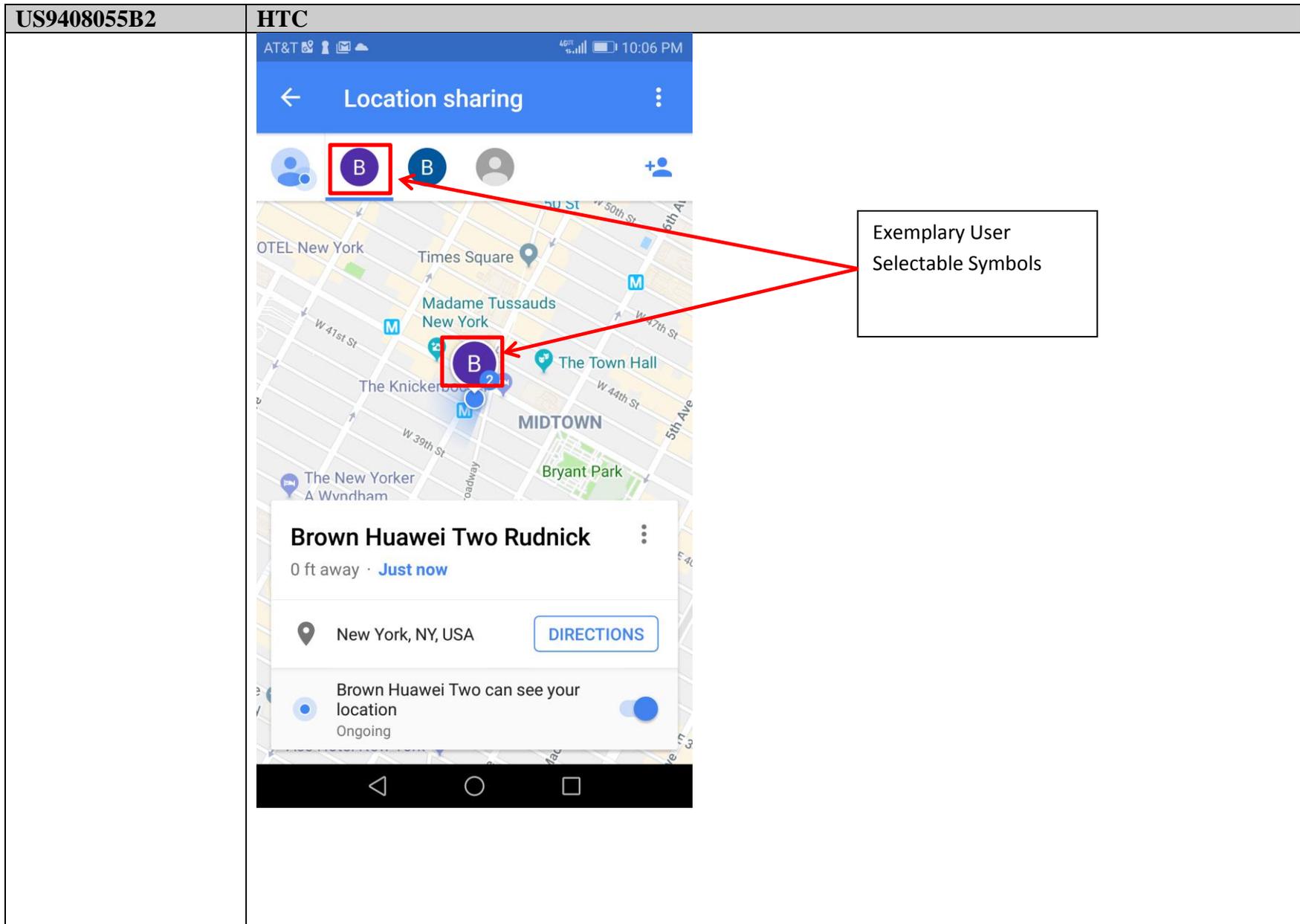
**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC
	 <p data-bbox="514 1144 1176 1177">People you've shared with will get a notification from Google Maps.</p> <p data-bbox="514 1218 1659 1258"><a href="https://arstechnica.com/gadgets/2017/03/location-sharing-finally-returns-to-google-maps/">https://arstechnica.com/gadgets/2017/03/location-sharing-finally-returns-to-google-maps/</a></p>

## Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products

US9408055B2	HTC
	 <p>The screenshot shows a Google Maps view of Echo Park, Los Angeles. Two circular profile pictures of contacts are overlaid on the map, indicating their current locations. One contact is near Tina's Place, and the other is near the intersection of Montrose St and Reservoir St. A text overlay at the bottom of the map reads: "With their permission, contacts will just start showing up on Google Maps." Below the map is a blue hyperlink: <a href="https://arstechnica.com/gadgets/2017/03/location-sharing-finally-returns-to-google-maps/">https://arstechnica.com/gadgets/2017/03/location-sharing-finally-returns-to-google-maps/</a>. Below the link is the text: <b><u>Exemplary Google Maps Screenshots:</u></b></p>

### Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products



**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC
	<p><b><u>Exemplary Source Code:</u></b></p> <p>The above functionality is performed at least in part by the following publicly available source code and/or source code that invokes or is invoked by the following source code (or a substantially similar copy compiled and loaded onto the Accused Products by HTC). AGIS reserves the right to supplement these contentions with additional source code as discovery progresses and as additional source code is made available.</p> <pre data-bbox="533 618 1738 662">public static LocationRequest create ()</pre> <p>Create a location request with default parameters.</p> <p>Default parameters are for a block accuracy, slowly updated location. It can then be adjusted as required by the applications before passing to the <code>FusedLocationProviderApi</code>.</p> <p><b>Returns</b></p> <ul style="list-style-type: none"> <li>• a new location request</li> </ul> <p><a href="https://developers.google.com/android/reference/com/google/android/gms/location/LocationRequest">https://developers.google.com/android/reference/com/google/android/gms/location/LocationRequest</a></p>

## Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products

US9408055B2	HTC
	<p data-bbox="533 248 1745 285"><code>public static final int PRIORITY_BALANCED_POWER_ACCURACY</code></p> <p data-bbox="533 315 1178 342">Used with <code>setPriority(int)</code> to request "block" level accuracy.</p> <p data-bbox="533 371 1644 431">Block level accuracy is considered to be about 100 meter accuracy. Using a coarse accuracy such as this often consumes less power.</p> <p data-bbox="533 456 730 483">Constant Value: 102</p> <p data-bbox="533 537 1745 574"><code>public static final int PRIORITY_HIGH_ACCURACY</code></p> <p data-bbox="533 604 1335 631">Used with <code>setPriority(int)</code> to request the most accurate locations available.</p> <p data-bbox="533 660 961 688">This will return the finest location available.</p> <p data-bbox="533 712 730 740">Constant Value: 100</p> <p data-bbox="533 794 1745 831"><code>public static final int PRIORITY_LOW_POWER</code></p> <p data-bbox="533 860 1157 888">Used with <code>setPriority(int)</code> to request "city" level accuracy.</p> <p data-bbox="533 917 1738 977">City level accuracy is considered to be about 10km accuracy. Using a coarse accuracy such as this often consumes less power.</p> <p data-bbox="533 1002 730 1029">Constant Value: 104</p> <p data-bbox="512 1078 1797 1105"><a href="https://developers.google.com/android/reference/com/google/android/gms/location/LocationRequest">https://developers.google.com/android/reference/com/google/android/gms/location/LocationRequest</a></p>

**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC
	<p data-bbox="533 248 1749 285"><code>public Task&lt;Location&gt; getLastLocation ()</code></p> <p data-bbox="533 315 1104 337">Returns the best most recent location currently available.</p> <p data-bbox="533 373 1696 431">If a location is not available, which should happen very rarely, null will be returned. The best accuracy available while respecting the location permissions will be returned.</p> <p data-bbox="533 467 1736 526">This method provides a simplified way to get location. It is particularly well suited for applications that do not require an accurate location and that do not want to maintain extra logic for location updates.</p> <p data-bbox="533 578 1749 615"><code>public Task&lt;LocationAvailability&gt; getLocationAvailability ()</code></p> <p data-bbox="533 651 1692 709">Returns the availability of location data. When <code>isLocationAvailable()</code> returns true, then the location returned by <code>getLastLocation()</code> will be reasonably up to date within the hints specified by the active <code>LocationRequest</code>s.</p> <p data-bbox="533 745 1472 768">If the client isn't connected to Google Play services and the request times out, null is returned.</p> <p data-bbox="533 803 1673 862">Note it's always possible for <code>getLastLocation()</code> to return null even when this method returns true (e.g. location settings were disabled between calls).</p> <p data-bbox="512 898 1900 972"><a href="https://developers.google.com/android/reference/com/google/android/gms/location/FusedLocationProviderClient">https://developers.google.com/android/reference/com/google/android/gms/location/FusedLocationProviderClient</a></p>

**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC						
	<p data-bbox="527 245 1749 326"> <code>public Task&lt;Void&gt; requestLocationUpdates (LocationRequest request, LocationCallback callback, Looper looper)</code> </p> <p data-bbox="527 354 1272 378">Requests location updates with a callback on the specified Looper thread.</p> <p data-bbox="527 412 1686 472">This method is suited for the foreground use cases. For background use cases, the <code>PendingIntent</code> version of the method is recommended, see <code>requestLocationUpdates(LocationRequest, PendingIntent)</code>.</p> <p data-bbox="527 506 1371 531">Any previous LocationRequests registered on this LocationListener will be replaced.</p> <p data-bbox="527 565 1686 656">This call will keep the Google Play services connection active, so make sure to call <code>removeLocationUpdates(LocationCallback)</code> when you no longer need it, otherwise you lose the benefits of the automatic connection management.</p> <p data-bbox="527 690 1745 714">Callbacks for <code>LocationCallback</code> will be made on the specified thread, which must already be a prepared looper thread.</p> <p data-bbox="527 740 663 764"><b>Parameters</b></p> <table border="1" data-bbox="527 792 1749 1008"> <tbody> <tr> <td data-bbox="527 802 625 859"><b>request</b></td> <td data-bbox="632 802 1749 859">The location request for the updates.</td> </tr> <tr> <td data-bbox="527 863 625 920"><b>callback</b></td> <td data-bbox="632 863 1749 920">The callback for the location updates.</td> </tr> <tr> <td data-bbox="527 925 625 1005"><b>looper</b></td> <td data-bbox="632 925 1749 1005">The Looper object whose message queue will be used to implement the callback mechanism, or null to make callbacks on the calling thread.</td> </tr> </tbody> </table> <p data-bbox="527 1057 1902 1133"> <a href="https://developers.google.com/android/reference/com/google/android/gms/location/FusedLocationProviderClient">https://developers.google.com/android/reference/com/google/android/gms/location/FusedLocationProviderClient</a> </p>	<b>request</b>	The location request for the updates.	<b>callback</b>	The callback for the location updates.	<b>looper</b>	The Looper object whose message queue will be used to implement the callback mechanism, or null to make callbacks on the calling thread.
<b>request</b>	The location request for the updates.						
<b>callback</b>	The callback for the location updates.						
<b>looper</b>	The Looper object whose message queue will be used to implement the callback mechanism, or null to make callbacks on the calling thread.						

## Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products

US9408055B2	HTC				
	<div style="background-color: #f0f0f0; padding: 5px; margin-bottom: 10px;"> <pre>public Task&lt;Void&gt; requestLocationUpdates (LocationRequest request, PendingIntent callbackIntent)</pre> </div> <p>Requests location updates with a callback on the specified PendingIntent.</p> <p>This method is suited for the background use cases, more specifically for receiving location updates, even when the app has been killed by the system. In order to do so, use a <code>PendingIntent</code> for a started service. For foreground use cases, the <code>LocationCallback</code> version of the method is recommended, see <code>requestLocationUpdates(LocationRequest, LocationCallback, Looper)</code>.</p> <p>Any previously registered requests that have the same PendingIntent (as defined by <code>equals(Object)</code>) will be replaced by this request.</p> <p>Both <code>LocationResult</code> and <code>LocationAvailability</code> are sent to the given PendingIntent. You can extract data from an Intent using <code>hasResult(Intent)</code>, <code>extractResult(Intent)</code>, <code>hasLocationAvailability(Intent)</code>, and <code>extractLocationAvailability(Intent)</code>.</p> <p><b>Parameters</b></p> <table border="1" style="width: 100%; border-collapse: collapse; margin-bottom: 10px;"> <tr> <td style="background-color: #cccccc; padding: 5px;"><code>request</code></td> <td style="padding: 5px;">The location request for the updates.</td> </tr> <tr> <td style="background-color: #cccccc; padding: 5px;"><code>callbackIntent</code></td> <td style="padding: 5px;">A pending intent to be sent for each location update.</td> </tr> </table> <p><b>Returns</b></p> <ul style="list-style-type: none"> <li>a Task for the call, check <code>isSuccessful()</code> to determine if it was successful.</li> </ul> <p><a href="https://developers.google.com/android/reference/com/google/android/gms/location/FusedLocationProviderClient">https://developers.google.com/android/reference/com/google/android/gms/location/FusedLocationProviderClient</a></p>	<code>request</code>	The location request for the updates.	<code>callbackIntent</code>	A pending intent to be sent for each location update.
<code>request</code>	The location request for the updates.				
<code>callbackIntent</code>	A pending intent to be sent for each location update.				



**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC				
	<p><code>public void onLocationAvailability (LocationAvailability locationAvailability)</code></p> <p>Called when there is a change in the availability of location data.</p> <p>When <code>isLocationAvailable()</code> returns <code>false</code> you can assume that location will not be returned in <code>onLocationResult(LocationResult)</code> until something changes in the device's settings or environment. Even when <code>isLocationAvailable()</code> returns <code>true</code> the <code>onLocationResult(LocationResult)</code> may not always be called regularly, however the device location is known and both the most recently delivered location and <code>getLastLocation(GoogleApiClient)</code> will be reasonably up to date given the hints specified by the active <code>LocationRequest</code>s.</p> <p><b>Parameters</b></p> <table border="1" data-bbox="527 639 1738 704"> <tr> <td><code>locationAvailability</code></td> <td>The current status of location availability.</td> </tr> </table> <p><code>public void onLocationResult (LocationResult result)</code></p> <p>Called when device location information is available.</p> <p>The most recent location returned by <code>getLastLocation()</code> is not guaranteed to be immediately fresh, but will be reasonably up to date given the hints specified by the active <code>LocationRequest</code>s.</p> <p><b>Parameters</b></p> <table border="1" data-bbox="527 1019 1738 1084"> <tr> <td><code>result</code></td> <td>The latest location result available.</td> </tr> </table> <p><a href="https://developers.google.com/android/reference/com/google/android/gms/location/LocationCallback">https://developers.google.com/android/reference/com/google/android/gms/location/LocationCallback</a></p>	<code>locationAvailability</code>	The current status of location availability.	<code>result</code>	The latest location result available.
<code>locationAvailability</code>	The current status of location availability.				
<code>result</code>	The latest location result available.				

**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC		
	<p>public abstract void <b>onLocationChanged</b> (<a href="#">Location</a> location)</p> <p>Called when the location has changed.</p> <p><b>Parameters</b></p> <table border="1" data-bbox="527 412 1740 477"> <tr> <td data-bbox="527 412 926 477"><b>location</b></td> <td data-bbox="932 412 1740 477">The updated location.</td> </tr> </table> <p><a href="https://developers.google.com/android/reference/com/google/android/gms/location/LocationListener">https://developers.google.com/android/reference/com/google/android/gms/location/LocationListener</a></p> <p>Public Constructors</p> <hr/> <p>public <b>MapView</b> (<a href="#">Context</a> context)</p> <p>public <b>MapView</b> (<a href="#">Context</a> context, <a href="#">AttributeSet</a> attrs)</p> <p>public <b>MapView</b> (<a href="#">Context</a> context, <a href="#">AttributeSet</a> attrs, int defStyle)</p> <p>public <b>MapView</b> (<a href="#">Context</a> context, <a href="#">GoogleMapOptions</a> options)</p> <p><a href="https://developers.google.com/android/reference/com/google/android/gms/maps/MapView">https://developers.google.com/android/reference/com/google/android/gms/maps/MapView</a></p>	<b>location</b>	The updated location.
<b>location</b>	The updated location.		

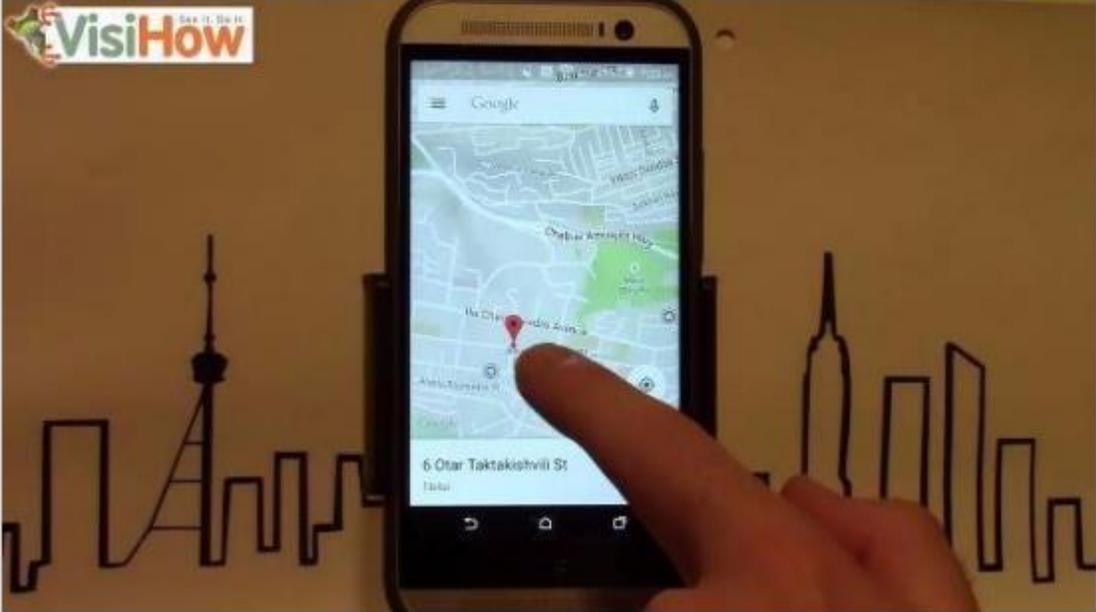
## Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products

US9408055B2	HTC		
<p>[1F] identifying user interaction with the interactive display selecting one or more of the user-selectable symbols corresponding to one or more of the second</p>	<pre>public void <b>getMapAsync</b> (<a href="#">OnMapReadyCallback</a> callback)</pre> <p>Returns a non-null instance of the <a href="#">GoogleMap</a> , ready to be used.</p> <p>Note that:</p> <ul style="list-style-type: none"> <li>• This method must be called from the main thread.</li> <li>• The callback will be executed in the main thread.</li> <li>• In the case where Google Play services is not installed on the user's device, the callback will not be triggered until the user installs it.</li> <li>• The <a href="#">GoogleMap</a> object provided by the callback is non-null.</li> </ul> <p><b>Parameters</b></p> <table border="1" style="width: 100%; border-collapse: collapse;"> <tr> <td style="background-color: #444; color: white; padding: 2px;"><b>callback</b></td> <td style="padding: 2px;">The callback object that will be triggered when the map is ready to be used.</td> </tr> </table> <pre>public final void <b>onCreate</b> (<a href="#">Bundle</a> savedInstanceState)</pre> <p>You must call this method from the parent Activity/Fragment's corresponding method.</p> <p><a href="https://developers.google.com/android/reference/com/google/android/gms/maps/MapView">https://developers.google.com/android/reference/com/google/android/gms/maps/MapView</a></p>	<b>callback</b>	The callback object that will be triggered when the map is ready to be used.
<b>callback</b>	The callback object that will be triggered when the map is ready to be used.		
<p>corresponding to one or more of the second</p>	<p>HTC infringes directly and/or indirectly by performing, inducing others to perform, and/or contributing to the performance of identifying user interaction with the interactive display selecting one or more of the user-selectable symbols corresponding to one or more of the second devices and user interaction with the display specifying an action and, based thereon, sending data to the one or more second devices.</p> <p><b>Regarding Google Maps</b>, Google Latitude, Google Plus, Google Hangouts, Google Messages, and Android Messenger, the Accused Products are configured to allow a user of the first device to interact with the</p>		

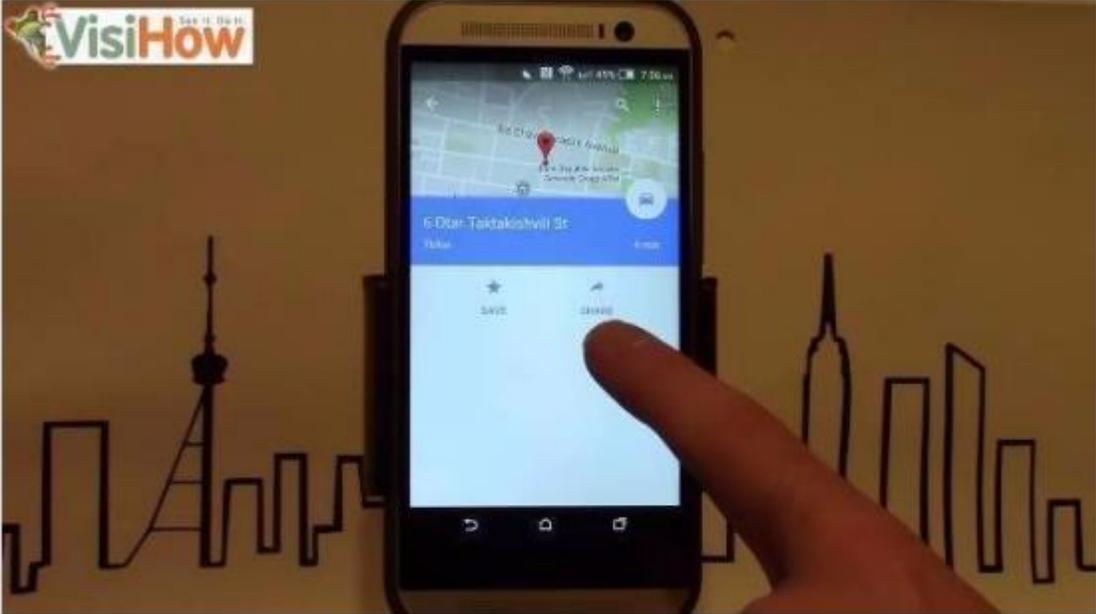
**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC
<p>devices and user interaction with the display specifying an action and, based thereon, sending data to the one or more second devices;</p>	<p>display, to select a user, contact, or device, and to select an action to be performed, such as: sending a message, initiating a call, initiating a data conference, sharing a location, stop sharing a location, block a user from location sharing, sending a location, requesting a location, or sending other data. When an action is specified, data is sent from the first device to the second device via a server.</p> <p><b><u>Exemplary Support for Google Maps:</u></b>                      Using Google Maps, a user may choose a symbol and send data to that device. For example, a user who is already sharing her location with another user can stop sharing by making a selection resulting in the second device no longer displaying the first device’s location. Additionally, a user can share an ETA message with another user or send another user a link in a message to share her location. Additionally, a user who is sharing a location until she arrives can make a selection to stop her location from showing on the second device.</p> <p><b>Share Locations from Google Maps on HTC One M8</b></p>

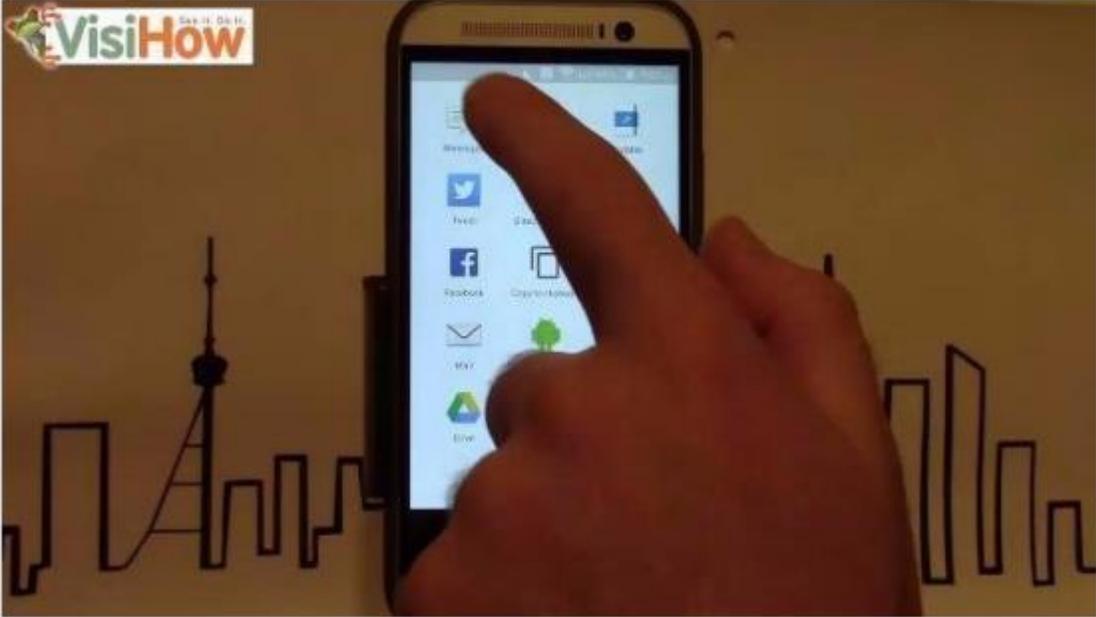
## Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products

US9408055B2	HTC
	<p data-bbox="531 233 827 256"><b>Place a pin on the map.</b></p> <p data-bbox="531 272 1640 380">We can do this anywhere in the entire world, anywhere, on any map that we find in the Google Maps application. We are going to choose a location, press down on the screen, and hold down until we can see that a red pin has been dropped.</p> 

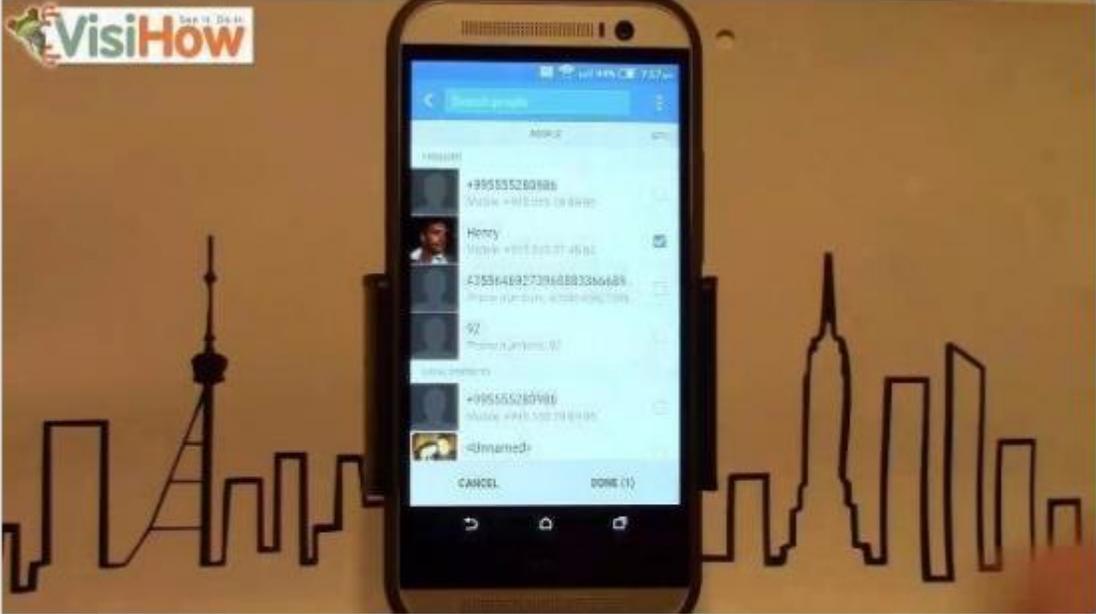
## Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products

US9408055B2	HTC
	<p data-bbox="527 240 1207 310"><b>We have two options:</b> "SAVE" and "SHARE". Press down on the word "SHARE".</p>  <hr/> <p data-bbox="527 1057 1633 1208"><b>A menu appears with all of the different ways that we can share this location.</b> We can share this location via messages, Bluetooth, Twitter, Facebook, mail, Gmail and Viber. If you have other applications, such as WhatsApp, or any other chat or email applications, you will be able to use them as well and will see them here.</p>

### Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products

US9408055B2	HTC
	<p data-bbox="527 240 863 264"><b>Let's choose "Messages".</b></p> <p data-bbox="527 277 1633 345">This option is located on the top left-hand corner of the screen. All options will have the same basic processes. They will ask us to send the location. Press down on the "Messages" icon.</p> 

## Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products

US9408055B2	HTC
	<p data-bbox="520 233 1260 256"><b>Press the box next to the contact who will be the recipient.</b></p> <p data-bbox="520 269 1549 292">A checkmark will appear in the box. We can select multiple contacts as well if we desire.</p>  <p data-bbox="520 1045 911 1068"><b>Press on the word "DONE (1)".</b></p> <p data-bbox="520 1081 1629 1153">This icon is located in the bottom right-hand corner of the screen. The bracketed number indicates the number of contacts we selected.</p> <p data-bbox="520 1166 1407 1188"><a href="http://visihow.com/Share_Locations_from_Google_Maps_on_HTC_One_M8">http://visihow.com/Share_Locations_from_Google_Maps_on_HTC_One_M8</a></p>



## Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products

US9408055B2	HTC
	<p data-bbox="548 240 968 261">COMPUTER <b>ANDROID</b> IPHONE &amp; IPAD</p> <hr data-bbox="533 289 1566 293"/> <h3 data-bbox="533 347 1024 383">If they have a Google Account</h3> <ol data-bbox="533 402 1419 699" style="list-style-type: none"><li data-bbox="533 402 1220 423">1. If you haven't already, add their Gmail address to your <a href="#">Google Contacts</a>.</li><li data-bbox="533 440 1419 461">2. On your Android phone or tablet, open the Google Maps app  and sign in. <a href="#">Learn how to sign in.</a></li><li data-bbox="533 477 1031 498">3. Tap Menu  &gt; <b>Location sharing</b> &gt; Add People .</li><li data-bbox="533 514 1003 535">4. Choose how long you want to share your location.</li><li data-bbox="533 552 1140 610">5. Tap <b>Select People</b>.<ul data-bbox="569 586 1140 610" style="list-style-type: none"><li data-bbox="569 586 1140 610">• If you're asked about your contacts, give Google Maps access.</li></ul></li><li data-bbox="533 634 884 656">6. Choose who you want to share with.</li><li data-bbox="533 672 663 693">7. Tap <b>Share</b>.</li></ol> <h3 data-bbox="533 753 1108 789">If they don't have a Google Account</h3> <ol data-bbox="533 813 1560 938" style="list-style-type: none"><li data-bbox="533 813 1419 834">1. On your Android phone or tablet, open the Google Maps app  and sign in. <a href="#">Learn how to sign in.</a></li><li data-bbox="533 850 1031 872">2. Tap Menu  &gt; <b>Location sharing</b> &gt; Add People .</li><li data-bbox="533 888 1560 938">3. Tap More  &gt; <b>Copy to clipboard</b>. People with this link can see your location for as long as you choose, up to 72 hours.</li></ol> <h3 data-bbox="533 987 869 1023">Share using another app</h3> <p data-bbox="533 1040 1205 1062">You can also share through messaging apps. Tap More  &gt; select an app.</p> <h3 data-bbox="533 1122 743 1157">Stop sharing</h3> <ol data-bbox="533 1182 1205 1279" style="list-style-type: none"><li data-bbox="533 1182 842 1203">1. Open the Google Maps app .</li><li data-bbox="533 1219 869 1240">2. Tap Menu  &gt; <b>Location sharing</b>.</li><li data-bbox="533 1256 1205 1279">3. Next to the person with whom you want to stop sharing, tap Remove .</li></ol> <p data-bbox="512 1295 1703 1328"><a href="https://support.google.com/maps/answer/7326816?co=GENIE.Platform%3DAndroid&amp;oco=1">https://support.google.com/maps/answer/7326816?co=GENIE.Platform%3DAndroid&amp;oco=1</a></p>

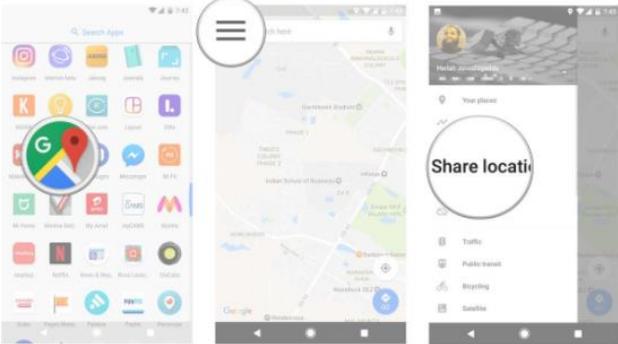
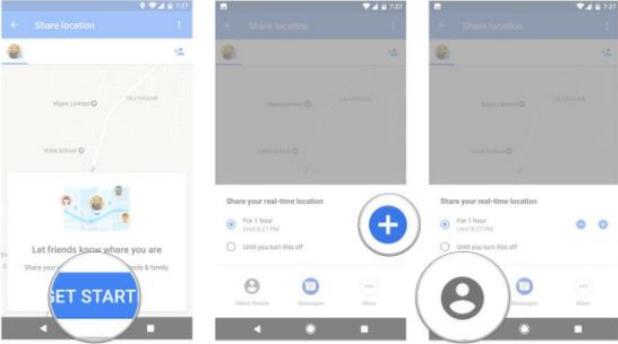
## Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products

US9408055B2	HTC
	<h3>Share your E.T.A</h3> <p>After you start your drive, you can share your destination, estimated arrival time, and where you are on the route.</p> <ol style="list-style-type: none"><li>1. Open the Google Maps app .</li><li>2. Set a driving destination. <a href="#">Learn how to navigate to a place.</a></li><li>3. After you start navigation, tap More  &gt; <b>Share trip progress.</b></li><li>4. Choose a person from the list.</li><li>5. Tap <b>Share.</b></li><li>6. Location Sharing will stop when you reach your destination or stop navigating.</li></ol> <ul style="list-style-type: none"><li>• To stop sharing before you arrive, tap More  &gt; <b>Stop sharing.</b></li></ul> <h3>See where someone is</h3> <p>If someone shares their location with you, you can see them on the map.</p> <ol style="list-style-type: none"><li>1. Open the Google Maps app .</li><li>2. Tap Menu  &gt; <b>Location sharing.</b></li><li>3. Choose someone.</li></ol> <ul style="list-style-type: none"><li>• To see an updated location, tap on a friend's icon &gt; More  &gt; <b>Refresh.</b></li></ul> <h3>Stop seeing someone's location</h3> <ol style="list-style-type: none"><li>1. Open the Google Maps app .</li><li>2. On the map, tap their icon.</li><li>3. At the bottom, tap More .</li><li>4. To temporarily hide someone, tap <b>Hide from map.</b> You can stop hiding them at any time.</li></ol> <p><b>Note:</b> You can stop someone's location from ever appearing on your map. <a href="#">Learn how to block another person's account.</a></p> <p><a href="https://support.google.com/maps/answer/7326816?co=GENIE.Platform%3DAndroid&amp;oco=1">https://support.google.com/maps/answer/7326816?co=GENIE.Platform%3DAndroid&amp;oco=1</a></p>

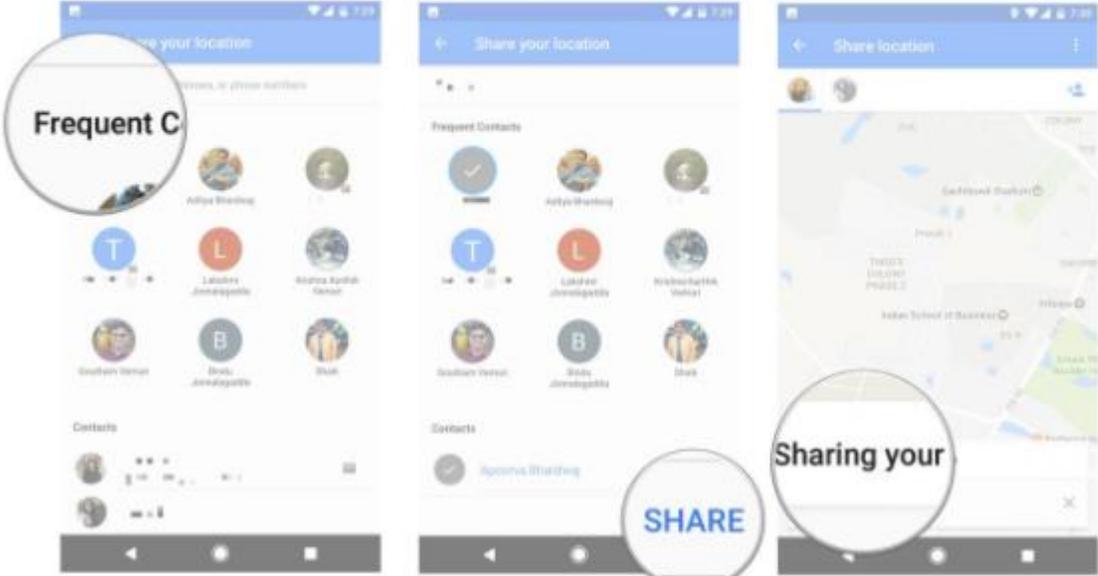
## Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products

US9408055B2	HTC
	<h3 data-bbox="541 280 877 318">Hide or share lists</h3> <p data-bbox="541 347 909 371"><b>Note:</b> You can't share starred places.</p> <ol data-bbox="554 402 1682 667" style="list-style-type: none"><li data-bbox="554 402 890 427">1. Open the Google Maps app .</li><li data-bbox="554 443 968 467">2. Tap Menu  &gt; <b>Your places</b> &gt; <b>Saved</b>.</li><li data-bbox="554 483 1682 667">3. Next to the list you want to share, tap More  &gt; choose an option:<ul data-bbox="583 524 1682 667" style="list-style-type: none"><li data-bbox="583 524 1444 548">• <b>Hide/Show on your map:</b> Display or hide your saved places when looking at the map.</li><li data-bbox="583 565 1056 589">• <b>Share list:</b> Allow others to see your saved list.</li><li data-bbox="583 605 1682 667">• <b>Sharing options:</b> You can make your list public, private, or shared. To let anyone with the link see your list, tap <b>Shared</b>. To let anyone find and follow your list, tap <b>Public</b>.</li></ul></li></ol> <h3 data-bbox="541 735 768 773">Follow a list</h3> <p data-bbox="541 802 1728 859">If you follow a list made by someone else, their saved places will show up in Your Places. The places will also appear as suggested locations in Google Maps.</p> <h3 data-bbox="541 914 915 951">Follow a list using a link</h3> <ol data-bbox="554 976 1356 1081" style="list-style-type: none"><li data-bbox="554 976 961 1000">1. Tap on the link you received to open it.</li><li data-bbox="554 1016 1272 1040">2. Tap <b>Follow</b>. This list will now be added to the group of lists you follow.</li><li data-bbox="554 1057 1356 1081">3. <b>Optional:</b> To unfollow a list someone shared with you, tap the list &gt; <b>Following</b>.</li></ol> <h3 data-bbox="541 1133 926 1170">See lists made by others</h3> <p data-bbox="541 1195 1335 1219">If a user has any Google Maps lists that were made public, you can follow them.</p> <ol data-bbox="554 1252 1136 1357" style="list-style-type: none"><li data-bbox="554 1252 1136 1276">1. Tap on the name of a user whose list you want to follow.</li><li data-bbox="554 1292 678 1317">2. Tap <b>Lists</b>.</li><li data-bbox="554 1333 1136 1357">3. Tap on the list you want to follow &gt; More  &gt; <b>Follow</b>.</li></ol> <p data-bbox="510 1373 1898 1398"><a href="https://support.google.com/maps/answer/7280933?hl=en&amp;ref_topic=7301134&amp;co=GENIE.Platform%3DAn">https://support.google.com/maps/answer/7280933?hl=en&amp;ref_topic=7301134&amp;co=GENIE.Platform%3DAn</a></p>

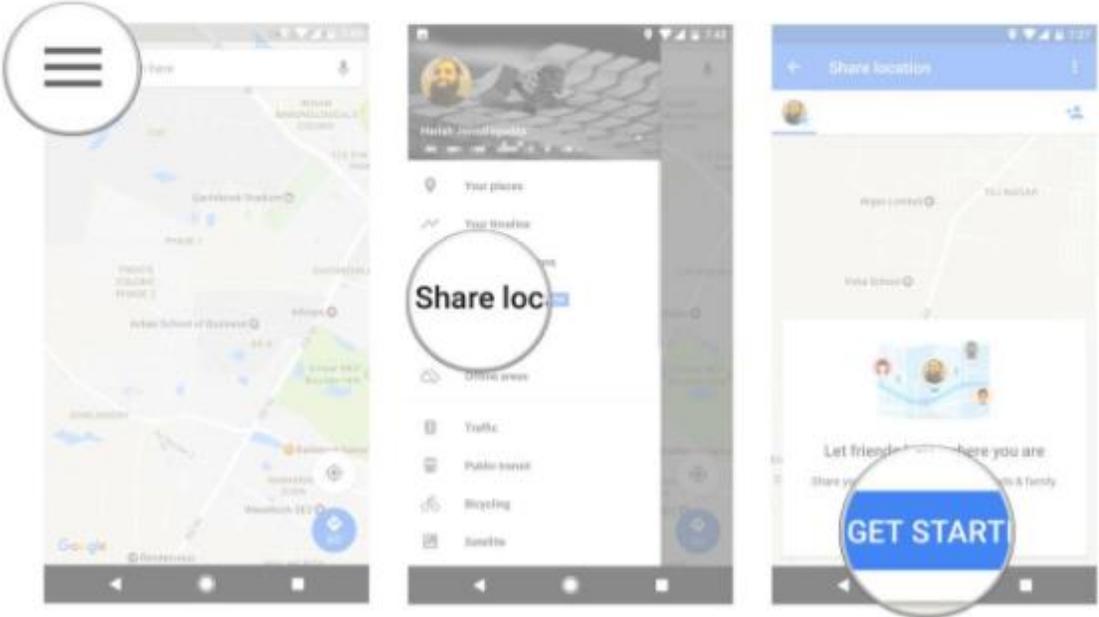
## Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products

US9408055B2	HTC
	<p data-bbox="512 233 684 261">droid&amp;oco=1</p> <h3 data-bbox="512 310 1104 342">How to share your location in Google Maps</h3> <ol data-bbox="512 367 1087 451" style="list-style-type: none"><li>1. Open Google Maps from the app drawer or the home screen.</li><li>2. Tap the <b>hamburger menu</b> (the three horizontal lines) on the top left corner of the screen.</li><li>3. Select <b>Share location</b>.</li></ol>  <ol data-bbox="512 854 1117 954" style="list-style-type: none"><li>4. Tap <b>Get Started</b>.</li><li>5. Use the <b>+</b> icon to select a time period or select the <b>Until you turn this off</b> setting to share your location indefinitely.</li><li>6. Tap <b>Select People</b>.</li></ol>  <p data-bbox="512 1333 1356 1365"><a href="https://www.androidcentral.com/how-share-location-google-maps">https://www.androidcentral.com/how-share-location-google-maps</a></p>

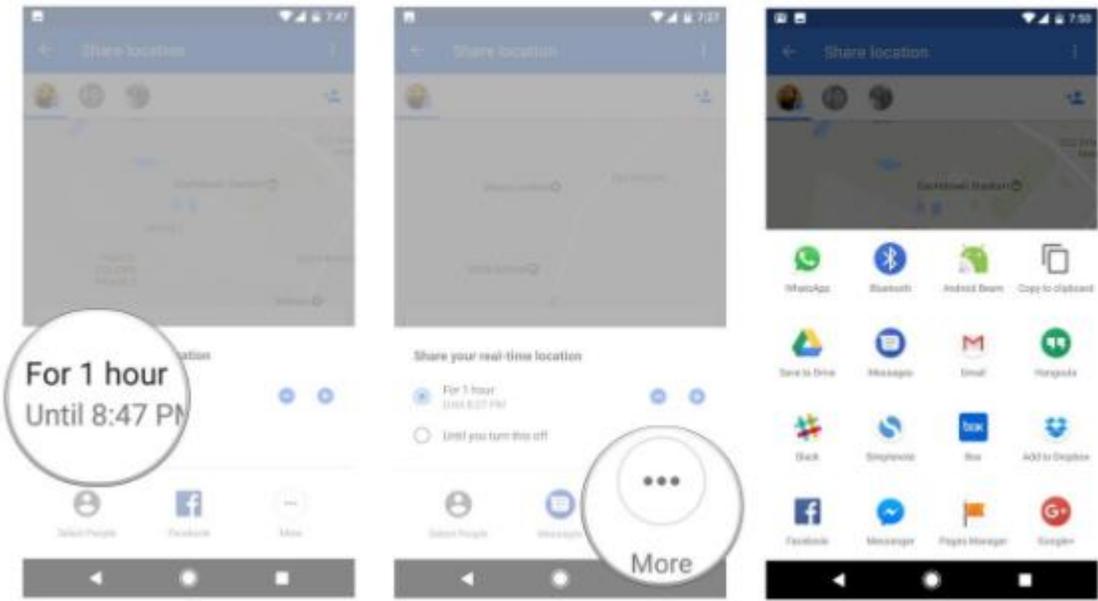
### Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products

US9408055B2	HTC
	<p data-bbox="527 253 1577 310">7. You'll see a list of your frequent contacts at the top, along with a full list of contacts. Pick the contacts by tapping their name.</p> <p data-bbox="527 339 1457 367">8. Once you've selected the contacts you want to share your location to, tap <b>Share</b>.</p> <p data-bbox="527 396 1419 423">9. You'll see a message saying that the selected contact can view your location.</p>  <p data-bbox="506 1101 1356 1133"><a href="https://www.androidcentral.com/how-share-location-google-maps">https://www.androidcentral.com/how-share-location-google-maps</a></p>

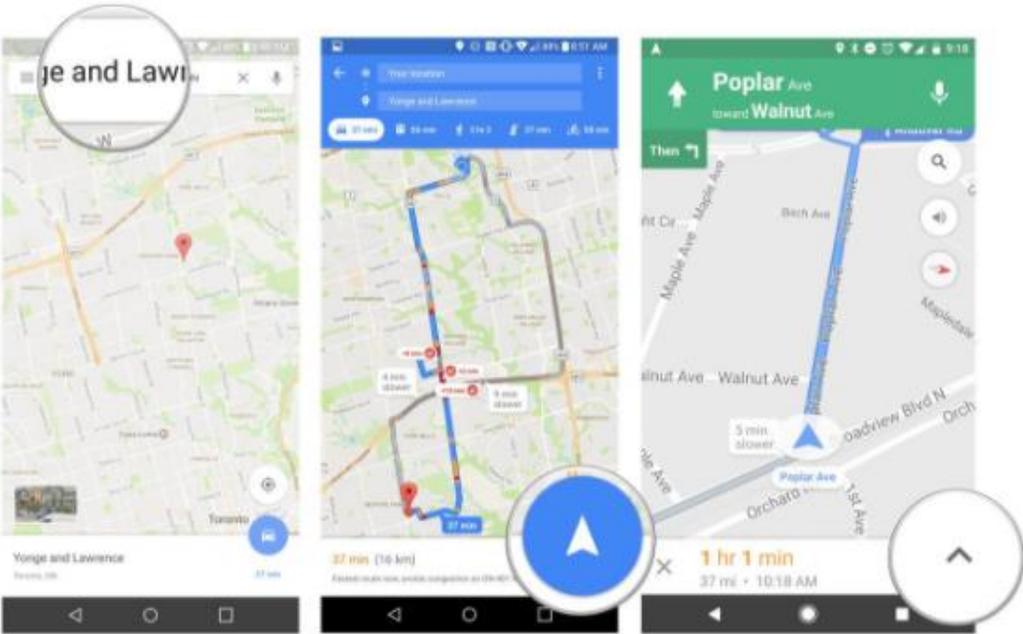
## Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products

US9408055B2	HTC
	<h3 data-bbox="520 245 1255 289">How to create a shareable link</h3> <p data-bbox="520 331 1461 358">You can also create a link and use it to share your location easily. Here's how to do it:</p> <ol data-bbox="520 410 1234 548" style="list-style-type: none"><li data-bbox="520 410 1234 438">1. Tap the hamburger menu on the top left corner of the screen.</li><li data-bbox="520 467 800 495">2. Select Share location.</li><li data-bbox="520 524 737 552">3. Tap Get Started.</li></ol>  <p data-bbox="506 1230 1356 1263"><a href="https://www.androidcentral.com/how-share-location-google-maps">https://www.androidcentral.com/how-share-location-google-maps</a></p>

### Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products

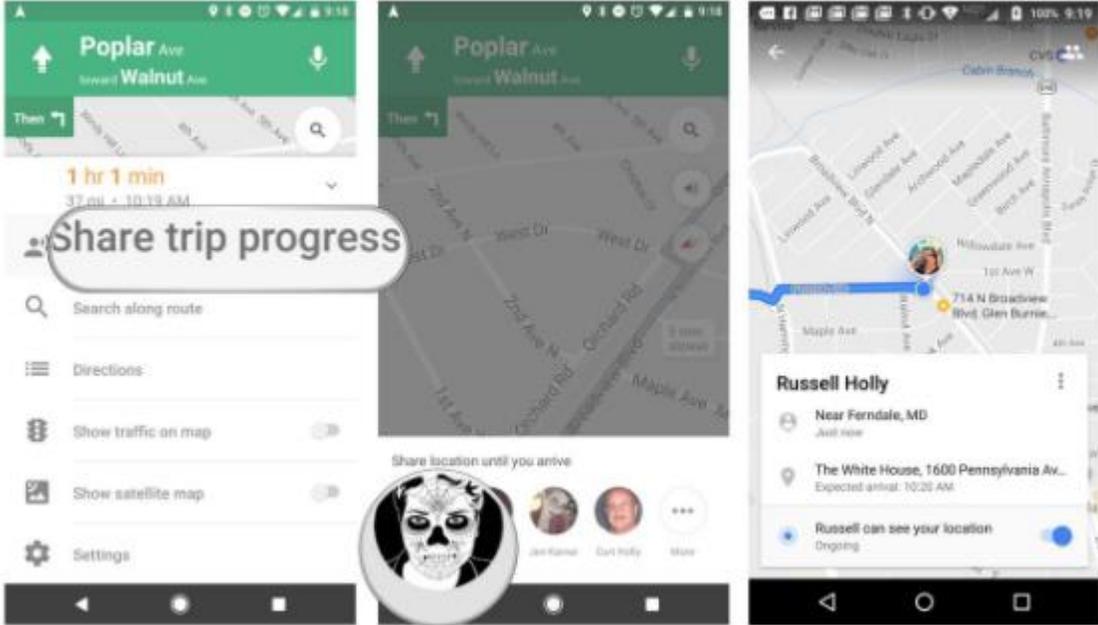
US9408055B2	HTC
	<p data-bbox="520 245 1213 277">4. Select the <b>amount of time</b> you want to share your location.</p> <p data-bbox="520 302 680 334">5. Tap <b>More</b>.</p> <p data-bbox="520 358 1633 423">6. Select <b>your app of choice</b> to create and send a <b>unique URL</b> that broadcasts your current location. You can email it, send the link via Messenger, or even tweet it to the <b>intended recipient</b>.</p> <div data-bbox="569 456 1667 1057"></div> <p data-bbox="506 1081 1360 1114"><a href="https://www.androidcentral.com/how-share-location-google-maps">https://www.androidcentral.com/how-share-location-google-maps</a></p>

**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

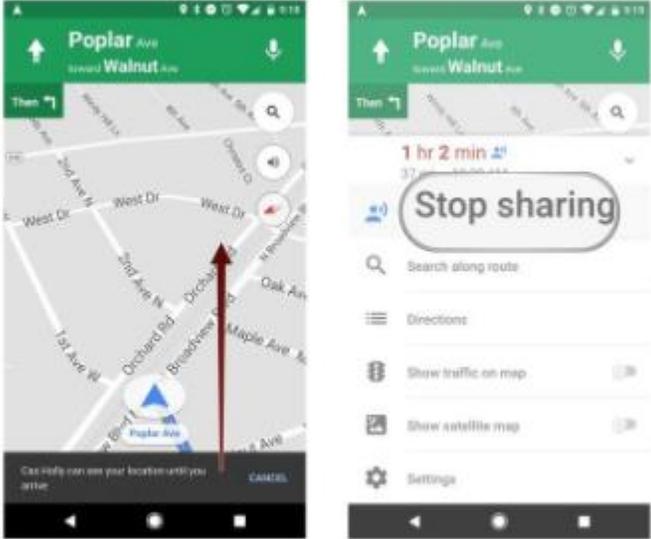
US9408055B2	HTC
	<p data-bbox="527 240 1428 337"><b>How to share your navigation directions while you walk, drive or transit</b></p> <p data-bbox="527 375 1556 464">One of the best ways to share your location with someone is during a drive, walk or transit. If you're meeting a friend or family member somewhere, or navigating towards their home, Google Maps lets you share your location with them for the duration of the trip. It's magic!</p> <ol data-bbox="527 513 1396 643" style="list-style-type: none"> <li>1. In the <b>search bar</b> enter your destination.</li> <li>2. Pick your navigation type (drive, transit, walk) and press the blue <b>navigate</b> button.</li> <li>3. Tap the arrow next to the <b>time-to-destination</b> number at the bottom of the screen.</li> </ol>  <p data-bbox="512 1328 1360 1360"><a href="https://www.androidcentral.com/how-share-location-google-maps">https://www.androidcentral.com/how-share-location-google-maps</a></p>



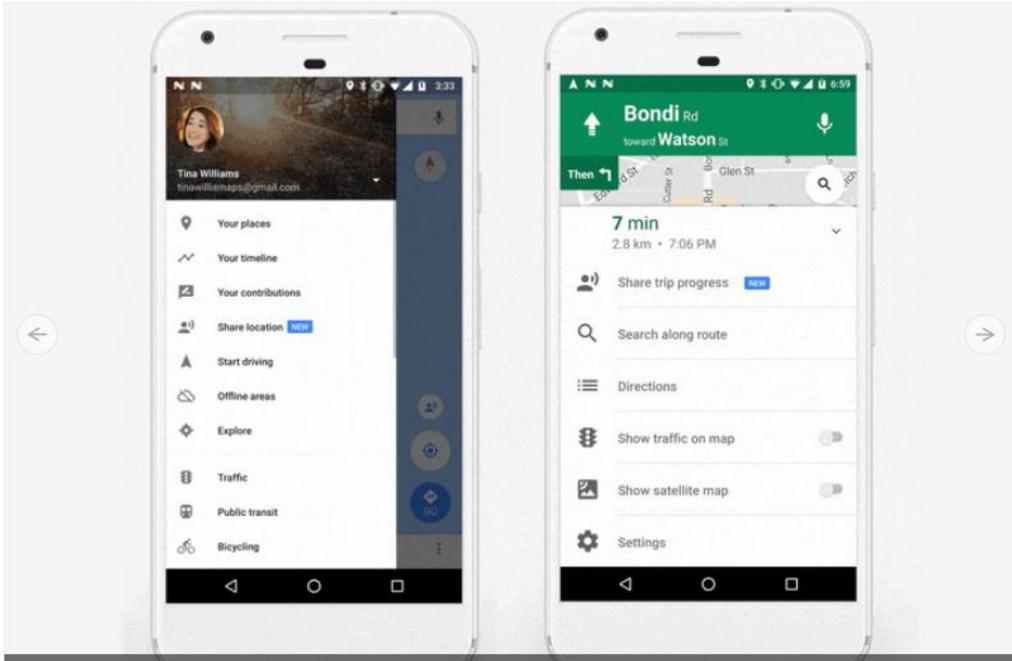
### Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products

US9408055B2	HTC
	<p data-bbox="527 277 835 305">4. Tap Share trip progress.</p> <p data-bbox="527 334 1150 362">5. Choose one or more contacts to share trip progress.</p>  <p data-bbox="527 1065 1360 1133">You can also stop sharing your location with someone before a trip ends. <a href="https://www.androidcentral.com/how-share-location-google-maps">https://www.androidcentral.com/how-share-location-google-maps</a></p>

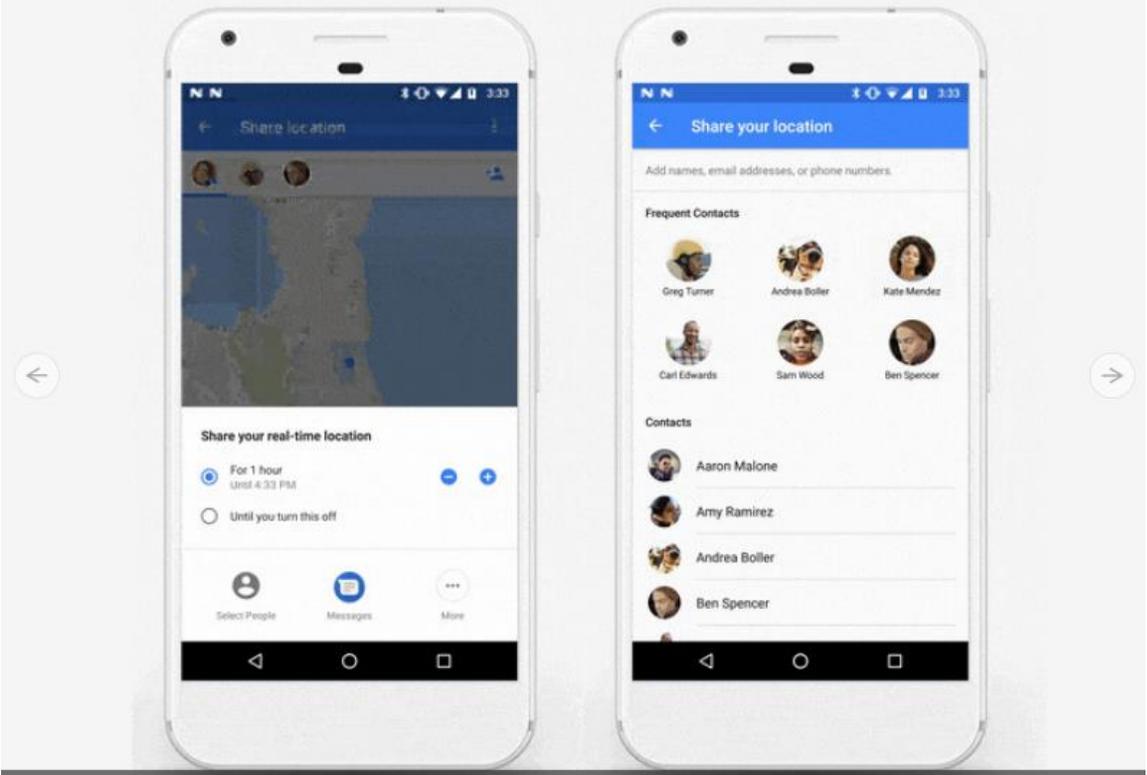
## Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products

US9408055B2	HTC
	<ol style="list-style-type: none"><li data-bbox="533 245 1465 272">1. Tap the arrow next to the <b>time-to-destination</b> number at the bottom of the screen.</li><li data-bbox="533 302 768 329">2. Tap Stop sharing.</li></ol> <div data-bbox="747 383 1398 922"></div> <p data-bbox="541 976 638 1003">That's it!</p> <p data-bbox="541 1045 1608 1073">Are you excited that location sharing is back in Google Maps? How often do you use the feature?</p> <p data-bbox="512 1084 1356 1112"><a href="https://www.androidcentral.com/how-share-location-google-maps">https://www.androidcentral.com/how-share-location-google-maps</a></p> <p data-bbox="512 1195 1419 1222">As shown below, a group may also be defined within Google Contacts.</p>

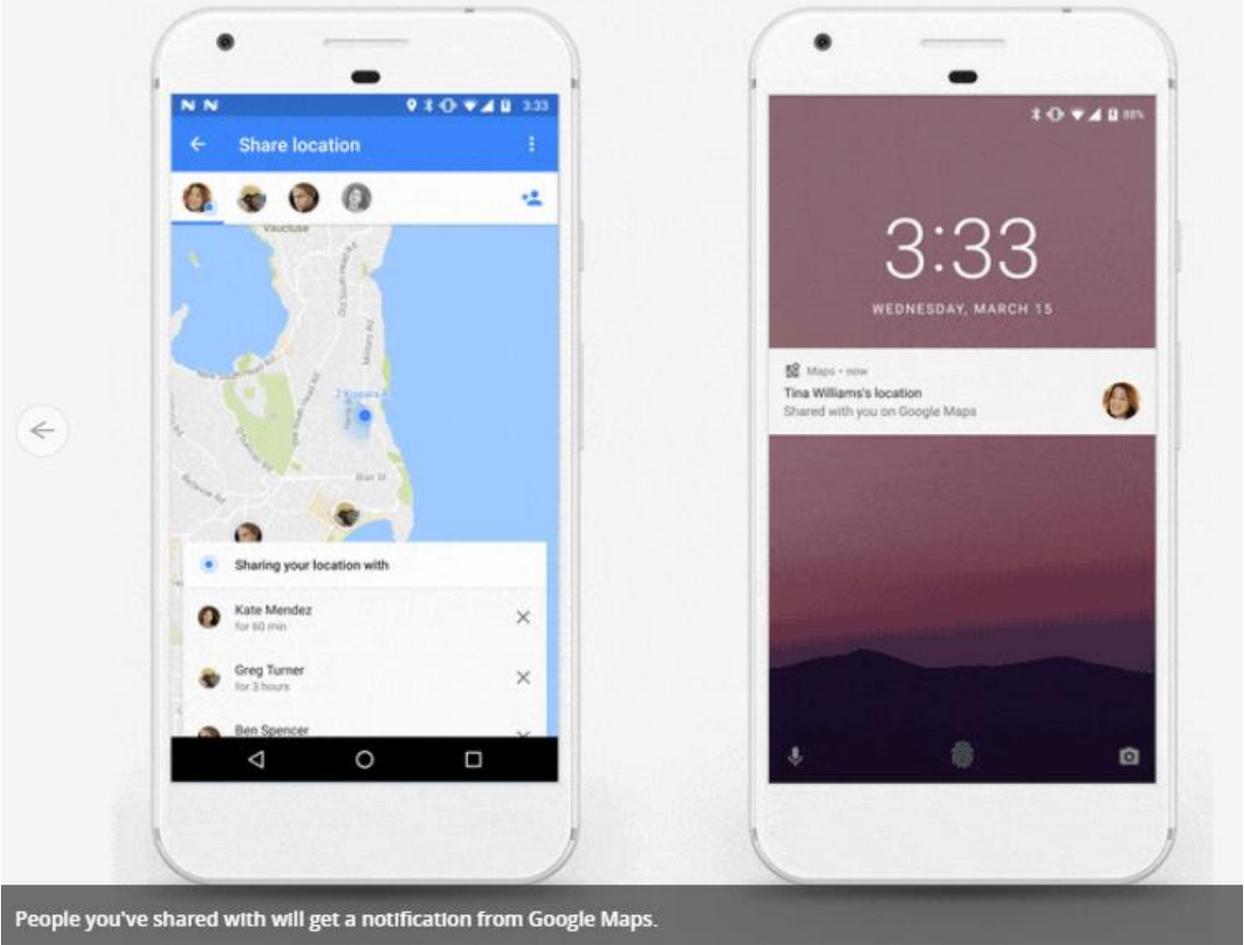
## Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products

US9408055B2	HTC
	<p data-bbox="548 245 940 289"><b>Share your contacts</b></p> <ol data-bbox="562 316 1045 479" style="list-style-type: none"><li>1. Open your device's Contacts app .</li><li>2. Tap a contact in the list.</li><li>3. Tap More  &gt; <b>Share</b>.</li><li>4. Choose how you want to share the contact.</li></ol> <p data-bbox="512 495 1535 527"><a href="https://support.google.com/android/answer/6118731?hl=en&amp;ref_topic=6118711">https://support.google.com/android/answer/6118731?hl=en&amp;ref_topic=6118711</a></p>  <p data-bbox="520 1230 1516 1284">Ways to start location sharing: (left) from the app navigation drawer, and (right) from the menu, you can drag up while navigating somewhere.</p> <p data-bbox="512 1295 1654 1325"><a href="https://arstechnica.com/gadgets/2017/03/location-sharing-finally-returns-to-google-maps/">https://arstechnica.com/gadgets/2017/03/location-sharing-finally-returns-to-google-maps/</a></p>

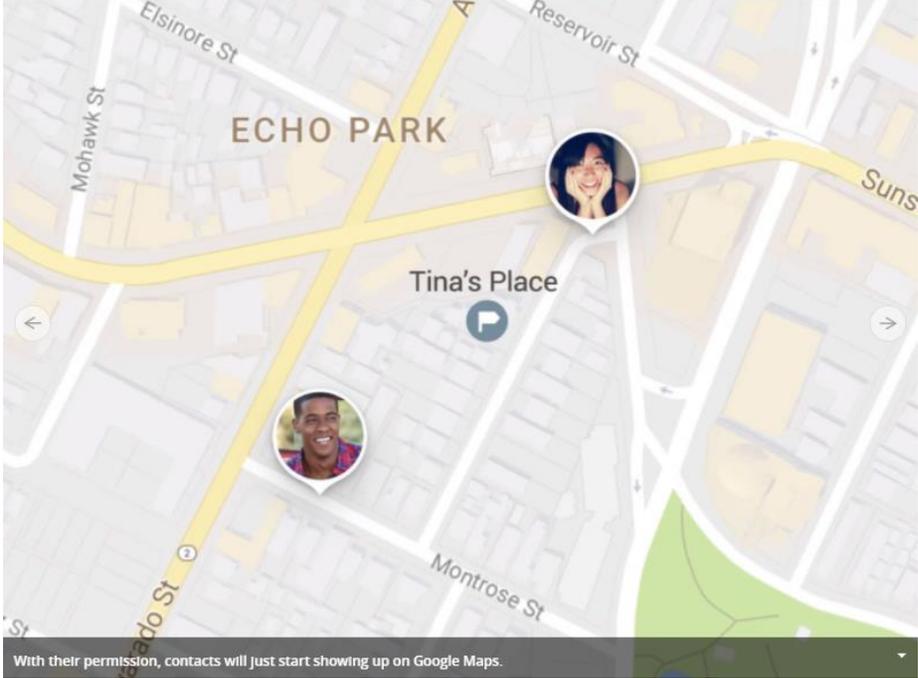
### Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products

US9408055B2	HTC
	 <p data-bbox="514 1023 1661 1063">You can pick how long you want to share your location for, and with whom you want to share it.</p> <p data-bbox="514 1063 1661 1096"><a href="https://arstechnica.com/gadgets/2017/03/location-sharing-finally-returns-to-google-maps/">https://arstechnica.com/gadgets/2017/03/location-sharing-finally-returns-to-google-maps/</a></p>

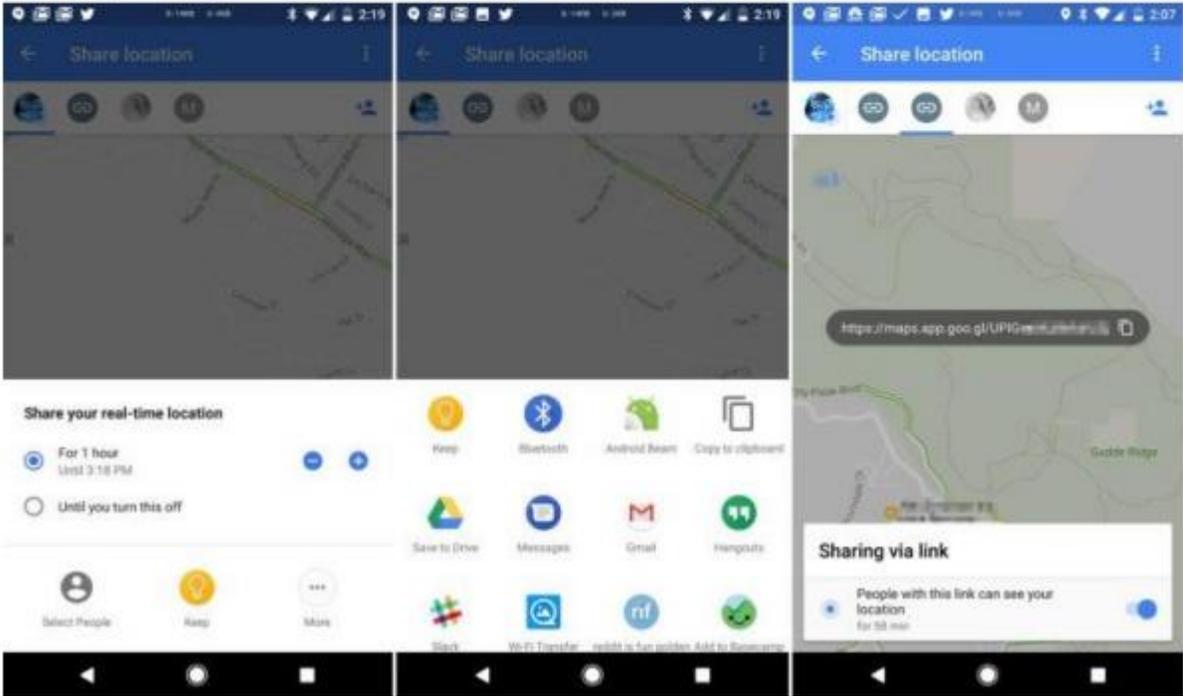
**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC
	 <p data-bbox="514 1144 1176 1177">People you've shared with will get a notification from Google Maps.</p> <p data-bbox="514 1188 1659 1226"><a href="https://arstechnica.com/gadgets/2017/03/location-sharing-finally-returns-to-google-maps/">https://arstechnica.com/gadgets/2017/03/location-sharing-finally-returns-to-google-maps/</a></p>

## Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products

US9408055B2	HTC
	 <p data-bbox="512 883 1430 911">With their permission, contacts will just start showing up on Google Maps.</p> <p data-bbox="512 915 1656 948"><a href="https://arstechnica.com/gadgets/2017/03/location-sharing-finally-returns-to-google-maps/">https://arstechnica.com/gadgets/2017/03/location-sharing-finally-returns-to-google-maps/</a></p> <h3 data-bbox="512 1019 779 1068">Stop sharing</h3> <ol data-bbox="512 1094 1352 1219" style="list-style-type: none"><li>1. Open the Google Maps app 📍.</li><li>2. Tap the Menu ☰ &gt; <b>Share location</b>.</li><li>3. Next to the person with whom you want to stop sharing, tap <b>Remove</b> ✕.</li></ol>

**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

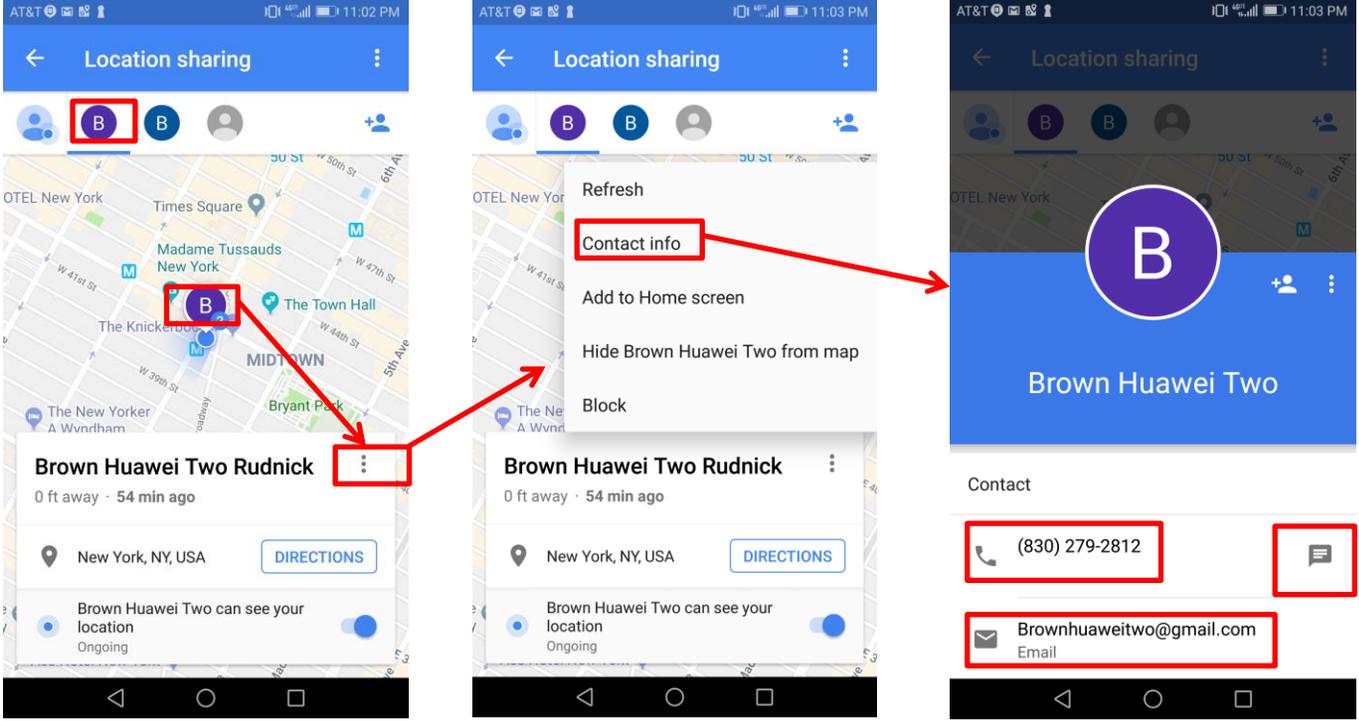
US9408055B2	HTC
	<p><b>Share your E.T.A</b></p> <p>After you start your drive, you can share your destination, estimated arrival time, and where you are on the route.</p> <ol style="list-style-type: none"> <li>1. Open the Google Maps app .</li> <li>2. Set a driving destination. <a href="#">Learn how to navigate to a place.</a></li> <li>3. After you start navigation, tap <b>More</b>  <b>&gt; Share trip progress.</b></li> <li>4. Choose a person from the list.</li> <li>5. Tap <b>Share.</b></li> <li>6. Location Sharing will stop when you reach your destination or stop navigating.</li> </ol> <ul style="list-style-type: none"> <li>• To stop sharing before you arrive, tap <b>More</b>  <b>&gt; Stop sharing.</b></li> </ul> 

**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC
	 <p><b><u>Exemplary Maps Screenshots:</u></b></p>



**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC
	 <p><b>Exemplary Source Code:</b></p> <p>The above functionality is performed at least in part by the following publicly available source code and/or source code that invokes or is invoked by the following source code (or a substantially similar copy compiled and loaded onto the Accused Products by HTC). AGIS reserves the right to supplement these contentions with additional source code as discovery progresses and as additional source code is made available. AGIS reserves the right to supplement these contentions with additional source code as discovery progresses and as additional source code is made available.</p>

**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC
	<pre> 44  * Class that sends chat message via SMS. 45  * 46  * The interface emulates a blocking sending similar to making an HTTP request. 47  * It calls the SmsManager to send a (potentially multipart) message and waits 48  * on the sent status on each part. The waiting has a timeout so it won't wait 49  * forever. Once the sent status of all parts received, the call returns. 50  * A successful sending requires success status for all parts. Otherwise, we 51  * pick the highest level of failure as the error for the whole message, which 52  * is used to determine if we need to retry the sending. 53  */ 54  public class SmsSender { 55      private static final String TAG = LogUtil.BUGLE_TAG; 56 57      public static final String EXTRA_PART_ID = "part_id"; 58 59      /* 60       * A map for pending sms messages. The key is the random request UUID. 61       */ 62      private static ConcurrentHashMap&lt;Uri, SendResult&gt; sPendingMessageMap = 63          new ConcurrentHashMap&lt;Uri, SendResult&gt;(); 64 65      private static final Random RANDOM = new Random(); 66 67      // Whether we should send multipart SMS as separate messages 68      private static Boolean sSendMultipartSmsAsSeparateMessages = null; 69 </pre> <p><a href="https://android.googlesource.com/platform/packages/apps/Messaging/+nougat-mr1-release/src/com/android/messaging/sms/SmsSender.java">https://android.googlesource.com/platform/packages/apps/Messaging/+nougat-mr1-release/src/com/android/messaging/sms/SmsSender.java</a></p>

**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC
	<pre> 253 // Actually sending the message using SmsManager 254 private static void sendInternal(final Context context, final int subId, String dest, 255     final ArrayList&lt;String&gt; messages, final String serviceCenter, 256     final boolean requireDeliveryReport, final Uri messageUri) throws SmsException { 257     Assert.notNull(context); 258     final SmsManager smsManager = PhoneUtils.get(subId).getSmsManager(); 259     final int messageCount = messages.size(); 260     final ArrayList&lt;PendingIntent&gt; deliveryIntents = new ArrayList&lt;PendingIntent&gt;(messageCount); 261     final ArrayList&lt;PendingIntent&gt; sentIntents = new ArrayList&lt;PendingIntent&gt;(messageCount); 262     for (int i = 0; i &lt; messageCount; i++) { 263         // Make pending intents different for each message part 264         final int partId = (messageCount &lt;= 1 ? 0 : i + 1); 265         if (requireDeliveryReport &amp;&amp; (i == (messageCount - 1))) { 266             // TODO we only care about the delivery status of the last part 267             // Shall we have better tracking of delivery status of all parts? 268             deliveryIntents.add(PendingIntent.getBroadcast( 269                 context, 270                 partId, 271                 getSendStatusIntent(context, SendStatusReceiver.MESSAGE_DELIVERED_ACTION, 272                     messageUri, partId, subId), 273                 0/*flag*/)); 274         } else { 275             deliveryIntents.add(null); 276         } 277         sentIntents.add(PendingIntent.getBroadcast( 278             context, 279             partId, 280             getSendStatusIntent(context, SendStatusReceiver.MESSAGE_SENT_ACTION, 281                 messageUri, partId, subId), 282             0/*flag*/)); 283     } 284     if (sSendMultipartSmsAsSeparateMessages == null) { 285         sSendMultipartSmsAsSeparateMessages = MmsConfig.get(subId) 286             .getSendMultipartSmsAsSeparateMessages(); 287     } </pre> <p><a href="https://android.googlesource.com/platform/packages/apps/Messaging/+nougat-mr1-release/src/com/android/messaging/sms/SmsSender.java">https://android.googlesource.com/platform/packages/apps/Messaging/+nougat-mr1-release/src/com/android/messaging/sms/SmsSender.java</a></p>

**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC
	<pre> 288     try { 289         if (sSendMultipartSmsAsSeparateMessages) { 290             // If multipart sms is not supported, send them as separate messages 291             for (int i = 0; i &lt; messageCount; i++) { 292                 smsManager.sendTextMessage(dest, 293                     serviceCenter, 294                     messages.get(i), 295                     sentIntents.get(i), 296                     deliveryIntents.get(i)); 297             } 298         } else { 299             smsManager.sendMultipartTextMessage( 300                 dest, serviceCenter, messages, sentIntents, deliveryIntents); 301         } 302     } catch (final Exception e) { 303         throw new SmsException("SmsSender: caught exception in sending " + e); 304     } 305 } </pre> <p><a href="https://android.googlesource.com/platform/packages/apps/Messaging/+nougat-mr1-release/src/com/android/messaging/sms/SmsSender.java">https://android.googlesource.com/platform/packages/apps/Messaging/+nougat-mr1-release/src/com/android/messaging/sms/SmsSender.java</a></p> <pre> 56  * Class that receives incoming SMS messages through android.provider.Telephony.SMS_RECEIVED 57  * 58  * This class serves two purposes: 59  * - Process phone verification SMS messages 60  * - Handle SMS messages when the user has enabled us to be the default SMS app (Pre-KLP) 61  */ 62  public final class SmsReceiver extends BroadcastReceiver { 63      private static final String TAG = LogUtil.BUGLE_TAG; 64 65      private static ArrayList&lt;Pattern&gt; sIgnoreSmsPatterns; 66 </pre> <p><a href="https://android.googlesource.com/platform/packages/apps/Messaging/+nougat-mr1-release/src/com/android/messaging/receiver/SmsReceiver.java">https://android.googlesource.com/platform/packages/apps/Messaging/+nougat-mr1-release/src/com/android/messaging/receiver/SmsReceiver.java</a></p>

**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC
	<pre> 200     public static void deliverSmsMessages(final Context context, final int subId, 201         final int errorCode, final android.telephony.SmsMessage[] messages) { 202         final ContentValues messageValues = 203             MmsUtils.parseReceivedSmsMessage(context, messages, errorCode); 204 205         LogUtil.v(TAG, "SmsReceiver.deliverSmsMessages"); 206 207         final long nowInMillis = System.currentTimeMillis(); 208         final long receivedTimestampMs = MmsUtils.getMessageDate(messages[0], nowInMillis); 209 210         messageValues.put(Sms.Inbox.DATE, receivedTimestampMs); 211         // Default to unread and unseen for us but ReceiveSmsMessageAction will override 212         // seen for the telephony db. 213         messageValues.put(Sms.Inbox.READ, 0); 214         messageValues.put(Sms.Inbox.SEEN, 0); 215         if (OsUtil.isAtLeastL_MR1()) { 216             messageValues.put(Sms.SUBSCRIPTION_ID, subId); 217         } 218 219         if (messages[0].getMessageClass() == android.telephony.SmsMessage.MessageClass.CLASS_0    220             DebugUtils.debugClassZeroSmsEnabled()) { 221             Factory.get().getUIIntents().launchClassZeroActivity(context, messageValues); 222         } else { 223             final ReceiveSmsMessageAction action = new ReceiveSmsMessageAction(messageValues); 224             action.start(); 225         } 226     } </pre> <p><a href="https://android.googlesource.com/platform/packages/apps/Messaging/+nougat-mr1-release/src/com/android/messaging/receiver/SmsReceiver.java">https://android.googlesource.com/platform/packages/apps/Messaging/+nougat-mr1-release/src/com/android/messaging/receiver/SmsReceiver.java</a></p>

**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC
	<pre>228     @Override 229     public void onReceive(final Context context, final Intent intent) { 230         LogUtil.v(TAG, "SmsReceiver.onReceive " + intent); 231         // On KLP+ we only take delivery of SMS messages in SmsDeliverReceiver. 232         if (PhoneUtils.getDefault().isSmsEnabled()) { 233             final String action = intent.getAction(); 234             if (OsUtil.isSecondaryUser() &amp;&amp; 235                 (Telephony.Sms.Intents.SMS_RECEIVED_ACTION.equals(action)    236                  // TODO: update this with the actual constant from Telephony 237                  "android.provider.Telephony.MMS_DOWNLOADED".equals(action))) { 238                 postNewMessageSecondaryUserNotification(); 239             } else if (!OsUtil.isAtLeastKLP()) { 240                 deliverSmsIntent(context, intent); 241             } 242         } 243     }</pre> <p><a href="https://android.googlesource.com/platform/packages/apps/Messaging/+/nougat-mr1-release/src/com/android/messaging/receiver/SmsReceiver.java">https://android.googlesource.com/platform/packages/apps/Messaging/+/nougat-mr1-release/src/com/android/messaging/receiver/SmsReceiver.java</a></p>

**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC
	<pre>52  * Class that sends chat message via MMS. 53  * 54  * The interface emulates a blocking send similar to making an HTTP request. 55  */ 56  public class MmsSender { 57      private static final String TAG = LogUtil.BUGLE_TAG; 58 59      /** 60       * Send an MMS message. 61       * 62       * @param context Context 63       * @param messageUri The unique URI of the message for identifying it during sending 64       * @param sendReq The SendReq PDU of the message 65       * @throws MmsFailureException 66       */ 67      public static void sendMms(final Context context, final int subId, final Uri messageUri, 68                               final SendReq sendReq, final Bundle sentIntentExtras) throws MmsFailureException { 69          sendMms(context, 70                subId, 71                messageUri, 72                null /* locationUri */, 73                sendReq, 74                true /* responseImportant */, 75                sentIntentExtras); 76      }</pre> <p><a href="https://android.googlesource.com/platform/packages/apps/Messaging/+/nougat-mr1-release/src/com/android/messaging/sms/MmsSender.java">https://android.googlesource.com/platform/packages/apps/Messaging/+/nougat-mr1-release/src/com/android/messaging/sms/MmsSender.java</a></p>

**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC
	<pre> 240      * Download an MMS message. 241      * 242      * @param context Context 243      * @param contentLocation The url of the MMS message 244      * @throws MmsFailureException 245      * @throws InvalidHeaderValueException 246      */ 247      public static void downloadMms(final Context context, final int subId, 248          final String contentLocation, Bundle extras) throws MmsFailureException, 249          InvalidHeaderValueException { 250          final Uri requestUri = Uri.parse(contentLocation); 251          final Uri contentUri = MmsFileProvider.buildRawMmsUri(); 252 253          final Intent downloadedIntent = new Intent(SendStatusReceiver.MMS_DOWNLOADED_ACTION, 254              requestUri, 255              context, 256              SendStatusReceiver.class); 257          downloadedIntent.putExtra(SendMessageAction.EXTRA_CONTENT_URI, contentUri); 258          if (extras != null) { 259              downloadedIntent.putExtras(extras); 260          } 261          final PendingIntent downloadedPendingIntent = PendingIntent.getBroadcast( 262              context, 263              0 /*request code*/, 264              downloadedIntent, 265              PendingIntent.FLAG_UPDATE_CURRENT); 266 267          MmsManager.downloadMultimediaMessage(subId, context, contentLocation, contentUri, 268              downloadedPendingIntent); 269      } </pre> <p><a href="https://android.googlesource.com/platform/packages/apps/Messaging/+nougat-mr1-release/src/com/android/messaging/sms/MmsSender.java">https://android.googlesource.com/platform/packages/apps/Messaging/+nougat-mr1-release/src/com/android/messaging/sms/MmsSender.java</a></p>



**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC
	<pre> 97      * Execute an MMS HTTP request, either a POST (sending) or a GET (downloading) 98      * 99      * @param urlString The request URL, for sending it is usually the MMSC, and for downloading 100     *           it is the message URL 101     * @param pdu For POST (sending) only, the PDU to send 102     * @param method HTTP method, POST for sending and GET for downloading 103     * @param isProxySet Is there a proxy for the MMSC 104     * @param proxyHost The proxy host 105     * @param proxyPort The proxy port 106     * @param mmsConfig The MMS config to use 107     * @param userAgent The user agent header value 108     * @param uaProfUrl The UA Prof URL header value 109     * @return The HTTP response body 110     * @throws MmsHttpException For any failures 111     */ 112     public byte[] execute(String urlString, byte[] pdu, String method, boolean isProxySet, 113                          String proxyHost, int proxyPort, Bundle mmsConfig, String userAgent, String uaProfUrl) 114         throws MmsHttpException { 115         Log.d(MmsService.TAG, "HTTP: " + method + " " + Utils.redactUrlForNonVerbose(urlString) 116             + (isProxySet ? (" proxy=" + proxyHost + ":" + proxyPort) : "")) 117             + ", PDU size=" + (pdu != null ? pdu.length : 0)); 118         checkMethod(method); 119         HttpURLConnection connection = null; 120         try { 121             Proxy proxy = Proxy.NO_PROXY; 122             if (isProxySet) { 123                 proxy = new Proxy(Proxy.Type.HTTP, new InetSocketAddress(proxyHost, proxyPort)); 124             } 125             final URL url = new URL(urlString); 126             // Now get the connection 127             connection = (HttpURLConnection) url.openConnection(proxy); 128             connection.setDoInput(true); 129             connection.setConnectTimeout( 130                 mmsConfig.getInt(CarrierConfigValuesLoader.CONFIG_HTTP_SOCKET_TIMEOUT, 131                     CarrierConfigValuesLoader.CONFIG_HTTP_SOCKET_TIMEOUT_DEFAULT)); </pre> <p data-bbox="506 1219 1596 1287"><a href="https://android.googlesource.com/platform/packages/apps/Messaging/+/nougat-mr1-release/src/android/support/v7/mms/MmsHttpClient.java">https://android.googlesource.com/platform/packages/apps/Messaging/+/nougat-mr1-release/src/android/support/v7/mms/MmsHttpClient.java</a></p>

**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC
	<pre> 132 // ----- COMMON HEADERS ----- 133 // Header: Accept 134 connection.setRequestProperty(HEADER_ACCEPT, HEADER_VALUE_ACCEPT); 135 // Header: Accept-Language 136 connection.setRequestProperty( 137     HEADER_ACCEPT_LANGUAGE, getCurrentAcceptLanguage(Locale.getDefault())); 138 // Header: User-Agent 139 Log.i(MmsService.TAG, "HTTP: User-Agent=" + userAgent); 140 connection.setRequestProperty(HEADER_USER_AGENT, userAgent); 141 // Header: x-wap-profile 142 final String uaProfUrlTagName = mmsConfig.getString( 143     CarrierConfigValuesLoader.CONFIG_UA_PROF_TAG_NAME, 144     CarrierConfigValuesLoader.CONFIG_UA_PROF_TAG_NAME_DEFAULT); 145 if (uaProfUrl != null) { 146     Log.i(MmsService.TAG, "HTTP: UaProfUrl=" + uaProfUrl); 147     connection.setRequestProperty(uaProfUrlTagName, uaProfUrl); 148 } 149 // Add extra headers specified by mms_config.xml's httpparams 150 addExtraHeaders(connection, mmsConfig); 151 // Different stuff for GET and POST 152 if (METHOD_POST.equals(method)) { 153     if (pdu == null    pdu.length &lt; 1) { 154         Log.e(MmsService.TAG, "HTTP: empty pdu"); 155         throw new MmsHttpException(0/*statusCode*/, "Sending empty PDU"); 156     } 157     connection.setDoOutput(true); 158     connection.setRequestMethod(METHOD_POST); 159     if (mmsConfig.getBoolean( 160         CarrierConfigValuesLoader.CONFIG_SUPPORT_HTTP_CHARSET_HEADER, 161         CarrierConfigValuesLoader.CONFIG_SUPPORT_HTTP_CHARSET_HEADER_DEFAULT)) { 162         connection.setRequestProperty(HEADER_CONTENT_TYPE, 163             HEADER_VALUE_CONTENT_TYPE_WITH_CHARSET); 164     } else { 165         connection.setRequestProperty(HEADER_CONTENT_TYPE, 166             HEADER_VALUE_CONTENT_TYPE_WITHOUT_CHARSET); </pre> <p><a href="https://android.googlesource.com/platform/packages/apps/Messaging/+nougat-mr1-release/src/android/support/v7/mms/MmsHttpClient.java">https://android.googlesource.com/platform/packages/apps/Messaging/+nougat-mr1-release/src/android/support/v7/mms/MmsHttpClient.java</a></p>

**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC
	<pre> 167     } 168     if (Log.isLoggable(MmsService.TAG, Log.VERBOSE)) { 169         logHttpHeaders(connection.getRequestProperties()); 170     } 171     connection.setFixedLengthStreamingMode(pdu.length); 172     // Sending request body 173     final OutputStream out = 174         new BufferedOutputStream(connection.getOutputStream()); 175     out.write(pdu); 176     out.flush(); 177     out.close(); 178 } else if (METHOD_GET.equals(method)) { 179     if (Log.isLoggable(MmsService.TAG, Log.VERBOSE)) { 180         logHttpHeaders(connection.getRequestProperties()); 181     } 182     connection.setRequestMethod(METHOD_GET); 183 } 184 // Get response 185 final int responseCode = connection.getResponseCode(); 186 final String responseMessage = connection.getResponseMessage(); 187 Log.d(MmsService.TAG, "HTTP: " + responseCode + " " + responseMessage); 188 if (Log.isLoggable(MmsService.TAG, Log.VERBOSE)) { 189     logHttpHeaders(connection.getHeaderFields()); 190 } 191 if (responseCode / 100 != 2) { 192     throw new MmsHttpException(responseCode, responseMessage); 193 } 194 final InputStream in = new BufferedInputStream(connection.getInputStream()); 195 final ByteArrayOutputStream byteOut = new ByteArrayOutputStream(); 196 final byte[] buf = new byte[4096]; 197 int count = 0; 198 while ((count = in.read(buf)) &gt; 0) { 199     byteOut.write(buf, 0, count); 200 } 201 in.close(); 202 final byte[] responseBody = byteOut.toByteArray(); 203 Log.d(MmsService.TAG, "HTTP: response size=" 204     + (responseBody != null ? responseBody.length : 0)); 205 return responseBody; </pre> <p><a href="https://android.goesource.com/platform/packages/apps/Messaging/+nougat-mr1-release/src/android/support/v7/mms/MmsHttpClient.java">https://android.goesource.com/platform/packages/apps/Messaging/+nougat-mr1-release/src/android/support/v7/mms/MmsHttpClient.java</a></p>

**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC
	<pre> 38  * Request to send an MMS 39  */ 40  class SendRequest extends MmsRequest { 41      // Max send response PDU size in bytes (exceeding this may cause problem with 42      // system intent delivery). 43      private static final int MAX_SEND_RESPONSE_SIZE = 1000 * 1024; 44 45      private byte[] mPduData; 46 47      SendRequest(final String locationUri, final Uri pduUri, final PendingIntent sentIntent) { 48          super(locationUri, pduUri, sentIntent); 49      } 50 51      @Override 52      protected boolean loadRequest(final Context context, final Bundle mmsConfig) { 53          mPduData = readPduFromContentUri( 54              context, 55              mPduUri, 56              mmsConfig.getInt( 57                  CarrierConfigValuesLoader.CONFIG_MAX_MESSAGE_SIZE, 58                  CarrierConfigValuesLoader.CONFIG_MAX_MESSAGE_SIZE_DEFAULT)); 59          return (mPduData != null); 60      } 61 62      @Override 63      protected boolean transferResponse(final Context context, final Intent fillIn, 64          final byte[] response) { 65          // SendConf pdus are always small and can be included in the intent 66          if (response != null &amp;&amp; fillIn != null) { 67              if (response.length &gt; MAX_SEND_RESPONSE_SIZE) { 68                  // If the response PDU is too large, it won't be able to fit in 69                  // the PendingIntent to be transferred via system IPC. 70                  return false; 71              } 72              fillIn.putExtra(SmsManager.EXTRA_MMS_DATA, response); 73          } 74          return true; 75      } </pre>

**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC
	<p><a href="https://android.googlesource.com/platform/packages/apps/Messaging/+/nougat-mr1-release/src/android/support/v7/mms/SendRequest.java">https://android.googlesource.com/platform/packages/apps/Messaging/+/nougat-mr1-release/src/android/support/v7/mms/SendRequest.java</a></p> <pre data-bbox="533 354 1738 397">public static LocationRequest create ()</pre> <p data-bbox="525 423 1029 451">Create a location request with default parameters.</p> <p data-bbox="525 482 1638 542">Default parameters are for a block accuracy, slowly updated location. It can then be adjusted as required by the applications before passing to the <code>FusedLocationProviderApi</code>.</p> <p data-bbox="548 565 630 589"><b>Returns</b></p> <ul data-bbox="554 610 810 634" style="list-style-type: none"> <li>• a new location request</li> </ul> <p><a href="https://developers.google.com/android/reference/com/google/android/gms/location/LocationRequest">https://developers.google.com/android/reference/com/google/android/gms/location/LocationRequest</a></p>

**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC
	<p><code>public static final int PRIORITY_BALANCED_POWER_ACCURACY</code></p> <p>Used with <code>setPriority(int)</code> to request "block" level accuracy.</p> <p>Block level accuracy is considered to be about 100 meter accuracy. Using a coarse accuracy such as this often consumes less power.</p> <p>Constant Value: 102</p> <hr/> <p><code>public static final int PRIORITY_HIGH_ACCURACY</code></p> <p>Used with <code>setPriority(int)</code> to request the most accurate locations available.</p> <p>This will return the finest location available.</p> <p>Constant Value: 100</p> <hr/> <p><code>public static final int PRIORITY_LOW_POWER</code></p> <p>Used with <code>setPriority(int)</code> to request "city" level accuracy.</p> <p>City level accuracy is considered to be about 10km accuracy. Using a coarse accuracy such as this often consumes less power.</p> <p>Constant Value: 104</p> <p><a href="https://developers.google.com/android/reference/com/google/android/gms/location/LocationRequest">https://developers.google.com/android/reference/com/google/android/gms/location/LocationRequest</a></p>

**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC
	<p data-bbox="533 248 1751 285"><code>public Task&lt;Location&gt; getLastLocation ()</code></p> <p data-bbox="533 315 1104 337">Returns the best most recent location currently available.</p> <p data-bbox="533 371 1696 430">If a location is not available, which should happen very rarely, null will be returned. The best accuracy available while respecting the location permissions will be returned.</p> <p data-bbox="533 464 1738 522">This method provides a simplified way to get location. It is particularly well suited for applications that do not require an accurate location and that do not want to maintain extra logic for location updates.</p> <p data-bbox="533 578 1751 615"><code>public Task&lt;LocationAvailability&gt; getLocationAvailability ()</code></p> <p data-bbox="533 646 1692 704">Returns the availability of location data. When <code>isLocationAvailable()</code> returns true, then the location returned by <code>getLastLocation()</code> will be reasonably up to date within the hints specified by the active <code>LocationRequest</code>s.</p> <p data-bbox="533 738 1472 761">If the client isn't connected to Google Play services and the request times out, null is returned.</p> <p data-bbox="533 795 1675 854">Note it's always possible for <code>getLastLocation()</code> to return null even when this method returns true (e.g. location settings were disabled between calls).</p> <p data-bbox="512 865 1898 930"><a href="https://developers.google.com/android/reference/com/google/android/gms/location/FusedLocationProviderClient">https://developers.google.com/android/reference/com/google/android/gms/location/FusedLocationProviderClient</a></p>

**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC						
	<p data-bbox="531 245 1749 326"> <code>public Task&lt;Void&gt; requestLocationUpdates (LocationRequest request, LocationCallback callback, Looper looper)</code> </p> <p data-bbox="531 354 1272 378">Requests location updates with a callback on the specified Looper thread.</p> <p data-bbox="531 410 1686 472">This method is suited for the foreground use cases. For background use cases, the <code>PendingIntent</code> version of the method is recommended, see <code>requestLocationUpdates(LocationRequest, PendingIntent)</code>.</p> <p data-bbox="531 505 1371 529">Any previous LocationRequests registered on this LocationListener will be replaced.</p> <p data-bbox="531 561 1686 656">This call will keep the Google Play services connection active, so make sure to call <code>removeLocationUpdates(LocationCallback)</code> when you no longer need it, otherwise you lose the benefits of the automatic connection management.</p> <p data-bbox="531 688 1745 712">Callbacks for <code>LocationCallback</code> will be made on the specified thread, which must already be a prepared looper thread.</p> <p data-bbox="531 737 667 761"><b>Parameters</b></p> <table border="1" data-bbox="531 792 1749 1008"> <tbody> <tr> <td data-bbox="531 800 625 857"><b>request</b></td> <td data-bbox="632 800 1749 857">The location request for the updates.</td> </tr> <tr> <td data-bbox="531 865 625 922"><b>callback</b></td> <td data-bbox="632 865 1749 922">The callback for the location updates.</td> </tr> <tr> <td data-bbox="531 930 625 1003"><b>looper</b></td> <td data-bbox="632 930 1749 1003">The Looper object whose message queue will be used to implement the callback mechanism, or null to make callbacks on the calling thread.</td> </tr> </tbody> </table> <p data-bbox="531 1024 1902 1089"> <a href="https://developers.google.com/android/reference/com/google/android/gms/location/FusedLocationProviderClient">https://developers.google.com/android/reference/com/google/android/gms/location/FusedLocationProviderClient</a> </p>	<b>request</b>	The location request for the updates.	<b>callback</b>	The callback for the location updates.	<b>looper</b>	The Looper object whose message queue will be used to implement the callback mechanism, or null to make callbacks on the calling thread.
<b>request</b>	The location request for the updates.						
<b>callback</b>	The callback for the location updates.						
<b>looper</b>	The Looper object whose message queue will be used to implement the callback mechanism, or null to make callbacks on the calling thread.						



**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC				
	<p data-bbox="533 245 1740 321"> <code>public Task&lt;Void&gt; requestLocationUpdates (LocationRequest request, PendingIntent callbackIntent)</code> </p> <p data-bbox="533 354 1268 378">Requests location updates with a callback on the specified PendingIntent.</p> <p data-bbox="533 410 1730 540">This method is suited for the background use cases, more specifically for receiving location updates, even when the app has been killed by the system. In order to do so, use a <code>PendingIntent</code> for a started service. For foreground use cases, the <code>LocationCallback</code> version of the method is recommended, see <code>requestLocationUpdates(LocationRequest, LocationCallback, Looper)</code>.</p> <p data-bbox="533 573 1730 630">Any previously registered requests that have the same PendingIntent (as defined by <code>equals(Object)</code>) will be replaced by this request.</p> <p data-bbox="533 662 1730 751">Both <code>LocationResult</code> and <code>LocationAvailability</code> are sent to the given PendingIntent. You can extract data from an Intent using <code>hasResult(Intent)</code>, <code>extractResult(Intent)</code>, <code>hasLocationAvailability(Intent)</code>, and <code>extractLocationAvailability(Intent)</code>.</p> <p data-bbox="533 784 667 808"><b>Parameters</b></p> <table border="1" data-bbox="533 833 1740 971"> <tr> <td data-bbox="533 833 835 906"><code>request</code></td> <td data-bbox="835 833 1740 906">The location request for the updates.</td> </tr> <tr> <td data-bbox="533 906 835 971"><code>callbackIntent</code></td> <td data-bbox="835 906 1740 971">A pending intent to be sent for each location update.</td> </tr> </table> <p data-bbox="533 1003 630 1027"><b>Returns</b></p> <ul data-bbox="533 1044 1360 1068" style="list-style-type: none"> <li>• a Task for the call, check <code>isSuccessful()</code> to determine if it was successful.</li> </ul> <p data-bbox="533 1084 1898 1146"> <a href="https://developers.google.com/android/reference/com/google/android/gms/location/FusedLocationProviderClient">https://developers.google.com/android/reference/com/google/android/gms/location/FusedLocationProviderClient</a> </p>	<code>request</code>	The location request for the updates.	<code>callbackIntent</code>	A pending intent to be sent for each location update.
<code>request</code>	The location request for the updates.				
<code>callbackIntent</code>	A pending intent to be sent for each location update.				

## Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products

US9408055B2	HTC						
	<div style="background-color: #f0f0f0; padding: 5px; margin-bottom: 10px;"> <p><code>public void onLocationAvailability (LocationAvailability locationAvailability)</code></p> </div> <p>Called when there is a change in the availability of location data.</p> <p>When <code>isLocationAvailable()</code> returns <code>false</code> you can assume that location will not be returned in <code>onLocationResult(LocationResult)</code> until something changes in the device's settings or environment. Even when <code>isLocationAvailable()</code> returns <code>true</code> the <code>onLocationResult(LocationResult)</code> may not always be called regularly, however the device location is known and both the most recently delivered location and <code>getLastLocation(GoogleApiClient)</code> will be reasonably up to date given the hints specified by the active <code>LocationRequest</code>s.</p> <p><b>Parameters</b></p> <table style="width: 100%; border-collapse: collapse; margin-top: 10px;"> <tr> <td style="background-color: #444; color: white; padding: 5px;"><code>locationAvailability</code></td> <td style="padding: 5px;">The current status of location availability.</td> </tr> </table> <div style="background-color: #f0f0f0; padding: 5px; margin-top: 10px;"> <p><code>public void onLocationResult (LocationResult result)</code></p> </div> <p>Called when device location information is available.</p> <p>The most recent location returned by <code>getLastLocation()</code> is not guaranteed to be immediately fresh, but will be reasonably up to date given the hints specified by the active <code>LocationRequest</code>s.</p> <p><b>Parameters</b></p> <table style="width: 100%; border-collapse: collapse; margin-top: 10px;"> <tr> <td style="background-color: #444; color: white; padding: 5px;"><code>result</code></td> <td style="padding: 5px;">The latest location result available.</td> </tr> </table> <p><a href="https://developers.google.com/android/reference/com/google/android/gms/location/LocationCallback">https://developers.google.com/android/reference/com/google/android/gms/location/LocationCallback</a></p> <div style="background-color: #f0f0f0; padding: 5px; margin-top: 10px;"> <p><code>public abstract void onLocationChanged (Location location)</code></p> </div> <p>Called when the location has changed.</p> <p><b>Parameters</b></p> <table style="width: 100%; border-collapse: collapse; margin-top: 10px;"> <tr> <td style="background-color: #444; color: white; padding: 5px;"><code>location</code></td> <td style="padding: 5px;">The updated location.</td> </tr> </table>	<code>locationAvailability</code>	The current status of location availability.	<code>result</code>	The latest location result available.	<code>location</code>	The updated location.
<code>locationAvailability</code>	The current status of location availability.						
<code>result</code>	The latest location result available.						
<code>location</code>	The updated location.						

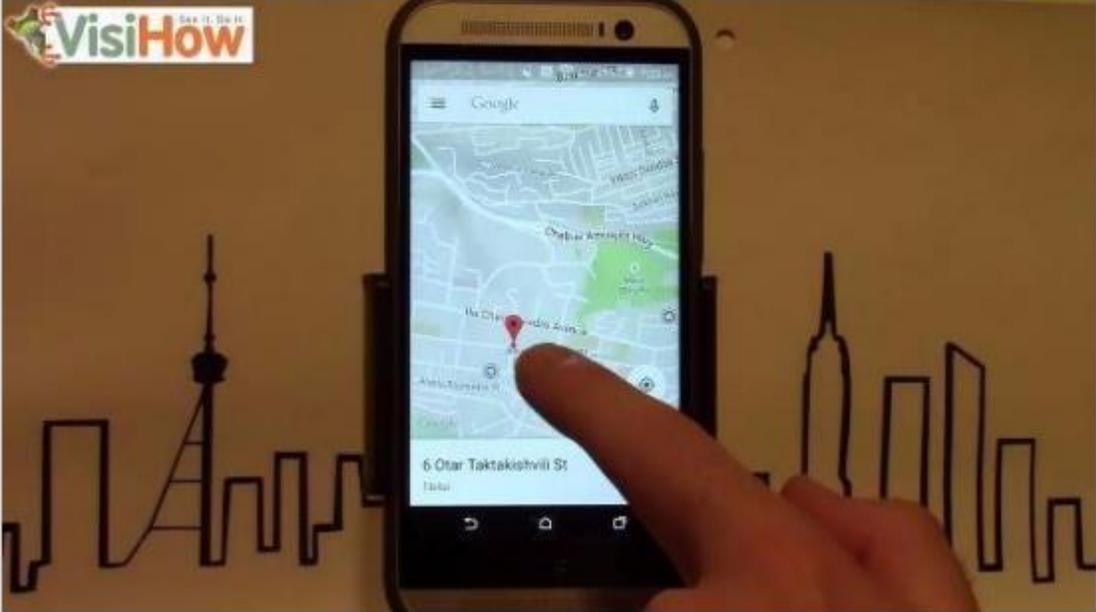
**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC
	<p data-bbox="512 233 1797 264"><a href="https://developers.google.com/android/reference/com/google/android/gms/location/LocationListener">https://developers.google.com/android/reference/com/google/android/gms/location/LocationListener</a></p> <p data-bbox="512 315 814 345">Public Constructors</p> <hr data-bbox="512 358 1740 362"/> <p data-bbox="512 412 926 443">public <b>MapView</b> (<a href="#">Context</a> context)</p> <p data-bbox="512 505 1129 535">public <b>MapView</b> (<a href="#">Context</a> context, <a href="#">AttributeSet</a> attrs)</p> <p data-bbox="512 597 1268 628">public <b>MapView</b> (<a href="#">Context</a> context, <a href="#">AttributeSet</a> attrs, int defStyle)</p> <p data-bbox="512 690 1247 721">public <b>MapView</b> (<a href="#">Context</a> context, <a href="#">GoogleMapOptions</a> options)</p> <p data-bbox="512 748 1671 779"><a href="https://developers.google.com/android/reference/com/google/android/gms/maps/MapView">https://developers.google.com/android/reference/com/google/android/gms/maps/MapView</a></p>

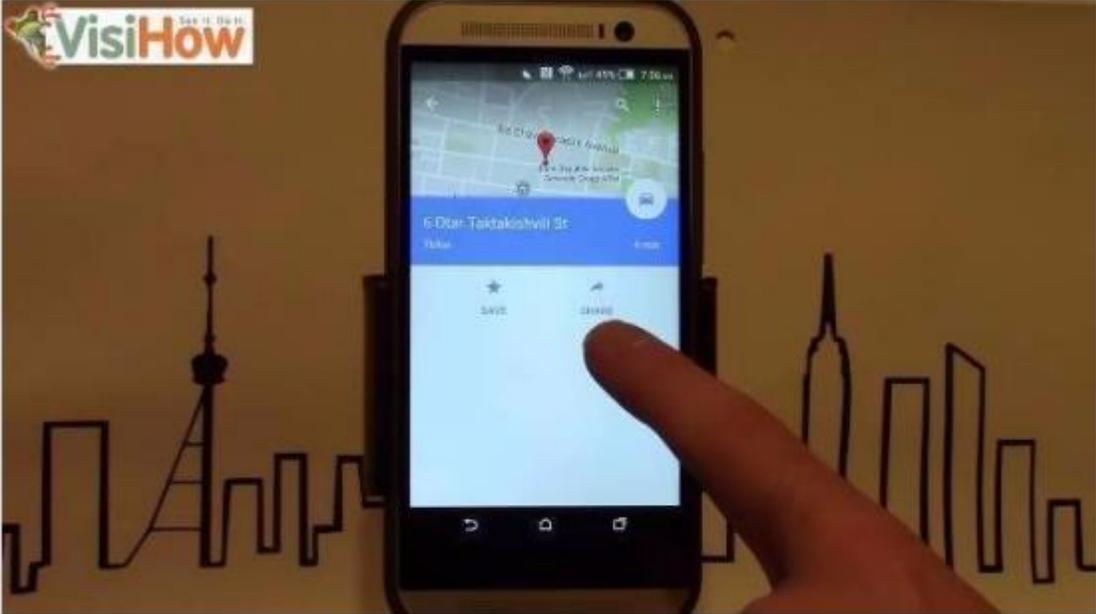
**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC		
	<p><code>public void <b>getMapAsync</b> (<a href="#">OnMapReadyCallback</a> callback)</code></p> <p>Returns a non-null instance of the <a href="#">GoogleMap</a>, ready to be used.</p> <p>Note that:</p> <ul style="list-style-type: none"> <li>• This method must be called from the main thread.</li> <li>• The callback will be executed in the main thread.</li> <li>• In the case where Google Play services is not installed on the user's device, the callback will not be triggered until the user installs it.</li> <li>• The <a href="#">GoogleMap</a> object provided by the callback is non-null.</li> </ul> <p><b>Parameters</b></p> <table border="1" data-bbox="527 686 1738 753"> <tr> <td><code>callback</code></td> <td>The callback object that will be triggered when the map is ready to be used.</td> </tr> </table> <p><code>public final void <b>onCreate</b> (<a href="#">Bundle</a> savedInstanceState)</code></p> <p>You must call this method from the parent Activity/Fragment's corresponding method.</p> <p><a href="https://developers.google.com/android/reference/com/google/android/gms/maps/MapView">https://developers.google.com/android/reference/com/google/android/gms/maps/MapView</a></p>	<code>callback</code>	The callback object that will be triggered when the map is ready to be used.
<code>callback</code>	The callback object that will be triggered when the map is ready to be used.		
<p>[1G] receiving user input via user interaction with the interactive display of the first device, the user input specifying a location and a symbol corresponding to an entity other than the first device and the second devices;</p>	<p>HTC infringes directly and/or indirectly by performing, inducing others to perform, and/or contributing to the performance of receiving user input via user interaction with the interactive display of the first device, the user input specifying a location and a symbol corresponding to an entity other than the first device and the second devices.</p> <p><b>Share Locations from Google Maps on HTC One M8</b></p>		

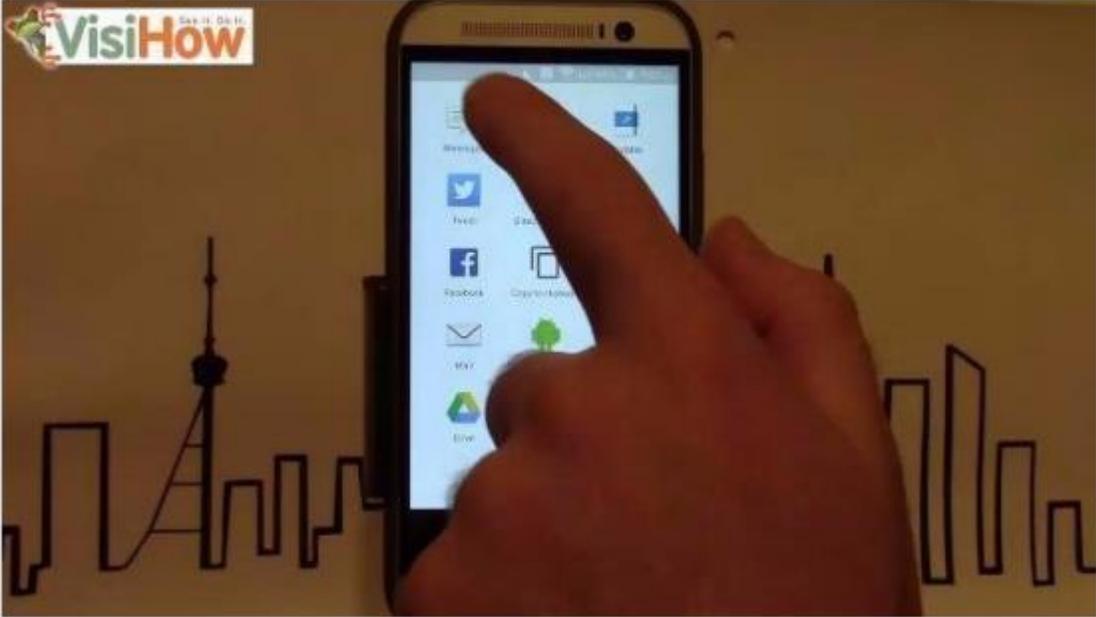
## Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products

US9408055B2	HTC
	<p data-bbox="531 233 829 256"><b>Place a pin on the map.</b></p> <p data-bbox="531 272 1640 380">We can do this anywhere in the entire world, anywhere, on any map that we find in the Google Maps application. We are going to choose a location, press down on the screen, and hold down until we can see that a red pin has been dropped.</p> 

## Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products

US9408055B2	HTC
	<p data-bbox="527 240 1207 310"><b>We have two options:</b> "SAVE" and "SHARE". Press down on the word "SHARE".</p>  <p data-bbox="527 1057 1633 1208"><b>A menu appears with all of the different ways that we can share this location.</b> We can share this location via messages, Bluetooth, Twitter, Facebook, mail, Gmail and Viber. If you have other applications, such as WhatsApp, or any other chat or email applications, you will be able to use them as well and will see them here.</p>

### Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products

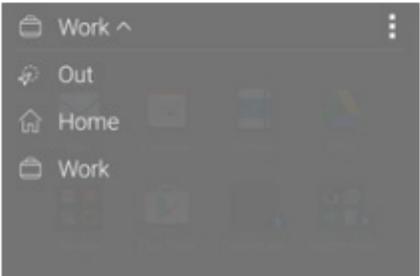
US9408055B2	HTC
	<p data-bbox="527 240 856 264"><b>Let's choose "Messages".</b></p> <p data-bbox="527 277 1633 345">This option is located on the top left-hand corner of the screen. All options will have the same basic processes. They will ask us to send the location. Press down on the "Messages" icon.</p> 

## Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products

US9408055B2	HTC
	<p data-bbox="520 233 1260 256"><b>Press the box next to the contact who will be the recipient.</b></p> <p data-bbox="520 269 1549 292">A checkmark will appear in the box. We can select multiple contacts as well if we desire.</p>  <p data-bbox="520 1045 911 1068"><b>Press on the word "DONE (1)".</b></p> <p data-bbox="520 1084 1629 1153">This icon is located in the bottom right-hand corner of the screen. The bracketed number indicates the number of contacts we selected.</p> <p data-bbox="520 1166 1407 1188"><a href="http://visihow.com/Share_Locations_from_Google_Maps_on_HTC_One_M8">http://visihow.com/Share_Locations_from_Google_Maps_on_HTC_One_M8</a></p>



## Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products

US9408055B2	HTC
	<h3 data-bbox="520 261 1115 310">Manually switching locations</h3> <p data-bbox="520 358 1413 467">The HTC Sense Home widget automatically changes locations based on where you are. You can also manually change the location in the HTC Sense Home widget.</p> <p data-bbox="520 505 1402 613">For the HTC Sense Home widget to change locations automatically, you need to make sure that location services is turned on. See <a href="#">Turning location services on or off</a>.</p> <ol data-bbox="583 651 1392 776" style="list-style-type: none"><li>1. On your Home screen, slide right or left until you see the HTC Sense Home widget.</li><li>2. Tap , and then tap the location you want.</li></ol>  <p data-bbox="527 1101 1409 1127">HTC One (M8) - Manually switching locations - Support   HTC United States</p>

## Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products

US9408055B2	HTC
	<h3 data-bbox="552 269 1331 318">Setting your home and work locations</h3> <p data-bbox="552 367 1415 435">In the HTC Sense Home widget, set your home and work locations based on your address, Wi-Fi network, or both.</p> <p data-bbox="552 475 1425 621">You can associate multiple addresses and Wi-Fi networks to each of these locations. Using your set addresses or Wi-Fi networks, the HTC Sense Home widget will be able to determine where you are and display the appropriate apps.</p> <ol data-bbox="621 662 1457 1146" style="list-style-type: none"><li data-bbox="621 662 1423 727">1. On the Home screen, swipe right or left until you see the HTC Sense Home widget.</li><li data-bbox="621 751 972 784">2. Tap  &gt;  &gt; <b>Set locations</b>.</li><li data-bbox="621 816 1062 849">3. Choose the location you want to set.</li><li data-bbox="621 865 1457 1092">4. Tap  and do one of the following:<ul data-bbox="701 930 1457 1092" style="list-style-type: none"><li data-bbox="701 930 1444 995">• Tap <b>Address</b> and then enter your street address or select it on the map.</li><li data-bbox="701 1027 1457 1092">• Tap <b>Wi-Fi network</b> and select one or more Wi-Fi networks you want to associate with the location.</li></ul></li><li data-bbox="621 1117 1457 1146">5. When you've finished setting your home and work locations, press .</li></ol> <p data-bbox="520 1182 1503 1214">HTC One (M8) - Setting your home and work locations - Support   HTC United States</p>

## Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products

US9408055B2	HTC
	<p data-bbox="533 233 1304 272"><b>HTC One V™ – Google Location Service &amp; GPS</b></p> <p data-bbox="533 318 1717 423">Google Maps lets you track your current location, view real-time traffic situations, and receive detailed directions to your destination. It also provides a search tool where you can locate a place of interest or an address on a vector or aerial map, or view locations in street level.</p> <p data-bbox="533 467 905 496"><b>Turning on Location Services</b></p> <div data-bbox="533 508 1717 943"> </div> <ol data-bbox="533 976 1717 1146" style="list-style-type: none"> <li>1. From the Home Screen, slide the <b>Notifications</b> panel open.</li> <li>2. In the top right corner, tap <b>Settings</b>.</li> <li>3. Tap <b>Location</b>.</li> <li>4. Make your selection by tapping <b>Google's location service</b>, <b>Use GPS satellites</b>, or both. <b>Note:</b> You will need to accept the location consent terms and conditions.</li> </ol> <p data-bbox="512 1203 1902 1414"><b>Regarding Google Maps</b>, Google Latitude, Google Plus, Google Hangouts, Google Messages, and Android Messenger, the Accused Products are configured to allow a user of the first device to interact with the display, to select a user, contact, or device, and to select an action to be performed, such as: sending a message, initiating a call, initiating a data conference, sharing a location, stop sharing a location, block a user from location sharing, sending a location, requesting a location, or sending other data. When an action is specified, data is sent from the first device to the second device via a server. In an example, using Google</p>

**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC
	<p>Maps, a user can interact with the display to specify a location that does not correspond to the first or second devices. Alternatively, a user can share an ETA which includes a route, where the route is not the location of any of the first or second devices.</p> <p>Selection with Markers: <a href="https://developers.google.com/maps/documentation/android-api/marker">https://developers.google.com/maps/documentation/android-api/marker</a></p> <p>Queries with GeoTagging database: <a href="https://developers.google.com/android/reference/com/google/android/gms/location/places/GeoDataClient">https://developers.google.com/android/reference/com/google/android/gms/location/places/GeoDataClient</a></p>

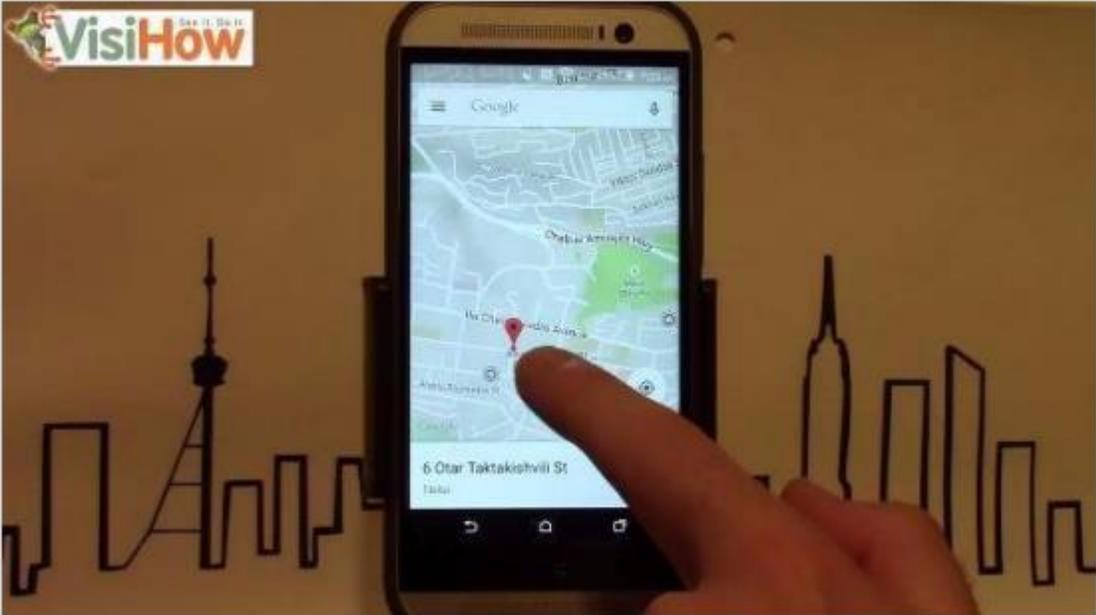
## Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products

US9408055B2	HTC
	<h3 data-bbox="541 245 1255 293">Embed a map or share a location</h3> <p data-bbox="541 313 1516 391">On your computer, you can embed a map, Street View image, driving directions, or search into your website or blog. On your computer, phone, or tablet, you can share a location or map with others over email, Google+, Facebook, Twitter, or text.</p> <p data-bbox="554 467 947 488"><a href="#">ANDROID</a> <a href="#">COMPUTER</a> <a href="#">IPHONE &amp; IPAD</a></p> <hr data-bbox="541 509 1528 513"/> <h3 data-bbox="541 565 848 591">Share a map or location</h3> <ol data-bbox="554 613 1230 773" style="list-style-type: none"><li>1. Open the Google Maps app .</li><li>2. Search for a place. Or find a place on the map then touch and hold to drop a pin.</li><li>3. At the bottom, tap the place's name or address.</li><li>4. Tap Share . If you don't see this, tap More  &gt; <b>Share</b>.</li><li>5. Select an app. It'll send a link that shows the place in Google Maps.</li></ol> <p data-bbox="525 805 1640 831"><a href="https://support.google.com/maps/answer/144361?co=GENIE.Platform%3DAndroid&amp;hl=en">https://support.google.com/maps/answer/144361?co=GENIE.Platform%3DAndroid&amp;hl=en</a></p> <h3 data-bbox="541 898 737 924">Share your E.T.A</h3> <p data-bbox="541 946 1461 967">After you start your drive, you can share your destination, estimated arrival time, and where you are on the route.</p> <ol data-bbox="546 989 1209 1170" style="list-style-type: none"><li>1. Open the Google Maps app .</li><li>2. Set a driving destination. <a href="#">Learn how to navigate to a place.</a></li><li>3. After you start navigation, tap <b>More</b>  &gt; <b>Share trip progress</b>.</li><li>4. Choose a person from the list.</li><li>5. Tap <b>Share</b>.</li><li>6. Location Sharing will stop when you reach your destination or stop navigating.</li></ol> <p data-bbox="541 1192 1083 1213">• To stop sharing before you arrive, tap <b>More</b>  &gt; <b>Stop sharing</b>.</p> <p data-bbox="512 1219 1696 1245"><a href="https://support.google.com/maps/answer/7326816?co=GENIE.Platform%3DAndroid&amp;hl=en">https://support.google.com/maps/answer/7326816?co=GENIE.Platform%3DAndroid&amp;hl=en</a></p> <p data-bbox="512 1292 1520 1318">Markers (adding location information to the link associated with the database):</p> <div data-bbox="512 1360 1915 1396" style="background-color: black; height: 22px;"></div>

**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC
	<pre>static final LatLng PERTH = new LatLng(-31.90, 115.86); Marker perth = mMap.addMarker(new MarkerOptions()     .position(PERTH)     .draggable(true));</pre>
<p>[1H] and based on the user input, adding the user-specified symbol to the interactive display at a position on the interactive map corresponding to the user-specified location, and transmitting the user-specified symbol and location to the second devices for addition of the user-specified symbol to respective interactive displays of the second devices at respective positions on respective interactive maps corresponding to the user-specified location.</p>	<p>HTC infringes directly and/or indirectly by performing, inducing others to perform, and/or contributing to the performance of: based on the user input, adding the user-specified symbol to the interactive display at a position on the interactive map corresponding to the user-specified location, and transmitting the user-specified symbol and location to the second devices for addition of the user-specified symbol to respective interactive displays of the second devices at respective positions on respective interactive maps corresponding to the user-specified location.</p> <p>A user can interact with the display to specify a location that does not correspond to the first or second devices. A user can drop a symbol pin on the specified location. A user can then share that location and transmit the location to one or more second devices using Android Messages, Google Hangouts, or another application.</p> <p>Alternatively, a user can share an ETA which includes a route, where the route is not the location of any of the first or second devices. Again, this route can be shared with users over Android Messages, Google Hangouts, or another application.</p> <p><b>Share Locations from Google Maps on HTC One M8</b></p>

## Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products

US9408055B2	HTC
	<p data-bbox="531 233 827 256"><b>Place a pin on the map.</b></p> <p data-bbox="531 272 1640 380">We can do this anywhere in the entire world, anywhere, on any map that we find in the Google Maps application. We are going to choose a location, press down on the screen, and hold down until we can see that a red pin has been dropped.</p>  <p data-bbox="512 1141 743 1164"><b>Placing a Marker:</b></p> <p data-bbox="512 1179 1430 1201"><a href="https://developers.google.com/maps/documentation/android-api/marker">https://developers.google.com/maps/documentation/android-api/marker</a></p> <p data-bbox="512 1252 1083 1274">based on queries with GeoTagging database:</p> <p data-bbox="512 1289 1860 1312"><a href="https://developers.google.com/android/reference/com/google/android/gms/location/places/GeoDataClient">https://developers.google.com/android/reference/com/google/android/gms/location/places/GeoDataClient</a></p>

**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

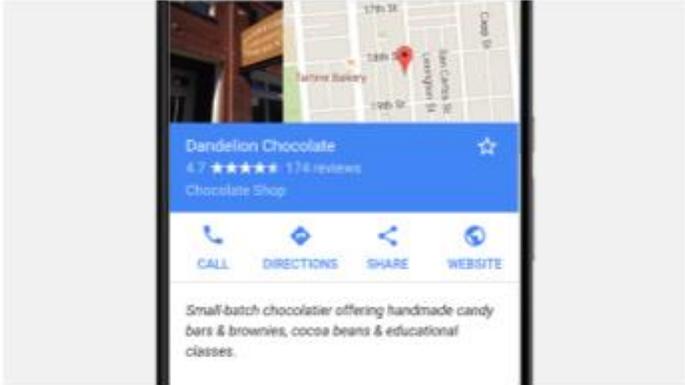
US9408055B2	HTC
	<p><b>Embed a map or share a location</b></p> <p>On your computer, you can embed a map, Street View image, driving directions, or search into your website or blog. On your computer, phone, or tablet, you can share a location or map with others over email, Google+, Facebook, Twitter, or text.</p> <p><b>ANDROID</b> COMPUTER IPHONE &amp; IPAD</p> <hr/> <p><b>Share a map or location</b></p> <ol style="list-style-type: none"> <li>1. Open the Google Maps app .</li> <li>2. Search for a place. Or find a place on the map then touch and hold to drop a pin.</li> <li>3. At the bottom, tap the place's name or address.</li> <li>4. Tap Share . If you don't see this, tap More  &gt; <b>Share</b>.</li> <li>5. Select an app. It'll send a link that shows the place in Google Maps.</li> </ol> <p><a href="https://support.google.com/maps/answer/144361?co=GENIE.Platform%3DAndroid&amp;hl=en">https://support.google.com/maps/answer/144361?co=GENIE.Platform%3DAndroid&amp;hl=en</a></p> <p><b>Share your E.T.A</b></p> <p>After you start your drive, you can share your destination, estimated arrival time, and where you are on the route.</p> <ol style="list-style-type: none"> <li>1. Open the Google Maps app .</li> <li>2. Set a driving destination. <a href="#">Learn how to navigate to a place.</a></li> <li>3. After you start navigation, tap <b>More</b>  &gt; <b>Share trip progress</b>.</li> <li>4. Choose a person from the list.</li> <li>5. Tap <b>Share</b>.</li> <li>6. Location Sharing will stop when you reach your destination or stop navigating.</li> </ol> <ul style="list-style-type: none"> <li>• To stop sharing before you arrive, tap <b>More</b>  &gt; <b>Stop sharing</b>.</li> </ul>
<p>2[A]. The method of claim 1, further comprising</p>	<p>HTC infringes directly and/or indirectly by performing, inducing others to perform, and/or contributing to the performance of performing, by the first device: presenting another symbol on the interactive map corresponding to a fixed location and associated with a telephone number. See claim 1, which is</p>



## Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products

US9408055B2	HTC
<p>performing, by the first device: presenting another symbol on the interactive map corresponding to a fixed location and associated with a telephone number;</p>	<p>incorporated herein by reference in its entirety.</p> <h3 data-bbox="520 342 1730 444">Share Locations from Google Maps on HTC One M8</h3> <p data-bbox="527 449 831 480">Place a pin on the map.</p> <p data-bbox="527 492 1640 602">We can do this anywhere in the entire world, anywhere, on any map that we find in the Google Maps application. We are going to choose a location, press down on the screen, and hold down until we can see that a red pin has been dropped.</p>  <p data-bbox="510 1360 1430 1427">See, e.g., Placing a Marker: <a href="https://developers.google.com/maps/documentation/android-api/marker">https://developers.google.com/maps/documentation/android-api/marker</a></p>

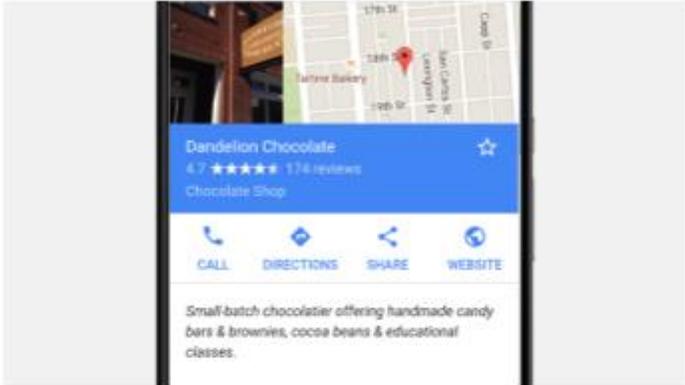
## Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products

US9408055B2	HTC
	<p>based on queries with GeoTagging database: <a href="https://developers.google.com/android/reference/com/google/android/gms/location/places/GeoDataClient">https://developers.google.com/android/reference/com/google/android/gms/location/places/GeoDataClient</a></p> <p>Sharing a link: <a href="https://support.google.com/maps/answer/144361?co=GENIE.Platform%3DDesktop&amp;hl=en">https://support.google.com/maps/answer/144361?co=GENIE.Platform%3DDesktop&amp;hl=en</a></p> <p>Users can use android maps to receive information associated with fixed locations such as restaurants, stadiums, transportation stations (e.g., bus and train stations), etc.</p>  <p>The image is a screenshot of an Android mobile application interface for Google Maps. It shows a map with a red location pin. Below the map, there is a blue information card for 'Dandelion Chocolate', a 'Chocolate Shop'. The card displays a 4.7-star rating with 174 reviews. Below the card are four icons: a phone for 'CALL', a location pin for 'DIRECTIONS', a share icon for 'SHARE', and a globe for 'WEBSITE'. At the bottom of the card, there is a short description: 'Small batch chocolatier offering handmade candy bars &amp; brownies, cocoa beans &amp; educational classes.'</p> <p>Place details</p> <p>Retrieve rich details about a place, including name, address, phone number, website link and more.</p> <p><a href="https://developers.google.com/places/android-api/">https://developers.google.com/places/android-api/</a></p>

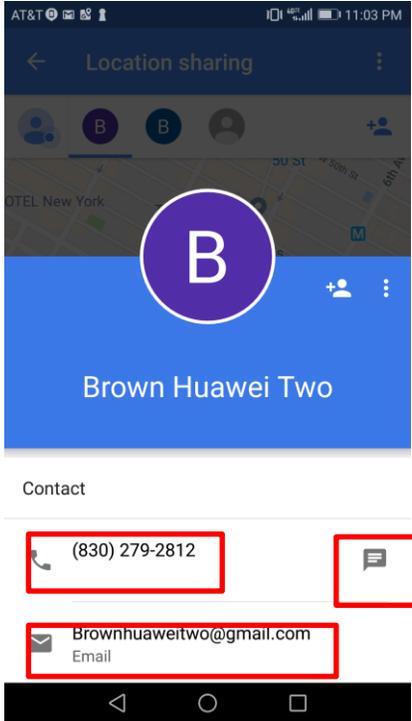
## Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products

US9408055B2	HTC
	<p><b>Business phone numbers</b></p> <p>Google is fantastic for tracking down business phone numbers.</p> <p>You can accomplish this in a number of ways, including:</p> <ul style="list-style-type: none"> <li>▪ <b>type of business plus zip code:</b> Perhaps you don't know the name of the business you're looking for, but you have something in mind. Type in the business genre, for example, "pizza restaurant", then the zip code. Google will return local listings that include maps, reviews, and contact information (phone numbers, addresses, <a href="#">website URLs</a>, even email addresses if available).</li> <li>▪ <b>type of business plus city:</b> Just like in the previous example, except you can substitute the name of a city for a zip code, i.e., "Seattle doctors".</li> </ul> <p><a href="https://www.lifewire.com/google-phone-number-search-3481892">https://www.lifewire.com/google-phone-number-search-3481892</a></p> <h3>Embed a map or share a location</h3> <p>On your computer, you can embed a map, Street View image, driving directions, or search into your website or blog. On your computer, phone, or tablet, you can share a location or map with others over email, Google+, Facebook, Twitter, or text.</p> <p><a href="#">ANDROID</a>   <a href="#">COMPUTER</a>   <a href="#">IPHONE &amp; IPAD</a></p> <hr/> <h3>Share a map or location</h3> <ol style="list-style-type: none"> <li>1. Open the Google Maps app .</li> <li>2. Search for a place. Or find a place on the map then touch and hold to drop a pin.</li> <li>3. At the bottom, tap the place's name or address.</li> <li>4. Tap Share . If you don't see this, tap More  &gt; <b>Share</b>.</li> <li>5. Select an app. It'll send a link that shows the place in Google Maps.</li> </ol> <p><a href="https://support.google.com/maps/answer/144361?co=GENIE.Platform%3DAndroid&amp;hl=en">https://support.google.com/maps/answer/144361?co=GENIE.Platform%3DAndroid&amp;hl=en</a></p>

**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC
<p>[2B] and receiving user selection of the other symbol and, based thereon, initiating a telephone call to the telephone number associated with the symbol.</p>	<p>HTC infringes directly and/or indirectly by performing, inducing others to perform, and/or contributing to the performance of [] receiving user selection of the other symbol and, based thereon, initiating a telephone call to the telephone number associated with the symbol. See claims 1 and 2[A], which are incorporated herein by reference in their entireties.</p> <p>A user can search for a fixed location using the maps application, which presents the user with a symbol(s) on the map corresponding to the fixed location’s geographical coordinates. The user can then select the symbol(s) to see the fixed location’s associated telephone number and select a “soft switch” having a symbol of a telephone handset to call the telephone number associated with the symbol</p>  <p>The image is a screenshot of a mobile application interface, likely Google Maps, showing details for a location named 'Dandelion Chocolate'. At the top, there is a blue header with the name 'Dandelion Chocolate', a star icon, and a rating of 4.7 stars with 174 reviews. Below the header, there are four icons: a telephone handset, a location pin, a share icon, and a globe icon, with the labels 'CALL', 'DIRECTIONS', 'SHARE', and 'WEBSITE' respectively. Below the icons, there is a short description: 'Small batch chocolatier offering handmade candy bars &amp; brownies, cocoa beans &amp; educational classes.'</p> <p>Place details</p> <p>Retrieve rich details about a place, including name, address, phone number, website link and more.</p> <p><a href="https://developers.google.com/places/android-api/">https://developers.google.com/places/android-api/</a></p>

**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC
	<p>For example, the Accused Products include software that registers touch events with an interactive display, where touching a “call” button starts a phone call. Furthermore, these phone calls can merge multiple parties into a conference call.</p>  <p>See also:  <a href="https://www.wikihow.com/Conference-Call-on-an-Android">https://www.wikihow.com/Conference-Call-on-an-Android</a></p>
<p>3. The method of claim 1 wherein the data comprises a text</p>	<p>HTC infringes directly and/or indirectly by performing, inducing others to perform, and/or contributing to the performance of wherein the data comprises a text message, an image, a video, or a command to cause the second devices corresponding to the selected symbols to convert text to speech.</p>

**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC
<p>message, an image, a video, or a command to cause the second devices corresponding to the selected symbols to convert text to speech.</p>	<p>The Accused Products include an a TextToSpeech class that allows Android devices to convert any text displayed on the device to speech which is then recited using an electronic voice. The user of a first device sends a text message, image, video, or command to a second device using the iMessage app. The second device receives the communication and converts all text to speech when, e.g., extToSpeech is enabled.</p> <p>Android allows you convert your text into voice. Not only you can convert it but it also allows you to speak text in variety of different languages.</p> <p>Android provides <b>TextToSpeech</b> class for this purpose. In order to use this class, you need to instantiate an object of this class and also specify the <b>initListener</b>. Its syntax is given below –</p> <pre data-bbox="520 724 1360 932">private EditText write; ttobj=new TextToSpeech(getApplicationContext(), new TextToSpeech.OnIn @Override public void onInit(int status) { } });</pre> <p>In this listener, you have to specify the properties for TextToSpeech object , such as its language ,pitch e.t.c. Language can be set by calling <b>setLanguage()</b> method. Its syntax is given below –</p> <pre data-bbox="520 1127 1360 1182">ttobj.setLanguage(Locale.UK);</pre> <p><a href="https://www.tutorialspoint.com/android/android_text_to_speech.htm">https://www.tutorialspoint.com/android/android_text_to_speech.htm</a></p>
<p>4[A]. The method of claim 1 wherein: the</p>	<p>HTC infringes 4 directly and/or indirectly by performing, inducing others to perform, and/or contributing to the performance of wherein the SMS messages include an Internet Protocol (IP) address of the first device.</p>

**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

<b>US9408055B2</b>	<b>HTC</b>
SMS messages include an Internet Protocol (IP) address of the first device;	See claim 1, which is incorporated herein by reference in its entirety.  Android devices can send SMS messages that include an IP address of a device using Google Voice text message services. Google Voice is a voice over IP technology over internet protocol (IP) networks which use IP addresses to route data. Accordingly, the SMS messages include an IP address.

## Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products

US9408055B2	HTC
	<h3 data-bbox="512 240 1192 300">Send &amp; get text messages</h3> <p data-bbox="512 321 1688 418">You can send text messages for free to U.S. and Canadian numbers using your Google Voice number. Texts sent using Google Voice will use Wi-Fi, or mobile data from your cell phone service plan if you're not connected to Wi-Fi. If you're outside the U.S. and are not using Wi-Fi, your cell phone company might charge you extra roaming fees to send a text.</p> <p data-bbox="512 444 1696 506"><b>Google Voice &amp; Hangouts:</b> You can choose to <a href="#">link Google Voice and Hangouts</a>. If you do, you'll have to use Hangouts to send texts, not the Google Voice website or apps. To stop using Hangouts for texts, <a href="#">turn off Google Voice in Hangouts</a>.</p> <p data-bbox="529 592 999 618"><a href="#">ANDROID</a>   <a href="#">COMPUTER</a>   <a href="#">IPHONE &amp; IPAD</a></p> <hr data-bbox="512 646 1726 652"/> <p data-bbox="512 711 1272 737">If you haven't yet, <a href="#">download the Google Voice app on your Android device</a>. <a href="#">↗</a></p> <h3 data-bbox="512 803 898 847">Send a text message</h3> <p data-bbox="512 868 1701 894">With the Google Voice website and apps, you can text people messages and photos and send texts to groups of people.</p> <p data-bbox="512 922 1633 948">If you send a text longer than 160 characters to a non-Google Voice number, it will be sent as multiple messages.</p> <p data-bbox="512 976 1180 1002"><b>Note:</b> You can't send texts to five- or six-digit "short code" numbers.</p> <ol data-bbox="512 1034 1289 1279" style="list-style-type: none"><li data-bbox="512 1034 1094 1060">1. On your Android device, open the Google Voice app .</li><li data-bbox="512 1076 863 1102">2. Open the tab for Messages .</li><li data-bbox="512 1118 814 1144">3. At the bottom, tap Add .</li><li data-bbox="512 1161 1289 1227">4. Enter a contact's name or phone number.<ul data-bbox="548 1198 1289 1224" style="list-style-type: none"><li data-bbox="548 1198 1289 1224">• To create a group text message, add up to 30 names or phone numbers.</li></ul></li><li data-bbox="512 1252 930 1278">5. Enter your message, and tap Send .</li></ol> <p data-bbox="512 1308 1696 1367">To include an image with your message, tap Select image . If your image is bigger than 2MB, it'll be sent as a smaller file. But GIFs over 2MB won't send.</p> <p data-bbox="512 1370 1682 1396"><a href="https://support.google.com/voice/answer/115116?co=GENIE.Platform%3DAndroid&amp;hl=en">https://support.google.com/voice/answer/115116?co=GENIE.Platform%3DAndroid&amp;hl=en</a></p>



**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC
<p>[4B] and the IP-based responses include respective IP addresses of the second devices.</p>	<p>HTC infringes directly and/or indirectly by performing, inducing others to perform, and/or contributing to the performance of [] the IP-based responses include respective IP addresses of the second devices. See claim 4[A], which is incorporated herein by reference in its entirety.</p> <p>Android devices can reply to SMS texts via, e.g., Google Voice that also include IP address to route the message to its destination device via an IP network.</p> <p><b>Reply to a text message</b></p> <ol style="list-style-type: none"> <li>1. On your Android device, open the Google Voice app .</li> <li>2. Open the tab for Messages .</li> <li>3. Tap the text message you want to reply to.</li> <li>4. At the bottom, enter your message, and tap Send .</li> </ol> <p><a href="https://support.google.com/voice/answer/115116?co=GENIE.Platform%3DAndroid&amp;hl=en">https://support.google.com/voice/answer/115116?co=GENIE.Platform%3DAndroid&amp;hl=en</a></p>
<p>5. The method of claim 1, further comprising performing by the first device: transmitting location information including an updated location of the first device to the second devices based on displacement of the first device by at least a predetermined distance relative to a</p>	<p>HTC infringes directly and/or indirectly by performing, inducing others to perform, and/or contributing to the performance of performing by the first device: transmitting location information including an updated location of the first device to the second devices based on displacement of the first device by at least a predetermined distance relative to a previous location of the first device, passage of at least a predetermined time interval since transmitting information including a location of the first device, or a combination of the displacement of the first device and the passage of time. See claim 1, which is incorporated herein by reference in its entirety.</p> <p>For example, Android devices include a LocationListener that allows the transmission and receipt of notification when a location of a device has changed.</p>

**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC
<p>previous location of the first device, passage of at least a predetermined time interval since transmitting information including a location of the first device, or a combination of the displacement of the first device and the passage of time.</p>	<div data-bbox="514 233 945 289"> <h2>LocationListener</h2> </div> <div data-bbox="514 300 840 373"> <p>public interface LocationListener android.location.LocationListener</p> </div> <hr/> <div data-bbox="514 454 1743 511"> <p>Used for receiving notifications from the LocationManager when the location has changed. These methods are called if the LocationListener has been registered with the location manager service using the <code>requestLocationUpdates(String, long, float, LocationListener)</code> method.</p> </div> <div data-bbox="514 544 766 576"> <h3>Developer Guides</h3> </div> <div data-bbox="514 600 1365 625"> <p>For more information about identifying user location, read the <a href="#">Obtaining User Location</a> developer guide.</p> </div> <div data-bbox="514 673 1533 714"> <p><a href="https://developer.android.com/reference/android/location/LocationListener.html">https://developer.android.com/reference/android/location/LocationListener.html</a></p> </div> <div data-bbox="535 771 714 941">  </div> <div data-bbox="535 1055 787 1096"> <h3>LocationManager</h3> </div> <div data-bbox="535 1128 871 1226"> <p>The location Manager manages the location engines and GPS satellites along with other providers that provide our applications with access to location.</p> </div> <div data-bbox="535 1250 871 1299"> <p>We can then trigger different services in our device, once we get a new location.</p> </div> <div data-bbox="924 966 1102 1307">  </div> <div data-bbox="1102 795 1648 950"> <p>Listeners are used to listen to the location updates.</p> <ol style="list-style-type: none"> <li>1. LocationListener</li> <li>2. PendingIntent</li> </ol> <p>These can be used and triggered to perform an action or start another activity or service.</p> </div> <div data-bbox="1113 1039 1396 1201">  </div> <div data-bbox="1438 990 1638 1250"> <div data-bbox="1438 990 1638 1250" style="border: 1px solid black; padding: 5px;"> <p><b>SMS sending service</b></p> <p>Shares the location details with clients.</p> </div> </div> <div data-bbox="514 1323 1743 1388"> <p><b>Figure 1:</b> Demonstration of our requirement and work around. Explains what happens and what objects are being used in this demonstration.</p> </div>

**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC
<p>6[A]. The method of claim 1, further comprising performing, by the first device: receiving second user selection of one or more of the symbols corresponding to one or more of the second devices;</p>	<p><a href="https://www.codeproject.com/Articles/1040389/Ok-Android-Broadcast-my-location">https://www.codeproject.com/Articles/1040389/Ok-Android-Broadcast-my-location</a></p> <p>HTC infringes directly and/or indirectly by performing, inducing others to perform, and/or contributing to the performance of performing, by the first device: receiving second user selection of one or more of the symbols corresponding to one or more of the second devices. See claim 1, which is incorporated herein by reference in its entirety.</p> <p>For example, a first android device can receive second user selection of one or more of the symbols of second devices to join a communication network. For example, an Accused Product joins a communication network by signing-in using a Google Account. Signing-in to a Google Account with credentials (e.g., Gmail address or a registered username) which identify the user as being associated with the Google network of services. A user enters her credentials and transmits the credentials in the form of a request, e.g. an HTTP request. Moreover, each Google Account is associated with groups within certain Google services, e.g. Hangouts, Google+, Google Groups. Thus the Google network itself constitutes a group and additional examples of groups include Groups, Circles, Communities, as well as associations and shares created in Google services such as Android Messages, Contacts, Calendar, Gmail, Drive, Docs, Sheets, Slides, Keep, Photos, Contacts, Allo, Youtube, Waze, Hangouts, Google+, and other Google services or applications.</p> <p>Using Google Maps on an Accused Product, a user joins the Google network (and any associated groups within the Google network) by signing-on with a Google Account, as shown below:</p>

## Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products

US9408055B2	HTC
	<p data-bbox="527 233 1134 277"><b>If they have a Google Account</b></p> <ol data-bbox="527 302 1627 667" style="list-style-type: none"><li>1. If you haven't already, add their Gmail address to your <a href="#">Google Contacts</a>.</li><li>2. On your Android phone or tablet, open the Google Maps app  and sign in. <a href="#">Learn how to sign in.</a></li><li>3. Tap the Menu  &gt; <b>Share location</b> &gt; Add People .</li><li>4. Choose how long you want to share your location.</li><li>5. Tap <b>Select People</b>.<ul data-bbox="569 529 1276 557" style="list-style-type: none"><li>• If you're asked about your contacts, give Google Maps access.</li></ul></li><li>6. Choose who you want to share with.</li><li>7. Tap <b>Share</b>.</li></ol> <p data-bbox="512 683 1654 711"><a href="https://support.google.com/plus/answer/3302509?hl=en&amp;co=GENIE.Platform%3DAndroid&amp;oco=1">https://support.google.com/plus/answer/3302509?hl=en&amp;co=GENIE.Platform%3DAndroid&amp;oco=1</a></p> <p data-bbox="512 764 1850 797">Instead of selecting multiple contacts, a user can create a group and share to that group, as shown below:</p> <p data-bbox="520 841 852 889"><b>Group contacts</b></p> <p data-bbox="520 906 1094 927">You can organize the people and businesses in Contacts using labels.</p> <p data-bbox="520 971 709 1003"><b>Create a group</b></p> <ol data-bbox="527 1019 1430 1114" style="list-style-type: none"><li>1. Go to <a href="#">Google Contacts</a>.</li><li>2. On the left under "Labels," click <b>Create label</b>. (If you don't see "Labels," go to <a href="#">group contacts in old Contacts</a>.)</li><li>3. Type a name, then click <b>OK</b>.</li></ol> <hr data-bbox="520 1133 1514 1138"/> <p data-bbox="541 1162 821 1183"><a href="#">Add contacts to a group label</a> </p> <ol data-bbox="575 1214 1461 1333" style="list-style-type: none"><li>1. To select contacts, check the boxes next to their names.</li><li>2. In the top right, click Label . (If you don't see Label , go to <a href="#">group contacts in old Contacts</a>.)</li><li>3. Choose the groups you want to add the contacts to. You'll see a checkmark appear next to the groups you chose.</li></ol> <p data-bbox="512 1344 1199 1372"><a href="https://support.google.com/mail/answer/30970?hl=en">https://support.google.com/mail/answer/30970?hl=en</a></p>

## Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products

US9408055B2	HTC
	<h3 data-bbox="541 233 1570 289">Use Gmail to access your Google Account</h3> <p data-bbox="541 310 1682 367">If you use Gmail, you already have a Google Account. With a Google Account, you have access to free Google products like Drive, Docs, Calendar, and <a href="#">more</a>.</p> <p data-bbox="541 394 1094 418">To sign in to your Google Account (or any Google product):</p> <ol data-bbox="554 446 1430 548" style="list-style-type: none"><li>1. Go to the sign in page of the product (for Google Accounts it is <a href="#">myaccount.google.com</a>).</li><li>2. Enter your Gmail username (everything that appears before '@gmail.com').</li><li>3. Enter your password.</li></ol> <p data-bbox="522 566 1455 594"><a href="https://support.google.com/accounts/answer/76194?hl=en&amp;ref_topic=7189242">https://support.google.com/accounts/answer/76194?hl=en&amp;ref_topic=7189242</a></p> <h3 data-bbox="531 638 999 680">Sign in or out of Google+</h3> <p data-bbox="531 695 1402 740">To sign in, enter your Google username and password. If you're on a public or shared device, don't forget to sign out so no one else can get in to your Google+ account.</p> <p data-bbox="531 760 1289 781"><b>Tip:</b> If you don't know your username or password, you can <a href="#">get help signing in to your Google account</a>.</p> <hr/> <p data-bbox="548 821 957 842">Sign in to your Google+ account for the first time</p> <p data-bbox="573 867 667 888"><b>Android app</b></p> <ol data-bbox="579 906 1052 959" style="list-style-type: none"><li>1. On your Android phone or tablet, open the the Google+ app.</li><li>2. Tap the account you want to use.</li></ol> <p data-bbox="573 979 653 1000"><b>Computer</b></p> <ol data-bbox="579 1018 982 1070" style="list-style-type: none"><li>1. On your computer, open <a href="#">Google+</a>.</li><li>2. Enter your username and password and click <b>Sign in</b>.</li></ol> <p data-bbox="522 1084 1430 1112"><a href="https://support.google.com/plus/answer/1301225?hl=en&amp;ref_topic=3049735">https://support.google.com/plus/answer/1301225?hl=en&amp;ref_topic=3049735</a></p> <p data-bbox="510 1162 1871 1230">The sign-in process includes a message because a request message (e.g. HTTP[S] request) is sent from the Accused Product, as shown below:</p>

**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC
	<p data-bbox="541 240 961 267"><b>Connect to the Online Service</b></p> <p data-bbox="541 298 1602 375">The example below shows how to connect to a Google server. Since Google uses the industry standard OAuth2 protocol to authenticate requests, the techniques discussed here are broadly applicable. Keep in mind, though, that every server is different. You may find yourself needing to make minor adjustments to these instructions to account for your specific situation.</p> <p data-bbox="541 396 1602 472">The Google APIs require you to supply four values with each request: the API key, the client ID, the client secret, and the auth key. The first three come from the Google API Console website. The last is the string value you obtained by calling <code>AccountManager.getAuthToken()</code>. You pass these to the Google Server as part of an HTTP request.</p> <pre data-bbox="554 488 1619 623"> URL url = new URL("https://www.googleapis.com/tasks/v1/users/@me/lists?key=" + your_api_key); URLConnection conn = (URLConnection) url.openConnection(); conn.addRequestProperty("client_id", your_client_id); conn.addRequestProperty("client_secret", your_client_secret); conn.setRequestProperty("Authorization", "OAuth " + token); </pre> <p data-bbox="541 644 1623 691">If the request returns an HTTP error code of 401, then your token has been denied. As mentioned in the last section, the most common reason for this is that the token has expired. The fix is simple: call <code>AccountManager.invalidateAuthToken()</code> and repeat the token acquisition dance one more time.</p> <p data-bbox="541 712 1608 789">Because expired tokens are such a common occurrence, and fixing them is so easy, many applications just assume the token has expired before even asking for it. If renewing a token is a cheap operation for your server, you might prefer to call <code>AccountManager.invalidateAuthToken()</code> before the first call to <code>AccountManager.getAuthToken()</code>, and spare yourself the need to request an auth token twice.</p> <p data-bbox="520 810 1304 837"><a href="https://developer.android.com/training/id-auth/authenticate.html">https://developer.android.com/training/id-auth/authenticate.html</a></p>

**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

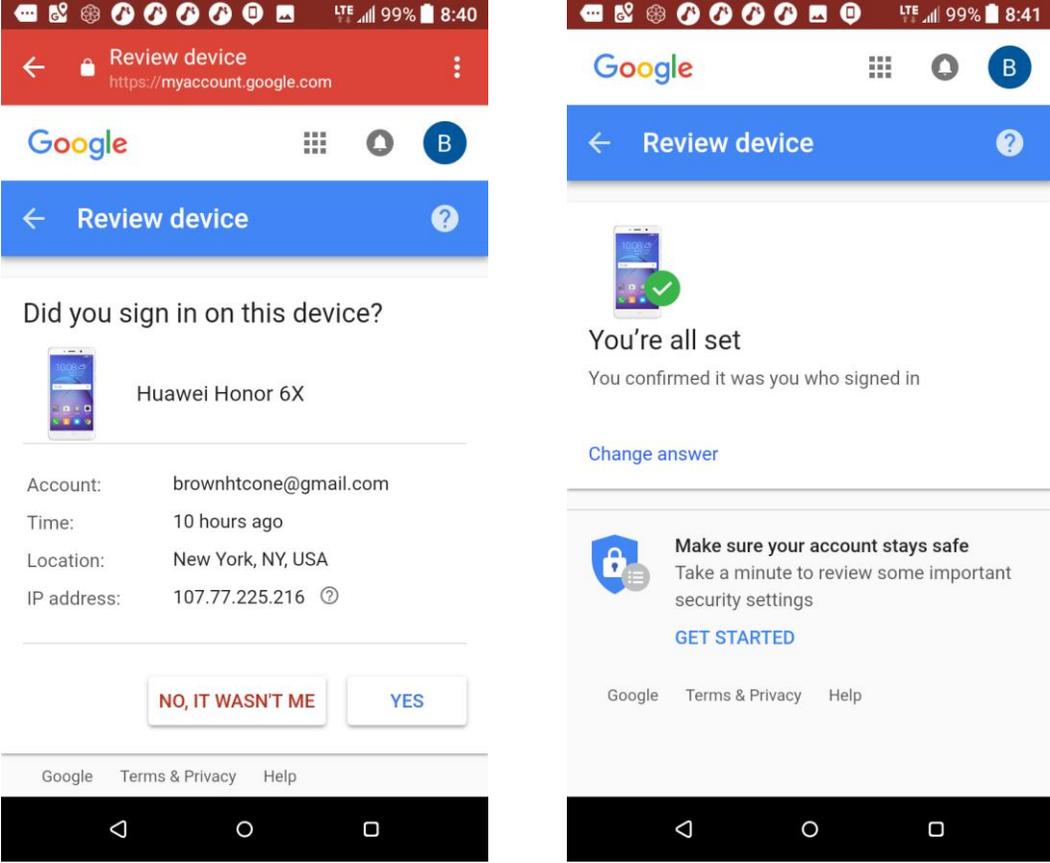
US9408055B2	HTC
	<p>2. In the activity's <code>onActivityResult</code> method, retrieve the sign-in result with <code>getSignInResultFromIntent</code>.</p> <pre data-bbox="569 289 1591 581"> @Override public void onActivityResult(int requestCode, int resultCode, Intent data) {     super.onActivityResult(requestCode, resultCode, data);      // Result returned from launching the Intent from GoogleSignInApi.getSignInIntent(...);     if (requestCode == RC_SIGN_IN) {         GoogleSignInResult result = Auth.GoogleSignInApi.getSignInResultFromIntent(data);         handleSignInResult(result);     } } </pre> <p style="text-align: right;"><a href="#">SignInActivity.java</a> </p> <p>After you retrieve the sign-in result, you can check if sign-in succeeded with the <code>isSuccess</code> method. If sign-in succeeded, you can call the <code>getSignInAccount</code> method to get a <code>GoogleSignInAccount</code> object that contains information about the signed-in user, such as the user's name.</p> <pre data-bbox="569 716 1591 1052"> private void handleSignInResult(GoogleSignInResult result) {     Log.d(TAG, "handleSignInResult:" + result.isSuccess());     if (result.isSuccess()) {         // Signed in successfully, show authenticated UI.         GoogleSignInAccount acct = result.getSignInAccount();         mStatusTextView.setText(getString(R.string.signed_in_fmt, acct.getDisplayName()));         updateUI(true);     } else {         // Signed out, show unauthenticated UI.         updateUI(false);     } } </pre> <p style="text-align: right;"><a href="#">SignInActivity.java</a> </p> <p>You can also get the user's email address with <code>getEmail</code>, the user's Google ID (for client-side use) with <code>getId</code>, and an ID token for the user with <code>getIdToken</code>. If you need to pass the currently signed-in user to a backend server, send the ID token to your backend server and validate the token on the server.</p> <p><a href="https://developers.google.com/identity/sign-in/android/sign-in">https://developers.google.com/identity/sign-in/android/sign-in</a></p>
<p>[6B] and receiving user input assigning the one or more</p>	<p>HTC infringes directly and/or indirectly by performing, inducing others to perform, and/or contributing to the performance of [] receiving user input assigning the one or more second devices corresponding to the second selected one or more symbols to a sub-net. See claims 1 and 6[A], which are incorporated herein by</p>

**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC
<p>second devices corresponding to the second selected one or more symbols to a sub-net.</p>	<p>reference in their entireties.</p> <p>For example, an Accused Product is joined to a group (e.g. Google services such as the Play and Play Protect services) using a Google ID (e.g, an email address). Alternatively, any device with a browser or a device-location application can sign on to a Google service. The Google ID corresponds to one or more groups. These groups include, for example, the Google network, “friends” or “families” as defined by association with device-location features and applications (e.g., Find My Device, Hangouts (including Allo and Duo), Google+, Maps, and Chrome.)</p> <p>In establishing these groups, each Accused Product receives messages from other devices, e.g. other Accused Products, related to forming a group.</p> <p>In other examples, a user participates in groups within services of the Google network, e.g. Android Device Manager, Google Maps, Google Contacts, Google+, Google Groups, among others. Using Android Device Manager, a group of devices is associated with the sign-in credentials. Each member of the group is represented by a symbol. Using Google Maps, as integrated with Google Contacts and Messages, a user shares her location with a group by sending her location to a group or multiple contacts in a group message. Groups are also established in many other services across Google’s network, as shown by Google+ and Google Groups. In fact, Google+ included the location-sharing feature prior to its inclusion in Google Maps.</p>



**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC
	 <p>e.g., above, the device receives a message indicating that the second device (HTC Honor 6X) has joined the group, i.e. the google account.</p> <p><b><u>Google Share Location</u></b></p> <p>Google Latitude, Google Plus, Google Hangouts, Google Messages, and Android Messenger, Google Duo, Google Allo, and Google Chrome, the Accused Products require a user to join the corresponding network by signing-in to the device with an identifier (e.g., Google Account). The sign-in process takes place within the</p>

**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC
	<p>Google Maps software on the Accused Product or by navigating to maps.google.com within the Google Chrome browser on the Accused Product. Alternatively, the sign-in process may partially or completely take place using credentials already provided when the user associates a Google Account with the Accused Product, e.g., during initial setup of the Accused Product. Subject to discovery, one or more additional or substitute identifiers may correspond to the group. The sign-in process involves a user entering its Google Account and additional authentication data on the interface of the Accused Product and sending a message containing the Google Account and additional authentication data over a network to members of a group. The group comprises the multiple identifiers, individuals, profiles, and/or devices associated with the group. Further regarding Google Maps, Google Latitude, Google Plus, Google Hangouts, Google Messages, and Android Messenger, Google Duo, Google Allo, and Google Chrome, the Accused Products alternatively require a user to send a request containing the identifier (e.g., Google Account). Subject to discovery, additional identifiers may be assigned or used to correspond to the group. The request may be an invitation or message that associates a Google Account with one or more Google Accounts for the purposes of sharing locations within the group. The group comprises the multiple identifiers, individuals, profiles, and/or devices associated with the group</p>

## Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products

US9408055B2	HTC
	<h3 data-bbox="562 235 991 277">Use circles on Google+</h3> <p data-bbox="562 293 1423 337">You can use circles to control who you share posts with. Add people to a circle, then share a Collection or a post with that circle so those people can see it.</p> <p data-bbox="562 358 1423 402">You can add someone to a circle even if they don't follow you, so they'll be able to see posts you share with that circle, but might not see those posts in their home stream.</p> <p data-bbox="562 423 1423 492">When you add someone to a circle, they might get a notification letting them know you added them to one of your circles. They can see anything you share with that circle, including posts you shared with that circle before you added them. If that person adds you to their circles, posts you've shared with them may appear on their Home page.</p> <p data-bbox="562 513 1192 532">You can see people you have in your circles by choosing <b>People</b> in your Navigation menu.</p> <p data-bbox="562 553 1423 621">The people and pages you add to circles are publicly visible by default, but you can change who can see the people and pages you add. Depending on your settings, information about the people or pages you've added to circles may also appear in shared endorsements.</p> <p data-bbox="562 643 1119 662">You can add someone to a circle or remove someone from a circle at any time.</p> <h3 data-bbox="562 708 751 734">Create a circle</h3> <ol data-bbox="562 755 898 862" style="list-style-type: none"><li>1. In the Navigation menu, click or tap <b>People</b>.</li><li>2. Click or tap <b>Following</b>.</li><li>3. Click or tap <b>NEW CIRCLE</b>.</li><li>4. Name your circle, then click or tap <b>CREATE</b>.</li></ol> <p data-bbox="520 878 1423 904"><a href="https://support.google.com/plus/answer/1301225?hl=en&amp;ref_topic=3049735">https://support.google.com/plus/answer/1301225?hl=en&amp;ref_topic=3049735</a></p> <h3 data-bbox="546 956 1192 1008">Find and join Communities</h3> <p data-bbox="546 1032 1675 1084">You can connect with people who share your interests by joining a public or private Community in Google+. Stay up to date on what's happening in your Communities by seeing Community posts in your stream and getting notifications.</p> <p data-bbox="562 1170 1003 1190">COMPUTER   <b>ANDROID</b>   IPHONE &amp; IPAD</p> <p data-bbox="520 1216 1115 1242"><a href="https://support.google.com/plus/answer/6320394">https://support.google.com/plus/answer/6320394</a></p>

## Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products

US9408055B2	HTC
	<p data-bbox="541 235 911 277"><b>Find &amp; join a group</b></p> <p data-bbox="541 297 1434 342">You can join a Google Group to have discussions about a topic or communicate with your team, organization, class, or other group. Some groups grant immediate membership, but others need you to request membership first.</p> <p data-bbox="541 394 716 427"><b>Join a group</b></p> <p data-bbox="541 443 1188 464">1. Sign in to Google Groups <a href="#">🔗</a> . <a href="#">Learn how to join if you don't have a Google Account.</a></p> <p data-bbox="541 516 1115 558"><b>Use groups to share content</b></p> <p data-bbox="541 578 1467 623">Google Groups makes it easy to share your Google documents, sites, videos, and calendars with multiple people. As you add new members to your groups, they'll automatically gain access to content you previously shared with that group.</p> <p data-bbox="541 651 1478 777">For example, if you create a group with the address "marketing-team@your_domain.com" and add five members, you can instantly share a Google document with them, just by sharing the document with the group's address. If you later add another member, that member automatically inherits permission to access the document or any other content you shared with the group. Similarly, if you remove a member from a group, that person no longer has access to any content you shared with the group.</p> <p data-bbox="541 794 1146 821"><b><a href="https://support.google.com/a/answer/167101?hl=en">https://support.google.com/a/answer/167101?hl=en</a></b></p> <p data-bbox="512 862 1881 967">In addition to the communication described above with respect to the Android Device Manager and Google Maps applications, messages can be received over messaging applications integrated in the Accused Products, such as Google Messages or Google Hangouts.</p> <p data-bbox="512 1045 1031 1073"><b><u>Exemplary Support for Google Maps:</u></b></p>

## Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products

US9408055B2	HTC
	<p>COMPUTER <b>ANDROID</b> IPHONE &amp; IPAD</p> <hr/> <h3>If they have a Google Account</h3> <ol style="list-style-type: none"><li>1. If you haven't already, add their Gmail address to your <a href="#">Google Contacts</a>.</li><li>2. On your Android phone or tablet, open the Google Maps app  and sign in. <a href="#">Learn how to sign in.</a></li><li>3. Tap Menu ≡ &gt; <b>Location sharing</b> &gt; Add People .</li><li>4. Choose how long you want to share your location.</li><li>5. Tap <b>Select People</b>.<ul style="list-style-type: none"><li>• If you're asked about your contacts, give Google Maps access.</li></ul></li><li>6. Choose who you want to share with.</li><li>7. Tap <b>Share</b>.</li></ol> <h3>If they don't have a Google Account</h3> <ol style="list-style-type: none"><li>1. On your Android phone or tablet, open the Google Maps app  and sign in. <a href="#">Learn how to sign in.</a></li><li>2. Tap Menu ≡ &gt; <b>Location sharing</b> &gt; Add People .</li><li>3. Tap More ... &gt; <b>Copy to clipboard</b>. People with this link can see your location for as long as you choose, up to 72 hours.</li></ol> <h3>Share using another app</h3> <p>You can also share through messaging apps. Tap More ... &gt; select an app.</p> <h3>Stop sharing</h3> <ol style="list-style-type: none"><li>1. Open the Google Maps app .</li><li>2. Tap Menu ≡ &gt; <b>Location sharing</b>.</li><li>3. Next to the person with whom you want to stop sharing, tap Remove ✕.</li></ol> <p><a href="https://support.google.com/maps/answer/7326816?co=GENIE.Platform%3DAndroid&amp;oco=1">https://support.google.com/maps/answer/7326816?co=GENIE.Platform%3DAndroid&amp;oco=1</a></p>

## Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products

US9408055B2	HTC
	<h3>Share your E.T.A</h3> <p>After you start your drive, you can share your destination, estimated arrival time, and where you are on the route.</p> <ol style="list-style-type: none"><li>1. Open the Google Maps app .</li><li>2. Set a driving destination. <a href="#">Learn how to navigate to a place.</a></li><li>3. After you start navigation, tap More ^ &gt; <b>Share trip progress.</b></li><li>4. Choose a person from the list.</li><li>5. Tap <b>Share.</b></li><li>6. Location Sharing will stop when you reach your destination or stop navigating.</li></ol> <ul style="list-style-type: none"><li>• To stop sharing before you arrive, tap More ^ &gt; <b>Stop sharing.</b></li></ul> <h3>See where someone is</h3> <p>If someone shares their location with you, you can see them on the map.</p> <ol style="list-style-type: none"><li>1. Open the Google Maps app .</li><li>2. Tap Menu ≡ &gt; <b>Location sharing.</b></li><li>3. Choose someone.</li></ol> <ul style="list-style-type: none"><li>• To see an updated location, tap on a friend's icon &gt; More ⋮ &gt; <b>Refresh.</b></li></ul> <h3>Stop seeing someone's location</h3> <ol style="list-style-type: none"><li>1. Open the Google Maps app .</li><li>2. On the map, tap their icon.</li><li>3. At the bottom, tap More ^ .</li><li>4. To temporarily hide someone, tap <b>Hide from map.</b> You can stop hiding them at any time.</li></ol> <p><b>Note:</b> You can stop someone's location from ever appearing on your map. <a href="#">Learn how to block another person's account.</a></p> <p><a href="https://support.google.com/maps/answer/7326816?co=GENIE.Platform%3DAndroid&amp;oco=1">https://support.google.com/maps/answer/7326816?co=GENIE.Platform%3DAndroid&amp;oco=1</a></p>

## Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products

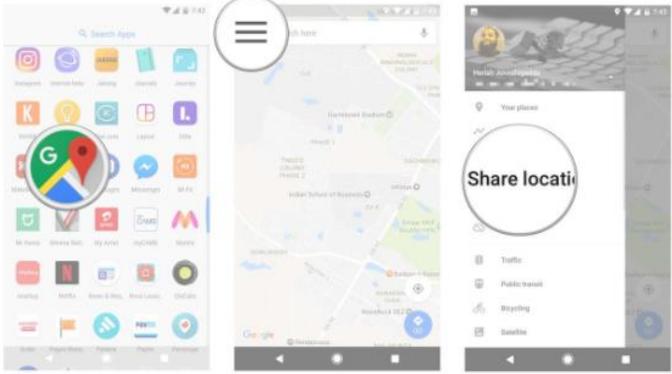
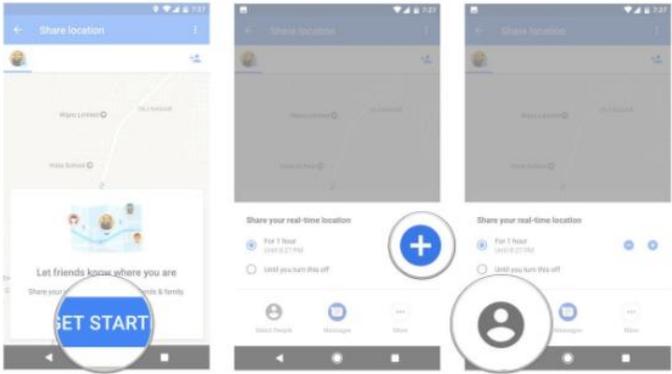
US9408055B2	HTC
	<h3>Create a list of places</h3> <p>In Google Maps, you can create a list of places, like your favorite places or places you want to visit.</p> <p>COMPUTER <b>ANDROID</b> IPHONE &amp; IPAD</p> <hr/> <h3>Make a new list</h3> <ol style="list-style-type: none"><li>1. On your Android phone or tablet, open the Google Maps app .</li><li>2. Tap Menu  &gt; <b>Your places</b> &gt; <b>Saved</b>.</li><li>3. In the bottom right, tap Add .</li><li>4. Enter a name and description.</li><li>5. Tap <b>Save</b>.</li></ol> <h3>Save a place to a list</h3> <ol style="list-style-type: none"><li>1. Open the Google Maps app .</li><li>2. Search for a place or tap it on the map.</li><li>3. At the bottom, tap the place's name or address.</li><li>4. Tap <b>Save</b>.</li><li>5. Choose a list. To create a new list, tap <b>New list</b> .</li></ol> <h3>See your lists</h3> <ol style="list-style-type: none"><li>1. Open the Google Maps app .</li><li>2. Tap Menu  &gt; <b>Your places</b> &gt; <b>Saved</b>.</li></ol> <p><a href="https://support.google.com/maps/answer/7280933?hl=en&amp;ref_topic=7301134&amp;co=GENIE.Platform%3DAndroid&amp;oco=1">https://support.google.com/maps/answer/7280933?hl=en&amp;ref_topic=7301134&amp;co=GENIE.Platform%3DAndroid&amp;oco=1</a></p>

## Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products

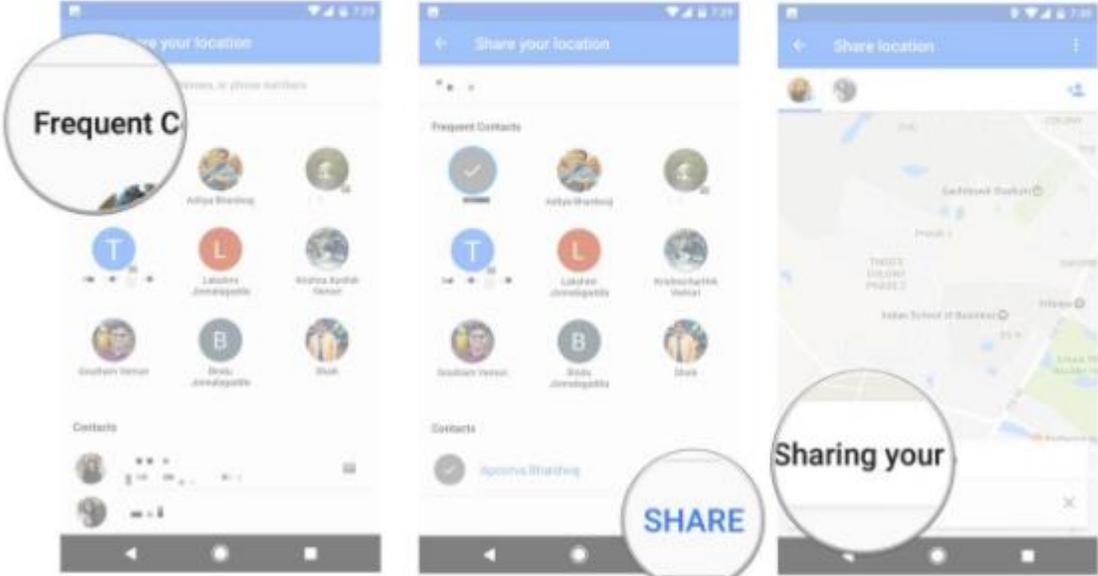
US9408055B2	HTC
	<h3 data-bbox="541 245 877 282">Hide or share lists</h3> <p data-bbox="541 310 909 334"><b>Note:</b> You can't share starred places.</p> <ol data-bbox="554 367 1251 472" style="list-style-type: none"><li data-bbox="554 367 890 391">1. Open the Google Maps app .</li><li data-bbox="554 407 968 431">2. Tap Menu  &gt; <b>Your places</b> &gt; <b>Saved</b>.</li><li data-bbox="554 448 1251 472">3. Next to the list you want to share, tap More  &gt; choose an option:</li></ol> <ul data-bbox="583 488 1682 630" style="list-style-type: none"><li data-bbox="583 488 1440 513">• <b>Hide/Show on your map:</b> Display or hide your saved places when looking at the map.</li><li data-bbox="583 529 1058 553">• <b>Share list:</b> Allow others to see your saved list.</li><li data-bbox="583 570 1682 630">• <b>Sharing options:</b> You can make your list public, private, or shared. To let anyone with the link see your list, tap <b>Shared</b>. To let anyone find and follow your list, tap <b>Public</b>.</li></ul> <h3 data-bbox="541 699 764 737">Follow a list</h3> <p data-bbox="541 764 1728 821">If you follow a list made by someone else, their saved places will show up in Your Places. The places will also appear as suggested locations in Google Maps.</p> <h3 data-bbox="541 878 915 915">Follow a list using a link</h3> <ol data-bbox="554 935 1356 1040" style="list-style-type: none"><li data-bbox="554 935 957 959">1. Tap on the link you received to open it.</li><li data-bbox="554 976 1272 1000">2. Tap <b>Follow</b>. This list will now be added to the group of lists you follow.</li><li data-bbox="554 1016 1356 1040">3. <b>Optional:</b> To unfollow a list someone shared with you, tap the list &gt; <b>Following</b>.</li></ol> <h3 data-bbox="541 1097 924 1135">See lists made by others</h3> <p data-bbox="541 1154 1335 1179">If a user has any Google Maps lists that were made public, you can follow them.</p> <ol data-bbox="554 1211 1136 1317" style="list-style-type: none"><li data-bbox="554 1211 1136 1235">1. Tap on the name of a user whose list you want to follow.</li><li data-bbox="554 1252 680 1276">2. Tap <b>Lists</b>.</li><li data-bbox="554 1292 1136 1317">3. Tap on the list you want to follow &gt; More  &gt; <b>Follow</b>.</li></ol> <p data-bbox="512 1333 1902 1401"><a href="https://support.google.com/maps/answer/7280933?hl=en&amp;ref_topic=7301134&amp;co=GENIE.Platform%3DAndroid&amp;oco=1">https://support.google.com/maps/answer/7280933?hl=en&amp;ref_topic=7301134&amp;co=GENIE.Platform%3DAndroid&amp;oco=1</a></p>



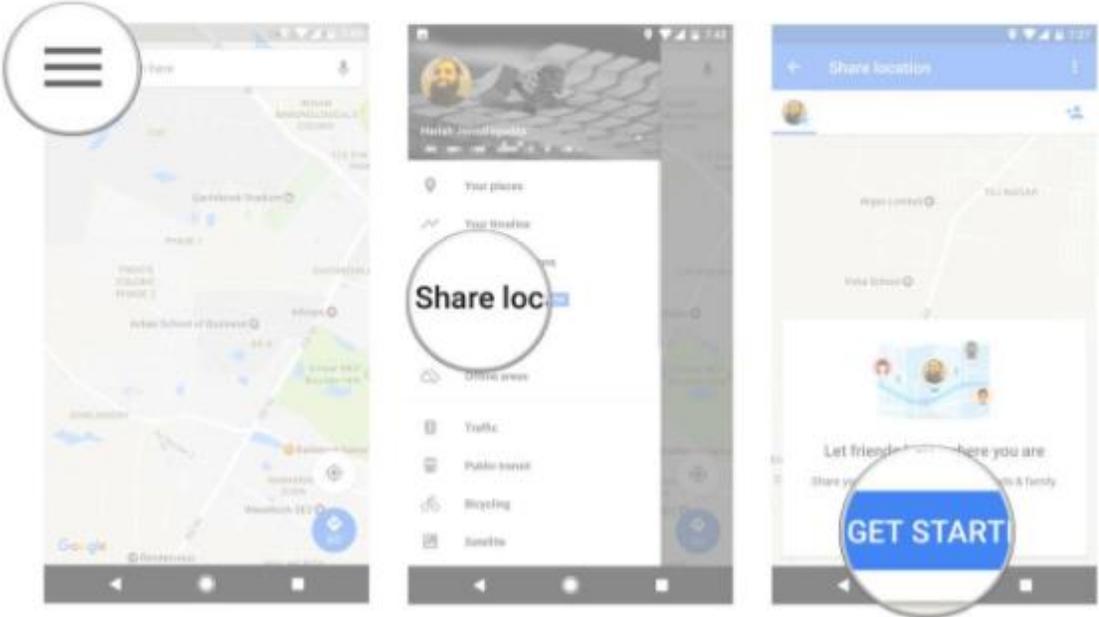
## Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products

US9408055B2	HTC
	<p data-bbox="520 240 1150 272"><b>How to share your location in Google Maps</b></p> <ol data-bbox="520 300 1134 389" style="list-style-type: none"> <li>1. Open Google Maps from the app drawer or the home screen.</li> <li>2. Tap the hamburger menu (the three horizontal lines) on the top left corner of the screen.</li> <li>3. Select Share location.</li> </ol>  <ol data-bbox="520 824 1165 933" style="list-style-type: none"> <li>4. Tap Get Started.</li> <li>5. Use the + Icon to select a time period or select the <b>Until you turn this off</b> setting to share your location indefinitely.</li> <li>6. Tap Select People.</li> </ol>  <p data-bbox="520 1339 1354 1372"><a href="https://www.androidcentral.com/how-share-location-google-maps">https://www.androidcentral.com/how-share-location-google-maps</a></p>

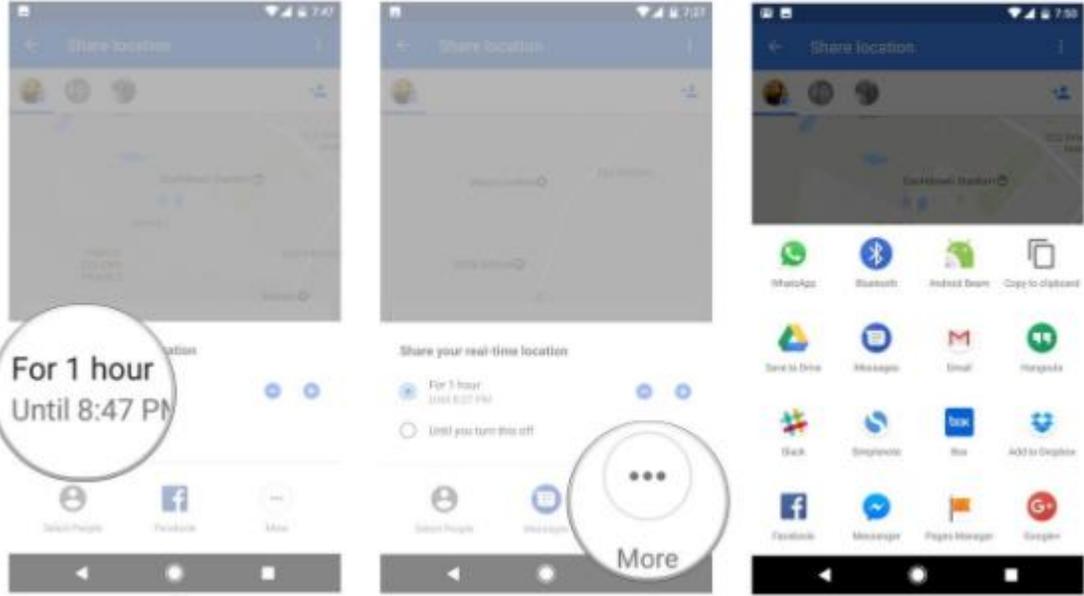
## Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products

US9408055B2	HTC
	<p data-bbox="527 253 1577 310">7. You'll see a list of your frequent contacts at the top, along with a full list of contacts. Pick the contacts by tapping their name.</p> <p data-bbox="527 339 1457 367">8. Once you've selected the contacts you want to share your location to, tap <b>Share</b>.</p> <p data-bbox="527 396 1419 423">9. You'll see a message saying that the selected contact can view your location.</p>  <p data-bbox="506 1065 1356 1092"><a href="https://www.androidcentral.com/how-share-location-google-maps">https://www.androidcentral.com/how-share-location-google-maps</a></p>

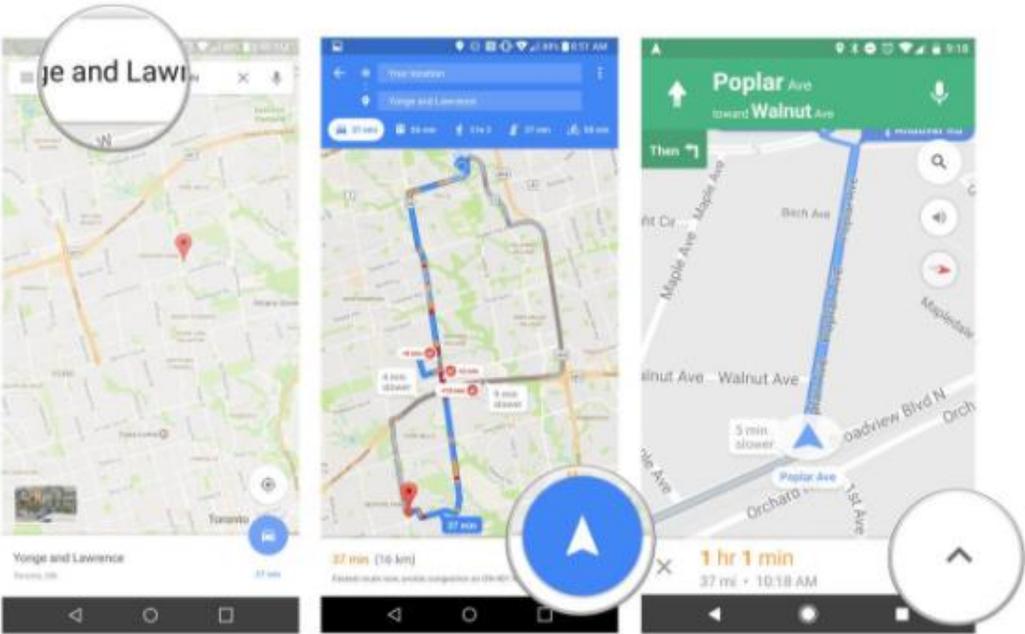
## Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products

US9408055B2	HTC
	<h3 data-bbox="520 245 1255 293">How to create a shareable link</h3> <p data-bbox="520 334 1461 362">You can also create a link and use it to share your location easily. Here's how to do it:</p> <ol data-bbox="520 412 1234 553" style="list-style-type: none"><li data-bbox="520 412 1234 440">1. Tap the hamburger menu on the top left corner of the screen.</li><li data-bbox="520 467 800 495">2. Select Share location.</li><li data-bbox="520 522 737 550">3. Tap Get Started.</li></ol>  <p data-bbox="506 1230 1356 1263"><a href="https://www.androidcentral.com/how-share-location-google-maps">https://www.androidcentral.com/how-share-location-google-maps</a></p>

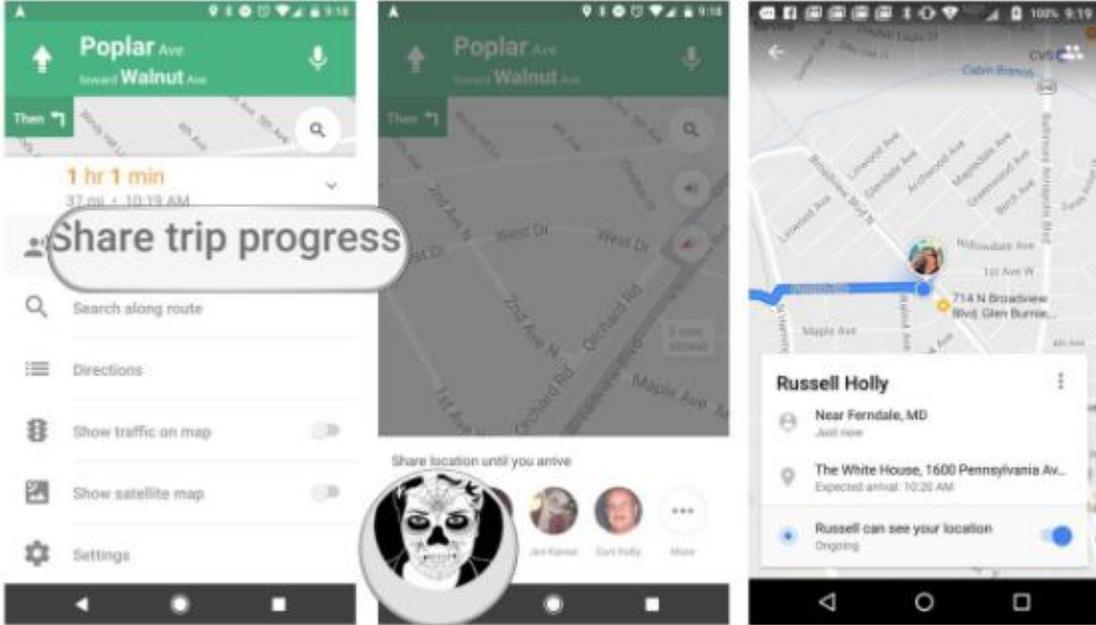
## Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products

US9408055B2	HTC
	<p data-bbox="520 245 1213 272">4. Select the <b>amount of time</b> you want to share your location.</p> <p data-bbox="520 302 680 329">5. Tap <b>More</b>.</p> <p data-bbox="520 358 1633 418">6. Select <b>your app of choice</b> to create and send a <b>unique URL</b> that broadcasts your current location. You can email it, send the link via Messenger, or even tweet it to the intended recipient.</p>  <p data-bbox="506 1084 1360 1112"><a href="https://www.androidcentral.com/how-share-location-google-maps">https://www.androidcentral.com/how-share-location-google-maps</a></p>

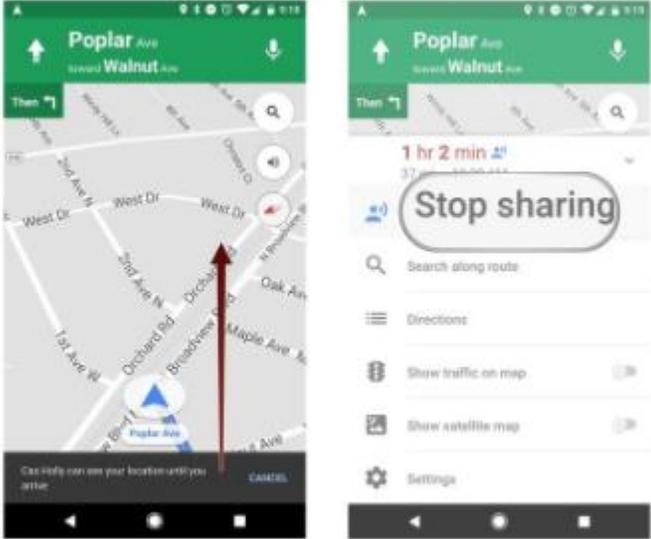
**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC
	<p data-bbox="527 240 1428 337"><b>How to share your navigation directions while you walk, drive or transit</b></p> <p data-bbox="527 375 1556 464">One of the best ways to share your location with someone is during a drive, walk or transit. If you're meeting a friend or family member somewhere, or navigating towards their home, Google Maps lets you share your location with them for the duration of the trip. It's magic!</p> <ol data-bbox="527 513 1394 643" style="list-style-type: none"> <li data-bbox="527 513 974 537">1. In the <b>search bar</b> enter your destination.</li> <li data-bbox="527 565 1394 589">2. Pick your navigation type (drive, transit, walk) and press the blue <b>navigate</b> button.</li> <li data-bbox="527 617 1394 641">3. Tap the arrow next to the <b>time-to-destination</b> number at the bottom of the screen.</li> </ol>  <p data-bbox="512 1328 1356 1360"><a href="https://www.androidcentral.com/how-share-location-google-maps">https://www.androidcentral.com/how-share-location-google-maps</a></p>

### Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products

US9408055B2	HTC
	<p data-bbox="527 277 835 305">4. Tap Share trip progress.</p> <p data-bbox="527 334 1150 362">5. Choose one or more contacts to share trip progress.</p>  <p data-bbox="527 1065 1360 1092">You can also stop sharing your location with someone before a trip ends.</p> <p data-bbox="512 1101 1360 1128"><a href="https://www.androidcentral.com/how-share-location-google-maps">https://www.androidcentral.com/how-share-location-google-maps</a></p>

## Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products

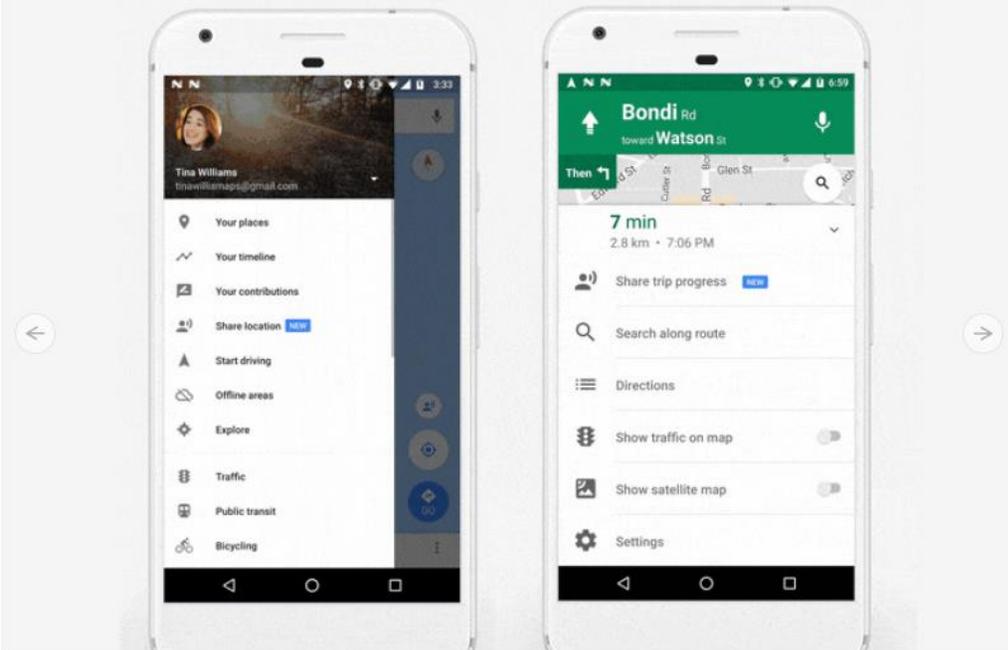
US9408055B2	HTC
	<ol style="list-style-type: none"><li data-bbox="533 245 1465 272">1. Tap the arrow next to the <b>time-to-destination</b> number at the bottom of the screen.</li><li data-bbox="533 302 768 329">2. Tap <b>Stop sharing</b>.</li></ol> <div data-bbox="747 383 1398 922"></div> <p data-bbox="541 976 638 1003">That's it!</p> <p data-bbox="541 1045 1608 1073">Are you excited that location sharing is back in Google Maps? How often do you use the feature?</p> <p data-bbox="512 1084 1356 1112"><a href="https://www.androidcentral.com/how-share-location-google-maps">https://www.androidcentral.com/how-share-location-google-maps</a></p> <p data-bbox="512 1195 1419 1222">As shown below, a group may also be defined within Google Contacts.</p>

## Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products

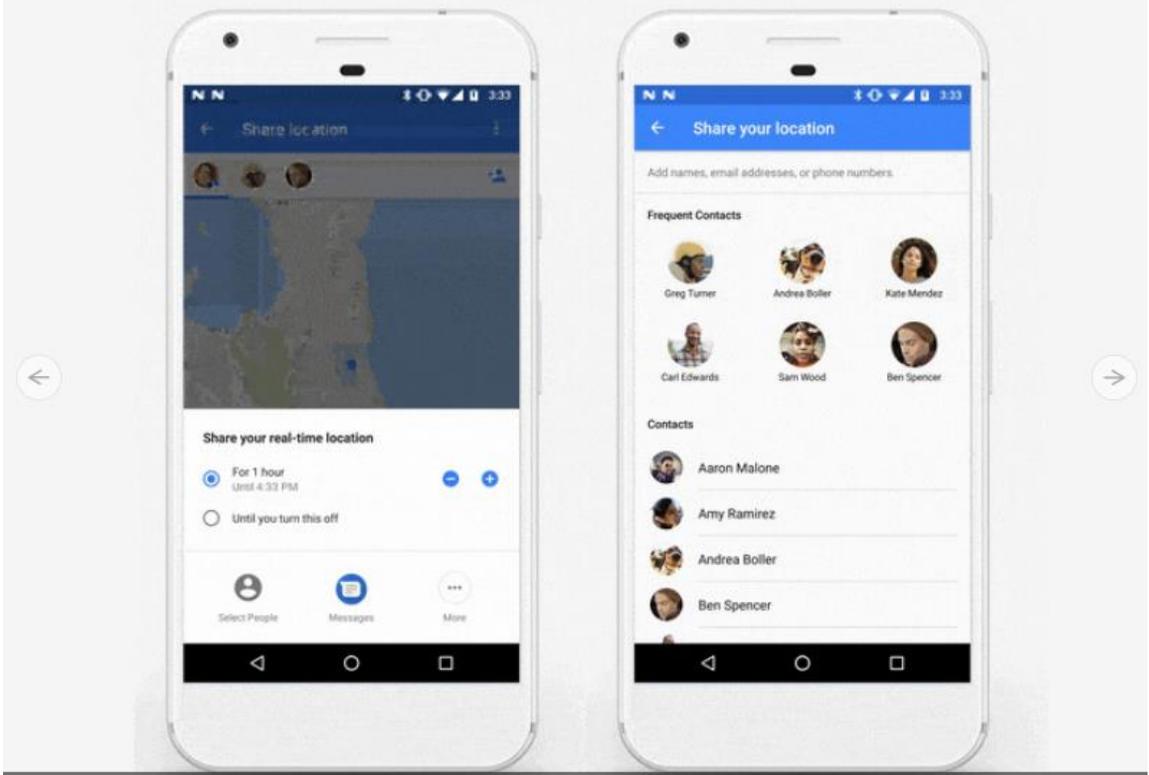
US9408055B2	HTC
	<h3 data-bbox="541 235 892 284">See your contacts</h3> <ol data-bbox="556 300 976 381" style="list-style-type: none"><li data-bbox="556 300 976 332">1. Open your device's Contacts app .</li><li data-bbox="556 349 724 381">2. Tap Menu .</li></ol> <ul data-bbox="546 406 1732 609" style="list-style-type: none"><li data-bbox="546 406 1113 438">• <b>See contacts by label:</b> Choose a label from the list.</li><li data-bbox="546 446 1365 479">• <b>See contacts for another account:</b> Tap Down arrow  &gt; pick an account.</li><li data-bbox="546 495 1218 527">• <b>See the contacts for all your accounts:</b> Choose <b>All contacts</b>.</li></ul> <p data-bbox="567 527 1732 560"><b>Tip:</b> If you have multiple contacts with the same information, the information will be grouped into one contact.</p> <li data-bbox="546 576 1354 609">• <b>See your Google Account contacts on the web:</b> Go to <a href="#">Google Contacts</a> .</li> <p data-bbox="514 633 1533 665"><a href="https://support.google.com/android/answer/6118731?hl=en&amp;ref_topic=6118711">https://support.google.com/android/answer/6118731?hl=en&amp;ref_topic=6118711</a></p> <h3 data-bbox="541 706 892 755">Label your contacts</h3> <p data-bbox="541 771 987 803">You can group contacts together using labels.</p> <ol data-bbox="556 828 934 933" style="list-style-type: none"><li data-bbox="556 828 934 860">1. Open your device's Contacts app .</li><li data-bbox="556 868 871 901">2. Tap Menu  &gt; <b>Create label</b>.</li><li data-bbox="556 909 871 933">3. Enter a label name and tap <b>Ok</b>.</li></ol> <ul data-bbox="546 958 1711 1031" style="list-style-type: none"><li data-bbox="546 958 1239 990">• <b>Add one contact to a label:</b> Tap Add contact  &gt; choose a contact.</li><li data-bbox="546 998 1711 1031">• <b>Add multiple contacts to a label:</b> Tap Add contact  &gt; touch and hold a contact &gt; tap the other contacts &gt; tap <b>Add</b>.</li></ul> <p data-bbox="514 1039 1533 1071"><a href="https://support.google.com/android/answer/6118731?hl=en&amp;ref_topic=6118711">https://support.google.com/android/answer/6118731?hl=en&amp;ref_topic=6118711</a></p> <h3 data-bbox="546 1120 945 1169">Share your contacts</h3> <ol data-bbox="556 1193 1050 1356" style="list-style-type: none"><li data-bbox="556 1193 976 1226">1. Open your device's Contacts app .</li><li data-bbox="556 1234 840 1266">2. Tap a contact in the list.</li><li data-bbox="556 1274 829 1307">3. Tap More  &gt; <b>Share</b>.</li><li data-bbox="556 1315 1050 1356">4. Choose how you want to share the contact.</li></ol> <p data-bbox="514 1372 1533 1404"><a href="https://support.google.com/android/answer/6118731?hl=en&amp;ref_topic=6118711">https://support.google.com/android/answer/6118731?hl=en&amp;ref_topic=6118711</a></p>



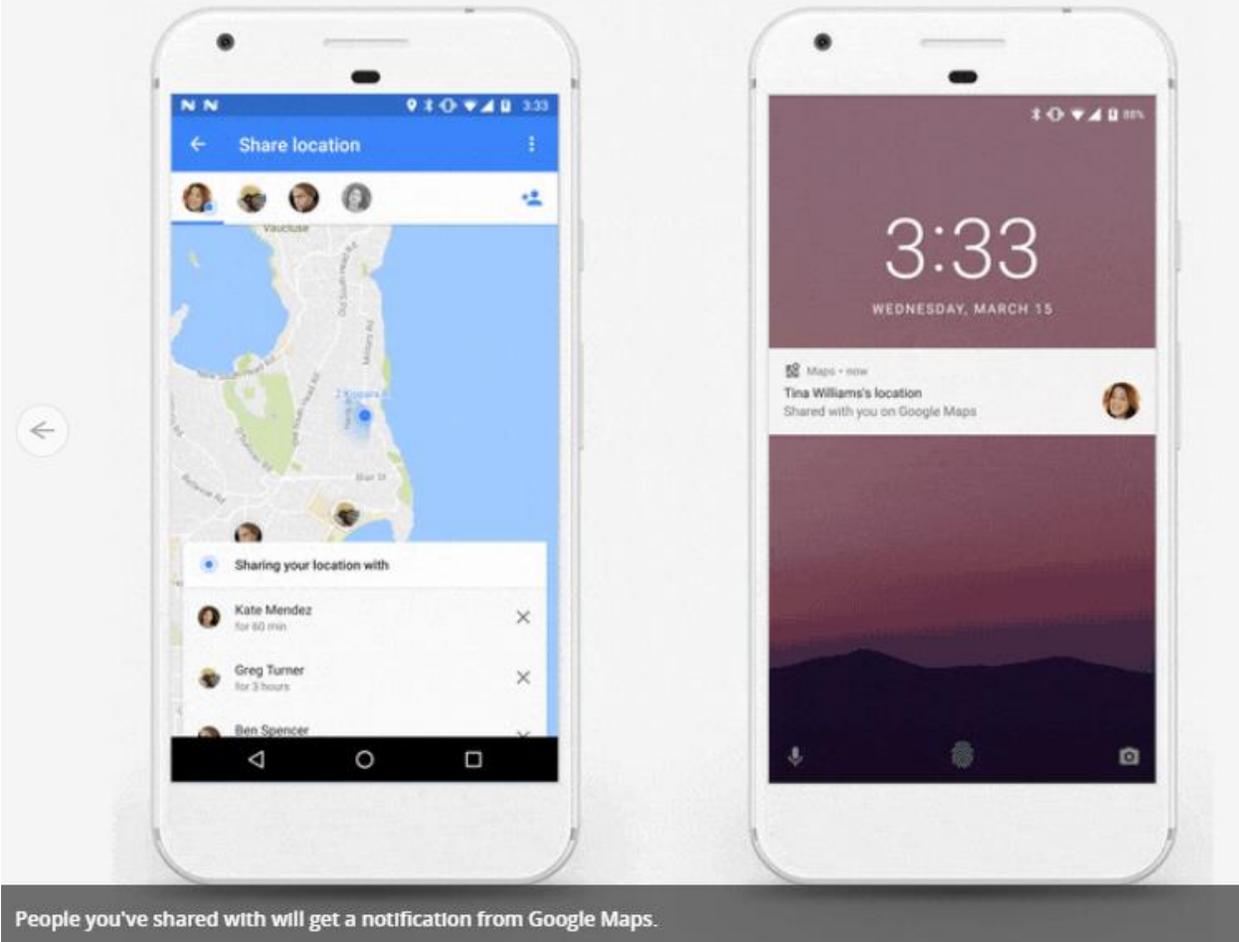
## Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products

US9408055B2	HTC
	 <p>Ways to start location sharing: (left) from the app navigation drawer, and (right) from the menu, you can drag up while navigating somewhere.</p> <p><a href="https://arstechnica.com/gadgets/2017/03/location-sharing-finally-returns-to-google-maps/">https://arstechnica.com/gadgets/2017/03/location-sharing-finally-returns-to-google-maps/</a></p>

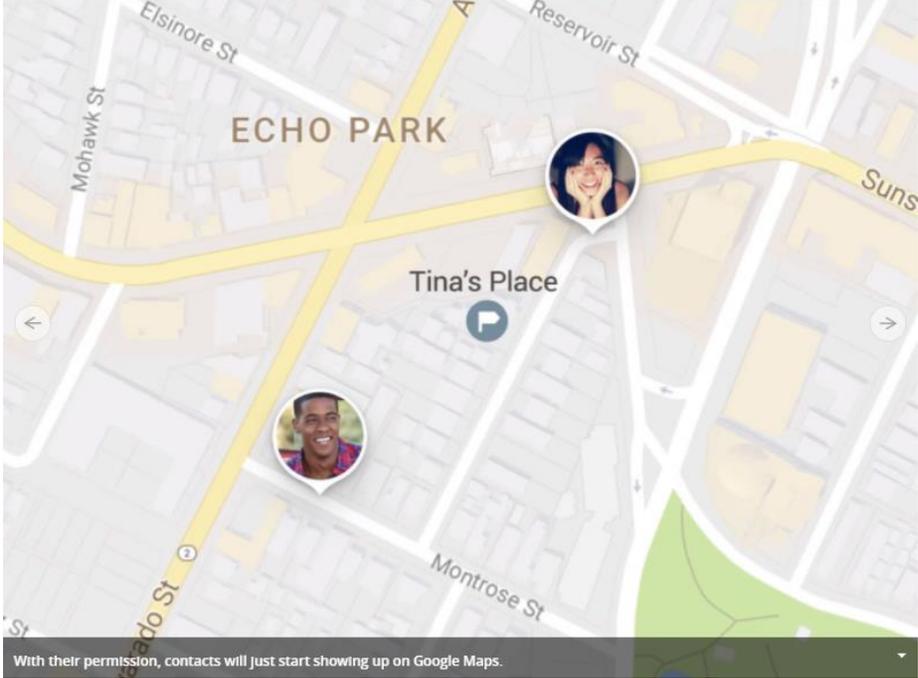
### Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products

US9408055B2	HTC
	 <p data-bbox="512 1024 1661 1057">You can pick how long you want to share your location for, and with whom you want to share it.</p> <p data-bbox="512 1065 1661 1097"><a href="https://arstechnica.com/gadgets/2017/03/location-sharing-finally-returns-to-google-maps/">https://arstechnica.com/gadgets/2017/03/location-sharing-finally-returns-to-google-maps/</a></p>

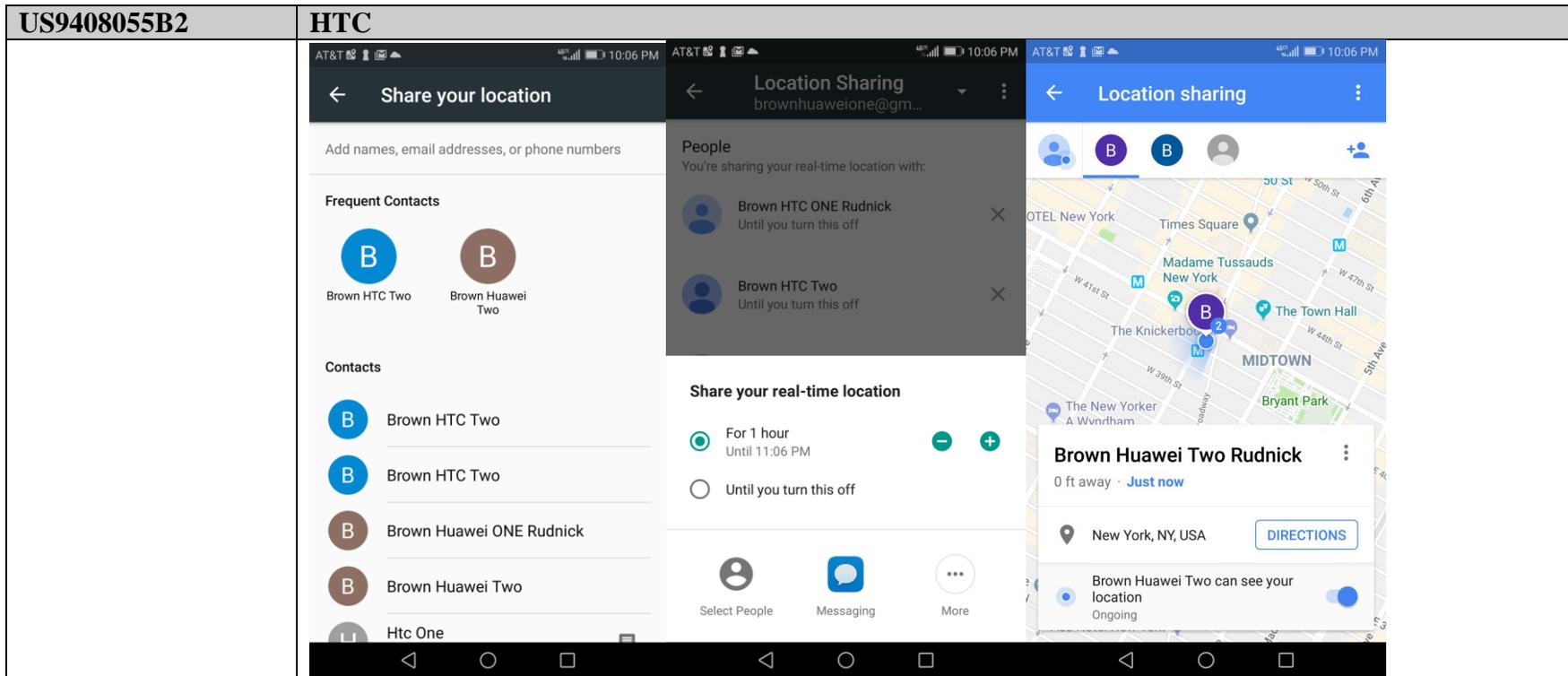
### Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products

US9408055B2	HTC
	 <p data-bbox="514 1144 1176 1177">People you've shared with will get a notification from Google Maps.</p> <p data-bbox="514 1185 1659 1226"><a href="https://arstechnica.com/gadgets/2017/03/location-sharing-finally-returns-to-google-maps/">https://arstechnica.com/gadgets/2017/03/location-sharing-finally-returns-to-google-maps/</a></p>

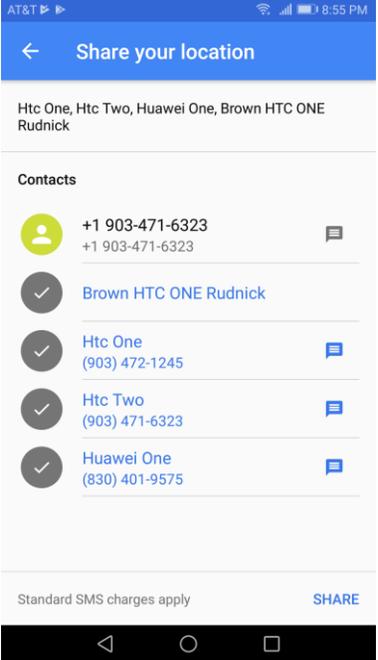
### Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products

US9408055B2	HTC
	 <p data-bbox="512 915 1656 948"><a href="https://arstechnica.com/gadgets/2017/03/location-sharing-finally-returns-to-google-maps/">https://arstechnica.com/gadgets/2017/03/location-sharing-finally-returns-to-google-maps/</a></p> <p data-bbox="512 987 1031 1019"><b><u>Exemplary Google Maps Screenshots:</u></b></p>

### Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products



## Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products

US9408055B2	HTC
	 <p>Exemplary Source Code: The above functionality is performed at least in part by the following publicly available source code and/or source code that invokes or is invoked by the following source code (or a substantially similar copy compiled and loaded onto the Accused Products by HTC): AGIS reserves the right to amend these contentions to include additional source code as discovery progresses and as additional source code is made available.</p>

## Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products

US9408055B2	HTC
	<h3 data-bbox="527 240 1016 293">Contacts Provider</h3> <p data-bbox="527 329 1472 589">The Contacts Provider is a powerful and flexible Android component that manages the device's central repository of data about people. The Contacts Provider is the source of data you see in the device's contacts application, and you can also access its data in your own application and transfer data between the device and online services. The provider accommodates a wide range of data sources and tries to manage as much data as possible for each person, with the result that its organization is complex. Because of this, the provider's API includes an extensive set of contract classes and interfaces that facilitate both data retrieval and modification.</p> <p data-bbox="527 625 835 646">This guide describes the following:</p> <ul data-bbox="527 675 1373 850" style="list-style-type: none"><li data-bbox="527 675 806 696">• The basic provider structure.</li><li data-bbox="527 725 894 747">• How to retrieve data from the provider.</li><li data-bbox="527 776 863 797">• How to modify data in the provider.</li><li data-bbox="527 826 1373 847">• How to write a sync adapter for synchronizing data from your server to the Contacts Provider.</li></ul> <p data-bbox="510 862 1486 889"><a href="https://developer.android.com/guide/topics/providers/contacts-provider.html">https://developer.android.com/guide/topics/providers/contacts-provider.html</a></p>

**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC
	<p><b>Overview</b></p> <p>ContactsContract defines an extensible database of contact-related information. Contact information is stored in a three-tier data model:</p> <ul style="list-style-type: none"> <li>• A row in the <code>ContactsContract.Data</code> table can store any kind of personal data, such as a phone number or email addresses. The set of data kinds that can be stored in this table is open-ended. There is a predefined set of common kinds, but any application can add its own data kinds.</li> <li>• A row in the <code>ContactsContract.RawContacts</code> table represents a set of data describing a person and associated with a single account (for example, one of the user's Gmail accounts).</li> <li>• A row in the <code>ContactsContract.Contacts</code> table represents an aggregate of one or more RawContacts presumably describing the same person. When data in or associated with the RawContacts table is changed, the affected aggregate contacts are updated as necessary.</li> </ul> <p>Other tables include:</p> <ul style="list-style-type: none"> <li>• <code>ContactsContract.Groups</code>, which contains information about raw contact groups such as Gmail contact groups. The current API does not support the notion of groups spanning multiple accounts.</li> <li>• <code>ContactsContract.StatusUpdates</code>, which contains social status updates including IM availability.</li> <li>• <code>ContactsContract.AggregationExceptions</code>, which is used for manual aggregation and disaggregation of raw contacts</li> <li>• <code>ContactsContract.Settings</code>, which contains visibility and sync settings for accounts and groups.</li> <li>• <code>ContactsContract.SyncState</code>, which contains free-form data maintained on behalf of sync adapters</li> <li>• <code>ContactsContract.PhoneLookup</code>, which is used for quick caller-ID lookup</li> </ul> <p><a href="https://developer.android.com/reference/android/provider/ContactsContract.html">https://developer.android.com/reference/android/provider/ContactsContract.html</a></p> <p><b>Data</b></p> <p>As noted previously, the data for a raw contact is stored in a <code>ContactsContract.Data</code> row that is linked to the raw contact's <code>_ID</code> value. This allows a single raw contact to have multiple instances of the same type of data such as email addresses or phone numbers. For example, if "Thomas Higginson" for <code>emilyd@gmail.com</code> (the raw contact row for Thomas Higginson associated with the Google account <code>emilyd@gmail.com</code>) has a home email address of <code>thigg@gmail.com</code> and a work email address of <code>thomas.higginson@gmail.com</code>, the Contacts Provider stores the two email address rows and links them both to the raw contact.</p> <p>Notice that different types of data are stored in this single table. Display name, phone number, email, postal address, photo, and website detail rows are all found in the <code>ContactsContract.Data</code> table. To help manage this, the <code>ContactsContract.Data</code> table has some columns with descriptive names, and others with generic names. The contents of a descriptive-name column have the same meaning regardless of the type of data in the row, while the contents of a generic-name column have different meanings depending on the type of data.</p> <p><a href="https://developer.android.com/guide/topics/providers/contacts-provider.html">https://developer.android.com/guide/topics/providers/contacts-provider.html</a></p>



**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC											
	<table border="1"> <thead> <tr> <th data-bbox="514 240 609 272">Task</th> <th data-bbox="615 240 852 272">Action</th> <th data-bbox="858 240 1188 272">Data</th> <th data-bbox="1194 240 1486 272">MIME type</th> <th data-bbox="1493 240 1745 272">Notes</th> </tr> </thead> <tbody> <tr> <td data-bbox="514 277 609 937">Pick a contact from a list</td> <td data-bbox="615 277 852 937">ACTION_PICK</td> <td data-bbox="858 277 1188 937">                     One of:                     <ul style="list-style-type: none"> <li data-bbox="867 321 1180 375">• <code>Contacts.CONTENT_URI</code>, which displays a list of contacts.</li> <li data-bbox="867 402 1180 488">• <code>Phone.CONTENT_URI</code>, which displays a list of phone numbers for a raw contact.</li> <li data-bbox="867 516 1180 602">• <code>StructuredPostal.CONTENT_URI</code>, which displays a list of postal addresses for a raw contact.</li> <li data-bbox="867 630 1180 716">• <code>Email.CONTENT_URI</code>, which displays a list of email addresses for a raw contact.</li> </ul> </td> <td data-bbox="1194 277 1486 937">Not used</td> <td data-bbox="1493 277 1745 937">                     Displays a list of raw contacts or a list of data from a raw contact, depending on the content URI type you supply. Call <code>startActivityForResult()</code>, which returns the content URI of the selected row. The form of the URI is the table's content URI with the row's <code>LOOKUP_ID</code> appended to it. The device's contacts app delegates read and write permissions to this content URI for the life of your activity. See the <a href="#">Content Provider Basics</a> guide for more details.                 </td> </tr> </tbody> </table>	Task	Action	Data	MIME type	Notes	Pick a contact from a list	ACTION_PICK	One of: <ul style="list-style-type: none"> <li data-bbox="867 321 1180 375">• <code>Contacts.CONTENT_URI</code>, which displays a list of contacts.</li> <li data-bbox="867 402 1180 488">• <code>Phone.CONTENT_URI</code>, which displays a list of phone numbers for a raw contact.</li> <li data-bbox="867 516 1180 602">• <code>StructuredPostal.CONTENT_URI</code>, which displays a list of postal addresses for a raw contact.</li> <li data-bbox="867 630 1180 716">• <code>Email.CONTENT_URI</code>, which displays a list of email addresses for a raw contact.</li> </ul>	Not used	Displays a list of raw contacts or a list of data from a raw contact, depending on the content URI type you supply. Call <code>startActivityForResult()</code> , which returns the content URI of the selected row. The form of the URI is the table's content URI with the row's <code>LOOKUP_ID</code> appended to it. The device's contacts app delegates read and write permissions to this content URI for the life of your activity. See the <a href="#">Content Provider Basics</a> guide for more details.	<a href="https://developer.android.com/guide/topics/providers/contacts-provider.html">https://developer.android.com/guide/topics/providers/contacts-provider.html</a>
Task	Action	Data	MIME type	Notes								
Pick a contact from a list	ACTION_PICK	One of: <ul style="list-style-type: none"> <li data-bbox="867 321 1180 375">• <code>Contacts.CONTENT_URI</code>, which displays a list of contacts.</li> <li data-bbox="867 402 1180 488">• <code>Phone.CONTENT_URI</code>, which displays a list of phone numbers for a raw contact.</li> <li data-bbox="867 516 1180 602">• <code>StructuredPostal.CONTENT_URI</code>, which displays a list of postal addresses for a raw contact.</li> <li data-bbox="867 630 1180 716">• <code>Email.CONTENT_URI</code>, which displays a list of email addresses for a raw contact.</li> </ul>	Not used	Displays a list of raw contacts or a list of data from a raw contact, depending on the content URI type you supply. Call <code>startActivityForResult()</code> , which returns the content URI of the selected row. The form of the URI is the table's content URI with the row's <code>LOOKUP_ID</code> appended to it. The device's contacts app delegates read and write permissions to this content URI for the life of your activity. See the <a href="#">Content Provider Basics</a> guide for more details.								

**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC
	<pre>25  * Parsed form of the intent sent to the Contacts application. 26  */ 27  public class ContactsRequest { 28 29      /** Default mode: browse contacts */ 30      public static final int ACTION_DEFAULT = 10; 31 32      /** Show all contacts */ 33      public static final int ACTION_ALL_CONTACTS = 15; 34 35      /** Show all contacts with phone numbers */ 36      public static final int ACTION_CONTACTS_WITH_PHONES = 17; 37 38      /** Show contents of a specific group */ 39      public static final int ACTION_GROUP = 20; 40 41      /** Show all starred contacts */ 42      public static final int ACTION_STARRED = 30; 43 44      /** Show frequently contacted contacts */ 45      public static final int ACTION_FREQUENT = 40; 46 47      /** Show starred and the frequent */ 48      public static final int ACTION_STREQUENT = 50; 49 50      /** Show all contacts and pick them when clicking */ 51      public static final int ACTION_PICK_CONTACT = 60; 52 53      /** Show all contacts as well as the option to create a new one */ 54      public static final int ACTION_PICK_OR_CREATE_CONTACT = 70; 55 56      /** Show all contacts and pick them for edit when clicking, and allow creating a new contact */ 57      public static final int ACTION_INSERT_OR_EDIT_CONTACT = 80;</pre> <p><a href="https://android.googlesource.com/platform/packages/apps/Contacts/+nougat-mr1-release/src/com/android/contacts/list/ContactsRequest.java">https://android.googlesource.com/platform/packages/apps/Contacts/+nougat-mr1-release/src/com/android/contacts/list/ContactsRequest.java</a></p>

**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC
	<pre>59  /** Show all phone numbers and pick them when clicking */ 60  public static final int ACTION_PICK_PHONE = 90; 61 62  /** Show all postal addresses and pick them when clicking */ 63  public static final int ACTION_PICK_POSTAL = 100; 64 65  /** Show all postal addresses and pick them when clicking */ 66  public static final int ACTION_PICK_EMAIL = 105; 67 68  /** Show all contacts and create a shortcut for the picked contact */ 69  public static final int ACTION_CREATE_SHORTCUT_CONTACT = 110; 70 71  /** Show all phone numbers and create a call shortcut for the picked number */ 72  public static final int ACTION_CREATE_SHORTCUT_CALL = 120; 73 74  /** Show all phone numbers and create an SMS shortcut for the picked number */ 75  public static final int ACTION_CREATE_SHORTCUT_SMS = 130; 76 77  /** Show all contacts and activate the specified one */ 78  public static final int ACTION_VIEW_CONTACT = 140; 79 80  /** Show contacts recommended for joining with a specified target contact */ 81  public static final int ACTION_PICK_JOIN = 150;</pre> <p><a href="https://android.googlesource.com/platform/packages/apps/Contacts+/nougat-mr1-release/src/com/android/contacts/list/ContactsRequest.java">https://android.googlesource.com/platform/packages/apps/Contacts+/nougat-mr1-release/src/com/android/contacts/list/ContactsRequest.java</a></p>

**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC
	<pre> 104  * Displays a list to browse contacts. 105  */ 106  public class PeopleActivity extends ContactsActivity implements 107      View.OnCreateContextMenuListener, 108      View.OnClickListener, 109      ActionBarAdapter.Listener, 110      DialogManager.DialogShowingViewActivity, 111      ContactListFilterController.ContactListFilterListener, 112      ProviderStatusListener, 113      MultiContactDeleteListener, 114      JoinContactsListener { <a href="https://android.googlesource.com/platform/packages/apps/Contacts+/nougat-mr1-release/src/com/android/contacts/activities/PeopleActivity.java">https://android.googlesource.com/platform/packages/apps/Contacts+/nougat-mr1- release/src/com/android/contacts/activities/PeopleActivity.java</a> 145      * Showing a list of Contacts. Also used for showing search results in search mode. 146      */ 147      private MultiSelectContactsListFragment mAllFragment; 148      private ContactTileListFragment mFavoritesFragment; <a href="https://android.googlesource.com/platform/packages/apps/Contacts+/nougat-mr1-release/src/com/android/contacts/activities/PeopleActivity.java">https://android.googlesource.com/platform/packages/apps/Contacts+/nougat-mr1- release/src/com/android/contacts/activities/PeopleActivity.java</a> </pre>

**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC
	<pre>458     private void configureFragments(boolean fromRequest) { 459         if (fromRequest) { 460             ContactListFilter filter = null; 461             int actionCode = mRequest.getActionCode(); 462             boolean searchMode = mRequest.isSearchMode(); 463             final int tabToOpen; 464             switch (actionCode) { 465                 case ContactsRequest.ACTION_ALL_CONTACTS: 466                     filter = ContactListFilter.createFilterWithType( 467                         ContactListFilter.FILTER_TYPE_ALL_ACCOUNTS); 468                     tabToOpen = TabState.ALL; 469                     break; 470                 case ContactsRequest.ACTION_CONTACTS_WITH_PHONES: 471                     filter = ContactListFilter.createFilterWithType( 472                         ContactListFilter.FILTER_TYPE_WITH_PHONE_NUMBERS_ONLY); 473                     tabToOpen = TabState.ALL; 474                     break; 475 476                 case ContactsRequest.ACTION_FREQUENT: 477                 case ContactsRequest.ACTION_STREQUENT: 478                 case ContactsRequest.ACTION_STARRED: 479                     tabToOpen = TabState.FAVORITES; 480                     break; 481                 case ContactsRequest.ACTION_VIEW_CONTACT: 482                     tabToOpen = TabState.ALL; 483                     break; 484                 default: 485                     tabToOpen = -1; 486                     break; 487             } </pre> <p data-bbox="506 1320 1566 1390"><a href="https://android.googlesource.com/platform/packages/apps/Contacts+/nougat-mr1-release/src/com/android/contacts/activities/PeopleActivity.java">https://android.googlesource.com/platform/packages/apps/Contacts+/nougat-mr1-release/src/com/android/contacts/activities/PeopleActivity.java</a></p>

**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC
	<pre>488         if (tabToOpen != -1) { 489             mActionBarAdapter.setCurrentTab(tabToOpen); 490         } 491 492         if (filter != null) { 493             mContactListFilterController.setContactListFilter(filter, false); 494             searchMode = false; 495         } 496 497         if (mRequest.getContactUri() != null) { 498             searchMode = false; 499         } 500 501         mActionBarAdapter.setSearchMode(searchMode); 502         configureContactListFragmentForRequest(); 503     } 504 505     configureContactListFragment(); 506 507     invalidateOptionsMenuIfNeeded(); 508 }</pre> <p data-bbox="506 1019 1570 1084"><a href="https://android.googlesource.com/platform/packages/apps/Contacts+/nougat-mr1-release/src/com/android/contacts/activities/PeopleActivity.java">https://android.googlesource.com/platform/packages/apps/Contacts+/nougat-mr1-release/src/com/android/contacts/activities/PeopleActivity.java</a></p>

**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC
	<pre>35 public class ProfileAndContactsLoader extends CursorLoader { 36 37     private boolean mLoadProfile; 38 39     private String[] mProjection; 40 41     private Uri mExtraUri; 42     private String[] mExtraProjection; 43     private String mExtraSelection; 44     private String[] mExtraSelectionArgs; 45     private boolean mMergeExtraContactsAfterPrimary; 46 47     public ProfileAndContactsLoader(Context context) { 48         super(context); 49     }</pre> <p><a href="https://android.googlesource.com/platform/packages/apps/ContactsCommon/+/nougat-mr1-release/src/com/android/contacts/common/list/ProfileAndContactsLoader.java">https://android.googlesource.com/platform/packages/apps/ContactsCommon/+/nougat-mr1-release/src/com/android/contacts/common/list/ProfileAndContactsLoader.java</a></p>

**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC
	<pre>35 public final class GroupMemberLoader extends CursorLoader { 36 37     public static class GroupEditorQuery { 38         private static final String[] PROJECTION = new String[] { 39             Data.CONTACT_ID,           // 0 40             Data.RAW_CONTACT_ID,      // 1 41             Data.DISPLAY_NAME_PRIMARY, // 2 42             Data.PHOTO_URI,           // 3 43             Data.LOOKUP_KEY,          // 4 44         }; 45 46         public static final int CONTACT_ID           = 0; 47         public static final int RAW_CONTACT_ID      = 1; 48         public static final int CONTACT_DISPLAY_NAME_PRIMARY = 2; 49         public static final int CONTACT_PHOTO_URI   = 3; 50         public static final int CONTACT_LOOKUP_KEY  = 4; 51     } 52 53     public static class GroupDetailQuery { 54         private static final String[] PROJECTION = new String[] { 55             Data.CONTACT_ID,           // 0 56             Data.PHOTO_URI,           // 1 57             Data.LOOKUP_KEY,          // 2 58             Data.DISPLAY_NAME_PRIMARY, // 3 59             Data.CONTACT_PRESENCE,    // 4 60             Data.CONTACT_STATUS,      // 5 61         }; 62 63         public static final int CONTACT_ID           = 0; 64         public static final int CONTACT_PHOTO_URI   = 1; 65         public static final int CONTACT_LOOKUP_KEY  = 2; 66         public static final int CONTACT_DISPLAY_NAME_PRIMARY = 3; 67         public static final int CONTACT_PRESENCE_STATUS = 4; 68         public static final int CONTACT_STATUS      = 5; 69     } 70 71     private final long mGroupId;</pre> <p><a href="https://android.googlesource.com/platform/packages/apps/Contacts/+/nougat-mr1-release/src/com/android/contacts/GroupMemberLoader.java">https://android.googlesource.com/platform/packages/apps/Contacts/+/nougat-mr1-release/src/com/android/contacts/GroupMemberLoader.java</a></p>



**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC
	<pre> 24  * Group loader for the group list that includes details such as the number of contacts per group 25  * and number of groups per account. This list is sorted by account type, account name, where the 26  * group names are in alphabetical order. Note that the list excludes default, favorite, and deleted 27  * groups. 28  */ 29  public final class GroupListLoader extends CursorLoader { 30 31      private final static String[] COLUMNS = new String[] { 32          Groups.ACCOUNT_NAME, 33          Groups.ACCOUNT_TYPE, 34          Groups.DATA_SET, 35          Groups._ID, 36          Groups.TITLE, 37          Groups.SUMMARY_COUNT, 38      }; 39 40      public final static int ACCOUNT_NAME = 0; 41      public final static int ACCOUNT_TYPE = 1; 42      public final static int DATA_SET = 2; 43      public final static int GROUP_ID = 3; 44      public final static int TITLE = 4; 45      public final static int MEMBER_COUNT = 5; 46 47      private static final Uri GROUP_LIST_URI = Groups.CONTENT_SUMMARY_URI; 48 49      public GroupListLoader(Context context) { 50          super(context, GROUP_LIST_URI, COLUMNS, Groups.ACCOUNT_TYPE + " NOT NULL AND " 51              + Groups.ACCOUNT_NAME + " NOT NULL AND " + Groups.AUTO_ADD + "=0 AND " + 52              Groups.FAVORITES + "=0 AND " + Groups.DELETED + "=0", null, 53              Groups.ACCOUNT_TYPE + ", " + Groups.ACCOUNT_NAME + ", " + Groups.DATA_SET + ", " + 54              Groups.TITLE + " COLLATE LOCALIZED ASC"); 55      } 56  } </pre> <p><a href="https://android.googlesource.com/platform/packages/apps/Contacts+/nougat-mr1-release/src/com/android/contacts/GroupListLoader.java">https://android.googlesource.com/platform/packages/apps/Contacts+/nougat-mr1-release/src/com/android/contacts/GroupListLoader.java</a></p>

**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC
	<pre> 24  * Group meta-data loader. Loads all groups or just a single group from the 25  * database (if given a {@link Uri}). 26  */ 27  public final class GroupMetaLoader extends CursorLoader { 28 29      private final static String[] COLUMNS = new String[] { 30          Groups.ACCOUNT_NAME, 31          Groups.ACCOUNT_TYPE, 32          Groups.DATA_SET, 33          Groups._ID, 34          Groups.TITLE, 35          Groups.AUTO_ADD, 36          Groups.FAVORITES, 37          Groups.GROUP_IS_READ_ONLY, 38          Groups.DELETED, 39      }; 40 41      public final static int ACCOUNT_NAME = 0; 42      public final static int ACCOUNT_TYPE = 1; 43      public final static int DATA_SET = 2; 44      public final static int GROUP_ID = 3; 45      public final static int TITLE = 4; 46      public final static int AUTO_ADD = 5; 47      public final static int FAVORITES = 6; 48      public final static int IS_READ_ONLY = 7; 49      public final static int DELETED = 8; 50 51      public GroupMetaLoader(Context context, Uri groupUri) { 52          super(context, ensureIsGroupUri(groupUri), COLUMNS, Groups.ACCOUNT_TYPE + " NOT NULL AND " 53              + Groups.ACCOUNT_NAME + " NOT NULL", null, null); 54      } 55 56      /** 57       * Ensures that this is a valid group URI. If invalid, then an exception is 58       * thrown. Otherwise, the original URI is returned. 59       */ 60      private static Uri ensureIsGroupUri(final Uri groupUri) { 61          // TODO: Fix ContactsProvider2 getType method to resolve the group Uris 62          if (groupUri == null) { 63              throw new IllegalArgumentException("Uri must not be null"); 64          } 65          if (!groupUri.toString().startsWith(Groups.CONTENT_URI.toString())) { 66              throw new IllegalArgumentException("Invalid group Uri: " + groupUri); 67          } 68          return groupUri; 69      } 70  } </pre> <p><a href="https://android.googlesource.com/platform/packages/apps/Contacts/+nougat-mr1-release/src/com/android/contacts/GroupMetaLoader.java">https://android.googlesource.com/platform/packages/apps/Contacts/+nougat-mr1-release/src/com/android/contacts/GroupMetaLoader.java</a></p>

**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC
	<pre> 19  * Meta-data for a contact group. We load all groups associated with the contact's 20  * constituent accounts. 21  */ 22  public final class GroupMetaData { 23      private String mAccountName; 24      private String mAccountType; 25      private String mDataSet; 26      private long mGroupId; 27      private String mTitle; 28      private boolean mDefaultGroup; 29      private boolean mFavorites; 30 31      public GroupMetaData(String accountName, String accountType, String dataSet, long groupId, 32                          String title, boolean defaultGroup, boolean favorites) { 33          this.mAccountName = accountName; 34          this.mAccountType = accountType; 35          this.mDataSet = dataSet; 36          this.mGroupId = groupId; 37          this.mTitle = title; 38          this.mDefaultGroup = defaultGroup; 39          this.mFavorites = favorites; 40      } 41 42      public String getAccountName() { 43          return mAccountName; 44      } 45 46      public String getAccountType() { 47          return mAccountType; 48      } 49 50      public String getDataSet() { 51          return mDataSet; 52      } 53 54      public long getGroupId() { 55          return mGroupId; 56      } 57 58      public String getTitle() { 59          return mTitle; 60      } 61 62      public boolean isDefaultGroup() { 63          return mDefaultGroup; 64      } 65 66      public boolean isFavorites() { 67          return mFavorites; </pre> <p><a href="https://android.googlesource.com/platform/packages/apps/ContactsCommon/+/nougat-mr1-release/src/com/android/contacts/common/GroupMetaData.java">https://android.googlesource.com/platform/packages/apps/ContactsCommon/+/nougat-mr1-release/src/com/android/contacts/common/GroupMetaData.java</a></p>

**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC
	<pre>44 * Class that sends chat message via SMS. 45 * 46 * The interface emulates a blocking sending similar to making an HTTP request. 47 * It calls the SmsManager to send a (potentially multipart) message and waits 48 * on the sent status on each part. The waiting has a timeout so it won't wait 49 * forever. Once the sent status of all parts received, the call returns. 50 * A successful sending requires success status for all parts. Otherwise, we 51 * pick the highest level of failure as the error for the whole message, which 52 * is used to determine if we need to retry the sending. 53 */ 54 public class SmsSender { 55     private static final String TAG = LogUtil.BUGLE_TAG; 56 57     public static final String EXTRA_PART_ID = "part_id"; 58 59     /* 60     * A map for pending sms messages. The key is the random request UUID. 61     */ 62     private static ConcurrentHashMap&lt;Uri, SendResult&gt; sPendingMessageMap = 63         new ConcurrentHashMap&lt;Uri, SendResult&gt;(); 64 65     private static final Random RANDOM = new Random(); 66 67     // Whether we should send multipart SMS as separate messages 68     private static Boolean sSendMultipartSmsAsSeparateMessages = null; 69</pre> <p><a href="https://android.googlesource.com/platform/packages/apps/Messaging/+nougat-mr1-release/src/com/android/messaging/sms/SmsSender.java">https://android.googlesource.com/platform/packages/apps/Messaging/+nougat-mr1-release/src/com/android/messaging/sms/SmsSender.java</a></p>

**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC
	<pre> 253 // Actually sending the message using SmsManager 254 private static void sendInternal(final Context context, final int subId, String dest, 255     final ArrayList&lt;String&gt; messages, final String serviceCenter, 256     final boolean requireDeliveryReport, final Uri messageUri) throws SmsException { 257     Assert.notNull(context); 258     final SmsManager smsManager = PhoneUtils.get(subId).getSmsManager(); 259     final int messageCount = messages.size(); 260     final ArrayList&lt;PendingIntent&gt; deliveryIntents = new ArrayList&lt;PendingIntent&gt;(messageCount); 261     final ArrayList&lt;PendingIntent&gt; sentIntents = new ArrayList&lt;PendingIntent&gt;(messageCount); 262     for (int i = 0; i &lt; messageCount; i++) { 263         // Make pending intents different for each message part 264         final int partId = (messageCount &lt;= 1 ? 0 : i + 1); 265         if (requireDeliveryReport &amp;&amp; (i == (messageCount - 1))) { 266             // TODO we only care about the delivery status of the last part 267             // Shall we have better tracking of delivery status of all parts? 268             deliveryIntents.add(PendingIntent.getBroadcast( 269                 context, 270                 partId, 271                 getSendStatusIntent(context, SendStatusReceiver.MESSAGE_DELIVERED_ACTION, 272                     messageUri, partId, subId), 273                 0/*flag*/)); 274         } else { 275             deliveryIntents.add(null); 276         } 277         sentIntents.add(PendingIntent.getBroadcast( 278             context, 279             partId, 280             getSendStatusIntent(context, SendStatusReceiver.MESSAGE_SENT_ACTION, 281                 messageUri, partId, subId), 282             0/*flag*/)); 283     } 284     if (sSendMultipartSmsAsSeparateMessages == null) { 285         sSendMultipartSmsAsSeparateMessages = MmsConfig.get(subId) 286             .getSendMultipartSmsAsSeparateMessages(); 287     } </pre> <p><a href="https://android.googlesource.com/platform/packages/apps/Messaging/+nougat-mr1-release/src/com/android/messaging/sms/SmsSender.java">https://android.googlesource.com/platform/packages/apps/Messaging/+nougat-mr1-release/src/com/android/messaging/sms/SmsSender.java</a></p>

**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC
	<pre> 288     try { 289         if (sSendMultipartSmsAsSeparateMessages) { 290             // If multipart sms is not supported, send them as separate messages 291             for (int i = 0; i &lt; messageCount; i++) { 292                 smsManager.sendTextMessage(dest, 293                     serviceCenter, 294                     messages.get(i), 295                     sentIntents.get(i), 296                     deliveryIntents.get(i)); 297             } 298         } else { 299             smsManager.sendMultipartTextMessage( 300                 dest, serviceCenter, messages, sentIntents, deliveryIntents); 301         } 302     } catch (final Exception e) { 303         throw new SmsException("SmsSender: caught exception in sending " + e); 304     } 305 } </pre> <p><a href="https://android.googlesource.com/platform/packages/apps/Messaging/+nougat-mr1-release/src/com/android/messaging/sms/SmsSender.java">https://android.googlesource.com/platform/packages/apps/Messaging/+nougat-mr1-release/src/com/android/messaging/sms/SmsSender.java</a></p> <pre> 56  * Class that receives incoming SMS messages through android.provider.Telephony.SMS_RECEIVED 57  * 58  * This class serves two purposes: 59  * - Process phone verification SMS messages 60  * - Handle SMS messages when the user has enabled us to be the default SMS app (Pre-KLP) 61  */ 62  public final class SmsReceiver extends BroadcastReceiver { 63      private static final String TAG = LogUtil.BUGLE_TAG; 64 65      private static ArrayList&lt;Pattern&gt; sIgnoreSmsPatterns; 66 </pre> <p><a href="https://android.googlesource.com/platform/packages/apps/Messaging/+nougat-mr1-release/src/com/android/messaging/receiver/SmsReceiver.java">https://android.googlesource.com/platform/packages/apps/Messaging/+nougat-mr1-release/src/com/android/messaging/receiver/SmsReceiver.java</a></p>

**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC
	<pre> 200     public static void deliverSmsMessages(final Context context, final int subId, 201         final int errorCode, final android.telephony.SmsMessage[] messages) { 202         final ContentValues messageValues = 203             MmsUtils.parseReceivedSmsMessage(context, messages, errorCode); 204 205         LogUtil.v(TAG, "SmsReceiver.deliverSmsMessages"); 206 207         final long nowInMillis = System.currentTimeMillis(); 208         final long receivedTimestampMs = MmsUtils.getMessageDate(messages[0], nowInMillis); 209 210         messageValues.put(Sms.Inbox.DATE, receivedTimestampMs); 211         // Default to unread and unseen for us but ReceiveSmsMessageAction will override 212         // seen for the telephony db. 213         messageValues.put(Sms.Inbox.READ, 0); 214         messageValues.put(Sms.Inbox.SEEN, 0); 215         if (OsUtil.isAtLeastL_MR1()) { 216             messageValues.put(Sms.SUBSCRIPTION_ID, subId); 217         } 218 219         if (messages[0].getMessageClass() == android.telephony.SmsMessage.MessageClass.CLASS_0    220             DebugUtils.debugClassZeroSmsEnabled()) { 221             Factory.get().getUIIntents().launchClassZeroActivity(context, messageValues); 222         } else { 223             final ReceiveSmsMessageAction action = new ReceiveSmsMessageAction(messageValues); 224             action.start(); 225         } 226     } </pre> <p><a href="https://android.googlesource.com/platform/packages/apps/Messaging/+nougat-mr1-release/src/com/android/messaging/receiver/SmsReceiver.java">https://android.googlesource.com/platform/packages/apps/Messaging/+nougat-mr1-release/src/com/android/messaging/receiver/SmsReceiver.java</a></p>

**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC
	<pre>228     @Override 229     public void onReceive(final Context context, final Intent intent) { 230         LogUtil.v(TAG, "SmsReceiver.onReceive " + intent); 231         // On KLP+ we only take delivery of SMS messages in SmsDeliverReceiver. 232         if (PhoneUtils.getDefault().isSmsEnabled()) { 233             final String action = intent.getAction(); 234             if (OsUtil.isSecondaryUser() &amp;&amp; 235                 (Telephony.Sms.Intents.SMS_RECEIVED_ACTION.equals(action)    236                  // TODO: update this with the actual constant from Telephony 237                  "android.provider.Telephony.MMS_DOWNLOADED".equals(action))) { 238                 postNewMessageSecondaryUserNotification(); 239             } else if (!OsUtil.isAtLeastKLP()) { 240                 deliverSmsIntent(context, intent); 241             } 242         } 243     }</pre> <p><a href="https://android.googlesource.com/platform/packages/apps/Messaging/+/nougat-mr1-release/src/com/android/messaging/receiver/SmsReceiver.java">https://android.googlesource.com/platform/packages/apps/Messaging/+/nougat-mr1-release/src/com/android/messaging/receiver/SmsReceiver.java</a></p>



**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC
	<pre>52  * Class that sends chat message via MMS. 53  * 54  * The interface emulates a blocking send similar to making an HTTP request. 55  */ 56  public class MmsSender { 57      private static final String TAG = LogUtil.BUGLE_TAG; 58 59      /** 60       * Send an MMS message. 61       * 62       * @param context Context 63       * @param messageUri The unique URI of the message for identifying it during sending 64       * @param sendReq The SendReq PDU of the message 65       * @throws MmsFailureException 66       */ 67      public static void sendMms(final Context context, final int subId, final Uri messageUri, 68                               final SendReq sendReq, final Bundle sentIntentExtras) throws MmsFailureException { 69          sendMms(context, 70                subId, 71                messageUri, 72                null /* locationUri */, 73                sendReq, 74                true /* responseImportant */, 75                sentIntentExtras); 76      }</pre> <p><a href="https://android.googlesource.com/platform/packages/apps/Messaging/+/nougat-mr1-release/src/com/android/messaging/sms/MmsSender.java">https://android.googlesource.com/platform/packages/apps/Messaging/+/nougat-mr1-release/src/com/android/messaging/sms/MmsSender.java</a></p>

**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC
	<pre> 240      * Download an MMS message. 241      * 242      * @param context Context 243      * @param contentLocation The url of the MMS message 244      * @throws MmsFailureException 245      * @throws InvalidHeaderValueException 246      */ 247      public static void downloadMms(final Context context, final int subId, 248          final String contentLocation, Bundle extras) throws MmsFailureException, 249          InvalidHeaderValueException { 250          final Uri requestUri = Uri.parse(contentLocation); 251          final Uri contentUri = MmsFileProvider.buildRawMmsUri(); 252 253          final Intent downloadedIntent = new Intent(SendStatusReceiver.MMS_DOWNLOADED_ACTION, 254              requestUri, 255              context, 256              SendStatusReceiver.class); 257          downloadedIntent.putExtra(SendMessageAction.EXTRA_CONTENT_URI, contentUri); 258          if (extras != null) { 259              downloadedIntent.putExtras(extras); 260          } 261          final PendingIntent downloadedPendingIntent = PendingIntent.getBroadcast( 262              context, 263              0 /*request code*/, 264              downloadedIntent, 265              PendingIntent.FLAG_UPDATE_CURRENT); 266 267          MmsManager.downloadMultimediaMessage(subId, context, contentLocation, contentUri, 268              downloadedPendingIntent); 269      } </pre> <p><a href="https://android.googlesource.com/platform/packages/apps/Messaging/+nougat-mr1-release/src/com/android/messaging/sms/MmsSender.java">https://android.googlesource.com/platform/packages/apps/Messaging/+nougat-mr1-release/src/com/android/messaging/sms/MmsSender.java</a></p>

**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC
	<pre> 97      * Execute an MMS HTTP request, either a POST (sending) or a GET (downloading) 98      * 99      * @param urlString The request URL, for sending it is usually the MMSC, and for downloading 100     *           it is the message URL 101     * @param pdu For POST (sending) only, the PDU to send 102     * @param method HTTP method, POST for sending and GET for downloading 103     * @param isProxySet Is there a proxy for the MMSC 104     * @param proxyHost The proxy host 105     * @param proxyPort The proxy port 106     * @param mmsConfig The MMS config to use 107     * @param userAgent The user agent header value 108     * @param uaProfUrl The UA Prof URL header value 109     * @return The HTTP response body 110     * @throws MmsHttpException For any failures 111     */ 112     public byte[] execute(String urlString, byte[] pdu, String method, boolean isProxySet, 113         String proxyHost, int proxyPort, Bundle mmsConfig, String userAgent, String uaProfUrl) 114         throws MmsHttpException { 115         Log.d(MmsService.TAG, "HTTP: " + method + " " + Utils.redactUrlForNonVerbose(urlString) 116             + (isProxySet ? (" proxy=" + proxyHost + ":" + proxyPort) : "")) 117             + ", PDU size=" + (pdu != null ? pdu.length : 0)); 118         checkMethod(method); 119         HttpURLConnection connection = null; 120         try { 121             Proxy proxy = Proxy.NO_PROXY; 122             if (isProxySet) { 123                 proxy = new Proxy(Proxy.Type.HTTP, new InetSocketAddress(proxyHost, proxyPort)); 124             } 125             final URL url = new URL(urlString); 126             // Now get the connection 127             connection = (HttpURLConnection) url.openConnection(proxy); 128             connection.setDoInput(true); 129             connection.setConnectTimeout( 130                 mmsConfig.getInt(CarrierConfigValuesLoader.CONFIG_HTTP_SOCKET_TIMEOUT, 131                     CarrierConfigValuesLoader.CONFIG_HTTP_SOCKET_TIMEOUT_DEFAULT)); </pre> <p data-bbox="506 1219 1596 1287"><a href="https://android.googlesource.com/platform/packages/apps/Messaging/+nougat-mr1-release/src/android/support/v7/mms/MmsHttpClient.java">https://android.googlesource.com/platform/packages/apps/Messaging/+nougat-mr1-release/src/android/support/v7/mms/MmsHttpClient.java</a></p>

**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC
	<pre> 132 // ----- COMMON HEADERS ----- 133 // Header: Accept 134 connection.setRequestProperty(HEADER_ACCEPT, HEADER_VALUE_ACCEPT); 135 // Header: Accept-Language 136 connection.setRequestProperty( 137     HEADER_ACCEPT_LANGUAGE, getCurrentAcceptLanguage(Locale.getDefault())); 138 // Header: User-Agent 139 Log.i(MmsService.TAG, "HTTP: User-Agent=" + userAgent); 140 connection.setRequestProperty(HEADER_USER_AGENT, userAgent); 141 // Header: x-wap-profile 142 final String uaProfUrlTagName = mmsConfig.getString( 143     CarrierConfigValuesLoader.CONFIG_UA_PROF_TAG_NAME, 144     CarrierConfigValuesLoader.CONFIG_UA_PROF_TAG_NAME_DEFAULT); 145 if (uaProfUrl != null) { 146     Log.i(MmsService.TAG, "HTTP: UaProfUrl=" + uaProfUrl); 147     connection.setRequestProperty(uaProfUrlTagName, uaProfUrl); 148 } 149 // Add extra headers specified by mms_config.xml's httpparams 150 addExtraHeaders(connection, mmsConfig); 151 // Different stuff for GET and POST 152 if (METHOD_POST.equals(method)) { 153     if (pdu == null    pdu.length &lt; 1) { 154         Log.e(MmsService.TAG, "HTTP: empty pdu"); 155         throw new MmsHttpException(0/*statusCode*/, "Sending empty PDU"); 156     } 157     connection.setDoOutput(true); 158     connection.setRequestMethod(METHOD_POST); 159     if (mmsConfig.getBoolean( 160         CarrierConfigValuesLoader.CONFIG_SUPPORT_HTTP_CHARSET_HEADER, 161         CarrierConfigValuesLoader.CONFIG_SUPPORT_HTTP_CHARSET_HEADER_DEFAULT)) { 162         connection.setRequestProperty(HEADER_CONTENT_TYPE, 163             HEADER_VALUE_CONTENT_TYPE_WITH_CHARSET); 164     } else { 165         connection.setRequestProperty(HEADER_CONTENT_TYPE, 166             HEADER_VALUE_CONTENT_TYPE_WITHOUT_CHARSET); </pre> <p><a href="https://android.googlesource.com/platform/packages/apps/Messaging/+nougat-mr1-release/src/android/support/v7/mms/MmsHttpClient.java">https://android.googlesource.com/platform/packages/apps/Messaging/+nougat-mr1-release/src/android/support/v7/mms/MmsHttpClient.java</a></p>

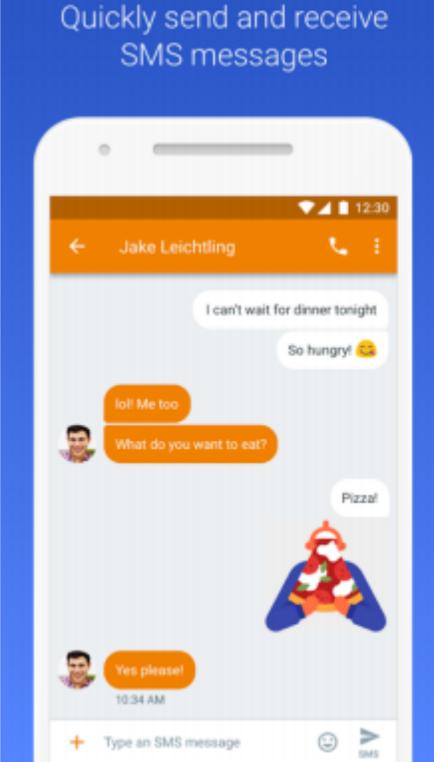
**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC
	<pre> 167     } 168     if (Log.isLoggable(MmsService.TAG, Log.VERBOSE)) { 169         logHttpHeaders(connection.getRequestProperties()); 170     } 171     connection.setFixedLengthStreamingMode(pdu.length); 172     // Sending request body 173     final OutputStream out = 174         new BufferedOutputStream(connection.getOutputStream()); 175     out.write(pdu); 176     out.flush(); 177     out.close(); 178 } else if (METHOD_GET.equals(method)) { 179     if (Log.isLoggable(MmsService.TAG, Log.VERBOSE)) { 180         logHttpHeaders(connection.getRequestProperties()); 181     } 182     connection.setRequestMethod(METHOD_GET); 183 } 184 // Get response 185 final int responseCode = connection.getResponseCode(); 186 final String responseMessage = connection.getResponseMessage(); 187 Log.d(MmsService.TAG, "HTTP: " + responseCode + " " + responseMessage); 188 if (Log.isLoggable(MmsService.TAG, Log.VERBOSE)) { 189     logHttpHeaders(connection.getHeaderFields()); 190 } 191 if (responseCode / 100 != 2) { 192     throw new MmsHttpException(responseCode, responseMessage); 193 } 194 final InputStream in = new BufferedInputStream(connection.getInputStream()); 195 final ByteArrayOutputStream byteOut = new ByteArrayOutputStream(); 196 final byte[] buf = new byte[4096]; 197 int count = 0; 198 while ((count = in.read(buf)) &gt; 0) { 199     byteOut.write(buf, 0, count); 200 } 201 in.close(); 202 final byte[] responseBody = byteOut.toByteArray(); 203 Log.d(MmsService.TAG, "HTTP: response size=" 204     + (responseBody != null ? responseBody.length : 0)); 205 return responseBody; </pre> <p><a href="https://android.googlesource.com/platform/packages/apps/Messaging/+nougat-mr1-release/src/android/support/v7/mms/MmsHttpClient.java">https://android.googlesource.com/platform/packages/apps/Messaging/+nougat-mr1-release/src/android/support/v7/mms/MmsHttpClient.java</a></p>

**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC
<p>7[A]. The method of claim 6, further comprising performing, by the first device: receiving user selection of the sub-net;</p>	<p>HTC infringes directly and/or indirectly by performing, inducing others to perform, and/or contributing to the performance of performing, by the first device: receiving user selection of the sub-net. See claims 1 and 6, which are incorporated herein by reference in their entireties.</p> <p>A user of the Accused Products selects a set of second devices assigned to a group to establish a group messaging session for sharing voice, text, photographs, or video via, e.g., Android’s messaging app, Google Hangouts, etc.. For example, the user can select a group name corresponding to the group in order to perform a subsequent action associated with the group such as conducting a conference communication session.</p> <p>Android Messages makes it easy to communicate with anyone by using SMS, MMS, and more. Stay in touch with friends and family, send group texts, and share your favorite pictures, videos, audio messages.</p> <p>• <b>Enhanced features:</b> On supported carriers, you can send messages over Wi-Fi or your data network, see when friends have read your message, and more.</p> <p><a href="https://play.google.com/store/apps/details?id=com.google.android.apps.messaging&amp;hl=en">https://play.google.com/store/apps/details?id=com.google.android.apps.messaging&amp;hl=en</a></p>

## Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products

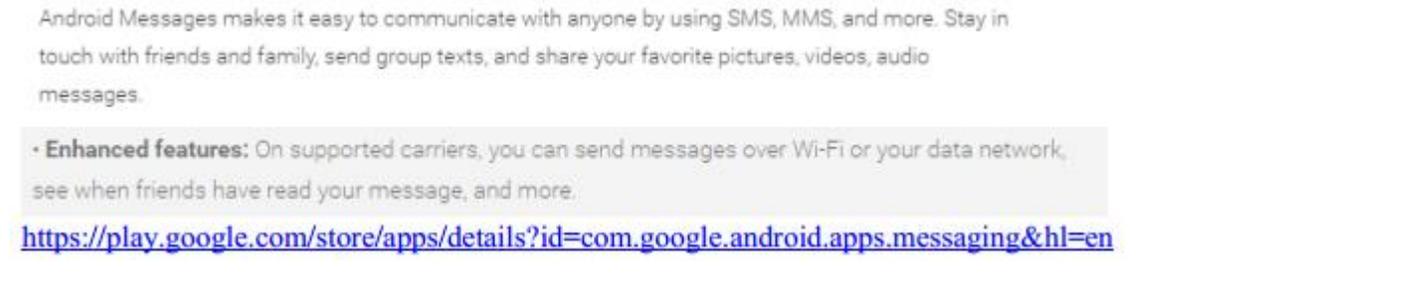
US9408055B2	HTC
	 <p data-bbox="527 1000 1623 1032"><a href="https://play.google.com/store/apps/details?id=com.google.android.apps.messaging&amp;hl=en">https://play.google.com/store/apps/details?id=com.google.android.apps.messaging&amp;hl=en</a></p> <p data-bbox="527 1081 806 1118"><b>Google Hangouts</b></p> <p data-bbox="527 1125 926 1157"><a href="https://hangouts.google.com/">https://hangouts.google.com/</a> ▼</p> <p data-bbox="527 1162 1682 1227">Hangouts bring <b>conversations</b> to life with photos, emoji, and even <b>group</b> video calls for free. Connect with friends across computers, <b>Android</b>, and Apple devices.</p> <p data-bbox="527 1234 772 1263">You visited this page.</p> <p data-bbox="527 1292 1902 1357"><a href="https://www.google.com/search?q=android+group+conversation&amp;oq=android+group+con&amp;aqs=chrome.0.0j69i57j0l4.10109j0j1&amp;sourceid=chrome&amp;ie=UTF-8">https://www.google.com/search?q=android+group+conversation&amp;oq=android+group+con&amp;aqs=chrome.0.0j69i57j0l4.10109j0j1&amp;sourceid=chrome&amp;ie=UTF-8</a></p>

## Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products

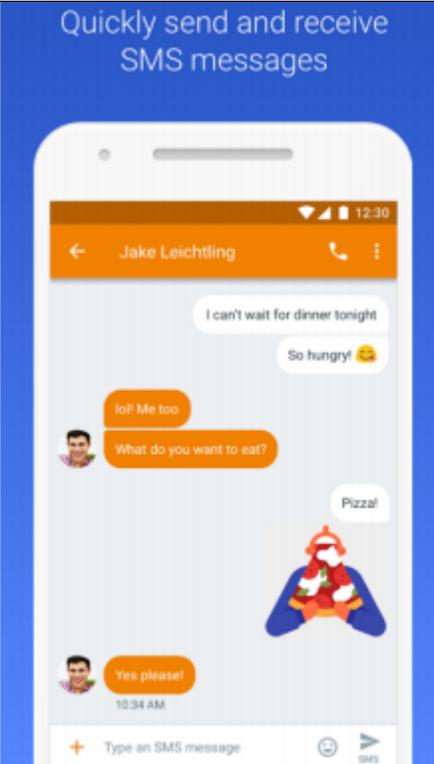
US9408055B2	HTC
	<p data-bbox="541 233 1087 277"><b>Get started with Hangouts</b></p> <p data-bbox="541 298 747 318">You can use Hangouts to:</p> <ul data-bbox="541 339 1134 427" style="list-style-type: none"><li>• Start a chat conversation or video call.</li><li>• Make phone calls using Wi-Fi or data.</li><li>• Send text messages with your <a href="#">Google Voice</a> or <a href="#">Project Fi</a> phone number.</li></ul> <p data-bbox="541 449 1461 496">Hangouts sync automatically across devices. If you start a Hangout on your computer, you can continue your chat on another device, like your phone.</p> <p data-bbox="516 508 1512 540"><a href="https://support.google.com/hangouts/answer/2944865?hl=en&amp;ref_topic=6386410">https://support.google.com/hangouts/answer/2944865?hl=en&amp;ref_topic=6386410</a></p> <p data-bbox="533 552 835 596"><b>Start a Hangout</b></p> <p data-bbox="533 610 1054 630">You can send and receive messages with one person or multiple people.</p> <p data-bbox="541 691 888 711">COMPUTER <b>ANDROID</b> IPHONE &amp; IPAD</p> <hr/> <p data-bbox="533 842 804 870"><b>Start a conversation</b></p> <ol data-bbox="541 889 1134 1027" style="list-style-type: none"><li>1. On your Android phone or tablet, open the Hangouts app .</li><li>2. At the bottom right, tap Add  &gt; New Conversation .</li><li>3. Type and select a person's name.</li><li>4. Enter your message. You can also add emojis, photos, your location, or a sticker.</li><li>5. Tap Send .</li></ol> <p data-bbox="516 1044 1751 1102"><a href="https://support.google.com/hangouts/answer/3115553?hl=en&amp;ref_topic=6386410&amp;co=GENIE.Platform%3DAndroid&amp;oco=0">https://support.google.com/hangouts/answer/3115553?hl=en&amp;ref_topic=6386410&amp;co=GENIE.Platform%3DAndroid&amp;oco=0</a></p> <ul data-bbox="554 1151 1566 1305" style="list-style-type: none"><li>• Connect your Google Voice account for phone calling, SMS texting, and voicemail integration.</li><li>• Keep in touch with contacts across Android, iOS, and the web, and sync chats across all your devices.</li><li>• Message contacts anytime, even if they're offline.</li></ul> <p data-bbox="516 1338 1400 1365"><a href="https://play.google.com/store/apps/details?id=com.google.android.talk&amp;hl=en">https://play.google.com/store/apps/details?id=com.google.android.talk&amp;hl=en</a></p>



**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC
<p>[7B] and establishing a conference among the one or more second devices of the sub-net for sharing voice, text, photographs, or video communications.</p>	<p>HTC infringes directly and/or indirectly by performing, inducing others to perform, and/or contributing to the performance of [] establishing a conference among the one or more second devices of the sub-net for sharing voice, text, photographs, or video communications. See claims 1, 6, and 7[A], which are incorporated herein by reference in their entireties.</p> <p>A user of the Accused Products selects a set of second devices assigned to a group to establish a group messaging session for sharing voice, text, photographs, or video via, e.g., Android’s messaging app, Google Hangouts, etc.. For example, the user can select a group name corresponding to the group in order to perform a subsequent action associated with the group such as conducting a conference communication session.</p>  <p>The screenshot shows the Android Messages app interface. At the top, it says "Android Messages makes it easy to communicate with anyone by using SMS, MMS, and more. Stay in touch with friends and family, send group texts, and share your favorite pictures, videos, audio messages." Below this, there is a section titled "Enhanced features" which lists: "On supported carriers, you can send messages over Wi-Fi or your data network, see when friends have read your message, and more." At the bottom of the screenshot, there is a blue hyperlink: <a href="https://play.google.com/store/apps/details?id=com.google.android.apps.messaging&amp;hl=en">https://play.google.com/store/apps/details?id=com.google.android.apps.messaging&amp;hl=en</a></p>

## Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products

US9408055B2	HTC
	 <p data-bbox="527 998 1623 1031"><a href="https://play.google.com/store/apps/details?id=com.google.android.apps.messaging&amp;hl=en">https://play.google.com/store/apps/details?id=com.google.android.apps.messaging&amp;hl=en</a></p> <p data-bbox="527 1079 806 1117"><b>Google Hangouts</b></p> <p data-bbox="527 1123 926 1156"><a href="https://hangouts.google.com/">https://hangouts.google.com/</a> ▼</p> <p data-bbox="527 1161 1680 1226">Hangouts bring <b>conversations</b> to life with photos, emoji, and even <b>group</b> video calls for free. Connect with friends across computers, <b>Android</b>, and Apple devices.</p> <p data-bbox="527 1230 772 1263">You visited this page.</p> <p data-bbox="527 1291 1902 1356"><a href="https://www.google.com/search?q=android+group+conversation&amp;oq=android+group+con&amp;aqs=chrome.0.0j69i57j0l4.10109j0j1&amp;sourceid=chrome&amp;ie=UTF-8">https://www.google.com/search?q=android+group+conversation&amp;oq=android+group+con&amp;aqs=chrome.0.0j69i57j0l4.10109j0j1&amp;sourceid=chrome&amp;ie=UTF-8</a></p>

## Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products

US9408055B2	HTC
	<p data-bbox="541 235 1087 284"><b>Get started with Hangouts</b></p> <p data-bbox="541 300 745 321">You can use Hangouts to:</p> <ul data-bbox="541 341 1134 430" style="list-style-type: none"><li>• Start a chat conversation or video call.</li><li>• Make phone calls using Wi-Fi or data.</li><li>• Send text messages with your <a href="#">Google Voice</a> or <a href="#">Project Fi</a> phone number.</li></ul> <p data-bbox="541 451 1459 500">Hangouts sync automatically across devices. If you start a Hangout on your computer, you can continue your chat on another device, like your phone.</p> <p data-bbox="514 511 1512 544"><a href="https://support.google.com/hangouts/answer/2944865?hl=en&amp;ref_topic=6386410">https://support.google.com/hangouts/answer/2944865?hl=en&amp;ref_topic=6386410</a></p> <p data-bbox="531 552 835 597"><b>Start a Hangout</b></p> <p data-bbox="531 609 1050 630">You can send and receive messages with one person or multiple people.</p> <p data-bbox="541 690 892 714">COMPUTER <b>ANDROID</b> IPHONE &amp; IPAD</p> <hr/> <p data-bbox="531 844 808 873"><b>Start a conversation</b></p> <ol data-bbox="541 889 1134 1031" style="list-style-type: none"><li>1. On your Android phone or tablet, open the Hangouts app .</li><li>2. At the bottom right, tap Add  &gt; New Conversation .</li><li>3. Type and select a person's name.</li><li>4. Enter your message. You can also add emojis, photos, your location, or a sticker.</li><li>5. Tap Send .</li></ol> <p data-bbox="514 1047 1753 1104"><a href="https://support.google.com/hangouts/answer/3115553?hl=en&amp;ref_topic=6386410&amp;co=GENIE.Platform%3DAndroid&amp;oco=0">https://support.google.com/hangouts/answer/3115553?hl=en&amp;ref_topic=6386410&amp;co=GENIE.Platform%3DAndroid&amp;oco=0</a></p> <ul data-bbox="550 1149 1564 1307" style="list-style-type: none"><li>• Connect your Google Voice account for phone calling, SMS texting, and voicemail integration.</li><li>• Keep in touch with contacts across Android, iOS, and the web, and sync chats across all your devices.</li><li>• Message contacts anytime, even if they're offline.</li></ul> <p data-bbox="514 1339 1396 1372"><a href="https://play.google.com/store/apps/details?id=com.google.android.talk&amp;hl=en">https://play.google.com/store/apps/details?id=com.google.android.talk&amp;hl=en</a></p>

**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC
<p>8. The method of claim 1, wherein the first device is a cellular phone or a personal digital assistant (PDA).</p>	<p>HTC infringes directly and/or indirectly by performing, inducing others to perform, and/or contributing to the performance of wherein the first device is a cellular phone or a personal digital assistant (PDA). See claim 1, which is incorporated herein by reference in its entirety.</p> <p>Upon information and belief, the Accused Products are forms of cellular devices or PDAs in that the functionality of a PDA has been subsumed into smartphones, tablets, and portable media players having the functionalities of a PDA that include cellular transceivers to enable cellular communications. To the extent that it is necessary, AGIS submits that the Accused Products meet the claim limitation “the first device is a personal digital assistant (PDA) or a cellular phone” under the doctrine of equivalents. For example, U.S. Cellular which is one of the largest mobile cellular service providers in the United States and categorizes HTC Android phones as smartphones for which U.S. Cellular provides cellular services.</p> <p><a href="https://www.uscellular.com/uscellular/cell-phones/showPhoneDetails.jsp?productId=prod1060051">https://www.uscellular.com/uscellular/cell-phones/showPhoneDetails.jsp?productId=prod1060051</a></p>
<p>9. The method of claim 1, further comprising performing, by the first device: identifying second user interaction with the interactive display selecting at least one of the user-selectable symbols corresponding to at least one of the second devices and user interaction with the display specifying an action and, based thereon, initiating a phone call or phone</p>	<p>HTC infringes directly and/or indirectly by performing, inducing others to perform, and/or contributing to the performance of performing, by the first device: identifying second user interaction with the interactive display selecting at least one of the user-selectable symbols corresponding to at least one of the second devices and user interaction with the display specifying an action and, based thereon, initiating a phone call or phone conference with the at least one second device. See claims 1 and 6-7, which are incorporated herein by reference in their entirety.</p> <p>For example, a user may choose a symbol representing a device and then choose a corresponding action. Using Google Maps, a user can select a device’s symbol and then effect an action to that device by specifying some action using the interface. Data is sent to the device based on that action.</p>

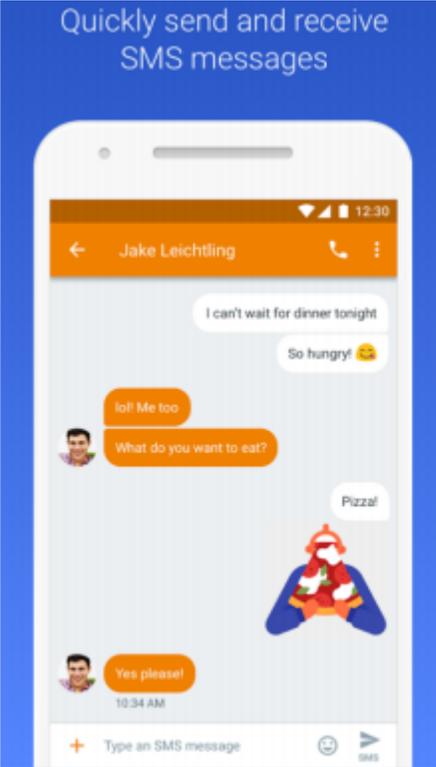
## Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products

US9408055B2	HTC
<p>conference with the at least one second device.</p>	<p><b>Group contacts</b></p> <p>You can organize the people and businesses in Contacts using labels.</p> <p><b>Create a group</b></p> <ol style="list-style-type: none"><li>1. Go to <a href="#">Google Contacts</a>.</li><li>2. On the left under "Labels," click <b>Create label</b>. (If you don't see "Labels," go to <a href="#">group contacts in old Contacts</a>.)</li><li>3. Type a name, then click <b>OK</b>.</li></ol> <hr/> <p><b>Add contacts to a group label</b></p> <ol style="list-style-type: none"><li>1. To select contacts, check the boxes next to their names.</li><li>2. In the top right, click <b>Label</b>. (If you don't see <b>Label</b>, go to <a href="#">group contacts in old Contacts</a>.)</li><li>3. Choose the groups you want to add the contacts to. You'll see a checkmark appear next to the groups you chose.</li></ol> <p><b>Google Hangouts</b></p> <p><a href="https://hangouts.google.com/">https://hangouts.google.com/</a></p> <p>Hangouts bring <b>conversations</b> to life with photos, emoji, and even <b>group</b> video calls for free. Connect with friends across computers, <b>Android</b>, and Apple devices.</p> <p>You visited this page.</p> <p><a href="https://www.google.com/search?q=android+group+conversation&amp;oq=android+group+con&amp;aqs=chrome.0.0j69i57j0l4.10109j0j1&amp;sourceid=chrome&amp;ie=UTF-8">https://www.google.com/search?q=android+group+conversation&amp;oq=android+group+con&amp;aqs=chrome.0.0j69i57j0l4.10109j0j1&amp;sourceid=chrome&amp;ie=UTF-8</a></p>

**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC
	<p>1 Phone the first person.</p> <p>2 After the call connects and you complete a few pleasantries, touch the Add Call icon.</p>  <p>The Add Call icon is shown. After touching that icon, or a similar icon, the first person is put on hold.</p> <p>3 Dial the second person.</p> <p>You can use the dialpad or choose the second person from the phone's address book or the recent calls log.</p> <p>Say your pleasantries and inform the party that the call is about to be merged.</p> <p>4 Touch the Merge or Merge Calls icon.</p>  <p>The two calls are now joined: The touchscreen says <i>Conference Call</i>, and the End Last Call icon appears. Everyone you've dialed can talk to and hear everyone else.</p> <p>5 Touch the End Call icon to end the conference call.</p> <p>All calls are disconnected.</p> <p><a href="http://www.dummies.com/consumer-electronics/smartphones/droid/how-to-make-a-conference-call-on-an-android-phone/">http://www.dummies.com/consumer-electronics/smartphones/droid/how-to-make-a-conference-call-on-an-android-phone/</a></p>
<p>10. The method of claim 3, wherein the video comprises a video clip or a video transmission.</p>	<p>HTC infringes directly and/or indirectly by performing, inducing others to perform, and/or contributing to the performance of wherein the video comprises a video clip or a video transmission. See claim 3, which is incorporated herein by reference in its entirety.</p>

## Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products

US9408055B2	HTC
	<p>Android Messages makes it easy to communicate with anyone by using SMS, MMS, and more. Stay in touch with friends and family, send group texts, and share your favorite pictures, videos, audio messages.</p> <p><b>Enhanced features:</b> On supported carriers, you can send messages over Wi-Fi or your data network, see when friends have read your message, and more.</p> <p><a href="https://play.google.com/store/apps/details?id=com.google.android.apps.messaging&amp;hl=en">https://play.google.com/store/apps/details?id=com.google.android.apps.messaging&amp;hl=en</a></p>  <p><a href="https://play.google.com/store/apps/details?id=com.google.android.apps.messaging&amp;hl=en">https://play.google.com/store/apps/details?id=com.google.android.apps.messaging&amp;hl=en</a></p>

**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

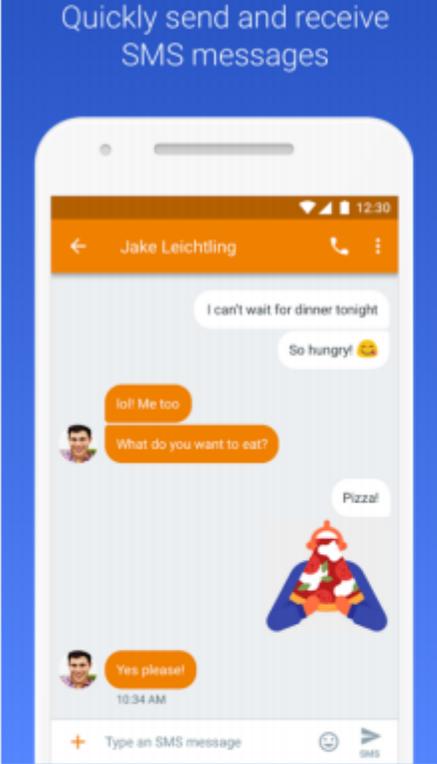
US9408055B2	HTC
	<p><b>Google Hangouts</b>  <a href="https://hangouts.google.com/">https://hangouts.google.com/</a> ▼                      Hangouts bring <b>conversations</b> to life with photos, emoji, and even <b>group</b> video calls for free. Connect with friends across computers, <b>Android</b>, and Apple devices.                      You visited this page.</p> <p><a href="https://www.google.com/search?q=android+group+conversation&amp;oq=android+group+con&amp;aqs=chrome.0.0j69i57j0l4.10109j0j1&amp;sourceid=chrome&amp;ie=UTF-8">https://www.google.com/search?q=android+group+conversation&amp;oq=android+group+con&amp;aqs=chrome.0.0j69i57j0l4.10109j0j1&amp;sourceid=chrome&amp;ie=UTF-8</a></p> <ol style="list-style-type: none"> <li>1. Wake your Android device and unlock the screen.</li> <li>2. Tap the "Application Menu" button at the bottom of the screen. Then tap the "Camera" icon.</li> <li>3. Move the camera function slider at the bottom of the window to the "Video Camera" option.</li> <li>4. Tap the "Resolution" icon at the lower left corner of the viewing window. Then tap the option that corresponds with how you wish to share the video. For example, if you wanted to send the video as an MMS message, tap the "MMS" option.</li> </ol> <p><a href="http://smallbusiness.chron.com/shrink-videos-big-android-28400.html">http://smallbusiness.chron.com/shrink-videos-big-android-28400.html</a></p>
<p>11[A]. The method of claim 6, further comprising performing, by the first device: receiving user selection of the sub-net;</p>	<p>HTC infringes directly and/or indirectly by performing, inducing others to perform, and/or contributing to the performance of performing, by the first device: receiving user selection of the sub-net. See claims 6 and 7, which are incorporated herein by reference in their entirety.</p>
<p>[11B] and causing the one or more second devices of the sub-net to place a call, make a verbal announcement, convert text to speech,</p>	<p>HTC infringes directly and/or indirectly by performing, inducing others to perform, and/or contributing to the performance of [] causing the one or more second devices of the sub-net to place a call, make a verbal announcement, convert text to speech, vibrate, or increase sound levels. See claims 3, 6, 7[A], 9, 11[A], which are incorporated herein by reference in their entirety.</p>



**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC
<p>vibrate, or increase sound levels.</p>	
<p>12. The method of claim 1, wherein first device sends the SMS messages and receives the IP-based responses via a cellular communications network.</p>	<p>HTC infringes directly and/or indirectly by performing, inducing others to perform, and/or contributing to the performance of wherein first device sends the SMS messages and receives the IP-based responses via a cellular communications network. See claims 1 and 8, which are incorporated herein by reference in their entirety.</p> <p>For example, the Accused Products are known to use IP-based communication over wireless or data connections. Both Android Messages and Google Hangouts utilize SMS messages, including group messages from one device to several devices, to send an SMS message, with additional information, to a contact. U.S. Cellular which is one of the largest mobile cellular service providers in the United States and categorizes HTC Android phones as smartphones for which U.S. Cellular provides cellular communications services to enable the IP-based communications over its cellular network.</p> <p><a href="https://www.uscellular.com/uscellular/cell-phones/showPhoneDetails.jsp?productId=prod1060051">https://www.uscellular.com/uscellular/cell-phones/showPhoneDetails.jsp?productId=prod1060051</a></p> <p>Android Messages makes it easy to communicate with anyone by using SMS, MMS, and more. Stay in touch with friends and family, send group texts, and share your favorite pictures, videos, audio messages.</p> <p>• <b>Enhanced features:</b> On supported carriers, you can send messages over Wi-Fi or your data network, see when friends have read your message, and more.</p> <p><a href="https://play.google.com/store/apps/details?id=com.google.android.apps.messaging&amp;hl=en">https://play.google.com/store/apps/details?id=com.google.android.apps.messaging&amp;hl=en</a></p>

## Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products

US9408055B2	HTC
	 <p data-bbox="520 1003 1619 1032"><a href="https://play.google.com/store/apps/details?id=com.google.android.apps.messaging&amp;hl=en">https://play.google.com/store/apps/details?id=com.google.android.apps.messaging&amp;hl=en</a></p> <h3 data-bbox="541 1089 1087 1133">Get started with Hangouts</h3> <p data-bbox="541 1154 751 1174">You can use Hangouts to:</p> <ul data-bbox="541 1198 1136 1284" style="list-style-type: none"><li>• Start a chat conversation or video call.</li><li>• Make phone calls using Wi-Fi or data.</li><li>• Send text messages with your <a href="#">Google Voice</a> or <a href="#">Project Fi</a> phone number.</li></ul> <p data-bbox="541 1308 1465 1354">Hangouts sync automatically across devices. If you start a Hangout on your computer, you can continue your chat on another device, like your phone.</p> <p data-bbox="520 1365 1514 1393"><a href="https://support.google.com/hangouts/answer/2944865?hl=en&amp;ref_topic=6386410">https://support.google.com/hangouts/answer/2944865?hl=en&amp;ref_topic=6386410</a></p>

**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC
<p>13. The method of claim 1, further comprising performing, by a first device: identifying second user interaction with the interactive display selecting a particular user-selectable symbol corresponding to a particular second device and user interaction with the display specifying an action and, based thereon, initiating a data call with the particular second device.</p>	<p>HTC infringes directly and/or indirectly by performing, inducing others to perform, and/or contributing to the performance of performing, by a first device: identifying second user interaction with the interactive display selecting a particular user-selectable symbol corresponding to a particular second device and user interaction with the display specifying an action and, based thereon, initiating a data call with the particular second device. See claims 1[F], 9, and 12, which are incorporated herein by reference in their entirety.</p>
<p>14. The method of claim 1, wherein the user input further specifies information associated with the entity, and wherein the method further comprises performing, by the first device: based on the user</p>	<p>HTC infringes directly and/or indirectly by performing, inducing others to perform, and/or contributing to the performance of wherein the user input further specifies information associated with the entity, and wherein the method further comprises performing, by the first device: based on the user input, transmitting the user-specified information associated with the entity to the second devices.</p> <p>A user shares a location corresponding to an entity (e.g. restaurant) that is not the device location. A user can search for or navigate to a location and then specify the location with a symbol corresponding to the entity. The user can also send information associated with the entity via the Android app as a comment in an MMS message.</p>

**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC
<p>input, transmitting the user-specified information associated with the entity to the second devices.</p>	<p><b>Share a map or location</b></p> <ol style="list-style-type: none"> <li>1. Open the Google Maps app .</li> <li>2. Search for a place. Or find a place on the map then touch and hold to drop a pin.</li> <li>3. At the bottom, tap the place's name or address.</li> <li>4. Tap Share . If you don't see this, tap More  &gt; <b>Share</b>.</li> <li>5. Select an app. It'll send a link that shows the place in Google Maps.</li> </ol> <p><a href="https://support.google.com/maps/answer/144361?co=GENIE.Platform%3DAndroid&amp;hl=en">https://support.google.com/maps/answer/144361?co=GENIE.Platform%3DAndroid&amp;hl=en</a></p>
<p>15. The method of claim 14, further comprising performing, by the first device: identifying user interaction with the interactive display selecting the symbol corresponding to the entity and, based, thereon, displaying the information associated with the entity.</p>	<p>HTC infringes directly and/or indirectly by performing, inducing others to perform, and/or contributing to the performance of performing, by the first device: identifying user interaction with the interactive display selecting the symbol corresponding to the entity and, based, thereon, displaying the information associated with the entity. See claims 1[G] and 14, which are incorporated herein by reference in their entirety.</p> <p>Using Google Maps, a user can interact with the display to specify a location that does not correspond to the first or second devices. Alternatively, a user can share an ETA which includes a route, where the route is not the location of any of the first or second devices.</p>

**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC
	<p><b>Embed a map or share a location</b></p> <p>On your computer, you can embed a map, Street View image, driving directions, or search into your website or blog. On your computer, phone, or tablet, you can share a location or map with others over email, Google+, Facebook, Twitter, or text.</p> <p><a href="#">ANDROID</a> <a href="#">COMPUTER</a> <a href="#">IPHONE &amp; IPAD</a></p> <hr/> <p><b>Share a map or location</b></p> <ol style="list-style-type: none"> <li>1. Open the Google Maps app .</li> <li>2. Search for a place. Or find a place on the map then touch and hold to drop a pin.</li> <li>3. At the bottom, tap the place's name or address.</li> <li>4. Tap Share . If you don't see this, tap More  &gt; <b>Share</b>.</li> <li>5. Select an app. It'll send a link that shows the place in Google Maps.</li> </ol> <p><a href="https://support.google.com/maps/answer/144361?co=GENIE.Platform%3DAndroid&amp;hl=en">https://support.google.com/maps/answer/144361?co=GENIE.Platform%3DAndroid&amp;hl=en</a></p> <p><b>Share your E.T.A</b></p> <p>After you start your drive, you can share your destination, estimated arrival time, and where you are on the route.</p> <ol style="list-style-type: none"> <li>1. Open the Google Maps app .</li> <li>2. Set a driving destination. <a href="#">Learn how to navigate to a place.</a></li> <li>3. After you start navigation, tap <b>More</b>  &gt; <b>Share trip progress</b>.</li> <li>4. Choose a person from the list.</li> <li>5. Tap <b>Share</b>.</li> <li>6. Location Sharing will stop when you reach your destination or stop navigating.</li> </ol> <p>• To stop sharing before you arrive, tap <b>More</b>  &gt; <b>Stop sharing</b>.</p> <p><a href="https://support.google.com/maps/answer/7326816?co=GENIE.Platform%3DAndroid&amp;hl=en">https://support.google.com/maps/answer/7326816?co=GENIE.Platform%3DAndroid&amp;hl=en</a></p>
<p>16. The method of claim 15, wherein the</p>	<p>HTC infringes directly and/or indirectly by performing, inducing others to perform, and/or contributing to the performance of wherein the information comprises a category of the entity. See claims 13-15, which are</p>

**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC
<p>information comprises a category of the entity.</p>	<p>incorporated herein by reference in their entirety.</p> <p>For example, the Google Maps app identifies restaurants that are categorized as providing breakfast, lunch, coffee, dinner, drinks, etc.</p>
<p>17. The method of claim 15, wherein the first device uses an Internet Protocol to transmit the user-specified symbol, location, and information associated with the entity.</p>	<p>HTC infringes directly and/or indirectly by performing, inducing others to perform, and/or contributing to the performance of wherein the first device uses an Internet Protocol to transmit the user-specified symbol, location, and information associated with the entity. See claims 1, 4, 12, and 15, which are incorporated herein by reference in their entirety.</p>

**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC
	<p><b>Embed a map or share a location</b></p> <p>On your computer, you can embed a map, Street View image, driving directions, or search into your website or blog. On your computer, phone, or tablet, you can share a location or map with others over email, Google+, Facebook, Twitter, or text.</p> <p><a href="#">ANDROID</a> <a href="#">COMPUTER</a> <a href="#">IPHONE &amp; IPAD</a></p> <hr/> <p><b>Share a map or location</b></p> <ol style="list-style-type: none"> <li>1. Open the Google Maps app .</li> <li>2. Search for a place. Or find a place on the map then touch and hold to drop a pin.</li> <li>3. At the bottom, tap the place's name or address.</li> <li>4. Tap Share . If you don't see this, tap More  &gt; <b>Share</b>.</li> <li>5. Select an app. It'll send a link that shows the place in Google Maps.</li> </ol> <p><a href="https://support.google.com/maps/answer/144361?co=GENIE.Platform%3DAndroid&amp;hl=en">https://support.google.com/maps/answer/144361?co=GENIE.Platform%3DAndroid&amp;hl=en</a></p> <p><b>Share your E.T.A</b></p> <p>After you start your drive, you can share your destination, estimated arrival time, and where you are on the route.</p> <ol style="list-style-type: none"> <li>1. Open the Google Maps app .</li> <li>2. Set a driving destination. <a href="#">Learn how to navigate to a place.</a></li> <li>3. After you start navigation, tap <b>More</b>  &gt; <b>Share trip progress</b>.</li> <li>4. Choose a person from the list.</li> <li>5. Tap <b>Share</b>.</li> <li>6. Location Sharing will stop when you reach your destination or stop navigating.</li> </ol> <p>• To stop sharing before you arrive, tap <b>More</b>  &gt; <b>Stop sharing</b>.</p> <p><a href="https://support.google.com/maps/answer/7326816?co=GENIE.Platform%3DAndroid&amp;hl=en">https://support.google.com/maps/answer/7326816?co=GENIE.Platform%3DAndroid&amp;hl=en</a></p>
<p>18. The method of claim 1, wherein the entity comprises a</p>	<p>HTC infringes directly and/or indirectly by performing, inducing others to perform, and/or contributing to the performance of wherein the entity comprises a vehicle, a person, an event, a site, a building, or a facility.</p>

**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC
<p>vehicle, a person, an event, a site, a building, or a facility.</p>	<p>For example, Google Maps allows a user to mark a location on its map and associate with a vehicle, person, event, site, building, or facility. For example, restaurants and train stations can be considered a site, a building, and a facility.</p> <p><b>Embed a map or share a location</b></p> <p>On your computer, you can embed a map, Street View image, driving directions, or search into your website or blog. On your computer, phone, or tablet, you can share a location or map with others over email, Google+, Facebook, Twitter, or text.</p> <p><b>ANDROID</b>    COMPUTER    IPHONE &amp; IPAD</p> <hr/> <p><b>Share a map or location</b></p> <ol style="list-style-type: none"> <li>1. Open the Google Maps app .</li> <li>2. Search for a place. Or find a place on the map then touch and hold to drop a pin.</li> <li>3. At the bottom, tap the place's name or address.</li> <li>4. Tap Share . If you don't see this, tap More  &gt; <b>Share</b>.</li> <li>5. Select an app. It'll send a link that shows the place in Google Maps.</li> </ol> <p><a href="https://support.google.com/maps/answer/144361?co=GENIE.Platform%3DAndroid&amp;hl=en">https://support.google.com/maps/answer/144361?co=GENIE.Platform%3DAndroid&amp;hl=en</a></p> <p><b>Share your E.T.A</b></p> <p>After you start your drive, you can share your destination, estimated arrival time, and where you are on the route.</p> <ol style="list-style-type: none"> <li>1. Open the Google Maps app .</li> <li>2. Set a driving destination. <a href="#">Learn how to navigate to a place.</a></li> <li>3. After you start navigation, tap <b>More</b>  &gt; <b>Share trip progress</b>.</li> <li>4. Choose a person from the list.</li> <li>5. Tap <b>Share</b>.</li> <li>6. Location Sharing will stop when you reach your destination or stop navigating.</li> </ol> <ul style="list-style-type: none"> <li>• To stop sharing before you arrive, tap <b>More</b>  &gt; <b>Stop sharing</b>.</li> </ul> <p><a href="https://support.google.com/maps/answer/7326816?co=GENIE.Platform%3DAndroid&amp;hl=en">https://support.google.com/maps/answer/7326816?co=GENIE.Platform%3DAndroid&amp;hl=en</a></p>



**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC
<p>19[A]. The method of claim 1, wherein the user-specified symbol is a first user-specified symbol, wherein the user-specified location is a first-user specified location, wherein the entity is a first entity, and wherein the method further comprises performing, by the first device: receiving user-specified information transmitted by a particular second device, the user-specified information including a second user-specified location and a second user-specified symbol corresponding to a second entity other than the first device and the second devices;</p>	<p>HTC infringes directly and/or indirectly by performing, inducing others to perform, and/or contributing to the performance of wherein the user-specified symbol is a first user-specified symbol, wherein the user-specified location is a first-user specified location, wherein the entity is a first entity, and wherein the method further comprises performing, by the first device: receiving user-specified information transmitted by a particular second device, the user-specified information including a second user-specified location and a second user-specified symbol corresponding to a second entity other than the first device and the second devices. See claims 1 and 13-15, which are incorporated herein by reference in their entirety.</p> <p>Using Google Maps, a user can interact with the display to specify a location that does not correspond to the first or second devices. A user can drop a symbol pin on the specified location. A user can then share that location and transmit the location to one or more second devices using Android Messages, Google Hangouts, or another application.</p> <p>Alternatively, a user can share an ETA which includes a route, where the route is not the location of any of the first or second devices. Again, this route can be shared with users over Android Messages, Google Hangouts, or another application.</p>

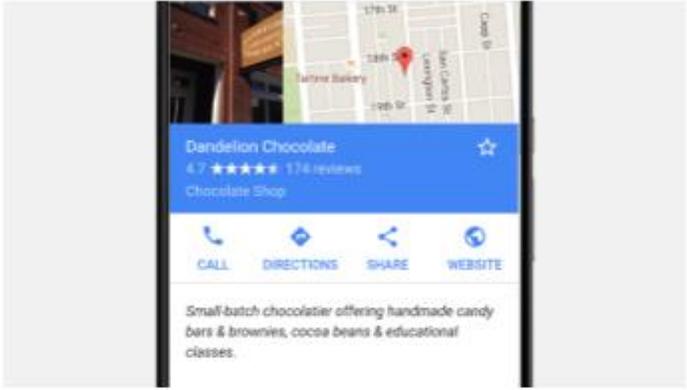
**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC
	<p><b>Embed a map or share a location</b></p> <p>On your computer, you can embed a map, Street View image, driving directions, or search into your website or blog. On your computer, phone, or tablet, you can share a location or map with others over email, Google+, Facebook, Twitter, or text.</p> <p><b>ANDROID</b>    COMPUTER    IPHONE &amp; IPAD</p> <hr/> <p><b>Share a map or location</b></p> <ol style="list-style-type: none"> <li>1. Open the Google Maps app .</li> <li>2. Search for a place. Or find a place on the map then touch and hold to drop a pin.</li> <li>3. At the bottom, tap the place's name or address.</li> <li>4. Tap Share . If you don't see this, tap More  &gt; <b>Share</b>.</li> <li>5. Select an app. It'll send a link that shows the place in Google Maps.</li> </ol> <p><a href="https://support.google.com/maps/answer/144361?co=GENIE.Platform%3DAndroid&amp;hl=en">https://support.google.com/maps/answer/144361?co=GENIE.Platform%3DAndroid&amp;hl=en</a></p> <p><b>Share your E.T.A</b></p> <p>After you start your drive, you can share your destination, estimated arrival time, and where you are on the route.</p> <ol style="list-style-type: none"> <li>1. Open the Google Maps app .</li> <li>2. Set a driving destination. <a href="#">Learn how to navigate to a place.</a></li> <li>3. After you start navigation, tap <b>More</b>  &gt; <b>Share trip progress</b>.</li> <li>4. Choose a person from the list.</li> <li>5. Tap <b>Share</b>.</li> <li>6. Location Sharing will stop when you reach your destination or stop navigating.</li> </ol> <ul style="list-style-type: none"> <li>• To stop sharing before you arrive, tap <b>More</b>  &gt; <b>Stop sharing</b>.</li> </ul> <p><a href="https://support.google.com/maps/answer/7326816?co=GENIE.Platform%3DAndroid&amp;hl=en">https://support.google.com/maps/answer/7326816?co=GENIE.Platform%3DAndroid&amp;hl=en</a></p>
[19B] and adding the second user-specified	HTC infringes directly and/or indirectly by performing, inducing others to perform, and/or contributing to the performance of [] adding the second user-specified symbol to the interactive display at a position on the

**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC
<p>symbol to the interactive display at a position on the interactive map corresponding to the second user-specified location.</p>	<p>interactive map corresponding to the second user-specified location. See claim 19[A], which is incorporated herein by reference in its entirety.</p> <p>Upon information and belief, when a user received shared map or location, the user’s device adds the received shared map or location on its interactive map.</p>
<p>20. The method of claim 19, wherein the user-specified information further includes information associated with the second entity, and wherein the method further comprises performing, by the first device: identifying user interaction with the interactive display selecting the second user-specified symbol corresponding to the second entity and, based, thereon, displaying the information associated with the second entity.</p>	<p>HTC infringes directly and/or indirectly by performing, inducing others to perform, and/or contributing to the performance of wherein the user-specified information further includes information associated with the second entity, and wherein the method further comprises performing, by the first device: identifying user interaction with the interactive display selecting the second user-specified symbol corresponding to the second entity and, based, thereon, displaying the information associated with the second entity. See claims 2, and 14-15, which are incorporated herein by reference in their entirety.</p> <p>The Accused Devices allow users to receive information associated with fixed locations such as restaurants, stadiums, transportation stations (e.g., bus and train stations), etc.</p>

## Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products

US9408055B2	HTC
	 <p>The screenshot shows a mobile application interface for a location named 'Dandelion Chocolate'. At the top, there is a map snippet with a red location pin. Below the map, the name 'Dandelion Chocolate' is displayed in a blue header, followed by a 4.7-star rating and '174 reviews'. Underneath, it says 'Chocolate Shop'. A row of four icons is visible: a phone icon for 'CALL', a location pin icon for 'DIRECTIONS', a share icon for 'SHARE', and a globe icon for 'WEBSITE'. At the bottom, a short description reads: 'Small batch chocolatier offering handmade candy bars &amp; brownies, cocoa beans &amp; educational classes.'</p> <p>Place details</p> <p>Retrieve rich details about a place, including name, address, phone number, website link and more.</p> <p><a href="https://developers.google.com/places/android-api/">https://developers.google.com/places/android-api/</a></p>

## Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products

US9408055B2	HTC
	<p><b>Business phone numbers</b></p> <p>Google is fantastic for tracking down business phone numbers.</p> <p>You can accomplish this in a number of ways, including:</p> <ul style="list-style-type: none"> <li>▪ <b>type of business plus zip code:</b> Perhaps you don't know the name of the business you're looking for, but you have something in mind. Type in the business genre, for example, "pizza restaurant", then the zip code. Google will return local listings that include maps, reviews, and contact information (phone numbers, addresses, <a href="#">website URLs</a>, even email addresses if available).</li> <li>▪ <b>type of business plus city:</b> Just like in the previous example, except you can substitute the name of a city for a zip code, i.e., "Seattle doctors".</li> </ul> <p><a href="https://www.lifewire.com/google-phone-number-search-3481892">https://www.lifewire.com/google-phone-number-search-3481892</a></p> <p><b>Embed a map or share a location</b></p> <p>On your computer, you can embed a map, Street View image, driving directions, or search into your website or blog. On your computer, phone, or tablet, you can share a location or map with others over email, Google+, Facebook, Twitter, or text.</p> <p><a href="#">ANDROID</a>   <a href="#">COMPUTER</a>   <a href="#">IPHONE &amp; IPAD</a></p> <hr/> <p><b>Share a map or location</b></p> <ol style="list-style-type: none"> <li>1. Open the Google Maps app .</li> <li>2. Search for a place. Or find a place on the map then touch and hold to drop a pin.</li> <li>3. At the bottom, tap the place's name or address.</li> <li>4. Tap Share . If you don't see this, tap More  &gt; <b>Share</b>.</li> <li>5. Select an app. It'll send a link that shows the place in Google Maps.</li> </ol> <p><a href="https://support.google.com/maps/answer/144361?co=GENIE.Platform%3DAndroid&amp;hl=en">https://support.google.com/maps/answer/144361?co=GENIE.Platform%3DAndroid&amp;hl=en</a></p>

**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC
<p>21. The method of claim 1, further comprising performing, by the first device: presenting a symbol corresponding to a facility selected from the group consisting of a hospital, a police station, and a fire station, wherein the symbol corresponding to the facility is positioned on the interactive map at a position corresponding to a location of the facility.</p>	<p>HTC infringes directly and/or indirectly by performing, inducing others to perform, and/or contributing to the performance of performing, by the first device: presenting a symbol corresponding to a facility selected from the group consisting of a hospital, a police station, and a fire station, wherein the symbol corresponding to the facility is positioned on the interactive map at a position corresponding to a location of the facility. See claim 1, which is incorporated herein by reference in its entirety.</p> <p>Upon information and belief, the Accused Products presents a symbol on the map corresponding to a location of a hospital, a police station, and a fire station in response to users to search for any one of those facilities. For example, using Google Maps, a user can interact with the display to specify a location that does not correspond to the first or second devices.</p>

**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

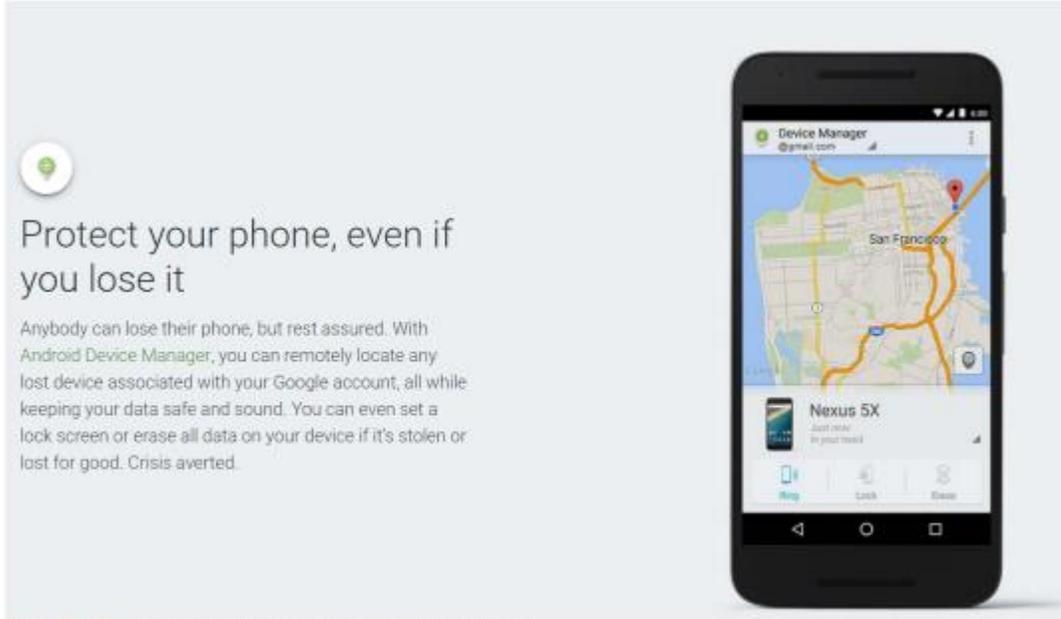
US9408055B2	HTC
	<p><b>Embed a map or share a location</b></p> <p>On your computer, you can embed a map, Street View image, driving directions, or search into your website or blog. On your computer, phone, or tablet, you can share a location or map with others over email, Google+, Facebook, Twitter, or text.</p> <p><a href="#">ANDROID</a> <a href="#">COMPUTER</a> <a href="#">IPHONE &amp; IPAD</a></p> <hr/> <p><b>Share a map or location</b></p> <ol style="list-style-type: none"> <li>1. Open the Google Maps app .</li> <li>2. Search for a place. Or find a place on the map then touch and hold to drop a pin.</li> <li>3. At the bottom, tap the place's name or address.</li> <li>4. Tap Share . If you don't see this, tap More  &gt; <b>Share</b>.</li> <li>5. Select an app. It'll send a link that shows the place in Google Maps.</li> </ol> <p><a href="https://support.google.com/maps/answer/144361?co=GENIE.Platform%3DAndroid&amp;hl=en">https://support.google.com/maps/answer/144361?co=GENIE.Platform%3DAndroid&amp;hl=en</a></p> <p><b>Share your E.T.A</b></p> <p>After you start your drive, you can share your destination, estimated arrival time, and where you are on the route.</p> <ol style="list-style-type: none"> <li>1. Open the Google Maps app .</li> <li>2. Set a driving destination. <a href="#">Learn how to navigate to a place.</a></li> <li>3. After you start navigation, tap <b>More</b>  &gt; <b>Share trip progress</b>.</li> <li>4. Choose a person from the list.</li> <li>5. Tap <b>Share</b>.</li> <li>6. Location Sharing will stop when you reach your destination or stop navigating.</li> </ol> <p>• To stop sharing before you arrive, tap <b>More</b>  &gt; <b>Stop sharing</b>.</p> <p><a href="https://support.google.com/maps/answer/7326816?co=GENIE.Platform%3DAndroid&amp;hl=en">https://support.google.com/maps/answer/7326816?co=GENIE.Platform%3DAndroid&amp;hl=en</a></p>
<p>22. The method of claim 21, further comprising performing, by the</p>	<p>HTC infringes directly and/or indirectly by performing, inducing others to perform, and/or contributing to the performance of performing, by the first device: identifying user interaction with the interactive display selecting the symbol corresponding to the facility and, based, thereon, displaying information associated with the facility. See claims 15-16, which are incorporated herein by reference in their entirety.</p>

**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

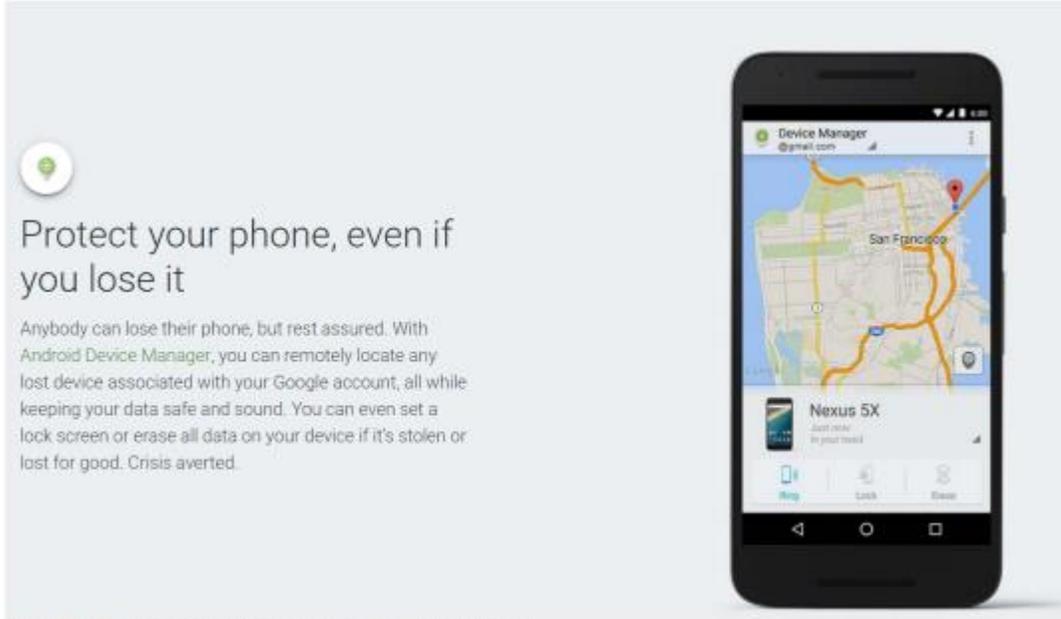
US9408055B2	HTC
<p>first device:                      identifying user                      interaction with the                      interactive display                      selecting the symbol                      corresponding to the                      facility and, based,                      thereon, displaying                      information associated                      with the facility.</p>	
<p>23[A]. The method of                      claim 1, further                      comprising                      performing, by the                      first device: receiving                      a message sent by a                      particular second                      device, wherein the                      message indicates an                      action to be performed                      by the first device;</p>	<p>HTC infringes directly and/or indirectly by performing, inducing others to perform, and/or contributing to the performance of performing, by the first device: receiving a message sent by a particular second device, wherein the message indicates an action to be performed by the first device. See claims 1, 6-7, 9, 12-13, which are incorporated herein by reference in their entirety.</p> <p>For example, a user may choose a symbol representing a device and then choose a corresponding action. Using Google Maps, a user can select a device’s symbol and then effect an action to that device by specifying some action using the interface. Data is sent to the device based on that action. For example, a user can send data causing the second device to: ring an audio alert, display a message and lock the screen, erase its contents</p>



**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC
	 <p data-bbox="533 448 974 532">Protect your phone, even if you lose it</p> <p data-bbox="533 553 989 704">Anybody can lose their phone, but rest assured. With Android Device Manager, you can remotely locate any lost device associated with your Google account, all while keeping your data safe and sound. You can even set a lock screen or erase all data on your device if it's stolen or lost for good. Crisis averted.</p> <p data-bbox="520 857 1045 883"><a href="https://www.android.com/security/overview/">https://www.android.com/security/overview/</a></p>
<p data-bbox="186 1008 474 1068">[23B] and performing the indicated action.</p>	<p data-bbox="510 1008 1866 1109">HTC infringes directly and/or indirectly by performing, inducing others to perform, and/or contributing to the performance of [] performing the indicated action. See claim [23A], which is incorporated herein by reference in its entirety.</p> <p data-bbox="510 1154 1902 1328">For example, a user may choose a symbol representing a device and then choose a corresponding action. Using Android Device Manager, a user can select a device's symbol and then effect an action to that device by specifying some action using the interface. Data is sent to the device based on that action. For example, a user can send data causing the second device to: ring an audio alert, display a message and lock the screen, erase its contents.</p>

**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC
	 <p data-bbox="533 448 974 532">Protect your phone, even if you lose it</p> <p data-bbox="533 553 989 706">Anybody can lose their phone, but rest assured. With Android Device Manager, you can remotely locate any lost device associated with your Google account, all while keeping your data safe and sound. You can even set a lock screen or erase all data on your device if it's stolen or lost for good. Crisis averted.</p> <p data-bbox="520 857 1045 885"><a href="https://www.android.com/security/overview/">https://www.android.com/security/overview/</a></p>
<p>24. The method of claim 23, wherein the indicated action is selected from the group consisting of playing audio, initiating a phone call, vibrating, converting text to speech,</p>	<p>HTC infringes directly and/or indirectly by performing, inducing others to perform, and/or contributing to the performance of wherein the indicated action is selected from the group consisting of playing audio, initiating a phone call, vibrating, converting text to speech, changing sound intensity, and displaying information. See claims 1 and 23, which are incorporated herein by reference in their entirety.</p>

**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC
changing sound intensity, and displaying information.	
25. The method of claim 1, further comprising performing, by the first device: remotely controlling a particular second device to perform an action by sending a message to the second device, wherein the message indicates the action to be performed.	HTC infringes directly and/or indirectly by performing, inducing others to perform, and/or contributing to the performance of performing, by the first device: remotely controlling a particular second device to perform an action by sending a message to the second device, wherein the message indicates the action to be performed.
26. The method of claim 25, wherein the indicated action is selected from the group consisting of playing audio, initiating a phone call, vibrating, converting text to speech, changing sound intensity, and displaying information.	HTC infringes directly and/or indirectly by performing, inducing others to perform, and/or contributing to the performance of wherein the indicated action is selected from the group consisting of playing audio, initiating a phone call, vibrating, converting text to speech, changing sound intensity, and displaying information. See claim 24, which is incorporated herein by reference in its entirety.
27. The method of	HTC infringes directly and/or indirectly by performing, inducing others to perform, and/or contributing to

**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC
<p>claim 1, further comprising performing, by the first device: identifying second user interaction with the interactive display selecting a particular user-selectable symbol corresponding to a particular second device and user interaction with the display specifying an action and, based thereon, initiating voice-over-IP (VOIP) communication with the particular second device.</p>	<p>the performance of performing, by the first device: identifying second user interaction with the interactive display selecting a particular user-selectable symbol corresponding to a particular second device and user interaction with the display specifying an action and, based thereon, initiating voice-over-IP (VOIP) communication with the particular second device. See claim 4, which is incorporated herein by reference in its entirety.</p> <p>Using, e.g., Android’s Google Voice app, users can initiate VOIP communications. For example, Google Voice is a voice over IP technology over internet protocol (IP) networks.</p>

**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC
	<p data-bbox="514 240 1192 298"><b>Send &amp; get text messages</b></p> <p data-bbox="514 321 1686 415">You can send text messages for free to U.S. and Canadian numbers using your Google Voice number. Texts sent using Google Voice will use Wi-Fi, or mobile data from your cell phone service plan if you're not connected to Wi-Fi. If you're outside the U.S. and are not using Wi-Fi, your cell phone company might charge you extra roaming fees to send a text.</p> <p data-bbox="514 444 1696 505"><b>Google Voice &amp; Hangouts:</b> You can choose to <a href="#">link Google Voice and Hangouts</a>. If you do, you'll have to use Hangouts to send texts, not the Google Voice website or apps. To stop using Hangouts for texts, <a href="#">turn off Google Voice in Hangouts</a>.</p> <p data-bbox="531 592 999 617"><a href="#">ANDROID</a>   <a href="#">COMPUTER</a>   <a href="#">IPHONE &amp; IPAD</a></p> <hr data-bbox="514 646 1724 651"/> <p data-bbox="514 711 1272 735">If you haven't yet, <a href="#">download the Google Voice app on your Android device</a>. <a href="#">↗</a></p> <p data-bbox="514 803 898 846"><b>Send a text message</b></p> <p data-bbox="514 867 1698 891">With the Google Voice website and apps, you can text people messages and photos and send texts to groups of people.</p> <p data-bbox="514 922 1629 946">If you send a text longer than 160 characters to a non-Google Voice number, it will be sent as multiple messages.</p> <p data-bbox="514 977 1180 1002"><b>Note:</b> You can't send texts to five- or six-digit "short code" numbers.</p> <ol data-bbox="514 1032 1289 1279" style="list-style-type: none"> <li>1. On your Android device, open the Google Voice app .</li> <li>2. Open the tab for Messages .</li> <li>3. At the bottom, tap Add .</li> <li>4. Enter a contact's name or phone number. <ul data-bbox="550 1198 1289 1222" style="list-style-type: none"> <li>• To create a group text message, add up to 30 names or phone numbers.</li> </ul> </li> <li>5. Enter your message, and tap Send .</li> </ol> <p data-bbox="514 1310 1696 1365">To include an image with your message, tap Select image . If your image is bigger than 2MB, it'll be sent as a smaller file. But GIFs over 2MB won't send.</p> <p data-bbox="514 1370 1682 1395"><a href="https://support.google.com/voice/answer/115116?co=GENIE.Platform%3DAndroid&amp;hl=en">https://support.google.com/voice/answer/115116?co=GENIE.Platform%3DAndroid&amp;hl=en</a></p>

**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC
<p>28[P]. A system comprising: a first device programmed to perform operations comprising:</p>	<p>HTC infringes either directly or indirectly, induces others to infringe, and/or contributes to the infringement of this system as set forth below. See claim 1[P], which is incorporated herein by reference in its entirety.</p> <p>The Accused Products meet the claim limitations by providing device-location tracking features such as those features described below. For example, the Accused Products meet the claim limitations because they are pre-installed with Android mobile operating systems containing code for providing device-location tracking features as provided in the claims limitations herein. For example, the Accused Products run applications and/or software that run within the Android mobile operating system and that use components of the Android mobile operating system to provide device-location tracking features. Upon information and belief, in addition to the components and features of the Android mobile operating system itself, the following applications and/or software run within the Android mobile operating system and use components of the Android mobile operating system to provide device-location tracking features: Google Latitude, Google Plus, Google Hangouts (including Allo and Duo), Google Maps, Google Chrome, Google Messages, and Android Messenger.</p> <p><b><u>Google Maps Share Location</u></b></p> <p>Share Location is currently included as a standard feature on the Accused Devices operating as a feature of Google Maps. Google Maps is a pre-installed software application in Android OS. The Accused Devices have included the Share Location functionalities since 2009 as part of Google Latitude, which was an opt-in feature for Google Maps on Android OS-based mobile devices, such as the Accused Products. Share Location functionalities were briefly shifted from Latitude for Google Maps to Google Plus and Google Hangouts, until reappearing as a standard feature in Google Maps. Upon information and belief, the Share Location method also uses and/or works in conjunction with functionalities associated with Google Maps, Google Messages, Android Messenger, Location Access, and other features, which are pre-installed on the Accused Products. For the purposes of these contentions, AGIS sets forth Google Maps' Share Location feature of the Accused Products as representative of this exemplary software. AGIS reserves the right to supplement these contentions to the extent that defendant requires additional information in accordance with P.R. 3-1 and for any other reason.</p> <p><i>See, e.g.,</i> <a href="https://techcrunch.com/2017/03/22/google-maps-now-lets-you-share-your-location-with-friends-and-family-for-a-specific-period-of-time/">https://techcrunch.com/2017/03/22/google-maps-now-lets-you-share-your-location-with-friends-and-family-for-a-specific-period-of-time/</a>; <a href="https://googleblog.blogspot.com/2009/02/see-where-your-friends-">https://googleblog.blogspot.com/2009/02/see-where-your-friends-</a></p>

**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC
	<p data-bbox="512 233 1724 337"> <a href="#">are-with-google.html</a>; <a href="https://googleblog.blogspot.jp/2013/07/a-new-google-maps-app-for-smartphones.html">https://googleblog.blogspot.jp/2013/07/a-new-google-maps-app-for-smartphones.html</a>; <a href="http://googleplusproject.blogspot.com/2013/05/google-for-android-42.html">http://googleplusproject.blogspot.com/2013/05/google-for-android-42.html</a>; <a href="https://googleblog.blogspot.com/2013/10/google-hangouts-and-photos-save-some.html">https://googleblog.blogspot.com/2013/10/google-hangouts-and-photos-save-some.html</a> </p> <p data-bbox="548 407 1205 440"> <b>Control within reach, even when your device isn't</b> </p> <p data-bbox="548 483 1591 699">           One of the biggest security risks you're likely to face is simply losing your phone. To help in these times of need, we're launching <a href="#">Find My Device</a> as part of Google Play Protect. With Find My Device you can locate, ring, lock and erase your Android devices—phones, tablets, and even watches. This feature is built in and enabled on all devices; visit <a href="http://android.com/find">android.com/find</a> or check out <a href="#">the app</a>.         </p> <p data-bbox="512 743 1444 776"> <i>See, e.g.,</i> <a href="https://www.blog.google/products/android/google-play-protect/">https://www.blog.google/products/android/google-play-protect/</a> </p> <p data-bbox="537 899 1310 997"> <b>Find your device using Android Device Manager</b> </p> <p data-bbox="537 1019 1493 1068">           If you've lost a device, you can use Android Device Manager to find its approximate location on a map and when it was last used. When Android Device Manager locates your device, that device will get a notification.         </p> <p data-bbox="537 1089 1482 1162"> <b>Before you can use Android Device Manager to locate your device:</b> Your device's <a href="#">location access need to be turned on</a> and be signed in to your Google Account. Android Device Manager won't work for devices that are turned off or that don't have a mobile data or Wi-Fi connection.         </p> <p data-bbox="537 1187 1461 1211"> <b>Tip:</b> If you've linked your phone to Google, you can locate or ring it by searching for <a href="#">find my phone on google.com</a>.         </p> <p data-bbox="520 1235 1192 1268"> <a href="https://support.google.com/pixelphone/answer/6160491">https://support.google.com/pixelphone/answer/6160491</a> </p>

## Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products

US9408055B2	HTC
	<p data-bbox="520 233 913 266"><b>Link your phone to Google</b></p> <p data-bbox="520 280 1201 318">You can connect your Android phone to Google, which lets you send information from your computer to your phone. For example, you can send directions you searched for on your computer to Google Maps on your phone.</p> <p data-bbox="520 342 737 363"><b>Link your Android phone</b></p> <p data-bbox="520 386 737 407"><b>Step 1: Update the Google app</b></p> <ol data-bbox="520 410 884 451" style="list-style-type: none"><li>1. On your phone, go to the <a href="#">Google app page on the Play Store</a>.</li><li>2. Tap <b>Update</b>.</li></ol> <p data-bbox="520 472 722 493"><b>Step 2: Turn on Google Now</b></p> <ol data-bbox="520 496 852 581" style="list-style-type: none"><li>1. On your phone, open the Google app .</li><li>2. At the top left, tap Menu  &gt; <b>Settings</b> &gt; <b>Now cards</b>.</li><li>3. Turn on <b>Show cards</b>.</li><li>4. Turn on <b>Show notifications</b>.</li></ol> <p data-bbox="520 602 770 623"><b>Step 3: Turn on Web &amp; App Activity</b></p> <ol data-bbox="520 626 741 667" style="list-style-type: none"><li>1. Visit the <a href="#">Account History page</a>.</li><li>2. Make sure the switch is on (green).</li></ol> <p data-bbox="520 688 741 709"><b>Step 4: Sign in to your browser</b></p> <ol data-bbox="520 712 978 842" style="list-style-type: none"><li>1. On your phone, open the Google app .</li><li>2. At the top left, tap the Menu .</li><li>3. At the top left, you'll see the email address you use for the Google app.</li><li>4. Visit <a href="http://www.google.com">www.google.com</a>  on your computer.</li><li>5. If you aren't signed in already, click <b>Sign in</b> in the top right corner of the page.</li><li>6. Sign in using the Google Account you use for the Google app.</li></ol> <p data-bbox="520 863 800 885"><b>Step 5: Send information to your phone</b></p> <ol data-bbox="520 888 1209 945" style="list-style-type: none"><li>1. Do one of the searches below, like <b>note to self</b>, or <b>send directions to my phone</b>.</li><li>2. If a box doesn't pop up with the option to send information to your phone, try refreshing the page. If you just turned on Google Now, it may take a few minutes for the box to show up</li></ol> <p data-bbox="520 953 1218 980"><a href="https://support.google.com/websearch/answer/6128427">https://support.google.com/websearch/answer/6128427</a></p>



## Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products

US9408055B2	HTC
	<p data-bbox="520 237 1010 261"><b>What you can do once your phone is linked</b></p> <hr/> <p data-bbox="537 297 667 318"><a href="#">Find my phone</a> <span style="float: right;">^</span></p> <p data-bbox="562 337 1041 358">You can get the current location of your phone if you can't find it.</p> <ol data-bbox="573 375 1388 472" style="list-style-type: none"><li>1. On your computer's browser, search on <a href="http://www.google.com">www.google.com</a> for <b>find my phone</b>.</li><li>2. If your phone is turned on and connected to the Internet, you'll see your phone's location.</li><li>3. If your phone's location is unavailable, you can still make it ring for 5 minutes on full volume by clicking <b>Ring</b>. You can stop the ringing from your phone when you find it.</li></ol> <p data-bbox="562 488 1388 529"><b>Tip:</b> You can also find your missing phone using the <a href="#">Android Device manager</a> which lets you find your device or remotely ring, lock, or erase it.</p> <hr/> <p data-bbox="537 589 785 610"><a href="#">Send directions to my phone</a> <span style="float: right;">^</span></p> <p data-bbox="562 630 1388 670">Once you've looked up directions on your computer, you can send them to your phone so you have them on your trip.</p> <ol data-bbox="573 686 1314 792" style="list-style-type: none"><li>1. On your computer's browser, search on <a href="http://www.google.com">www.google.com</a> for <b>send directions to my phone</b>.</li><li>2. Enter in your destination.</li><li>3. Click <b>Send directions to your phone</b>.</li><li>4. You'll get a notification on your phone. Tap to navigate to your destination using Google Maps.</li></ol> <hr/> <p data-bbox="537 849 751 870"><a href="#">Send a note to my phone</a> <span style="float: right;">^</span></p> <ol data-bbox="573 889 1367 995" style="list-style-type: none"><li>1. On your computer's browser, search on <a href="http://www.google.com">www.google.com</a> for <b>send a note to my phone</b>.</li><li>2. Type your note in the box.</li><li>3. Click <b>Send note to your phone</b>.</li><li>4. You'll get a notification on your phone with your note that you can either save to one of your apps or copy.</li></ol> <hr/> <p data-bbox="537 1052 646 1073"><a href="#">Set an alarm</a> <span style="float: right;">^</span></p> <ol data-bbox="573 1092 1178 1198" style="list-style-type: none"><li>1. On your computer's browser, search on <a href="http://www.google.com">www.google.com</a> for <b>set an alarm</b>.</li><li>2. Choose the time you want the alarm to go off.</li><li>3. Click <b>Set an alarm on your phone</b>.</li><li>4. An alarm will now be set on your phone's Clock app.</li></ol> <hr/> <p data-bbox="537 1255 667 1276"><a href="#">Set a reminder</a> <span style="float: right;">^</span></p> <ol data-bbox="573 1295 1335 1369" style="list-style-type: none"><li>1. On your computer's browser, search on <a href="http://www.google.com">www.google.com</a> for <b>set an reminder</b>.</li><li>2. Type what you want to be reminded about, and either when or where you want the reminder to go off.</li><li>3. Click <b>Remind me on my devices</b>.</li></ol> <hr/> <p data-bbox="520 1373 1220 1398"><a href="https://support.google.com/websearch/answer/6128427">https://support.google.com/websearch/answer/6128427</a></p>

**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC
	<p data-bbox="548 233 1325 277"><b>Share your location using Google Maps</b></p> <p data-bbox="548 297 1465 342">You can't share your location in Google+ anymore. If you used to share your location in Google+ and want to keep sharing it, you'll need to share it again in Google Maps.</p> <p data-bbox="520 358 1633 386"><a href="https://support.google.com/plus/answer/3302509?hl=en&amp;co=GENIE.Platform%3DAndroid&amp;oco=1">https://support.google.com/plus/answer/3302509?hl=en&amp;co=GENIE.Platform%3DAndroid&amp;oco=1</a></p> <p data-bbox="533 402 625 430"><b>Location</b></p> <p data-bbox="533 443 1409 537">Turn on location service, your phone determines your approximate location using Wi-Fi and mobile networks. When you select this option, you're asked whether you consent to allowing Google to use your location when providing these services.</p> <ul data-bbox="562 548 1367 764" style="list-style-type: none"><li>• <b>Mode</b> – Sets the how your current location information is determined.</li><li>• <b>Recent Location Request</b> – Displays applications and services that have recently requested your location information.</li><li>• <b>Camera</b> – Checkmark to tag photos or videos with their locations.</li><li>• <b>Google Location History</b> – Allows you to view and manage your Google location history.</li></ul> <p data-bbox="533 776 716 803"><b>Accounts &amp; sync</b></p> <p data-bbox="533 816 1402 943">Use the Accounts &amp; sync settings menu to add, remove, and manage your Google and other supported accounts. You also use these settings to control how and whether all applications send, receive, and sync data on their own schedules and whether all applications can synchronize user data automatically.</p> <p data-bbox="533 959 1373 1053">Gmail™, Calendar, and other applications may also have their own settings to control how they synchronize data; see the sections on those applications for details. Touch <b>Add account</b> to add new account.</p> <p data-bbox="533 1057 653 1073">-</p>

**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC
	 <p><a href="http://googlemobile.blogspot.com/2009/02/locate-your-friends-in-real-time-with.html">http://googlemobile.blogspot.com/2009/02/locate-your-friends-in-real-time-with.html</a></p> <p>Google’s location-sharing feature also appeared in Google+, Google Trust Contacts, and Google Hangouts services until its current integration in Google Maps.</p> <p>HTC makes, uses, sells, and otherwise provides this first device by making, using, selling, and importing Android devices such as HTC mobile devices, HTC tablets, and HTC Smartwatches as well as by providing its servers or using third party servers (e.g., Google servers) for use with Android devices to enable features such as Maps. Below are example HTC Android devices that perform each step of this method as set forth below.</p>

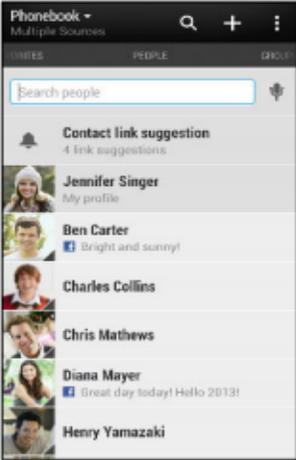
**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC
	<p style="text-align: right;">Sort by: <a href="#">Popularity</a> <a href="#">Date</a> <a href="#">Price</a></p> <p>HTC            LG            Samsung            Motorola            Fly            Sony-Ericsson            Apple            Nokia            Mobiado            Vertu            BenQ-Siemens            Sagem            Alcatel            Philips</p> <p style="text-align: center;"><a href="#">All brands</a></p> <div style="display: flex; flex-wrap: wrap;"> <div style="width: 50%;"> <p><b>HTC Desire 816</b></p>  <p>Mobile phone            2014 year            Touchscreen: 720 x 1280            5.5 inch.            Android 4.4</p> </div> <div style="width: 50%;"> <p><b>HTC One M8</b></p>  <p>Mobile phone            2014 year            Touchscreen: 1080 x 1920            5 inch.            Android 4.4</p> </div> <div style="width: 50%;"> <p><b>HTC Desire 300</b></p>  <p>Mobile phone            2013 year            Touchscreen: 480 x 800            4.3 inch.            Android 4.2</p> </div> <div style="width: 50%;"> <p><b>HTC Desire 601</b></p>  <p>Mobile phone            2013 year            Touchscreen: 540 x 960            4.5 inch.            Android 4.4</p> </div> <div style="width: 50%;"> <p><b>HTC Desire 700</b></p>  <p>Mobile phone            2013 year            Touchscreen: 540 x 960            5 inch.            Android 4.2</p> </div> <div style="width: 50%;"> <p><b>HTC Desire 400 Dual Sim</b></p>  <p>Mobile phone            2013 year            Touchscreen: 480 x 800            4.3 inch.            Android 4.1</p> </div> <div style="width: 50%;"> <p><b>HTC One Max</b></p>  <p>Mobile phone            2013 year            Touchscreen: 1080 x 1920            5.9 inch.            Android 4.3</p> </div> </div> <p><a href="https://mob.org/phone/htc/page_3/sort_date_down/">https://mob.org/phone/htc/page_3/sort_date_down/</a></p>
[28A] obtaining	HTC makes, uses, sells, and otherwise provides this second device by making, using, selling, and importing

**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

<b>US9408055B2</b>	<b>HTC</b>
<p>contact information of a plurality of second devices, wherein the contact information comprises respective telephone numbers of the second devices;</p>	<p>Android devices, and Smartwatches as well as by providing its servers for use with Android devices to enable features such as Find My Device.</p> <p>HTC infringes directly and/or indirectly by performing, inducing others to perform, and/or contributing to the performance of a first device programmed to perform operations comprising: obtaining contact information of a plurality of second devices, wherein the contact information comprises respective telephone numbers of the second devices. See claim 1[A] and 28[P], which are incorporated herein by reference in their entirety.</p> <p>For example, the Accused products include a contacts app to access contact information for second users using respective second devices.</p>

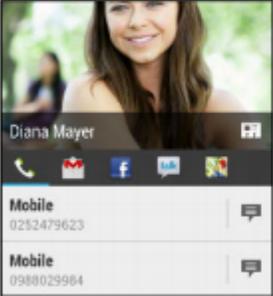
## Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products

US9408055B2	HTC
	<p data-bbox="562 233 947 269"><b>Your contacts list</b></p> <p data-bbox="562 305 1587 391">The Contacts app lists all contacts you've stored on HTC One and from online accounts you're logged in to. Use the Contacts app to easily manage communications with people that matter to you.</p> <ol style="list-style-type: none"> <li data-bbox="594 415 919 440">1. Open the Contacts app.</li> </ol>  <ol style="list-style-type: none"> <li data-bbox="594 943 995 967">2. On your contacts list, you can: <ul style="list-style-type: none"> <li data-bbox="657 987 1308 1011">▪ View your profile and edit your contact information.</li> <li data-bbox="657 1024 1104 1049">▪ Create, edit, find, or send contacts.</li> <li data-bbox="657 1062 919 1086">▪ See status updates.</li> <li data-bbox="657 1099 1520 1123">▪ Tap a contact photo to find ways to quickly connect with the contact.</li> <li data-bbox="657 1136 1486 1161">▪ See a notification icon when a contact has sent you new messages.</li> <li data-bbox="657 1174 1581 1214">▪ Check out who's online in Google Talk™. Online status icons are displayed if you're signed in to Google Talk.</li> </ul> </li> </ol> <p data-bbox="520 1239 1587 1304">💡 To sort your contacts by their first or last name, tap <b>☰</b> &gt; Settings &gt; Sort contact list.</p>

**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC
	<p data-bbox="527 245 926 277"><b>Filtering your contacts list</b></p> <p data-bbox="527 302 1575 329">When your contacts list gets long, you can choose which contact accounts to show.</p> <ol data-bbox="562 358 1444 488" style="list-style-type: none"> <li data-bbox="562 358 961 386">1. On the Contacts tab, tap ▼.</li> <li data-bbox="562 407 1444 435">2. Choose the accounts that contain the contacts you want to display.</li> <li data-bbox="562 456 716 483">3. Press &lt;.</li> </ol> <p data-bbox="527 529 758 561"><b>Finding people</b></p> <p data-bbox="527 583 1518 647">Search for contacts stored on HTC One, your company directory if you have an Exchange ActiveSync account, or social networks you've signed into.</p> <ol data-bbox="562 677 1608 1097" style="list-style-type: none"> <li data-bbox="562 677 905 704">1. Open the Contacts app.</li> <li data-bbox="562 725 1608 1097">2. On the Contacts tab, you can: <ul data-bbox="625 781 1608 1097" style="list-style-type: none"> <li data-bbox="625 781 1608 846">▪ Find people in your contacts list. Tap the Search people box, and then enter the first few letters of the contact name.</li> <li data-bbox="625 854 1608 951">▪ Find people on your company directory. Tap the Search people box, enter the first few letters of the contact name, and then tap Search contacts in your Company Directory.</li> <li data-bbox="625 959 1608 1097">▪ Search for people you know on your social networks. Tap ☰ &gt; Settings &gt; Find people you know on, and then select the social networks you're signed in to. The Contacts app then uploads your contacts to the selected social networks to help you find friends.</li> </ul> </li> </ol> <p data-bbox="527 1138 1608 1235">) Aside from searching for a contact by name, you can search using a contact's email address or company name. On the Contacts tab, tap ☰ &gt; Settings &gt; Search contacts by, and then choose a search criteria.</p>

## Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products

US9408055B2	HTC
	<p data-bbox="569 240 1283 284"><b>Getting in touch with a contact</b></p> <ol data-bbox="600 329 1625 440" style="list-style-type: none"><li data-bbox="600 329 940 358">1. Open the Contacts app.</li><li data-bbox="600 378 1625 440">2. Tap a contact's photo (not the name), and then choose how you want to get in touch with that contact.</li></ol>  <p data-bbox="520 789 1646 883"> For more ways of getting in touch with your contact, tap an icon below the contact photo.</p> <p data-bbox="512 927 1871 995">In another example, the Accused products run Android Messages and Google Hangouts which both access contact information for second users using respective second devices.</p>



## Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products

US9408055B2	HTC
	<h3 data-bbox="527 240 1016 293">Contacts Provider</h3> <p data-bbox="527 329 1472 589">The Contacts Provider is a powerful and flexible Android component that manages the device's central repository of data about people. The Contacts Provider is the source of data you see in the device's contacts application, and you can also access its data in your own application and transfer data between the device and online services. The provider accommodates a wide range of data sources and tries to manage as much data as possible for each person, with the result that its organization is complex. Because of this, the provider's API includes an extensive set of contract classes and interfaces that facilitate both data retrieval and modification.</p> <p data-bbox="527 625 835 646">This guide describes the following:</p> <ul data-bbox="527 675 1373 850" style="list-style-type: none"><li data-bbox="527 675 806 696">• The basic provider structure.</li><li data-bbox="527 725 894 747">• How to retrieve data from the provider.</li><li data-bbox="527 776 863 797">• How to modify data in the provider.</li><li data-bbox="527 826 1373 847">• How to write a sync adapter for synchronizing data from your server to the Contacts Provider.</li></ul> <p data-bbox="512 862 1486 889"><a href="https://developer.android.com/guide/topics/providers/contacts-provider.html">https://developer.android.com/guide/topics/providers/contacts-provider.html</a></p>

**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC
	<p><b>Overview</b></p> <p>ContactsContract defines an extensible database of contact-related information. Contact information is stored in a three-tier data model:</p> <ul style="list-style-type: none"> <li>• A row in the <code>ContactsContract.Data</code> table can store any kind of personal data, such as a phone number or email addresses. The set of data kinds that can be stored in this table is open-ended. There is a predefined set of common kinds, but any application can add its own data kinds.</li> <li>• A row in the <code>ContactsContract.RawContacts</code> table represents a set of data describing a person and associated with a single account (for example, one of the user's Gmail accounts).</li> <li>• A row in the <code>ContactsContract.Contacts</code> table represents an aggregate of one or more RawContacts presumably describing the same person. When data in or associated with the RawContacts table is changed, the affected aggregate contacts are updated as necessary.</li> </ul> <p>Other tables include:</p> <ul style="list-style-type: none"> <li>• <code>ContactsContract.Groups</code>, which contains information about raw contact groups such as Gmail contact groups. The current API does not support the notion of groups spanning multiple accounts.</li> <li>• <code>ContactsContract.StatusUpdates</code>, which contains social status updates including IM availability.</li> <li>• <code>ContactsContract.AggregationExceptions</code>, which is used for manual aggregation and disaggregation of raw contacts</li> <li>• <code>ContactsContract.Settings</code>, which contains visibility and sync settings for accounts and groups.</li> <li>• <code>ContactsContract.SyncState</code>, which contains free-form data maintained on behalf of sync adapters</li> <li>• <code>ContactsContract.PhoneLookup</code>, which is used for quick caller-ID lookup</li> </ul> <p><a href="https://developer.android.com/reference/android/provider/ContactsContract.html">https://developer.android.com/reference/android/provider/ContactsContract.html</a></p> <p><b>Data</b></p> <p>As noted previously, the data for a raw contact is stored in a <code>ContactsContract.Data</code> row that is linked to the raw contact's <code>_ID</code> value. This allows a single raw contact to have multiple instances of the same type of data such as email addresses or phone numbers. For example, if "Thomas Higginson" for <code>emilyd@gmail.com</code> (the raw contact row for Thomas Higginson associated with the Google account <code>emilyd@gmail.com</code>) has a home email address of <code>thigg@gmail.com</code> and a work email address of <code>thomas.higginson@gmail.com</code>, the Contacts Provider stores the two email address rows and links them both to the raw contact.</p> <p>Notice that different types of data are stored in this single table. Display name, phone number, email, postal address, photo, and website detail rows are all found in the <code>ContactsContract.Data</code> table. To help manage this, the <code>ContactsContract.Data</code> table has some columns with descriptive names, and others with generic names. The contents of a descriptive-name column have the same meaning regardless of the type of data in the row, while the contents of a generic-name column have different meanings depending on the type of data.</p> <p><a href="https://developer.android.com/guide/topics/providers/contacts-provider.html">https://developer.android.com/guide/topics/providers/contacts-provider.html</a></p>

**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC											
	<table border="1"> <thead> <tr> <th data-bbox="514 240 609 272">Task</th> <th data-bbox="615 240 852 272">Action</th> <th data-bbox="858 240 1188 272">Data</th> <th data-bbox="1194 240 1486 272">MIME type</th> <th data-bbox="1493 240 1745 272">Notes</th> </tr> </thead> <tbody> <tr> <td data-bbox="514 277 609 946">Pick a contact from a list</td> <td data-bbox="615 277 852 946">ACTION_PICK</td> <td data-bbox="858 277 1188 946">                     One of:                     <ul style="list-style-type: none"> <li>• <code>Contacts.CONTENT_URI</code>, which displays a list of contacts.</li> <li>• <code>Phone.CONTENT_URI</code>, which displays a list of phone numbers for a raw contact.</li> <li>• <code>StructuredPostal.CONTENT_URI</code>, which displays a list of postal addresses for a raw contact.</li> <li>• <code>Email.CONTENT_URI</code>, which displays a list of email addresses for a raw contact.</li> </ul> </td> <td data-bbox="1194 277 1486 946">Not used</td> <td data-bbox="1493 277 1745 946">                     Displays a list of raw contacts or a list of data from a raw contact, depending on the content URI type you supply. Call <code>startActivityForResult()</code>, which returns the content URI of the selected row. The form of the URI is the table's content URI with the row's <code>LOOKUP_ID</code> appended to it. The device's contacts app delegates read and write permissions to this content URI for the life of your activity. See the <a href="#">Content Provider Basics</a> guide for more details.                 </td> </tr> </tbody> </table>	Task	Action	Data	MIME type	Notes	Pick a contact from a list	ACTION_PICK	One of: <ul style="list-style-type: none"> <li>• <code>Contacts.CONTENT_URI</code>, which displays a list of contacts.</li> <li>• <code>Phone.CONTENT_URI</code>, which displays a list of phone numbers for a raw contact.</li> <li>• <code>StructuredPostal.CONTENT_URI</code>, which displays a list of postal addresses for a raw contact.</li> <li>• <code>Email.CONTENT_URI</code>, which displays a list of email addresses for a raw contact.</li> </ul>	Not used	Displays a list of raw contacts or a list of data from a raw contact, depending on the content URI type you supply. Call <code>startActivityForResult()</code> , which returns the content URI of the selected row. The form of the URI is the table's content URI with the row's <code>LOOKUP_ID</code> appended to it. The device's contacts app delegates read and write permissions to this content URI for the life of your activity. See the <a href="#">Content Provider Basics</a> guide for more details.	<a href="https://developer.android.com/guide/topics/providers/contacts-provider.html">https://developer.android.com/guide/topics/providers/contacts-provider.html</a>
Task	Action	Data	MIME type	Notes								
Pick a contact from a list	ACTION_PICK	One of: <ul style="list-style-type: none"> <li>• <code>Contacts.CONTENT_URI</code>, which displays a list of contacts.</li> <li>• <code>Phone.CONTENT_URI</code>, which displays a list of phone numbers for a raw contact.</li> <li>• <code>StructuredPostal.CONTENT_URI</code>, which displays a list of postal addresses for a raw contact.</li> <li>• <code>Email.CONTENT_URI</code>, which displays a list of email addresses for a raw contact.</li> </ul>	Not used	Displays a list of raw contacts or a list of data from a raw contact, depending on the content URI type you supply. Call <code>startActivityForResult()</code> , which returns the content URI of the selected row. The form of the URI is the table's content URI with the row's <code>LOOKUP_ID</code> appended to it. The device's contacts app delegates read and write permissions to this content URI for the life of your activity. See the <a href="#">Content Provider Basics</a> guide for more details.								

**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC
	<pre>25  * Parsed form of the intent sent to the Contacts application. 26  */ 27  public class ContactsRequest { 28 29      /** Default mode: browse contacts */ 30      public static final int ACTION_DEFAULT = 10; 31 32      /** Show all contacts */ 33      public static final int ACTION_ALL_CONTACTS = 15; 34 35      /** Show all contacts with phone numbers */ 36      public static final int ACTION_CONTACTS_WITH_PHONES = 17; 37 38      /** Show contents of a specific group */ 39      public static final int ACTION_GROUP = 20; 40 41      /** Show all starred contacts */ 42      public static final int ACTION_STARRED = 30; 43 44      /** Show frequently contacted contacts */ 45      public static final int ACTION_FREQUENT = 40; 46 47      /** Show starred and the frequent */ 48      public static final int ACTION_STREQUENT = 50; 49 50      /** Show all contacts and pick them when clicking */ 51      public static final int ACTION_PICK_CONTACT = 60; 52 53      /** Show all contacts as well as the option to create a new one */ 54      public static final int ACTION_PICK_OR_CREATE_CONTACT = 70; 55 56      /** Show all contacts and pick them for edit when clicking, and allow creating a new contact */ 57      public static final int ACTION_INSERT_OR_EDIT_CONTACT = 80;</pre> <p><a href="https://android.googlesource.com/platform/packages/apps/Contacts/+nougat-mr1-release/src/com/android/contacts/list/ContactsRequest.java">https://android.googlesource.com/platform/packages/apps/Contacts/+nougat-mr1-release/src/com/android/contacts/list/ContactsRequest.java</a></p>

**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC
	<pre>59  /** Show all phone numbers and pick them when clicking */ 60  public static final int ACTION_PICK_PHONE = 90; 61 62  /** Show all postal addresses and pick them when clicking */ 63  public static final int ACTION_PICK_POSTAL = 100; 64 65  /** Show all postal addresses and pick them when clicking */ 66  public static final int ACTION_PICK_EMAIL = 105; 67 68  /** Show all contacts and create a shortcut for the picked contact */ 69  public static final int ACTION_CREATE_SHORTCUT_CONTACT = 110; 70 71  /** Show all phone numbers and create a call shortcut for the picked number */ 72  public static final int ACTION_CREATE_SHORTCUT_CALL = 120; 73 74  /** Show all phone numbers and create an SMS shortcut for the picked number */ 75  public static final int ACTION_CREATE_SHORTCUT_SMS = 130; 76 77  /** Show all contacts and activate the specified one */ 78  public static final int ACTION_VIEW_CONTACT = 140; 79 80  /** Show contacts recommended for joining with a specified target contact */ 81  public static final int ACTION_PICK_JOIN = 150;</pre> <p data-bbox="506 1040 1566 1109"><a href="https://android.googlesource.com/platform/packages/apps/Contacts+/nougat-mr1-release/src/com/android/contacts/list/ContactsRequest.java">https://android.googlesource.com/platform/packages/apps/Contacts+/nougat-mr1-release/src/com/android/contacts/list/ContactsRequest.java</a></p>

**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC
	<pre> 104      * Displays a list to browse contacts. 105      */ 106      public class PeopleActivity extends ContactsActivity implements 107          View.OnCreateContextMenuListener, 108          View.OnClickListener, 109          ActionBarAdapter.Listener, 110          DialogManager.DialogShowingViewActivity, 111          ContactListFilterController.ContactListFilterListener, 112          ProviderStatusListener, 113          MultiContactDeleteListener, 114          JoinContactsListener { <a href="https://android.googlesource.com/platform/packages/apps/Contacts+/nougat-mr1-release/src/com/android/contacts/activities/PeopleActivity.java">https://android.googlesource.com/platform/packages/apps/Contacts+/nougat-mr1- release/src/com/android/contacts/activities/PeopleActivity.java</a> 145          * Showing a list of Contacts. Also used for showing search results in search mode. 146          */ 147          private MultiSelectContactsListFragment mAllFragment; 148          private ContactTileListFragment mFavoritesFragment; <a href="https://android.googlesource.com/platform/packages/apps/Contacts+/nougat-mr1-release/src/com/android/contacts/activities/PeopleActivity.java">https://android.googlesource.com/platform/packages/apps/Contacts+/nougat-mr1- release/src/com/android/contacts/activities/PeopleActivity.java</a> </pre>

**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC
	<pre>458     private void configureFragments(boolean fromRequest) { 459         if (fromRequest) { 460             ContactListFilter filter = null; 461             int actionCode = mRequest.getActionCode(); 462             boolean searchMode = mRequest.isSearchMode(); 463             final int tabToOpen; 464             switch (actionCode) { 465                 case ContactsRequest.ACTION_ALL_CONTACTS: 466                     filter = ContactListFilter.createFilterWithType( 467                         ContactListFilter.FILTER_TYPE_ALL_ACCOUNTS); 468                     tabToOpen = TabState.ALL; 469                     break; 470                 case ContactsRequest.ACTION_CONTACTS_WITH_PHONES: 471                     filter = ContactListFilter.createFilterWithType( 472                         ContactListFilter.FILTER_TYPE_WITH_PHONE_NUMBERS_ONLY); 473                     tabToOpen = TabState.ALL; 474                     break; 475 476                 case ContactsRequest.ACTION_FREQUENT: 477                 case ContactsRequest.ACTION_STREQUENT: 478                 case ContactsRequest.ACTION_STARRED: 479                     tabToOpen = TabState.FAVORITES; 480                     break; 481                 case ContactsRequest.ACTION_VIEW_CONTACT: 482                     tabToOpen = TabState.ALL; 483                     break; 484                 default: 485                     tabToOpen = -1; 486                     break; 487             } </pre> <p data-bbox="506 1321 1566 1390"><a href="https://android.googlesource.com/platform/packages/apps/Contacts+/nougat-mr1-release/src/com/android/contacts/activities/PeopleActivity.java">https://android.googlesource.com/platform/packages/apps/Contacts+/nougat-mr1-release/src/com/android/contacts/activities/PeopleActivity.java</a></p>

**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC
	<pre>488         if (tabToOpen != -1) { 489             mActionBarAdapter.setCurrentTab(tabToOpen); 490         } 491 492         if (filter != null) { 493             mContactListFilterController.setContactListFilter(filter, false); 494             searchMode = false; 495         } 496 497         if (mRequest.getContactUri() != null) { 498             searchMode = false; 499         } 500 501         mActionBarAdapter.setSearchMode(searchMode); 502         configureContactListFragmentForRequest(); 503     } 504 505     configureContactListFragment(); 506 507     invalidateOptionsMenuIfNeeded(); 508 }</pre> <p data-bbox="506 1019 1570 1084"><a href="https://android.googlesource.com/platform/packages/apps/Contacts+/nougat-mr1-release/src/com/android/contacts/activities/PeopleActivity.java">https://android.googlesource.com/platform/packages/apps/Contacts+/nougat-mr1-release/src/com/android/contacts/activities/PeopleActivity.java</a></p>



**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC
	<pre>35 public class ProfileAndContactsLoader extends CursorLoader { 36 37     private boolean mLoadProfile; 38 39     private String[] mProjection; 40 41     private Uri mExtraUri; 42     private String[] mExtraProjection; 43     private String mExtraSelection; 44     private String[] mExtraSelectionArgs; 45     private boolean mMergeExtraContactsAfterPrimary; 46 47     public ProfileAndContactsLoader(Context context) { 48         super(context); 49     }</pre> <p><a href="https://android.googlesource.com/platform/packages/apps/ContactsCommon/+/nougat-mr1-release/src/com/android/contacts/common/list/ProfileAndContactsLoader.java">https://android.googlesource.com/platform/packages/apps/ContactsCommon/+/nougat-mr1-release/src/com/android/contacts/common/list/ProfileAndContactsLoader.java</a></p>

**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC
	<pre> 35 public final class GroupMemberLoader extends CursorLoader { 36 37     public static class GroupEditorQuery { 38         private static final String[] PROJECTION = new String[] { 39             Data.CONTACT_ID,           // 0 40             Data.RAW_CONTACT_ID,      // 1 41             Data.DISPLAY_NAME_PRIMARY, // 2 42             Data.PHOTO_URI,           // 3 43             Data.LOOKUP_KEY,          // 4 44         }; 45 46         public static final int CONTACT_ID           = 0; 47         public static final int RAW_CONTACT_ID      = 1; 48         public static final int CONTACT_DISPLAY_NAME_PRIMARY = 2; 49         public static final int CONTACT_PHOTO_URI   = 3; 50         public static final int CONTACT_LOOKUP_KEY  = 4; 51     } 52 53     public static class GroupDetailQuery { 54         private static final String[] PROJECTION = new String[] { 55             Data.CONTACT_ID,           // 0 56             Data.PHOTO_URI,           // 1 57             Data.LOOKUP_KEY,          // 2 58             Data.DISPLAY_NAME_PRIMARY, // 3 59             Data.CONTACT_PRESENCE,     // 4 60             Data.CONTACT_STATUS,      // 5 61         }; 62 63         public static final int CONTACT_ID           = 0; 64         public static final int CONTACT_PHOTO_URI   = 1; 65         public static final int CONTACT_LOOKUP_KEY  = 2; 66         public static final int CONTACT_DISPLAY_NAME_PRIMARY = 3; 67         public static final int CONTACT_PRESENCE_STATUS = 4; 68         public static final int CONTACT_STATUS      = 5; 69     } 70 71     private final long mGroupId; </pre> <p><a href="https://android.googlesource.com/platform/packages/apps/Contacts/+nougat-mr1-release/src/com/android/contacts/GroupMemberLoader.java">https://android.googlesource.com/platform/packages/apps/Contacts/+nougat-mr1-release/src/com/android/contacts/GroupMemberLoader.java</a></p>

**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC
	<pre> 24  * Group loader for the group list that includes details such as the number of contacts per group 25  * and number of groups per account. This list is sorted by account type, account name, where the 26  * group names are in alphabetical order. Note that the list excludes default, favorite, and deleted 27  * groups. 28  */ 29  public final class GroupListLoader extends CursorLoader { 30 31      private final static String[] COLUMNS = new String[] { 32          Groups.ACCOUNT_NAME, 33          Groups.ACCOUNT_TYPE, 34          Groups.DATA_SET, 35          Groups._ID, 36          Groups.TITLE, 37          Groups.SUMMARY_COUNT, 38      }; 39 40      public final static int ACCOUNT_NAME = 0; 41      public final static int ACCOUNT_TYPE = 1; 42      public final static int DATA_SET = 2; 43      public final static int GROUP_ID = 3; 44      public final static int TITLE = 4; 45      public final static int MEMBER_COUNT = 5; 46 47      private static final Uri GROUP_LIST_URI = Groups.CONTENT_SUMMARY_URI; 48 49      public GroupListLoader(Context context) { 50          super(context, GROUP_LIST_URI, COLUMNS, Groups.ACCOUNT_TYPE + " NOT NULL AND " 51              + Groups.ACCOUNT_NAME + " NOT NULL AND " + Groups.AUTO_ADD + "=0 AND " + 52              Groups.FAVORITES + "=0 AND " + Groups.DELETED + "=0", null, 53              Groups.ACCOUNT_TYPE + ", " + Groups.ACCOUNT_NAME + ", " + Groups.DATA_SET + ", " + 54              Groups.TITLE + " COLLATE LOCALIZED ASC"); 55      } 56  } </pre> <p><a href="https://android.googlesource.com/platform/packages/apps/Contacts+/nougat-mr1-release/src/com/android/contacts/GroupListLoader.java">https://android.googlesource.com/platform/packages/apps/Contacts+/nougat-mr1-release/src/com/android/contacts/GroupListLoader.java</a></p>

**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC
	<pre> 24  * Group meta-data loader. Loads all groups or just a single group from the 25  * database (if given a {@link Uri}). 26  */ 27  public final class GroupMetaLoader extends CursorLoader { 28 29      private final static String[] COLUMNS = new String[] { 30          Groups.ACCOUNT_NAME, 31          Groups.ACCOUNT_TYPE, 32          Groups.DATA_SET, 33          Groups._ID, 34          Groups.TITLE, 35          Groups.AUTO_ADD, 36          Groups.FAVORITES, 37          Groups.GROUP_IS_READ_ONLY, 38          Groups.DELETED, 39      }; 40 41      public final static int ACCOUNT_NAME = 0; 42      public final static int ACCOUNT_TYPE = 1; 43      public final static int DATA_SET = 2; 44      public final static int GROUP_ID = 3; 45      public final static int TITLE = 4; 46      public final static int AUTO_ADD = 5; 47      public final static int FAVORITES = 6; 48      public final static int IS_READ_ONLY = 7; 49      public final static int DELETED = 8; 50 51      public GroupMetaLoader(Context context, Uri groupUri) { 52          super(context, ensureIsGroupUri(groupUri), COLUMNS, Groups.ACCOUNT_TYPE + " NOT NULL AND " 53              + Groups.ACCOUNT_NAME + " NOT NULL", null, null); 54      } 55 56      /** 57       * Ensures that this is a valid group URI. If invalid, then an exception is 58       * thrown. Otherwise, the original URI is returned. 59       */ 60      private static Uri ensureIsGroupUri(final Uri groupUri) { 61          // TODO: Fix ContactsProvider2 getType method to resolve the group Uris 62          if (groupUri == null) { 63              throw new IllegalArgumentException("Uri must not be null"); 64          } 65          if (!groupUri.toString().startsWith(Groups.CONTENT_URI.toString())) { 66              throw new IllegalArgumentException("Invalid group Uri: " + groupUri); 67          } 68          return groupUri; 69      } 70  } </pre> <p><a href="https://android.googlesource.com/platform/packages/apps/Contacts/+nougat-mr1-release/src/com/android/contacts/GroupMetaLoader.java">https://android.googlesource.com/platform/packages/apps/Contacts/+nougat-mr1-release/src/com/android/contacts/GroupMetaLoader.java</a></p>

**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC
	<pre> 19  * Meta-data for a contact group. We load all groups associated with the contact's 20  * constituent accounts. 21  */ 22  public final class GroupMetaData { 23      private String mAccountName; 24      private String mAccountType; 25      private String mDataSet; 26      private long mGroupId; 27      private String mTitle; 28      private boolean mDefaultGroup; 29      private boolean mFavorites; 30 31      public GroupMetaData(String accountName, String accountType, String dataSet, long groupId, 32                          String title, boolean defaultGroup, boolean favorites) { 33          this.mAccountName = accountName; 34          this.mAccountType = accountType; 35          this.mDataSet = dataSet; 36          this.mGroupId = groupId; 37          this.mTitle = title; 38          this.mDefaultGroup = defaultGroup; 39          this.mFavorites = favorites; 40      } 41 42      public String getAccountName() { 43          return mAccountName; 44      } 45 46      public String getAccountType() { 47          return mAccountType; 48      } 49 50      public String getDataSet() { 51          return mDataSet; 52      } 53 54      public long getGroupId() { 55          return mGroupId; 56      } 57 58      public String getTitle() { 59          return mTitle; 60      } 61 62      public boolean isDefaultGroup() { 63          return mDefaultGroup; 64      } 65 66      public boolean isFavorites() { 67          return mFavorites; </pre> <p><a href="https://android.googlesource.com/platform/packages/apps/ContactsCommon/+/nougat-mr1-release/src/com/android/contacts/common/GroupMetaData.java">https://android.googlesource.com/platform/packages/apps/ContactsCommon/+/nougat-mr1-release/src/com/android/contacts/common/GroupMetaData.java</a></p>

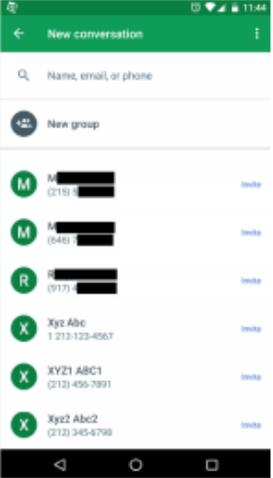
**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC
	<p data-bbox="541 277 1297 370"><b>Send &amp; receive text messages in Android Messages</b></p> <p data-bbox="541 386 1178 407">You can send and receive text messages with friends and contacts on Android Messages.</p> <p data-bbox="531 423 846 456"><b>Start a conversation</b></p> <ol data-bbox="541 480 1528 638" style="list-style-type: none"><li>1. Open the Android Messages app .</li><li>2. Tap Compose .</li><li>3. In "To," enter the names, phone numbers, or email addresses that you'd like to message. You can also pick from your top contacts or your whole contact list.</li><li>4. Tap Next .</li></ol> <p data-bbox="516 659 1472 691"><a href="https://support.google.com/nexus/answer/6080324?hl=en&amp;ref_topic=6080329">https://support.google.com/nexus/answer/6080324?hl=en&amp;ref_topic=6080329</a></p> <p data-bbox="541 781 894 821"><b>See your contacts</b></p> <ol data-bbox="552 846 968 919" style="list-style-type: none"><li>1. Open your device's Contacts app .</li><li>2. Tap Menu .</li></ol> <ul data-bbox="548 951 1738 1146" style="list-style-type: none"><li>• <b>See contacts by label:</b> Choose a label from the list.</li><li>• <b>See contacts for another account:</b> Tap Down arrow  &gt; pick an account.</li><li>• <b>See the contacts for all your accounts:</b> Choose <b>All contacts</b>. <b>Tip:</b> If you have multiple contacts with the same information, the information will be grouped into one contact.</li><li>• <b>See your Google Account contacts on the web:</b> Go to <a href="#">Google Contacts</a> .</li></ul> <p data-bbox="516 1179 1535 1211"><a href="https://support.google.com/android/answer/6118731?hl=en&amp;ref_topic=6118711">https://support.google.com/android/answer/6118731?hl=en&amp;ref_topic=6118711</a></p>

## Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products

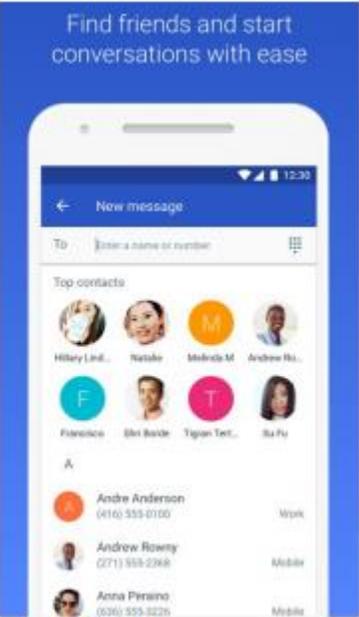
US9408055B2	HTC
	<p data-bbox="541 240 894 277"><b>Label your contacts</b></p> <p data-bbox="541 305 982 326">You can group contacts together using labels.</p> <ol data-bbox="552 358 930 461" style="list-style-type: none"><li data-bbox="552 358 930 380">1. Open your device's Contacts app .</li><li data-bbox="552 399 863 420">2. Tap Menu  &gt; <b>Create label</b>.</li><li data-bbox="552 440 869 461">3. Enter a label name and tap <b>Ok</b>.</li></ol> <ul data-bbox="552 493 1717 553" style="list-style-type: none"><li data-bbox="552 493 1234 514">• <b>Add one contact to a label:</b> Tap Add contact  &gt; choose a contact.</li><li data-bbox="552 534 1717 555">• <b>Add multiple contacts to a label:</b> Tap Add contact  &gt; touch and hold a contact &gt; tap the other contacts &gt; tap <b>Add</b>.</li></ul> <p data-bbox="512 570 1535 597"><a href="https://support.google.com/android/answer/6118731?hl=en&amp;ref_topic=6118711">https://support.google.com/android/answer/6118731?hl=en&amp;ref_topic=6118711</a></p> <p data-bbox="552 656 940 693"><b>Share your contacts</b></p> <ol data-bbox="562 725 1045 883" style="list-style-type: none"><li data-bbox="562 725 978 747">1. Open your device's Contacts app .</li><li data-bbox="562 766 842 787">2. Tap a contact in the list.</li><li data-bbox="562 807 831 828">3. Tap More  &gt; <b>Share</b>.</li><li data-bbox="562 847 1045 868">4. Choose how you want to share the contact.</li></ol> <p data-bbox="512 901 1535 928"><a href="https://support.google.com/android/answer/6118731?hl=en&amp;ref_topic=6118711">https://support.google.com/android/answer/6118731?hl=en&amp;ref_topic=6118711</a></p>

**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

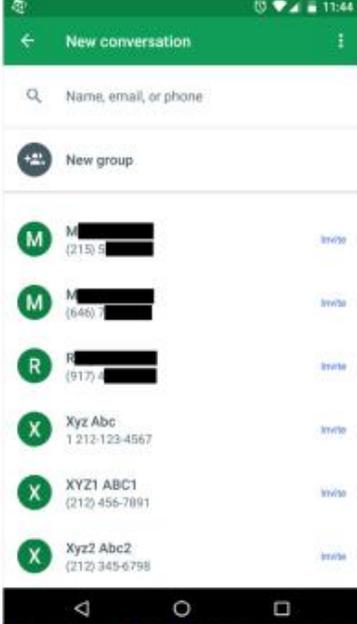
US9408055B2	HTC
	<div data-bbox="533 233 1041 277"> <h2>Start a Hangout</h2> </div> <div data-bbox="533 289 1041 310"> <p>You can send and receive messages with one person or multiple people.</p> </div> <div data-bbox="548 370 879 391"> <p>COMPUTER <b>ANDROID</b> IPHONE &amp; IPAD</p> </div> <div data-bbox="533 516 800 548"> <h3>Start a conversation</h3> </div> <div data-bbox="533 565 1121 699"> <ol style="list-style-type: none"> <li>1. On your Android phone or tablet, open the Hangouts app .</li> <li>2. At the bottom right, tap Add  &gt; New Conversation .</li> <li>3. Type and select a person's name.</li> <li>4. Enter your message. You can also add emojis, photos, your location, or a sticker.</li> <li>5. Tap Send .</li> </ol> </div> <div data-bbox="520 716 1751 776"> <p><a href="https://support.google.com/hangouts/answer/3115553?hl=en&amp;ref_topic=6386410&amp;co=GENIE.Platform%3DAndroid&amp;oco=0">https://support.google.com/hangouts/answer/3115553?hl=en&amp;ref_topic=6386410&amp;co=GENIE.Platform%3DAndroid&amp;oco=0</a></p> </div> <div data-bbox="533 818 879 857"> <h2>Contact someone</h2> </div> <div data-bbox="533 883 1150 911"> <p>You can call, email, or send text messages to your contacts.</p> </div> <div data-bbox="548 938 953 1182"> <ol style="list-style-type: none"> <li>1. Open your device's Contacts app .</li> <li>2. Tap a contact in the list.</li> <li>3. Choose an option:             <ul style="list-style-type: none"> <li>• Call </li> <li>• Email </li> <li>• New message </li> </ul> </li> </ol> </div> <div data-bbox="520 1198 1402 1230"> <p><a href="https://support.google.com/nexus/answer/6118731?hl=en&amp;ref_topic=6118711">https://support.google.com/nexus/answer/6118731?hl=en&amp;ref_topic=6118711</a></p> </div> <div data-bbox="1129 233 1402 711">  </div>



## Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products

US9408055B2	HTC
	<p data-bbox="527 688 762 716">Start a conversation</p> <ol data-bbox="527 732 1272 862" style="list-style-type: none"><li>1. Open the Android Messages app</li><li>2. Tap Compose</li><li>3. In "To," enter the names, phone numbers, or email addresses that you'd like to message. You can also pick from your top contacts or your whole contact list.</li><li>4. Tap Next</li></ol> <p data-bbox="516 878 1612 943"><a href="https://support.google.com/nexus/answer/6080324?hl=en&amp;ref_topic=6080329">https://support.google.com/nexus/answer/6080324?hl=en&amp;ref_topic=6080329</a> <a href="https://play.google.com/store/apps/details?id=com.google.android.apps.messaging&amp;hl=en">https://play.google.com/store/apps/details?id=com.google.android.apps.messaging&amp;hl=en</a></p> 

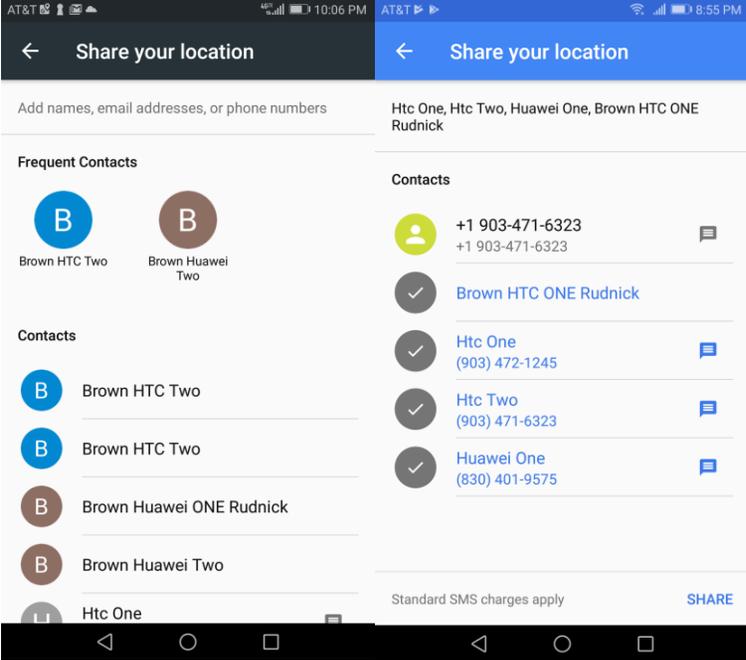
## Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products

US9408055B2	HTC
	<p data-bbox="533 618 842 651"><b>Start a conversation</b></p> <ol data-bbox="533 672 1220 829" style="list-style-type: none"><li>1. On your Android phone or tablet, open the Hangouts app .</li><li>2. At the bottom right, tap Add  &gt; New Conversation .</li><li>3. Type and select a person's name.</li><li>4. Enter your message. You can also add emojis, photos, your location, or a sticker.</li><li>5. Tap Send .</li></ol> <p data-bbox="520 862 1751 919"><a href="https://support.google.com/hangouts/answer/3115553?hl=en&amp;ref_topic=6386410&amp;co=GENIE.Platform%3DAndroid&amp;oco=0">https://support.google.com/hangouts/answer/3115553?hl=en&amp;ref_topic=6386410&amp;co=GENIE.Platform%3DAndroid&amp;oco=0</a></p> <ol data-bbox="533 980 1352 1138" style="list-style-type: none"><li>1. Open the Hangouts app .</li><li>2. At the bottom, tap Add  &gt; <b>New conversation</b> &gt; <b>New group</b>.</li><li>3. Enter and select the names, phone numbers, or email addresses of people in your group.</li><li>4. Tap Done .</li></ol> <p data-bbox="520 1149 1751 1206"><a href="https://support.google.com/hangouts/answer/3111943?hl=en&amp;ref_topic=6386410&amp;co=GENIE.Platform%3DAndroid&amp;oco=1">https://support.google.com/hangouts/answer/3111943?hl=en&amp;ref_topic=6386410&amp;co=GENIE.Platform%3DAndroid&amp;oco=1</a></p> 

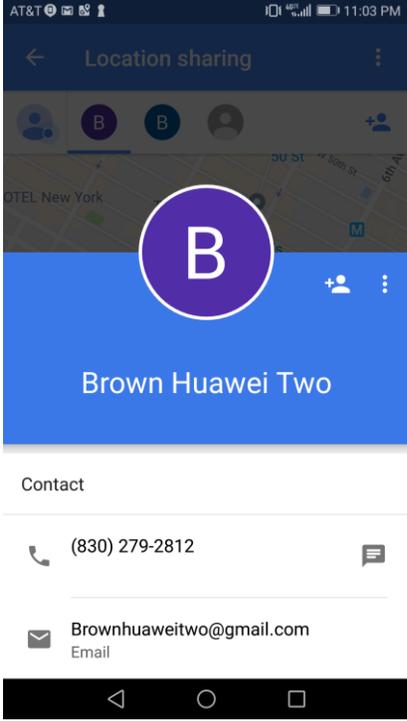
**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC
	<p data-bbox="541 235 903 276"><b>Contact someone</b></p> <p data-bbox="541 305 1192 332">You can call, email, or send text messages to your contacts.</p> <ol data-bbox="552 365 982 625" style="list-style-type: none"><li data-bbox="552 365 982 397">1. Open your device's Contacts app .</li><li data-bbox="552 410 840 443">2. Tap a contact in the list.</li><li data-bbox="552 456 781 488">3. Choose an option:<ul data-bbox="583 503 808 625" style="list-style-type: none"><li data-bbox="583 503 688 535">• Call </li><li data-bbox="583 548 709 581">• Email </li><li data-bbox="583 594 808 625">• New message </li></ul></li></ol> <p data-bbox="520 641 1470 673"><a href="https://support.google.com/nexus/answer/6118731?hl=en&amp;ref_topic=6118711">https://support.google.com/nexus/answer/6118731?hl=en&amp;ref_topic=6118711</a></p> <p data-bbox="510 711 1029 743"><b><u>Exemplary Google Maps Screenshots:</u></b></p>

**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC
	 <p>For example, the Accused Products include software that obtains contact information including the phone numbers . Furthermore, these phone calls can merge multiple parties into a conference call.</p>

**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC
	
<p>[28B] facilitating initiation of Internet Protocol (IP) based communication between the first device and the respective second devices by using the respective telephone numbers to send, from</p>	<p>HTC infringes directly and/or indirectly by performing, inducing others to perform, and/or contributing to the performance of facilitating initiation of Internet Protocol (IP) based communication between the first device and the respective second devices by using the respective telephone numbers to send, from the first device to the second devices, respective Short Message Service (SMS) messages including a telephone number of the first device and information usable by the respective second device to send IP-based communication to the first device. See claim 1[B], which is incorporated herein by reference in its entirety.</p> <p>For example, the Accused Products utilize SMS-based messages to initiate IP communication between participants of Maps location sharing. For example, both Android Messages and Hangouts, in conjunction with Maps, utilize SMS messages, including group messages from one device to several devices, to send an</p>

**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC
<p>the first device to the second devices, respective Short Message Service (SMS) messages including a telephone number of the first device and information usable by the respective second device to send IP-based communication to the first device;</p>	<p>SMS message, with additional information, to a contact.</p> <p>D2 Technologies Showcases its mCUE IP Communications Interface over WiMAX on HTC's EVO 4G Android Smartphone</p> <p><small>Amsterdam, Netherlands (WiMAX Forum Global Congress) and Santa Barbara, CA – June 14, 2010 —D2 Technologies, the market leader in embedded IP communications software platforms, today announced that it is holding private demonstrations of its mCUE® converged communications client for mobile devices and handsets on the HTC EVO™ 4G smartphone on Thursday, June 17 at the WiMAX Forum® Global Congress in Amsterdam. D2's mCUE on the HTC EVO, the first 4G phone to be introduced in the United States, was configured and installed in less than a week – clearly illustrating how OEMs and ODMs can more rapidly develop Android™-based devices by choosing to incorporate the converged presence-based communications user interface (CUI).</small></p> <p><a href="http://www.d2tech.com/press-releases-year.html?Y=2010">http://www.d2tech.com/press-releases-year.html?Y=2010</a></p>

**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

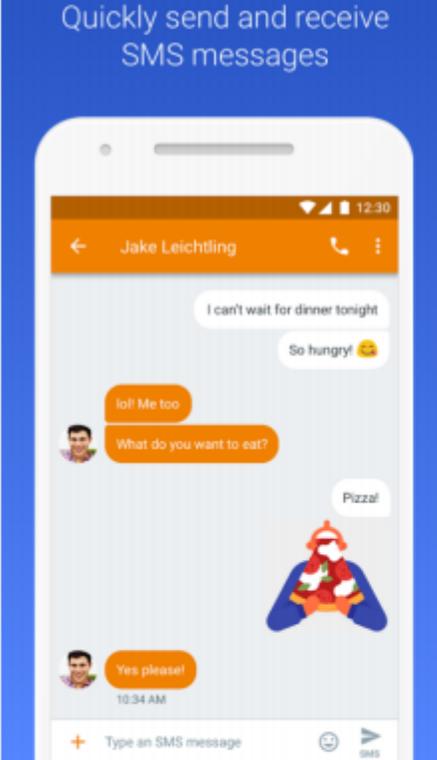
US9408055B2	HTC
	<p data-bbox="527 233 1289 302">Siemens Enterprise Communications and HTC Simplify Device Choice for Mobile UC   </p> <p data-bbox="527 329 1035 350">Reston, VA and Frankfurt, Germany, Feb 19, 2013</p> <p data-bbox="527 380 1451 464"><b>Enterasys' Mobile IAM and MDM connect™ BYOD solutions now support AirWatch to onboard and manage mobile devices and applications</b></p> <p data-bbox="527 482 1514 643">In an ongoing effort to support today's increasingly mobile workforce, Siemens Enterprise Communications and HTC Corporation today announced a strategic global partnership to make it easier for enterprises to embrace mobile unified communications (UC) on HTC enterprise-enabled devices. This partnership makes it even simpler for enterprises to embrace a BYOD strategy for mobile UC, since HTC's popular consumer Android smartphones now fully support Siemens Enterprise Communications' OpenScape Mobile and OpenScape Web Collaboration solutions.</p> <p data-bbox="527 693 625 714"><b>Key Facts</b></p> <ul data-bbox="527 730 1514 959" style="list-style-type: none"> <li>▪ Siemens Enterprise Communications OpenScape Mobile and OpenScape Web Collaboration solutions will be validated on select HTC devices to increase users' confidence that their chosen device will work seamlessly with their mobile communication tools</li> <li>▪ Siemens Enterprise Communications customers will have a simplified process to secure validated HTCPro devices supporting Siemens Enterprise Communications solutions</li> <li>▪ Siemens Enterprise Communications and HTC will collaborate to simplify deployment of mobile UC through joint marketing and fulfillment efforts</li> <li>▪ This collaboration will take place through HTCPro, a program that provides mobile solutions for companies and their employees and ensures that HTC's entire portfolio is business-ready</li> </ul> <p data-bbox="527 1000 1514 1068">Siemens Enterprise Communications OpenScape Mobile and OpenScape Web Collaboration solutions will be validated on select HTC devices to increase users' confidence that their chosen device will work seamlessly with their mobile communication tools.</p> <p data-bbox="512 1083 1713 1114"><a href="http://www.unify.com/us/news/E853B94A-F94F-4ADA-98DA-8C80BB965953/?isarchive=1">http://www.unify.com/us/news/E853B94A-F94F-4ADA-98DA-8C80BB965953/?isarchive=1</a></p>

## Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products

US9408055B2	HTC
	<p>Android Messages makes it easy to communicate with anyone by using SMS, MMS, and more. Stay in touch with friends and family, send group texts, and share your favorite pictures, videos, audio messages.</p> <p>· <b>Enhanced features:</b> On supported carriers, you can send messages over Wi-Fi or your data network, see when friends have read your message, and more.</p> <p><a href="https://play.google.com/store/apps/details?id=com.google.android.apps.messaging&amp;hl=en">https://play.google.com/store/apps/details?id=com.google.android.apps.messaging&amp;hl=en</a></p>



## Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products

US9408055B2	HTC
	 <p data-bbox="520 998 1617 1031"><a href="https://play.google.com/store/apps/details?id=com.google.android.apps.messaging&amp;hl=en">https://play.google.com/store/apps/details?id=com.google.android.apps.messaging&amp;hl=en</a></p> <h3 data-bbox="541 1084 1087 1133">Get started with Hangouts</h3> <p data-bbox="541 1149 751 1172">You can use Hangouts to:</p> <ul data-bbox="541 1193 1134 1282" style="list-style-type: none"><li>• Start a chat conversation or video call.</li><li>• Make phone calls using Wi-Fi or data.</li><li>• Send text messages with your <a href="#">Google Voice</a> or <a href="#">Project Fi</a> phone number.</li></ul> <p data-bbox="541 1302 1465 1351">Hangouts sync automatically across devices. If you start a Hangout on your computer, you can continue your chat on another device, like your phone.</p> <p data-bbox="520 1360 1512 1393"><a href="https://support.google.com/hangouts/answer/2944865?hl=en&amp;ref_topic=6386410">https://support.google.com/hangouts/answer/2944865?hl=en&amp;ref_topic=6386410</a></p>

## Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products

US9408055B2	HTC
	<p data-bbox="541 264 840 305"><b>Start a Hangout</b></p> <p data-bbox="541 321 1056 339">You can send and receive messages with one person or multiple people.</p> <p data-bbox="554 402 890 420">COMPUTER <b>ANDROID</b> IPHONE &amp; IPAD</p> <hr data-bbox="541 443 1142 444"/> <p data-bbox="541 553 808 581"><b>Start a conversation</b></p> <ol data-bbox="548 599 1136 737" style="list-style-type: none"><li>1. On your Android phone or tablet, open the Hangouts app .</li><li>2. At the bottom right, tap Add  &gt; New Conversation .</li><li>3. Type and select a person's name.</li><li>4. Enter your message. You can also add emojis, photos, your location, or a sticker.</li><li>5. Tap Send .</li></ol> <p data-bbox="525 753 1749 810"><a href="https://support.google.com/hangouts/answer/3115553?hl=en&amp;ref_topic=6386410&amp;co=GENIE.Platform%3DAndroid&amp;oco=0">https://support.google.com/hangouts/answer/3115553?hl=en&amp;ref_topic=6386410&amp;co=GENIE.Platform%3DAndroid&amp;oco=0</a></p> <ul data-bbox="560 862 1566 1013" style="list-style-type: none"><li>• Connect your Google Voice account for phone calling, SMS texting, and voicemail integration.</li><li>• Keep in touch with contacts across Android, iOS, and the web, and sync chats across all your devices.</li><li>• Message contacts anytime, even if they're offline.</li></ul> <p data-bbox="525 1045 1402 1073"><a href="https://play.google.com/store/apps/details?id=com.google.android.talk&amp;hl=en">https://play.google.com/store/apps/details?id=com.google.android.talk&amp;hl=en</a></p> <ol data-bbox="548 1083 1499 1344" style="list-style-type: none"><li>1. Open the Hangouts app .</li><li>2. At the bottom right, tap Add .</li><li>3. Choose <b>New SMS</b>.</li><li>4. Type the name or phone number. If you're traveling, use the "+" sign and country code when texting.</li><li>5. Tap the number or contact.</li><li>6. Tap Continue .</li><li>7. Type your message and tap Send .</li></ol> <p data-bbox="525 1357 1262 1385"><a href="https://support.google.com/hangouts/answer/3441321?hl=en">https://support.google.com/hangouts/answer/3441321?hl=en</a></p> <p data-bbox="514 1390 911 1417"><b><u>Google Maps Share Location</u></b></p>

**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC
	<p>Google Latitude, Google Plus, Google Hangouts, Google Messages, and Android Messenger, Google Duo, Google Allo, and Google Chrome, the Accused Products require a user to join the corresponding network by signing-in to the device with an identifier (e.g., Google Account). The sign-in process takes place within the Google Maps software on the Accused Product or by navigating to maps.google.com within the Google Chrome browser on the Accused Product. Alternatively, the sign-in process may partially or completely take place using credentials already provided when the user associates a Google Account with the Accused Product, e.g., during initial setup of the Accused Product. Subject to discovery, one or more additional or substitute identifiers may correspond to the group. The sign-in process involves a user entering its Google Account and additional authentication data on the interface of the Accused Product and sending a message containing the Google Account and additional authentication data over a network to members of a group. The group comprises the multiple identifiers, individuals, profiles, and/or devices associated with the group. Further regarding Google Maps, Google Latitude, Google Plus, Google Hangouts, Google Messages, and Android Messenger, Google Duo, Google Allo, and Google Chrome, the Accused Products alternatively require a user to send a request containing the identifier (e.g., Google Account). Subject to discovery, additional identifiers may be assigned or used to correspond to the group. The request may be an invitation or message that associates a Google Account with one or more Google Accounts for the purposes of sharing locations within the group. The group comprises the multiple identifiers, individuals, profiles, and/or devices associated with the group.</p> <p><b><u>Exemplary Support for Google Maps:</u></b></p>

## Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products

US9408055B2	HTC
	<p data-bbox="541 240 903 259">COMPUTER <b>ANDROID</b> IPHONE &amp; IPAD</p> <hr data-bbox="541 276 1417 279"/> <h3 data-bbox="531 329 949 362">If they have a Google Account</h3> <ol data-bbox="541 378 1291 630" style="list-style-type: none"><li data-bbox="541 378 1117 397">1. If you haven't already, add their Gmail address to your <a href="#">Google Contacts</a>.</li><li data-bbox="541 410 1291 430">2. On your Android phone or tablet, open the Google Maps app  and sign in. <a href="#">Learn how to sign in.</a></li><li data-bbox="541 443 961 462">3. Tap Menu ≡ &gt; <b>Location sharing</b> &gt; Add People .</li><li data-bbox="541 475 934 495">4. Choose how long you want to share your location.</li><li data-bbox="541 508 1050 557">5. Tap <b>Select People</b>.<ul data-bbox="562 535 1050 557" style="list-style-type: none"><li data-bbox="562 535 1050 557">• If you're asked about your contacts, give Google Maps access.</li></ul></li><li data-bbox="541 573 829 592">6. Choose who you want to share with.</li><li data-bbox="541 605 640 625">7. Tap <b>Share</b>.</li></ol> <h3 data-bbox="531 678 1024 711">If they don't have a Google Account</h3> <ol data-bbox="541 727 1411 841" style="list-style-type: none"><li data-bbox="541 727 1291 747">1. On your Android phone or tablet, open the Google Maps app  and sign in. <a href="#">Learn how to sign in.</a></li><li data-bbox="541 760 961 779">2. Tap Menu ≡ &gt; <b>Location sharing</b> &gt; Add People .</li><li data-bbox="541 792 1411 841">3. Tap More ... &gt; <b>Copy to clipboard</b>. People with this link can see your location for as long as you choose, up to 72 hours.</li></ol> <h3 data-bbox="531 881 819 914">Share using another app</h3> <p data-bbox="531 922 1102 941">You can also share through messaging apps. Tap More ... &gt; select an app.</p> <h3 data-bbox="531 995 709 1027">Stop sharing</h3> <ol data-bbox="541 1044 1102 1133" style="list-style-type: none"><li data-bbox="541 1044 793 1063">1. Open the Google Maps app .</li><li data-bbox="541 1076 819 1096">2. Tap Menu ≡ &gt; <b>Location sharing</b>.</li><li data-bbox="541 1109 1102 1128">3. Next to the person with whom you want to stop sharing, tap Remove ✕.</li></ol> <p data-bbox="510 1141 1703 1174"><a href="https://support.google.com/maps/answer/7326816?co=GENIE.Platform%3DAndroid&amp;oco=1">https://support.google.com/maps/answer/7326816?co=GENIE.Platform%3DAndroid&amp;oco=1</a></p>

## Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products

US9408055B2	HTC
	<h3>Share your E.T.A</h3> <p>After you start your drive, you can share your destination, estimated arrival time, and where you are on the route.</p> <ol style="list-style-type: none"><li>1. Open the Google Maps app .</li><li>2. Set a driving destination. <a href="#">Learn how to navigate to a place.</a></li><li>3. After you start navigation, tap More ^ &gt; <b>Share trip progress.</b></li><li>4. Choose a person from the list.</li><li>5. Tap <b>Share.</b></li><li>6. Location Sharing will stop when you reach your destination or stop navigating.</li></ol> <ul style="list-style-type: none"><li>• To stop sharing before you arrive, tap More ^ &gt; <b>Stop sharing.</b></li></ul> <h3>See where someone is</h3> <p>If someone shares their location with you, you can see them on the map.</p> <ol style="list-style-type: none"><li>1. Open the Google Maps app .</li><li>2. Tap Menu ≡ &gt; <b>Location sharing.</b></li><li>3. Choose someone.</li></ol> <ul style="list-style-type: none"><li>• To see an updated location, tap on a friend's icon &gt; More ⋮ &gt; <b>Refresh.</b></li></ul> <h3>Stop seeing someone's location</h3> <ol style="list-style-type: none"><li>1. Open the Google Maps app .</li><li>2. On the map, tap their icon.</li><li>3. At the bottom, tap More ^ .</li><li>4. To temporarily hide someone, tap <b>Hide from map.</b> You can stop hiding them at any time.</li></ol> <p><b>Note:</b> You can stop someone's location from ever appearing on your map. <a href="#">Learn how to block another person's account.</a></p> <p><a href="https://support.google.com/maps/answer/7326816?co=GENIE.Platform%3DAndroid&amp;oco=1">https://support.google.com/maps/answer/7326816?co=GENIE.Platform%3DAndroid&amp;oco=1</a></p>

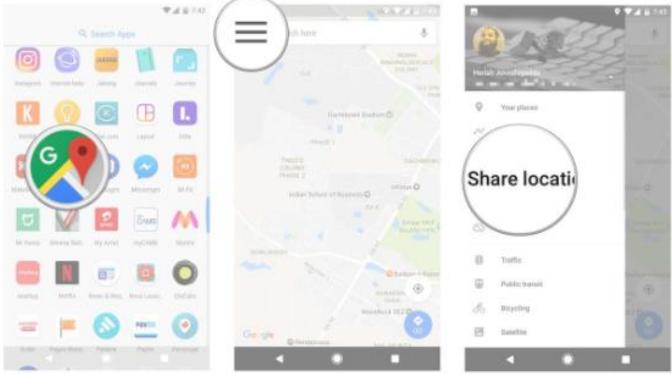
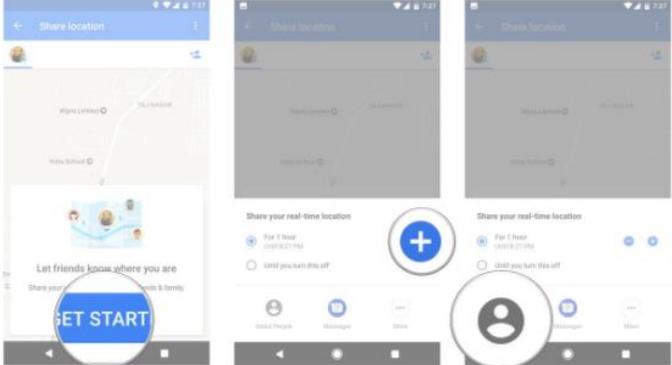
## Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products

US9408055B2	HTC
	<h3>Create a list of places</h3> <p>In Google Maps, you can create a list of places, like your favorite places or places you want to visit.</p> <p>COMPUTER <b>ANDROID</b> IPHONE &amp; IPAD</p> <hr/> <h3>Make a new list</h3> <ol style="list-style-type: none"><li>1. On your Android phone or tablet, open the Google Maps app .</li><li>2. Tap Menu  &gt; <b>Your places</b> &gt; <b>Saved</b>.</li><li>3. In the bottom right, tap Add .</li><li>4. Enter a name and description.</li><li>5. Tap <b>Save</b>.</li></ol> <h3>Save a place to a list</h3> <ol style="list-style-type: none"><li>1. Open the Google Maps app .</li><li>2. Search for a place or tap it on the map.</li><li>3. At the bottom, tap the place's name or address.</li><li>4. Tap <b>Save</b>.</li><li>5. Choose a list. To create a new list, tap <b>New list</b> .</li></ol> <h3>See your lists</h3> <ol style="list-style-type: none"><li>1. Open the Google Maps app .</li><li>2. Tap Menu  &gt; <b>Your places</b> &gt; <b>Saved</b>.</li></ol> <p><a href="https://support.google.com/maps/answer/7280933?hl=en&amp;ref_topic=7301134&amp;co=GENIE.Platform%3DAndroid&amp;oco=1">https://support.google.com/maps/answer/7280933?hl=en&amp;ref_topic=7301134&amp;co=GENIE.Platform%3DAndroid&amp;oco=1</a></p>

## Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products

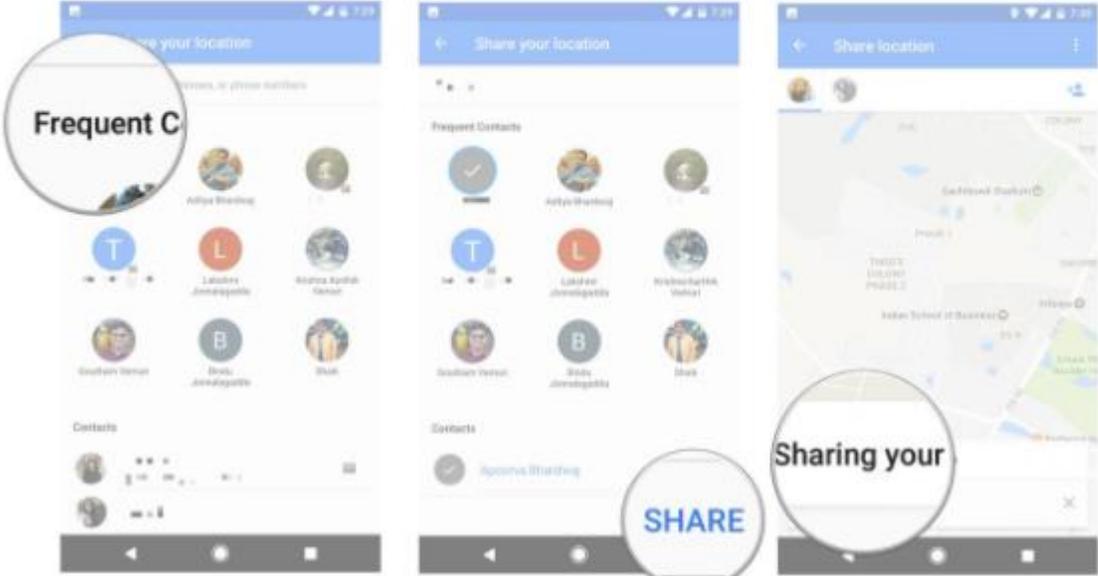
US9408055B2	HTC
	<h3 data-bbox="541 245 877 282">Hide or share lists</h3> <p data-bbox="541 310 909 334"><b>Note:</b> You can't share starred places.</p> <ol data-bbox="554 367 1251 472" style="list-style-type: none"><li data-bbox="554 367 890 391">1. Open the Google Maps app .</li><li data-bbox="554 407 968 431">2. Tap Menu  &gt; <b>Your places</b> &gt; <b>Saved</b>.</li><li data-bbox="554 448 1251 472">3. Next to the list you want to share, tap More  &gt; choose an option:</li></ol> <ul data-bbox="583 488 1682 626" style="list-style-type: none"><li data-bbox="583 488 1440 513">• <b>Hide/Show on your map:</b> Display or hide your saved places when looking at the map.</li><li data-bbox="583 529 1058 553">• <b>Share list:</b> Allow others to see your saved list.</li><li data-bbox="583 570 1682 626">• <b>Sharing options:</b> You can make your list public, private, or shared. To let anyone with the link see your list, tap <b>Shared</b>. To let anyone find and follow your list, tap <b>Public</b>.</li></ul> <h3 data-bbox="541 699 768 737">Follow a list</h3> <p data-bbox="541 764 1728 821">If you follow a list made by someone else, their saved places will show up in Your Places. The places will also appear as suggested locations in Google Maps.</p> <h3 data-bbox="541 878 915 915">Follow a list using a link</h3> <ol data-bbox="554 935 1356 1040" style="list-style-type: none"><li data-bbox="554 935 957 959">1. Tap on the link you received to open it.</li><li data-bbox="554 976 1272 1000">2. Tap <b>Follow</b>. This list will now be added to the group of lists you follow.</li><li data-bbox="554 1016 1356 1040">3. <b>Optional:</b> To unfollow a list someone shared with you, tap the list &gt; <b>Following</b>.</li></ol> <h3 data-bbox="541 1097 926 1135">See lists made by others</h3> <p data-bbox="541 1154 1335 1179">If a user has any Google Maps lists that were made public, you can follow them.</p> <ol data-bbox="554 1211 1136 1317" style="list-style-type: none"><li data-bbox="554 1211 1136 1235">1. Tap on the name of a user whose list you want to follow.</li><li data-bbox="554 1252 674 1276">2. Tap <b>Lists</b>.</li><li data-bbox="554 1292 1136 1317">3. Tap on the list you want to follow &gt; More  &gt; <b>Follow</b>.</li></ol> <p data-bbox="512 1333 1902 1398"><a href="https://support.google.com/maps/answer/7280933?hl=en&amp;ref_topic=7301134&amp;co=GENIE.Platform%3DAndroid&amp;oco=1">https://support.google.com/maps/answer/7280933?hl=en&amp;ref_topic=7301134&amp;co=GENIE.Platform%3DAndroid&amp;oco=1</a></p>

## Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products

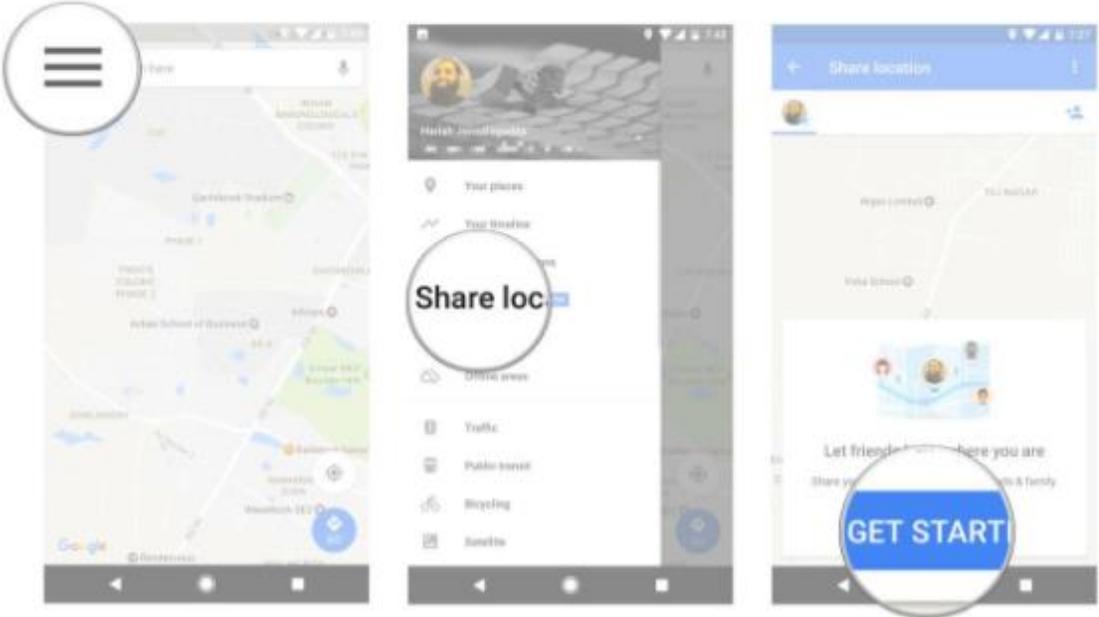
US9408055B2	HTC
	<h3 data-bbox="520 240 1150 272">How to share your location in Google Maps</h3> <ol data-bbox="520 305 1150 389" style="list-style-type: none"><li>1. Open Google Maps from the app drawer or the home screen.</li><li>2. Tap the hamburger menu (the three horizontal lines) on the top left corner of the screen.</li><li>3. Select Share location.</li></ol>  <ol data-bbox="520 828 1150 933" style="list-style-type: none"><li>4. Tap Get Started.</li><li>5. Use the + icon to select a time period or select the Until you turn this off setting to share your location indefinitely.</li><li>6. Tap Select People.</li></ol>  <p data-bbox="520 1339 1360 1372"><a href="https://www.androidcentral.com/how-share-location-google-maps">https://www.androidcentral.com/how-share-location-google-maps</a></p>



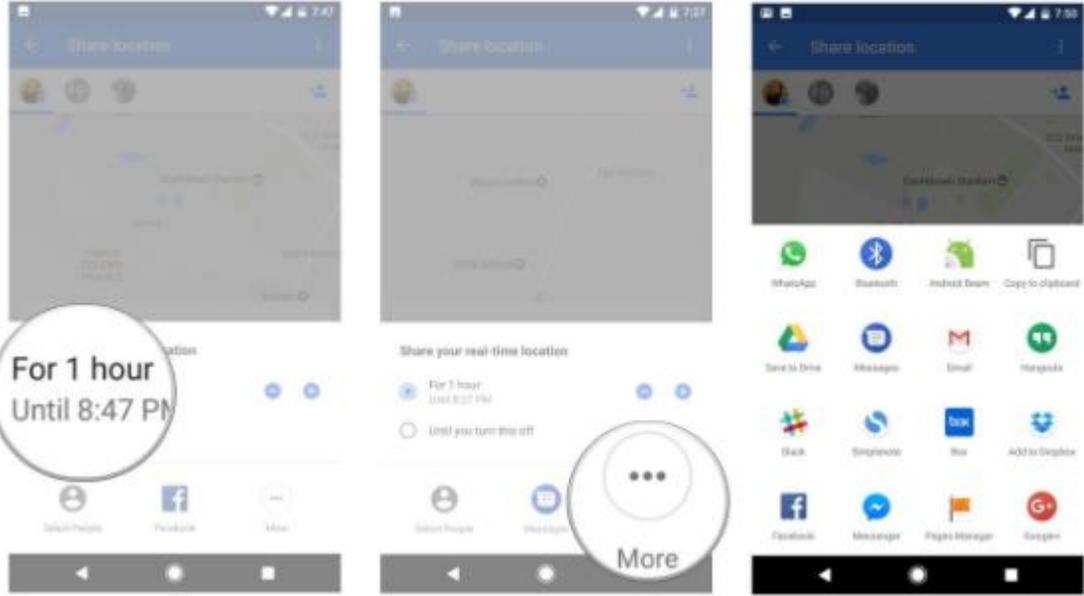
## Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products

US9408055B2	HTC
	<p data-bbox="527 253 1577 310">7. You'll see a list of your frequent contacts at the top, along with a full list of contacts. Pick the contacts by tapping their name.</p> <p data-bbox="527 339 1457 367">8. Once you've selected the contacts you want to share your location to, tap <b>Share</b>.</p> <p data-bbox="527 396 1419 423">9. You'll see a message saying that the selected contact can view your location.</p>  <p data-bbox="506 1065 1356 1092"><a href="https://www.androidcentral.com/how-share-location-google-maps">https://www.androidcentral.com/how-share-location-google-maps</a></p>

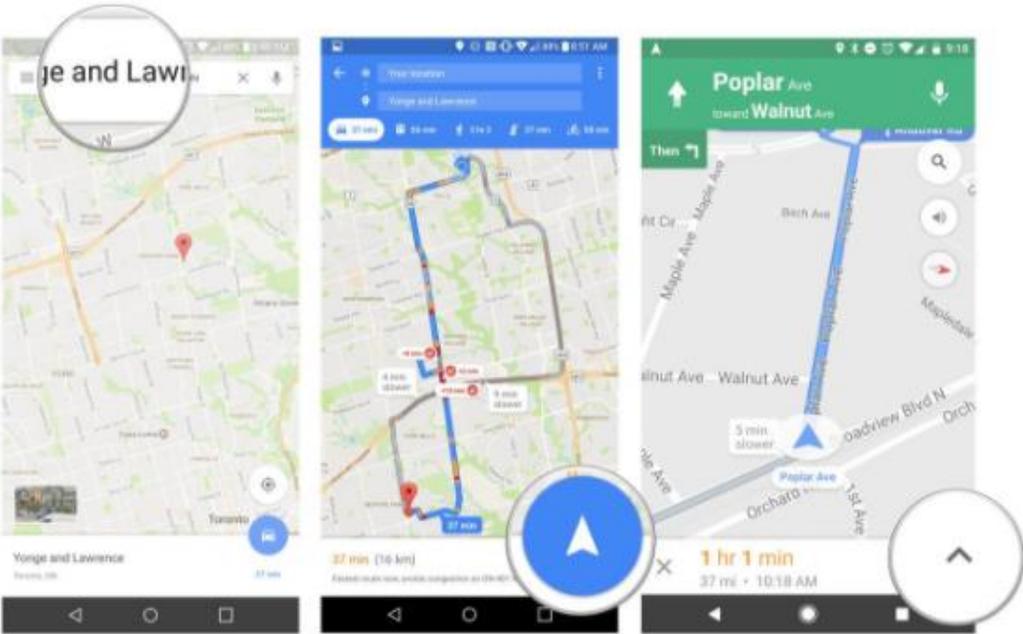
## Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products

US9408055B2	HTC
	<h3 data-bbox="527 245 1255 289">How to create a shareable link</h3> <p data-bbox="527 334 1461 358">You can also create a link and use it to share your location easily. Here's how to do it:</p> <ol data-bbox="527 412 1234 548" style="list-style-type: none"><li data-bbox="527 412 1234 440">1. Tap the hamburger menu on the top left corner of the screen.</li><li data-bbox="527 467 800 495">2. Select Share location.</li><li data-bbox="527 522 737 550">3. Tap Get Started.</li></ol>  <p data-bbox="512 1232 1356 1263"><a href="https://www.androidcentral.com/how-share-location-google-maps">https://www.androidcentral.com/how-share-location-google-maps</a></p>

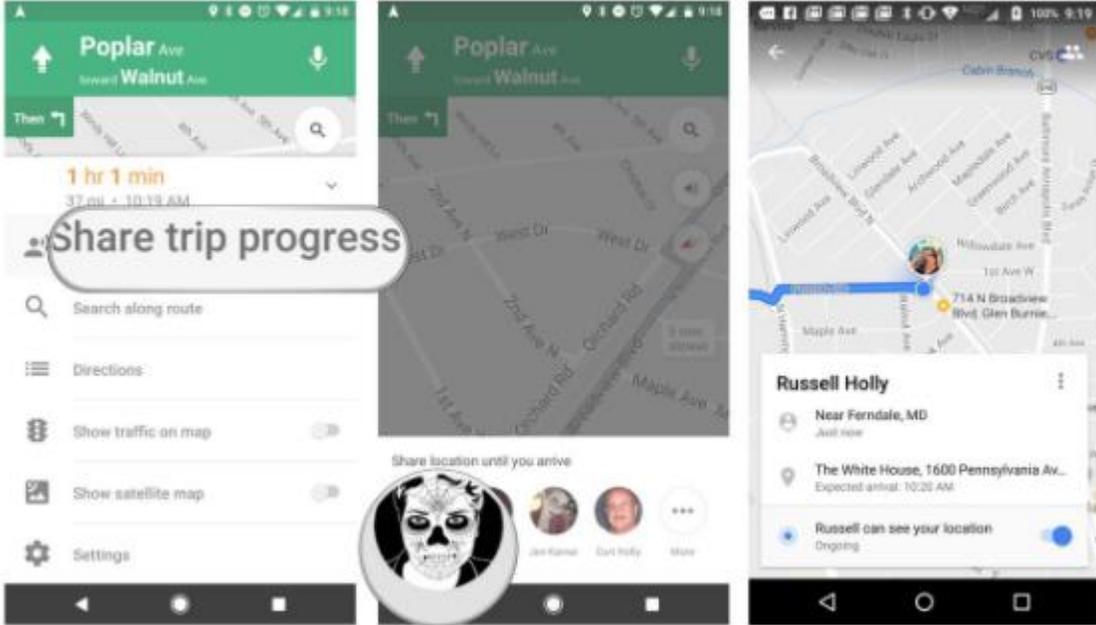
## Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products

US9408055B2	HTC
	<p data-bbox="520 245 1213 272">4. Select the <b>amount of time</b> you want to share your location.</p> <p data-bbox="520 302 680 329">5. Tap <b>More</b>.</p> <p data-bbox="520 358 1633 418">6. Select <b>your app of choice</b> to create and send a <b>unique URL</b> that broadcasts your current location. You can email it, send the link via Messenger, or even tweet it to the intended recipient.</p>  <p data-bbox="506 1084 1360 1117"><a href="https://www.androidcentral.com/how-share-location-google-maps">https://www.androidcentral.com/how-share-location-google-maps</a></p>

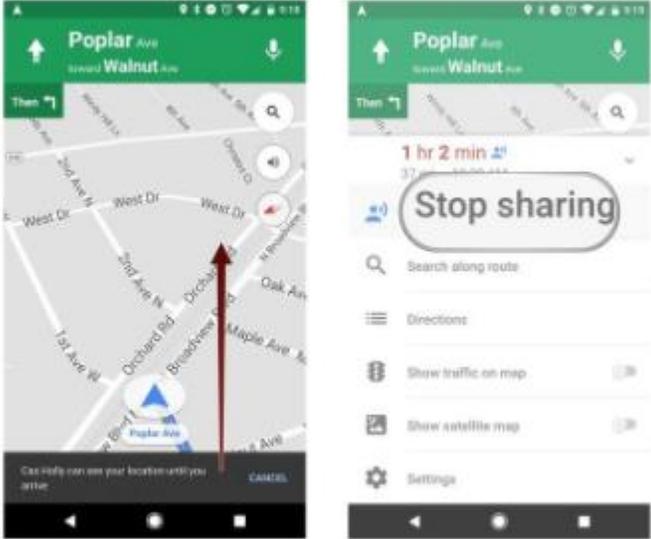
## Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products

US9408055B2	HTC
	<h3 data-bbox="527 240 1428 337">How to share your navigation directions while you walk, drive or transit</h3> <p data-bbox="527 375 1556 464">One of the best ways to share your location with someone is during a drive, walk or transit. If you're meeting a friend or family member somewhere, or navigating towards their home, Google Maps lets you share your location with them for the duration of the trip. It's magic!</p> <ol data-bbox="527 513 1396 643" style="list-style-type: none"><li>1. In the <b>search bar</b> enter your destination.</li><li>2. Pick your navigation type (drive, transit, walk) and press the blue <b>navigate</b> button.</li><li>3. Tap the arrow next to the <b>time-to-destination</b> number at the bottom of the screen.</li></ol>  <p data-bbox="512 1328 1356 1360"><a href="https://www.androidcentral.com/how-share-location-google-maps">https://www.androidcentral.com/how-share-location-google-maps</a></p>

## Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products

US9408055B2	HTC
	<p data-bbox="527 277 835 305">4. Tap Share trip progress.</p> <p data-bbox="527 334 1150 362">5. Choose one or more contacts to share trip progress.</p>  <p data-bbox="527 1065 1360 1133">You can also stop sharing your location with someone before a trip ends. <a href="https://www.androidcentral.com/how-share-location-google-maps">https://www.androidcentral.com/how-share-location-google-maps</a></p>

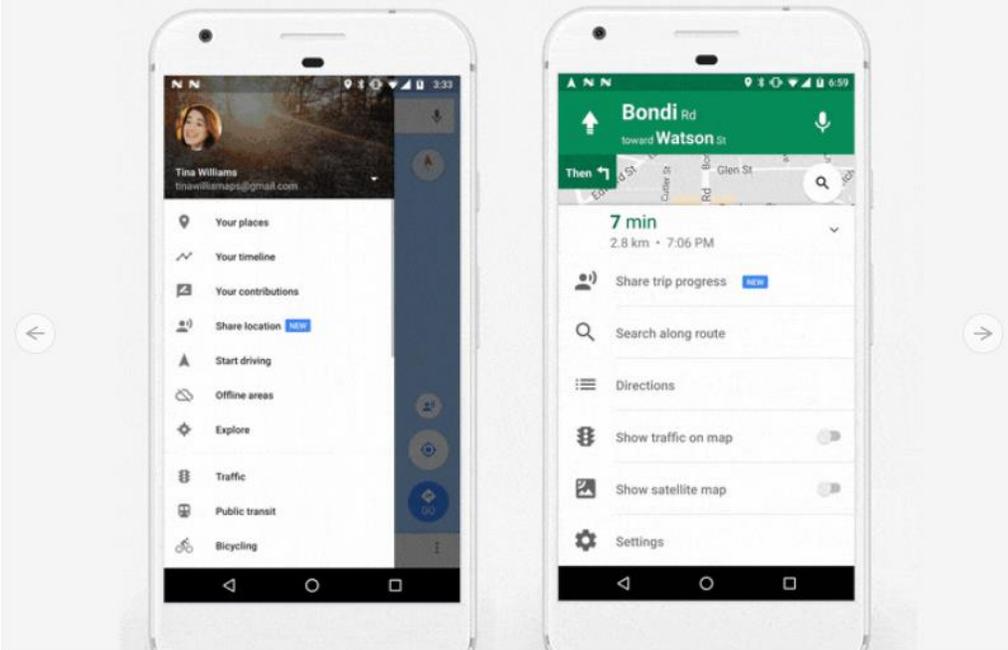
## Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products

US9408055B2	HTC
	<ol style="list-style-type: none"><li data-bbox="533 245 1465 272">1. Tap the arrow next to the <b>time-to-destination</b> number at the bottom of the screen.</li><li data-bbox="533 302 768 329">2. Tap <b>Stop sharing</b>.</li></ol> <div data-bbox="747 383 1398 922"></div> <p data-bbox="541 976 638 1003">That's it!</p> <p data-bbox="541 1045 1608 1073">Are you excited that location sharing is back in Google Maps? How often do you use the feature?</p> <p data-bbox="512 1084 1356 1112"><a href="https://www.androidcentral.com/how-share-location-google-maps">https://www.androidcentral.com/how-share-location-google-maps</a></p> <p data-bbox="512 1195 1419 1222">As shown below, a group may also be defined within Google Contacts.</p>

## Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products

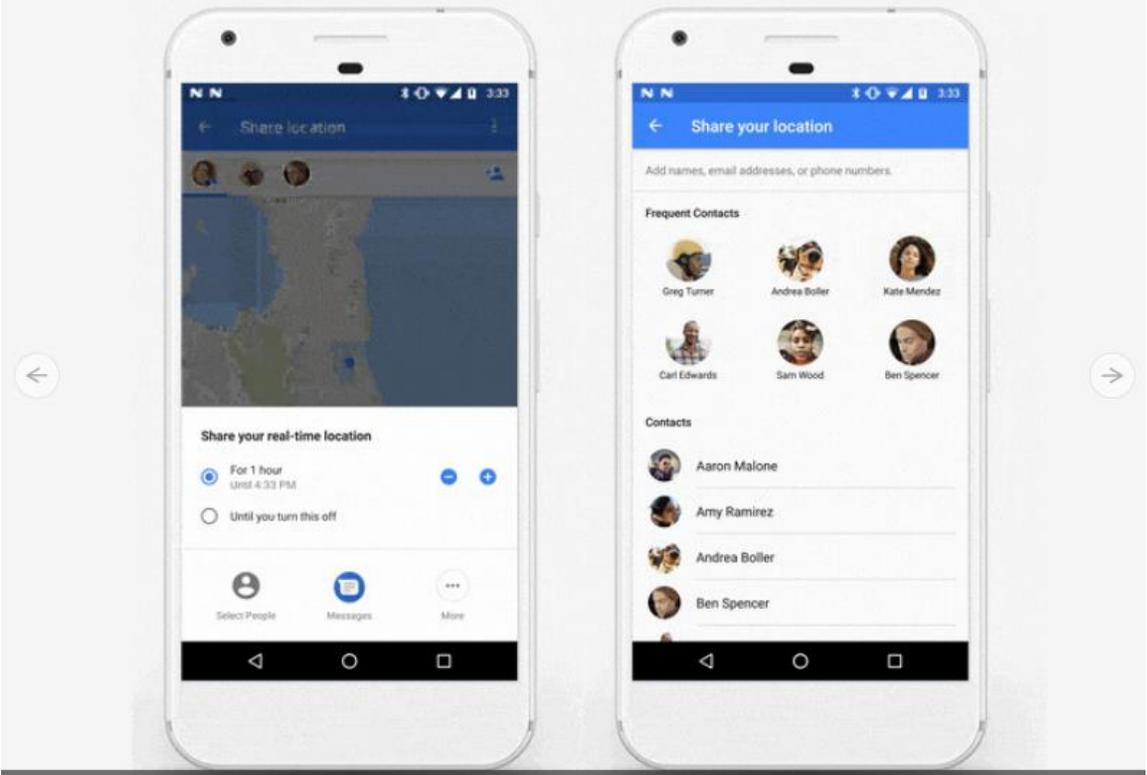
US9408055B2	HTC
	<p data-bbox="541 237 894 277"><b>See your contacts</b></p> <ol data-bbox="552 305 968 378" style="list-style-type: none"><li>1. Open your device's Contacts app .</li><li>2. Tap Menu .</li></ol> <ul data-bbox="552 410 1734 605" style="list-style-type: none"><li>• <b>See contacts by label:</b> Choose a label from the list.</li><li>• <b>See contacts for another account:</b> Tap Down arrow  &gt; pick an account.</li><li>• <b>See the contacts for all your accounts:</b> Choose <b>All contacts</b>. <b>Tip:</b> If you have multiple contacts with the same information, the information will be grouped into one contact.</li><li>• <b>See your Google Account contacts on the web:</b> Go to <a href="#">Google Contacts</a> .</li></ul> <p data-bbox="516 638 1535 670"><a href="https://support.google.com/android/answer/6118731?hl=en&amp;ref_topic=6118711">https://support.google.com/android/answer/6118731?hl=en&amp;ref_topic=6118711</a></p> <p data-bbox="541 716 894 756"><b>Label your contacts</b></p> <p data-bbox="541 781 982 805">You can group contacts together using labels.</p> <ol data-bbox="552 833 926 938" style="list-style-type: none"><li>1. Open your device's Contacts app .</li><li>2. Tap Menu  &gt; <b>Create label</b>.</li><li>3. Enter a label name and tap <b>Ok</b>.</li></ol> <ul data-bbox="552 971 1713 1036" style="list-style-type: none"><li>• <b>Add one contact to a label:</b> Tap Add contact  &gt; choose a contact.</li><li>• <b>Add multiple contacts to a label:</b> Tap Add contact  &gt; touch and hold a contact &gt; tap the other contacts &gt; tap <b>Add</b>.</li></ul> <p data-bbox="516 1044 1535 1076"><a href="https://support.google.com/android/answer/6118731?hl=en&amp;ref_topic=6118711">https://support.google.com/android/answer/6118731?hl=en&amp;ref_topic=6118711</a></p> <p data-bbox="552 1130 940 1170"><b>Share your contacts</b></p> <ol data-bbox="562 1198 1045 1360" style="list-style-type: none"><li>1. Open your device's Contacts app .</li><li>2. Tap a contact in the list.</li><li>3. Tap More  &gt; <b>Share</b>.</li><li>4. Choose how you want to share the contact.</li></ol> <p data-bbox="516 1385 1535 1417"><a href="https://support.google.com/android/answer/6118731?hl=en&amp;ref_topic=6118711">https://support.google.com/android/answer/6118731?hl=en&amp;ref_topic=6118711</a></p>

## Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products

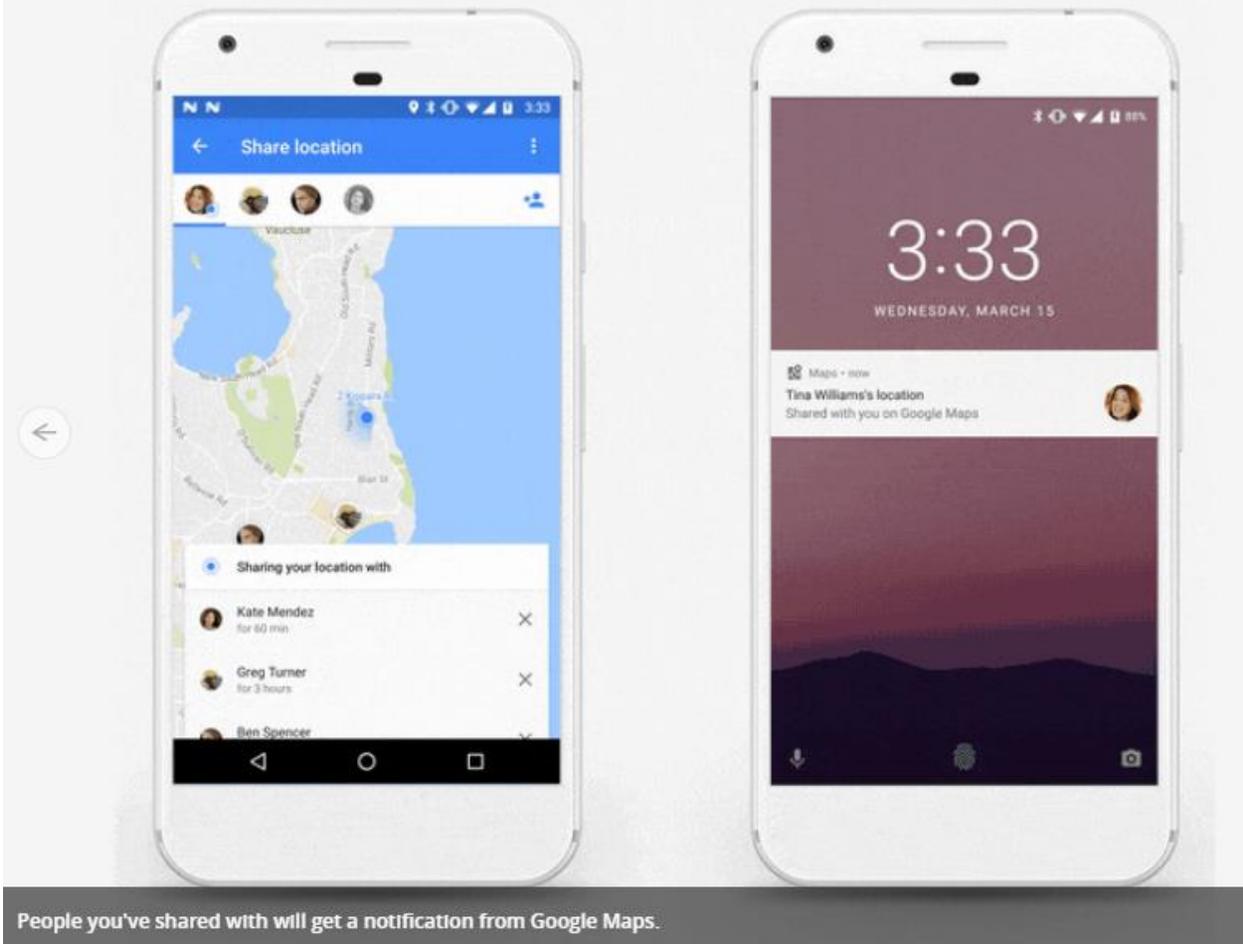
US9408055B2	HTC
	 <p>Ways to start location sharing: (left) from the app navigation drawer, and (right) from the menu, you can drag up while navigating somewhere.</p> <p><a href="https://arstechnica.com/gadgets/2017/03/location-sharing-finally-returns-to-google-maps/">https://arstechnica.com/gadgets/2017/03/location-sharing-finally-returns-to-google-maps/</a></p>



### Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products

US9408055B2	HTC
	 <p data-bbox="514 1023 1661 1063">You can pick how long you want to share your location for, and with whom you want to share it.</p> <p data-bbox="514 1068 1661 1096"><a href="https://arstechnica.com/gadgets/2017/03/location-sharing-finally-returns-to-google-maps/">https://arstechnica.com/gadgets/2017/03/location-sharing-finally-returns-to-google-maps/</a></p>

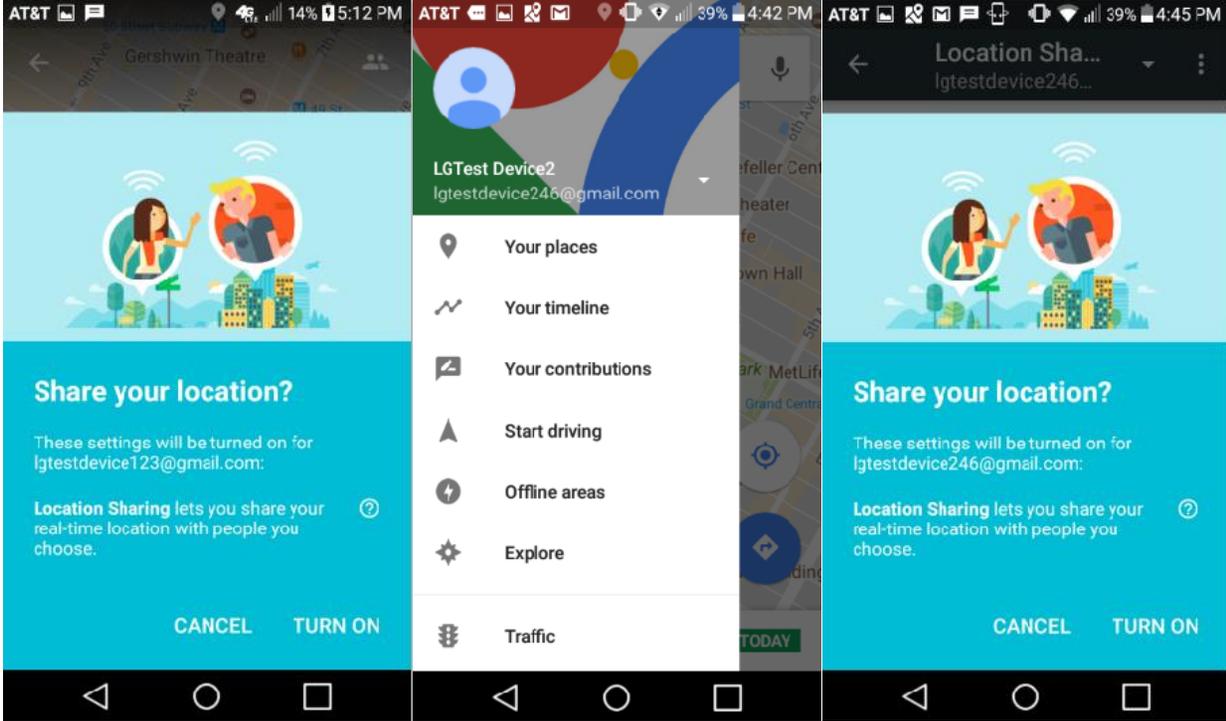
**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC
	 <p data-bbox="512 1143 1176 1170">People you've shared with will get a notification from Google Maps.</p> <p data-bbox="512 1187 1656 1219"><a href="https://arstechnica.com/gadgets/2017/03/location-sharing-finally-returns-to-google-maps/">https://arstechnica.com/gadgets/2017/03/location-sharing-finally-returns-to-google-maps/</a></p>

**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC
	<p>B-369</p>

**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC
	 <p>The image displays three screenshots from an HTC mobile application. The first screenshot (left) shows a location sharing prompt for 'lgtestdevice123@gmail.com' with 'CANCEL' and 'TURN ON' buttons. The second screenshot (middle) shows a menu with options: 'Your places', 'Your timeline', 'Your contributions', 'Start driving', 'Offline areas', 'Explore', and 'Traffic'. The third screenshot (right) shows a location sharing prompt for 'lgtestdevice246@gmail.com' with 'CANCEL' and 'TURN ON' buttons. All screenshots feature a blue header with an illustration of two people and a cityscape.</p> <p>Exemplary Source Code:  The above functionality is performed at least in part by the following publicly available source code and/or source code that invokes or is invoked by the following source code (or a substantially similar copy compiled and loaded onto the Accused Products by HTC): AGIS reserves the right to amend these contentions to include additional source code as discovery progresses and as additional source code is made available.</p>

## Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products

US9408055B2	HTC
	<h3 data-bbox="527 240 1016 293">Contacts Provider</h3> <p data-bbox="527 329 1472 589">The Contacts Provider is a powerful and flexible Android component that manages the device's central repository of data about people. The Contacts Provider is the source of data you see in the device's contacts application, and you can also access its data in your own application and transfer data between the device and online services. The provider accommodates a wide range of data sources and tries to manage as much data as possible for each person, with the result that its organization is complex. Because of this, the provider's API includes an extensive set of contract classes and interfaces that facilitate both data retrieval and modification.</p> <p data-bbox="527 625 835 646">This guide describes the following:</p> <ul data-bbox="527 675 1373 850" style="list-style-type: none"><li data-bbox="527 675 806 696">• The basic provider structure.</li><li data-bbox="527 725 894 747">• How to retrieve data from the provider.</li><li data-bbox="527 776 863 797">• How to modify data in the provider.</li><li data-bbox="527 826 1373 847">• How to write a sync adapter for synchronizing data from your server to the Contacts Provider.</li></ul> <p data-bbox="510 862 1486 889"><a href="https://developer.android.com/guide/topics/providers/contacts-provider.html">https://developer.android.com/guide/topics/providers/contacts-provider.html</a></p>

**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC
	<p data-bbox="533 240 657 264"><b>Overview</b></p> <p data-bbox="533 297 1608 313">ContactsContract defines an extensible database of contact-related information. Contact information is stored in a three-tier data model:</p> <ul data-bbox="533 342 1713 545" style="list-style-type: none"> <li data-bbox="533 342 1713 391">• A row in the <code>ContactsContract.Data</code> table can store any kind of personal data, such as a phone number or email addresses. The set of data kinds that can be stored in this table is open-ended. There is a predefined set of common kinds, but any application can add its own data kinds.</li> <li data-bbox="533 418 1713 467">• A row in the <code>ContactsContract.RawContacts</code> table represents a set of data describing a person and associated with a single account (for example, one of the user's Gmail accounts).</li> <li data-bbox="533 495 1713 545">• A row in the <code>ContactsContract.Contacts</code> table represents an aggregate of one or more RawContacts presumably describing the same person. When data in or associated with the RawContacts table is changed, the affected aggregate contacts are updated as necessary.</li> </ul> <p data-bbox="533 574 695 591">Other tables include:</p> <ul data-bbox="533 620 1713 899" style="list-style-type: none"> <li data-bbox="533 620 1713 669">• <code>ContactsContract.Groups</code>, which contains information about raw contact groups such as Gmail contact groups. The current API does not support the notion of groups spanning multiple accounts.</li> <li data-bbox="533 696 1325 712">• <code>ContactsContract.StatusUpdates</code>, which contains social status updates including IM availability.</li> <li data-bbox="533 740 1493 756">• <code>ContactsContract.AggregationExceptions</code>, which is used for manual aggregation and disaggregation of raw contacts</li> <li data-bbox="533 784 1325 800">• <code>ContactsContract.Settings</code>, which contains visibility and sync settings for accounts and groups.</li> <li data-bbox="533 828 1346 844">• <code>ContactsContract.SyncState</code>, which contains free-form data maintained on behalf of sync adapters</li> <li data-bbox="533 872 1129 888">• <code>ContactsContract.PhoneLookup</code>, which is used for quick caller-ID lookup</li> </ul> <p data-bbox="512 917 1541 950"><a href="https://developer.android.com/reference/android/provider/ContactsContract.html">https://developer.android.com/reference/android/provider/ContactsContract.html</a></p> <p data-bbox="533 963 615 987"><b>Data</b></p> <p data-bbox="533 1027 1745 1170">As noted previously, the data for a raw contact is stored in a <code>ContactsContract.Data</code> row that is linked to the raw contact's <code>_ID</code> value. This allows a single raw contact to have multiple instances of the same type of data such as email addresses or phone numbers. For example, if "Thomas Higginson" for <code>emilyd@gmail.com</code> (the raw contact row for Thomas Higginson associated with the Google account <code>emilyd@gmail.com</code>) has a home email address of <code>thigg@gmail.com</code> and a work email address of <code>thomas.higginson@gmail.com</code>, the Contacts Provider stores the two email address rows and links them both to the raw contact.</p> <p data-bbox="533 1200 1745 1310">Notice that different types of data are stored in this single table. Display name, phone number, email, postal address, photo, and website detail rows are all found in the <code>ContactsContract.Data</code> table. To help manage this, the <code>ContactsContract.Data</code> table has some columns with descriptive names, and others with generic names. The contents of a descriptive-name column have the same meaning regardless of the type of data in the row, while the contents of a generic-name column have different meanings depending on the type of data.</p> <p data-bbox="512 1323 1488 1356"><a href="https://developer.android.com/guide/topics/providers/contacts-provider.html">https://developer.android.com/guide/topics/providers/contacts-provider.html</a></p>

**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC													
	<table border="1"> <thead> <tr> <th>Task</th> <th>Action</th> <th>Data</th> <th>MIME type</th> <th>Notes</th> </tr> </thead> <tbody> <tr> <td>Pick a contact from a list</td> <td><a href="#">ACTION_PICK</a></td> <td>                     One of:                     <ul style="list-style-type: none"> <li>• <a href="#">Contacts.CONTENT_URI</a>, which displays a list of contacts.</li> <li>• <a href="#">Phone.CONTENT_URI</a>, which displays a list of phone numbers for a raw contact.</li> <li>• <a href="#">StructuredPostal.CONTENT_URI</a>, which displays a list of postal addresses for a raw contact.</li> <li>• <a href="#">Email.CONTENT_URI</a>, which displays a list of email addresses for a raw contact.</li> </ul> </td> <td>Not used</td> <td>                     Displays a list of raw contacts or a list of data from a raw contact, depending on the content URI type you supply. Call <a href="#">startActivityForResult()</a>, which returns the content URI of the selected row. The form of the URI is the table's content URI with the row's <a href="#">LOOKUP_ID</a> appended to it. The device's contacts app delegates read and write permissions to this content URI for the life of your activity. See the <a href="#">Content Provider Basics</a> guide for more details.                 </td> </tr> </tbody> </table>	Task	Action	Data	MIME type	Notes	Pick a contact from a list	<a href="#">ACTION_PICK</a>	One of: <ul style="list-style-type: none"> <li>• <a href="#">Contacts.CONTENT_URI</a>, which displays a list of contacts.</li> <li>• <a href="#">Phone.CONTENT_URI</a>, which displays a list of phone numbers for a raw contact.</li> <li>• <a href="#">StructuredPostal.CONTENT_URI</a>, which displays a list of postal addresses for a raw contact.</li> <li>• <a href="#">Email.CONTENT_URI</a>, which displays a list of email addresses for a raw contact.</li> </ul>	Not used	Displays a list of raw contacts or a list of data from a raw contact, depending on the content URI type you supply. Call <a href="#">startActivityForResult()</a> , which returns the content URI of the selected row. The form of the URI is the table's content URI with the row's <a href="#">LOOKUP_ID</a> appended to it. The device's contacts app delegates read and write permissions to this content URI for the life of your activity. See the <a href="#">Content Provider Basics</a> guide for more details.			
Task	Action	Data	MIME type	Notes										
Pick a contact from a list	<a href="#">ACTION_PICK</a>	One of: <ul style="list-style-type: none"> <li>• <a href="#">Contacts.CONTENT_URI</a>, which displays a list of contacts.</li> <li>• <a href="#">Phone.CONTENT_URI</a>, which displays a list of phone numbers for a raw contact.</li> <li>• <a href="#">StructuredPostal.CONTENT_URI</a>, which displays a list of postal addresses for a raw contact.</li> <li>• <a href="#">Email.CONTENT_URI</a>, which displays a list of email addresses for a raw contact.</li> </ul>	Not used	Displays a list of raw contacts or a list of data from a raw contact, depending on the content URI type you supply. Call <a href="#">startActivityForResult()</a> , which returns the content URI of the selected row. The form of the URI is the table's content URI with the row's <a href="#">LOOKUP_ID</a> appended to it. The device's contacts app delegates read and write permissions to this content URI for the life of your activity. See the <a href="#">Content Provider Basics</a> guide for more details.										
<a href="https://developer.android.com/guide/topics/providers/contacts-provider.html">https://developer.android.com/guide/topics/providers/contacts-provider.html</a>														

**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC
	<pre>25  * Parsed form of the intent sent to the Contacts application. 26  */ 27  public class ContactsRequest { 28 29      /** Default mode: browse contacts */ 30      public static final int ACTION_DEFAULT = 10; 31 32      /** Show all contacts */ 33      public static final int ACTION_ALL_CONTACTS = 15; 34 35      /** Show all contacts with phone numbers */ 36      public static final int ACTION_CONTACTS_WITH_PHONES = 17; 37 38      /** Show contents of a specific group */ 39      public static final int ACTION_GROUP = 20; 40 41      /** Show all starred contacts */ 42      public static final int ACTION_STARRED = 30; 43 44      /** Show frequently contacted contacts */ 45      public static final int ACTION_FREQUENT = 40; 46 47      /** Show starred and the frequent */ 48      public static final int ACTION_STREQUENT = 50; 49 50      /** Show all contacts and pick them when clicking */ 51      public static final int ACTION_PICK_CONTACT = 60; 52 53      /** Show all contacts as well as the option to create a new one */ 54      public static final int ACTION_PICK_OR_CREATE_CONTACT = 70; 55 56      /** Show all contacts and pick them for edit when clicking, and allow creating a new contact */ 57      public static final int ACTION_INSERT_OR_EDIT_CONTACT = 80;</pre> <p><a href="https://android.googlesource.com/platform/packages/apps/Contacts/+nougat-mr1-release/src/com/android/contacts/list/ContactsRequest.java">https://android.googlesource.com/platform/packages/apps/Contacts/+nougat-mr1-release/src/com/android/contacts/list/ContactsRequest.java</a></p>



**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC
	<pre> 59      /** Show all phone numbers and pick them when clicking */ 60      public static final int ACTION_PICK_PHONE = 90; 61 62      /** Show all postal addresses and pick them when clicking */ 63      public static final int ACTION_PICK_POSTAL = 100; 64 65      /** Show all postal addresses and pick them when clicking */ 66      public static final int ACTION_PICK_EMAIL = 105; 67 68      /** Show all contacts and create a shortcut for the picked contact */ 69      public static final int ACTION_CREATE_SHORTCUT_CONTACT = 110; 70 71      /** Show all phone numbers and create a call shortcut for the picked number */ 72      public static final int ACTION_CREATE_SHORTCUT_CALL = 120; 73 74      /** Show all phone numbers and create an SMS shortcut for the picked number */ 75      public static final int ACTION_CREATE_SHORTCUT_SMS = 130; 76 77      /** Show all contacts and activate the specified one */ 78      public static final int ACTION_VIEW_CONTACT = 140; 79 80      /** Show contacts recommended for joining with a specified target contact */ 81      public static final int ACTION_PICK_JOIN = 150; </pre> <p><a href="https://android.googlesource.com/platform/packages/apps/Contacts+/nougat-mr1-release/src/com/android/contacts/list/ContactsRequest.java">https://android.googlesource.com/platform/packages/apps/Contacts+/nougat-mr1-release/src/com/android/contacts/list/ContactsRequest.java</a></p>

**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC
	<pre> 104  * Displays a list to browse contacts. 105  */ 106  public class PeopleActivity extends ContactsActivity implements 107      View.OnCreateContextMenuListener, 108      View.OnClickListener, 109      ActionBarAdapter.Listener, 110      DialogManager.DialogShowingViewActivity, 111      ContactListFilterController.ContactListFilterListener, 112      ProviderStatusListener, 113      MultiContactDeleteListener, 114      JoinContactsListener { <a href="https://android.googlesource.com/platform/packages/apps/Contacts+/nougat-mr1-release/src/com/android/contacts/activities/PeopleActivity.java">https://android.googlesource.com/platform/packages/apps/Contacts+/nougat-mr1- release/src/com/android/contacts/activities/PeopleActivity.java</a> 145      * Showing a list of Contacts. Also used for showing search results in search mode. 146      */ 147      private MultiSelectContactsListFragment mAllFragment; 148      private ContactTileListFragment mFavoritesFragment; <a href="https://android.googlesource.com/platform/packages/apps/Contacts+/nougat-mr1-release/src/com/android/contacts/activities/PeopleActivity.java">https://android.googlesource.com/platform/packages/apps/Contacts+/nougat-mr1- release/src/com/android/contacts/activities/PeopleActivity.java</a> </pre>

**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC
	<pre>458     private void configureFragments(boolean fromRequest) { 459         if (fromRequest) { 460             ContactListFilter filter = null; 461             int actionCode = mRequest.getActionCode(); 462             boolean searchMode = mRequest.isSearchMode(); 463             final int tabToOpen; 464             switch (actionCode) { 465                 case ContactsRequest.ACTION_ALL_CONTACTS: 466                     filter = ContactListFilter.createFilterWithType( 467                         ContactListFilter.FILTER_TYPE_ALL_ACCOUNTS); 468                     tabToOpen = TabState.ALL; 469                     break; 470                 case ContactsRequest.ACTION_CONTACTS_WITH_PHONES: 471                     filter = ContactListFilter.createFilterWithType( 472                         ContactListFilter.FILTER_TYPE_WITH_PHONE_NUMBERS_ONLY); 473                     tabToOpen = TabState.ALL; 474                     break; 475 476                 case ContactsRequest.ACTION_FREQUENT: 477                 case ContactsRequest.ACTION_STREQUENT: 478                 case ContactsRequest.ACTION_STARRED: 479                     tabToOpen = TabState.FAVORITES; 480                     break; 481                 case ContactsRequest.ACTION_VIEW_CONTACT: 482                     tabToOpen = TabState.ALL; 483                     break; 484                 default: 485                     tabToOpen = -1; 486                     break; 487             } </pre> <p data-bbox="506 1320 1566 1390"><a href="https://android.googlesource.com/platform/packages/apps/Contacts+/nougat-mr1-release/src/com/android/contacts/activities/PeopleActivity.java">https://android.googlesource.com/platform/packages/apps/Contacts+/nougat-mr1-release/src/com/android/contacts/activities/PeopleActivity.java</a></p>

**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC
	<pre>488         if (tabToOpen != -1) { 489             mActionBarAdapter.setCurrentTab(tabToOpen); 490         } 491 492         if (filter != null) { 493             mContactListFilterController.setContactListFilter(filter, false); 494             searchMode = false; 495         } 496 497         if (mRequest.getContactUri() != null) { 498             searchMode = false; 499         } 500 501         mActionBarAdapter.setSearchMode(searchMode); 502         configureContactListFragmentForRequest(); 503     } 504 505     configureContactListFragment(); 506 507     invalidateOptionsMenuIfNeeded(); 508 }</pre> <p data-bbox="510 1019 1566 1084"><a href="https://android.googlesource.com/platform/packages/apps/Contacts+/nougat-mr1-release/src/com/android/contacts/activities/PeopleActivity.java">https://android.googlesource.com/platform/packages/apps/Contacts+/nougat-mr1-release/src/com/android/contacts/activities/PeopleActivity.java</a></p>

**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC
	<pre>35 public class ProfileAndContactsLoader extends CursorLoader { 36 37     private boolean mLoadProfile; 38 39     private String[] mProjection; 40 41     private Uri mExtraUri; 42     private String[] mExtraProjection; 43     private String mExtraSelection; 44     private String[] mExtraSelectionArgs; 45     private boolean mMergeExtraContactsAfterPrimary; 46 47     public ProfileAndContactsLoader(Context context) { 48         super(context); 49     }</pre> <p><a href="https://android.googlesource.com/platform/packages/apps/ContactsCommon/+/nougat-mr1-release/src/com/android/contacts/common/list/ProfileAndContactsLoader.java">https://android.googlesource.com/platform/packages/apps/ContactsCommon/+/nougat-mr1-release/src/com/android/contacts/common/list/ProfileAndContactsLoader.java</a></p>

**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC
	<pre> 35 public final class GroupMemberLoader extends CursorLoader { 36 37     public static class GroupEditorQuery { 38         private static final String[] PROJECTION = new String[] { 39             Data.CONTACT_ID,           // 0 40             Data.RAW_CONTACT_ID,      // 1 41             Data.DISPLAY_NAME_PRIMARY, // 2 42             Data.PHOTO_URI,           // 3 43             Data.LOOKUP_KEY,          // 4 44         }; 45 46         public static final int CONTACT_ID           = 0; 47         public static final int RAW_CONTACT_ID      = 1; 48         public static final int CONTACT_DISPLAY_NAME_PRIMARY = 2; 49         public static final int CONTACT_PHOTO_URI   = 3; 50         public static final int CONTACT_LOOKUP_KEY  = 4; 51     } 52 53     public static class GroupDetailQuery { 54         private static final String[] PROJECTION = new String[] { 55             Data.CONTACT_ID,           // 0 56             Data.PHOTO_URI,           // 1 57             Data.LOOKUP_KEY,          // 2 58             Data.DISPLAY_NAME_PRIMARY, // 3 59             Data.CONTACT_PRESENCE,    // 4 60             Data.CONTACT_STATUS,      // 5 61         }; 62 63         public static final int CONTACT_ID           = 0; 64         public static final int CONTACT_PHOTO_URI   = 1; 65         public static final int CONTACT_LOOKUP_KEY  = 2; 66         public static final int CONTACT_DISPLAY_NAME_PRIMARY = 3; 67         public static final int CONTACT_PRESENCE_STATUS = 4; 68         public static final int CONTACT_STATUS      = 5; 69     } 70 71     private final long mGroupId; </pre> <p><a href="https://android.googlesource.com/platform/packages/apps/Contacts/+/nougat-mr1-release/src/com/android/contacts/GroupMemberLoader.java">https://android.googlesource.com/platform/packages/apps/Contacts/+/nougat-mr1-release/src/com/android/contacts/GroupMemberLoader.java</a></p>

**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC
	<pre> 24  * Group loader for the group list that includes details such as the number of contacts per group 25  * and number of groups per account. This list is sorted by account type, account name, where the 26  * group names are in alphabetical order. Note that the list excludes default, favorite, and deleted 27  * groups. 28  */ 29  public final class GroupListLoader extends CursorLoader { 30 31      private final static String[] COLUMNS = new String[] { 32          Groups.ACCOUNT_NAME, 33          Groups.ACCOUNT_TYPE, 34          Groups.DATA_SET, 35          Groups._ID, 36          Groups.TITLE, 37          Groups.SUMMARY_COUNT, 38      }; 39 40      public final static int ACCOUNT_NAME = 0; 41      public final static int ACCOUNT_TYPE = 1; 42      public final static int DATA_SET = 2; 43      public final static int GROUP_ID = 3; 44      public final static int TITLE = 4; 45      public final static int MEMBER_COUNT = 5; 46 47      private static final Uri GROUP_LIST_URI = Groups.CONTENT_SUMMARY_URI; 48 49      public GroupListLoader(Context context) { 50          super(context, GROUP_LIST_URI, COLUMNS, Groups.ACCOUNT_TYPE + " NOT NULL AND " 51              + Groups.ACCOUNT_NAME + " NOT NULL AND " + Groups.AUTO_ADD + "=0 AND " + 52              Groups.FAVORITES + "=0 AND " + Groups.DELETED + "=0", null, 53              Groups.ACCOUNT_TYPE + ", " + Groups.ACCOUNT_NAME + ", " + Groups.DATA_SET + ", " + 54              Groups.TITLE + " COLLATE LOCALIZED ASC"); 55      } 56  } </pre> <p><a href="https://android.googlesource.com/platform/packages/apps/Contacts+/nougat-mr1-release/src/com/android/contacts/GroupListLoader.java">https://android.googlesource.com/platform/packages/apps/Contacts+/nougat-mr1-release/src/com/android/contacts/GroupListLoader.java</a></p>

**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC
	<pre> 24  * Group meta-data loader. Loads all groups or just a single group from the 25  * database (if given a {@link Uri}). 26  */ 27  public final class GroupMetaLoader extends CursorLoader { 28 29      private final static String[] COLUMNS = new String[] { 30          Groups.ACCOUNT_NAME, 31          Groups.ACCOUNT_TYPE, 32          Groups.DATA_SET, 33          Groups._ID, 34          Groups.TITLE, 35          Groups.AUTO_ADD, 36          Groups.FAVORITES, 37          Groups.GROUP_IS_READ_ONLY, 38          Groups.DELETED, 39      }; 40 41      public final static int ACCOUNT_NAME = 0; 42      public final static int ACCOUNT_TYPE = 1; 43      public final static int DATA_SET = 2; 44      public final static int GROUP_ID = 3; 45      public final static int TITLE = 4; 46      public final static int AUTO_ADD = 5; 47      public final static int FAVORITES = 6; 48      public final static int IS_READ_ONLY = 7; 49      public final static int DELETED = 8; 50 51      public GroupMetaLoader(Context context, Uri groupUri) { 52          super(context, ensureIsGroupUri(groupUri), COLUMNS, Groups.ACCOUNT_TYPE + " NOT NULL AND " 53              + Groups.ACCOUNT_NAME + " NOT NULL", null, null); 54      } 55 56      /** 57       * Ensures that this is a valid group URI. If invalid, then an exception is 58       * thrown. Otherwise, the original URI is returned. 59       */ 60      private static Uri ensureIsGroupUri(final Uri groupUri) { 61          // TODO: Fix ContactsProvider2 getType method to resolve the group Uris 62          if (groupUri == null) { 63              throw new IllegalArgumentException("Uri must not be null"); 64          } 65          if (!groupUri.toString().startsWith(Groups.CONTENT_URI.toString())) { 66              throw new IllegalArgumentException("Invalid group Uri: " + groupUri); 67          } 68          return groupUri; 69      } 70  } </pre> <p><a href="https://android.googlesource.com/platform/packages/apps/Contacts/+nougat-mr1-release/src/com/android/contacts/GroupMetaLoader.java">https://android.googlesource.com/platform/packages/apps/Contacts/+nougat-mr1-release/src/com/android/contacts/GroupMetaLoader.java</a></p>



**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC
	<pre> 19  * Meta-data for a contact group. We load all groups associated with the contact's 20  * constituent accounts. 21  */ 22  public final class GroupMetaData { 23      private String mAccountName; 24      private String mAccountType; 25      private String mDataSet; 26      private long mGroupId; 27      private String mTitle; 28      private boolean mDefaultGroup; 29      private boolean mFavorites; 30 31      public GroupMetaData(String accountName, String accountType, String dataSet, long groupId, 32                          String title, boolean defaultGroup, boolean favorites) { 33          this.mAccountName = accountName; 34          this.mAccountType = accountType; 35          this.mDataSet = dataSet; 36          this.mGroupId = groupId; 37          this.mTitle = title; 38          this.mDefaultGroup = defaultGroup; 39          this.mFavorites = favorites; 40      } 41 42      public String getAccountName() { 43          return mAccountName; 44      } 45 46      public String getAccountType() { 47          return mAccountType; 48      } 49 50      public String getDataSet() { 51          return mDataSet; 52      } 53 54      public long getGroupId() { 55          return mGroupId; 56      } 57 58      public String getTitle() { 59          return mTitle; 60      } 61 62      public boolean isDefaultGroup() { 63          return mDefaultGroup; 64      } 65 66      public boolean isFavorites() { 67          return mFavorites; </pre> <p><a href="https://android.googlesource.com/platform/packages/apps/ContactsCommon/+/nougat-mr1-release/src/com/android/contacts/common/GroupMetaData.java">https://android.googlesource.com/platform/packages/apps/ContactsCommon/+/nougat-mr1-release/src/com/android/contacts/common/GroupMetaData.java</a></p>

**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC
	<pre>44 * Class that sends chat message via SMS. 45 * 46 * The interface emulates a blocking sending similar to making an HTTP request. 47 * It calls the SmsManager to send a (potentially multipart) message and waits 48 * on the sent status on each part. The waiting has a timeout so it won't wait 49 * forever. Once the sent status of all parts received, the call returns. 50 * A successful sending requires success status for all parts. Otherwise, we 51 * pick the highest level of failure as the error for the whole message, which 52 * is used to determine if we need to retry the sending. 53 */ 54 public class SmsSender { 55     private static final String TAG = LogUtil.BUGLE_TAG; 56 57     public static final String EXTRA_PART_ID = "part_id"; 58 59     /* 60     * A map for pending sms messages. The key is the random request UUID. 61     */ 62     private static ConcurrentHashMap&lt;Uri, SendResult&gt; sPendingMessageMap = 63         new ConcurrentHashMap&lt;Uri, SendResult&gt;(); 64 65     private static final Random RANDOM = new Random(); 66 67     // Whether we should send multipart SMS as separate messages 68     private static Boolean sSendMultipartSmsAsSeparateMessages = null; 69</pre> <p><a href="https://android.googlesource.com/platform/packages/apps/Messaging/+nougat-mr1-release/src/com/android/messaging/sms/SmsSender.java">https://android.googlesource.com/platform/packages/apps/Messaging/+nougat-mr1-release/src/com/android/messaging/sms/SmsSender.java</a></p>

**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC
	<pre> 253 // Actually sending the message using SmsManager 254 private static void sendInternal(final Context context, final int subId, String dest, 255     final ArrayList&lt;String&gt; messages, final String serviceCenter, 256     final boolean requireDeliveryReport, final Uri messageUri) throws SmsException { 257     Assert.notNull(context); 258     final SmsManager smsManager = PhoneUtils.get(subId).getSmsManager(); 259     final int messageCount = messages.size(); 260     final ArrayList&lt;PendingIntent&gt; deliveryIntents = new ArrayList&lt;PendingIntent&gt;(messageCount); 261     final ArrayList&lt;PendingIntent&gt; sentIntents = new ArrayList&lt;PendingIntent&gt;(messageCount); 262     for (int i = 0; i &lt; messageCount; i++) { 263         // Make pending intents different for each message part 264         final int partId = (messageCount &lt;= 1 ? 0 : i + 1); 265         if (requireDeliveryReport &amp;&amp; (i == (messageCount - 1))) { 266             // TODO we only care about the delivery status of the last part 267             // Shall we have better tracking of delivery status of all parts? 268             deliveryIntents.add(PendingIntent.getBroadcast( 269                 context, 270                 partId, 271                 getSendStatusIntent(context, SendStatusReceiver.MESSAGE_DELIVERED_ACTION, 272                     messageUri, partId, subId), 273                 0/*flag*/)); 274         } else { 275             deliveryIntents.add(null); 276         } 277         sentIntents.add(PendingIntent.getBroadcast( 278             context, 279             partId, 280             getSendStatusIntent(context, SendStatusReceiver.MESSAGE_SENT_ACTION, 281                 messageUri, partId, subId), 282             0/*flag*/)); 283     } 284     if (sSendMultipartSmsAsSeparateMessages == null) { 285         sSendMultipartSmsAsSeparateMessages = MmsConfig.get(subId) 286             .getSendMultipartSmsAsSeparateMessages(); 287     } </pre> <p><a href="https://android.googlesource.com/platform/packages/apps/Messaging/+nougat-mr1-release/src/com/android/messaging/sms/SmsSender.java">https://android.googlesource.com/platform/packages/apps/Messaging/+nougat-mr1-release/src/com/android/messaging/sms/SmsSender.java</a></p>

**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC
	<pre> 288     try { 289         if (sSendMultipartSmsAsSeparateMessages) { 290             // If multipart sms is not supported, send them as separate messages 291             for (int i = 0; i &lt; messageCount; i++) { 292                 smsManager.sendTextMessage(dest, 293                     serviceCenter, 294                     messages.get(i), 295                     sentIntents.get(i), 296                     deliveryIntents.get(i)); 297             } 298         } else { 299             smsManager.sendMultipartTextMessage( 300                 dest, serviceCenter, messages, sentIntents, deliveryIntents); 301         } 302     } catch (final Exception e) { 303         throw new SmsException("SmsSender: caught exception in sending " + e); 304     } 305 } </pre> <p><a href="https://android.googlesource.com/platform/packages/apps/Messaging/+nougat-mr1-release/src/com/android/messaging/sms/SmsSender.java">https://android.googlesource.com/platform/packages/apps/Messaging/+nougat-mr1-release/src/com/android/messaging/sms/SmsSender.java</a></p> <pre> 56  * Class that receives incoming SMS messages through android.provider.Telephony.SMS_RECEIVED 57  * 58  * This class serves two purposes: 59  * - Process phone verification SMS messages 60  * - Handle SMS messages when the user has enabled us to be the default SMS app (Pre-KLP) 61  */ 62  public final class SmsReceiver extends BroadcastReceiver { 63      private static final String TAG = LogUtil.BUGLE_TAG; 64 65      private static ArrayList&lt;Pattern&gt; sIgnoreSmsPatterns; 66 </pre> <p><a href="https://android.googlesource.com/platform/packages/apps/Messaging/+nougat-mr1-release/src/com/android/messaging/receiver/SmsReceiver.java">https://android.googlesource.com/platform/packages/apps/Messaging/+nougat-mr1-release/src/com/android/messaging/receiver/SmsReceiver.java</a></p>

**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC
	<pre> 200     public static void deliverSmsMessages(final Context context, final int subId, 201         final int errorCode, final android.telephony.SmsMessage[] messages) { 202         final ContentValues messageValues = 203             MmsUtils.parseReceivedSmsMessage(context, messages, errorCode); 204 205         LogUtil.v(TAG, "SmsReceiver.deliverSmsMessages"); 206 207         final long nowInMillis = System.currentTimeMillis(); 208         final long receivedTimestampMs = MmsUtils.getMessageDate(messages[0], nowInMillis); 209 210         messageValues.put(Sms.Inbox.DATE, receivedTimestampMs); 211         // Default to unread and unseen for us but ReceiveSmsMessageAction will override 212         // seen for the telephony db. 213         messageValues.put(Sms.Inbox.READ, 0); 214         messageValues.put(Sms.Inbox.SEEN, 0); 215         if (OsUtil.isAtLeastL_MR1()) { 216             messageValues.put(Sms.SUBSCRIPTION_ID, subId); 217         } 218 219         if (messages[0].getMessageClass() == android.telephony.SmsMessage.MessageClass.CLASS_0    220             DebugUtils.debugClassZeroSmsEnabled()) { 221             Factory.get().getUIIntents().launchClassZeroActivity(context, messageValues); 222         } else { 223             final ReceiveSmsMessageAction action = new ReceiveSmsMessageAction(messageValues); 224             action.start(); 225         } 226     } </pre> <p><a href="https://android.googlesource.com/platform/packages/apps/Messaging/+nougat-mr1-release/src/com/android/messaging/receiver/SmsReceiver.java">https://android.googlesource.com/platform/packages/apps/Messaging/+nougat-mr1-release/src/com/android/messaging/receiver/SmsReceiver.java</a></p>

**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC
	<pre>228     @Override 229     public void onReceive(final Context context, final Intent intent) { 230         LogUtil.v(TAG, "SmsReceiver.onReceive " + intent); 231         // On KLP+ we only take delivery of SMS messages in SmsDeliverReceiver. 232         if (PhoneUtils.getDefault().isSmsEnabled()) { 233             final String action = intent.getAction(); 234             if (OsUtil.isSecondaryUser() &amp;&amp; 235                 (Telephony.Sms.Intents.SMS_RECEIVED_ACTION.equals(action)    236                  // TODO: update this with the actual constant from Telephony 237                  "android.provider.Telephony.MMS_DOWNLOADED".equals(action))) { 238                 postNewMessageSecondaryUserNotification(); 239             } else if (!OsUtil.isAtLeastKLP()) { 240                 deliverSmsIntent(context, intent); 241             } 242         } 243     }</pre> <p><a href="https://android.googlesource.com/platform/packages/apps/Messaging/+/nougat-mr1-release/src/com/android/messaging/receiver/SmsReceiver.java">https://android.googlesource.com/platform/packages/apps/Messaging/+/nougat-mr1-release/src/com/android/messaging/receiver/SmsReceiver.java</a></p>

**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC
	<pre>52  * Class that sends chat message via MMS. 53  * 54  * The interface emulates a blocking send similar to making an HTTP request. 55  */ 56  public class MmsSender { 57      private static final String TAG = LogUtil.BUGLE_TAG; 58 59      /** 60       * Send an MMS message. 61       * 62       * @param context Context 63       * @param messageUri The unique URI of the message for identifying it during sending 64       * @param sendReq The SendReq PDU of the message 65       * @throws MmsFailureException 66       */ 67      public static void sendMms(final Context context, final int subId, final Uri messageUri, 68                               final SendReq sendReq, final Bundle sentIntentExtras) throws MmsFailureException { 69          sendMms(context, 70                subId, 71                messageUri, 72                null /* locationUri */, 73                sendReq, 74                true /* responseImportant */, 75                sentIntentExtras); 76      }</pre> <p><a href="https://android.googlesource.com/platform/packages/apps/Messaging/+nougat-mr1-release/src/com/android/messaging/sms/MmsSender.java">https://android.googlesource.com/platform/packages/apps/Messaging/+nougat-mr1-release/src/com/android/messaging/sms/MmsSender.java</a></p>

**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC
	<pre> 240      * Download an MMS message. 241      * 242      * @param context Context 243      * @param contentLocation The url of the MMS message 244      * @throws MmsFailureException 245      * @throws InvalidHeaderValueException 246      */ 247      public static void downloadMms(final Context context, final int subId, 248          final String contentLocation, Bundle extras) throws MmsFailureException, 249          InvalidHeaderValueException { 250          final Uri requestUri = Uri.parse(contentLocation); 251          final Uri contentUri = MmsFileProvider.buildRawMmsUri(); 252 253          final Intent downloadedIntent = new Intent(SendStatusReceiver.MMS_DOWNLOADED_ACTION, 254              requestUri, 255              context, 256              SendStatusReceiver.class); 257          downloadedIntent.putExtra(SendMessageAction.EXTRA_CONTENT_URI, contentUri); 258          if (extras != null) { 259              downloadedIntent.putExtras(extras); 260          } 261          final PendingIntent downloadedPendingIntent = PendingIntent.getBroadcast( 262              context, 263              0 /*request code*/, 264              downloadedIntent, 265              PendingIntent.FLAG_UPDATE_CURRENT); 266 267          MmsManager.downloadMultimediaMessage(subId, context, contentLocation, contentUri, 268              downloadedPendingIntent); 269      } </pre> <p><a href="https://android.googlesource.com/platform/packages/apps/Messaging/+nougat-mr1-release/src/com/android/messaging/sms/MmsSender.java">https://android.googlesource.com/platform/packages/apps/Messaging/+nougat-mr1-release/src/com/android/messaging/sms/MmsSender.java</a></p>



**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC
	<pre> 97      * Execute an MMS HTTP request, either a POST (sending) or a GET (downloading) 98      * 99      * @param urlString The request URL, for sending it is usually the MMSC, and for downloading 100     *           it is the message URL 101     * @param pdu For POST (sending) only, the PDU to send 102     * @param method HTTP method, POST for sending and GET for downloading 103     * @param isProxySet Is there a proxy for the MMSC 104     * @param proxyHost The proxy host 105     * @param proxyPort The proxy port 106     * @param mmsConfig The MMS config to use 107     * @param userAgent The user agent header value 108     * @param uaProfUrl The UA Prof URL header value 109     * @return The HTTP response body 110     * @throws MmsHttpException For any failures 111     */ 112     public byte[] execute(String urlString, byte[] pdu, String method, boolean isProxySet, 113                          String proxyHost, int proxyPort, Bundle mmsConfig, String userAgent, String uaProfUrl) 114         throws MmsHttpException { 115         Log.d(MmsService.TAG, "HTTP: " + method + " " + Utils.redactUrlForNonVerbose(urlString) 116             + (isProxySet ? (" proxy=" + proxyHost + ":" + proxyPort) : "")) 117             + ", PDU size=" + (pdu != null ? pdu.length : 0)); 118         checkMethod(method); 119         HttpURLConnection connection = null; 120         try { 121             Proxy proxy = Proxy.NO_PROXY; 122             if (isProxySet) { 123                 proxy = new Proxy(Proxy.Type.HTTP, new InetSocketAddress(proxyHost, proxyPort)); 124             } 125             final URL url = new URL(urlString); 126             // Now get the connection 127             connection = (HttpURLConnection) url.openConnection(proxy); 128             connection.setDoInput(true); 129             connection.setConnectTimeout( 130                 mmsConfig.getInt(CarrierConfigValuesLoader.CONFIG_HTTP_SOCKET_TIMEOUT, 131                     CarrierConfigValuesLoader.CONFIG_HTTP_SOCKET_TIMEOUT_DEFAULT)); </pre> <p data-bbox="506 1219 1596 1287"><a href="https://android.googlesource.com/platform/packages/apps/Messaging/+nougat-mr1-release/src/android/support/v7/mms/MmsHttpClient.java">https://android.googlesource.com/platform/packages/apps/Messaging/+nougat-mr1-release/src/android/support/v7/mms/MmsHttpClient.java</a></p>

**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC
	<pre> 132 // ----- COMMON HEADERS ----- 133 // Header: Accept 134 connection.setRequestProperty(HEADER_ACCEPT, HEADER_VALUE_ACCEPT); 135 // Header: Accept-Language 136 connection.setRequestProperty( 137     HEADER_ACCEPT_LANGUAGE, getCurrentAcceptLanguage(Locale.getDefault())); 138 // Header: User-Agent 139 Log.i(MmsService.TAG, "HTTP: User-Agent=" + userAgent); 140 connection.setRequestProperty(HEADER_USER_AGENT, userAgent); 141 // Header: x-wap-profile 142 final String uaProfUrlTagName = mmsConfig.getString( 143     CarrierConfigValuesLoader.CONFIG_UA_PROF_TAG_NAME, 144     CarrierConfigValuesLoader.CONFIG_UA_PROF_TAG_NAME_DEFAULT); 145 if (uaProfUrl != null) { 146     Log.i(MmsService.TAG, "HTTP: UaProfUrl=" + uaProfUrl); 147     connection.setRequestProperty(uaProfUrlTagName, uaProfUrl); 148 } 149 // Add extra headers specified by mms_config.xml's httpparams 150 addExtraHeaders(connection, mmsConfig); 151 // Different stuff for GET and POST 152 if (METHOD_POST.equals(method)) { 153     if (pdu == null    pdu.length &lt; 1) { 154         Log.e(MmsService.TAG, "HTTP: empty pdu"); 155         throw new MmsHttpException(0/*statusCode*/, "Sending empty PDU"); 156     } 157     connection.setDoOutput(true); 158     connection.setRequestMethod(METHOD_POST); 159     if (mmsConfig.getBoolean( 160         CarrierConfigValuesLoader.CONFIG_SUPPORT_HTTP_CHARSET_HEADER, 161         CarrierConfigValuesLoader.CONFIG_SUPPORT_HTTP_CHARSET_HEADER_DEFAULT)) { 162         connection.setRequestProperty(HEADER_CONTENT_TYPE, 163             HEADER_VALUE_CONTENT_TYPE_WITH_CHARSET); 164     } else { 165         connection.setRequestProperty(HEADER_CONTENT_TYPE, 166             HEADER_VALUE_CONTENT_TYPE_WITHOUT_CHARSET); </pre> <p><a href="https://android.googlesource.com/platform/packages/apps/Messaging/+nougat-mr1-release/src/android/support/v7/mms/MmsHttpClient.java">https://android.googlesource.com/platform/packages/apps/Messaging/+nougat-mr1-release/src/android/support/v7/mms/MmsHttpClient.java</a></p>

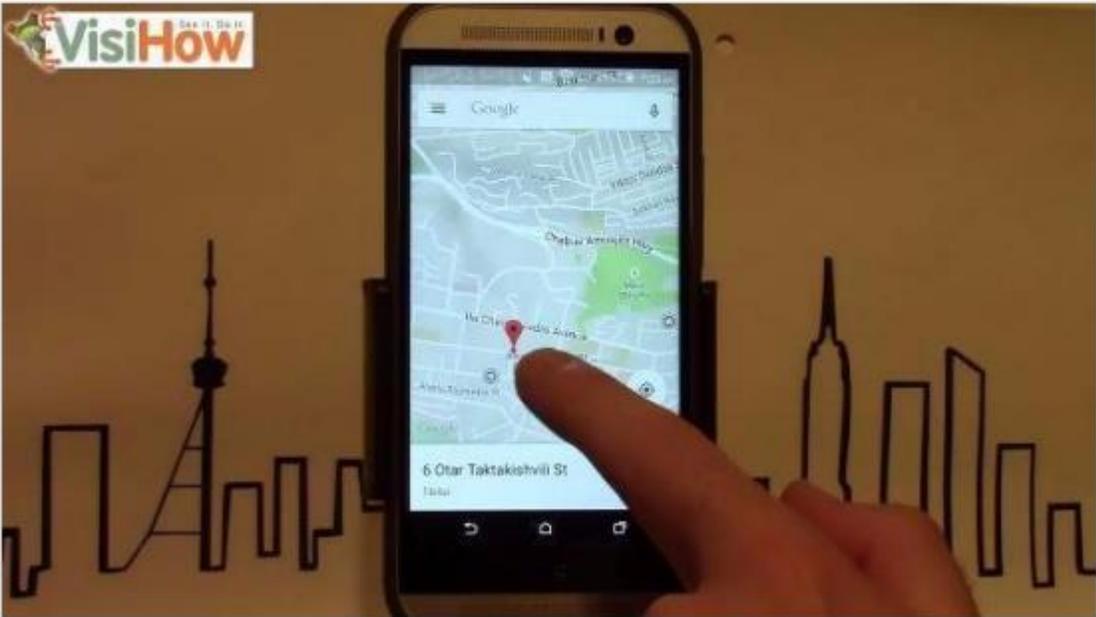
**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC
	<pre> 167     } 168     if (Log.isLoggable(MmsService.TAG, Log.VERBOSE)) { 169         logHttpHeaders(connection.getRequestProperties()); 170     } 171     connection.setFixedLengthStreamingMode(pdu.length); 172     // Sending request body 173     final OutputStream out = 174         new BufferedOutputStream(connection.getOutputStream()); 175     out.write(pdu); 176     out.flush(); 177     out.close(); 178 } else if (METHOD_GET.equals(method)) { 179     if (Log.isLoggable(MmsService.TAG, Log.VERBOSE)) { 180         logHttpHeaders(connection.getRequestProperties()); 181     } 182     connection.setRequestMethod(METHOD_GET); 183 } 184 // Get response 185 final int responseCode = connection.getResponseCode(); 186 final String responseMessage = connection.getResponseMessage(); 187 Log.d(MmsService.TAG, "HTTP: " + responseCode + " " + responseMessage); 188 if (Log.isLoggable(MmsService.TAG, Log.VERBOSE)) { 189     logHttpHeaders(connection.getHeaderFields()); 190 } 191 if (responseCode / 100 != 2) { 192     throw new MmsHttpException(responseCode, responseMessage); 193 } 194 final InputStream in = new BufferedInputStream(connection.getInputStream()); 195 final ByteArrayOutputStream byteOut = new ByteArrayOutputStream(); 196 final byte[] buf = new byte[4096]; 197 int count = 0; 198 while ((count = in.read(buf)) &gt; 0) { 199     byteOut.write(buf, 0, count); 200 } 201 in.close(); 202 final byte[] responseBody = byteOut.toByteArray(); 203 Log.d(MmsService.TAG, "HTTP: response size=" 204     + (responseBody != null ? responseBody.length : 0)); 205 return responseBody; </pre> <p><a href="https://android.googlesource.com/platform/packages/apps/Messaging/+nougat-mr1-release/src/android/support/v7/mms/MmsHttpClient.java">https://android.googlesource.com/platform/packages/apps/Messaging/+nougat-mr1-release/src/android/support/v7/mms/MmsHttpClient.java</a></p>

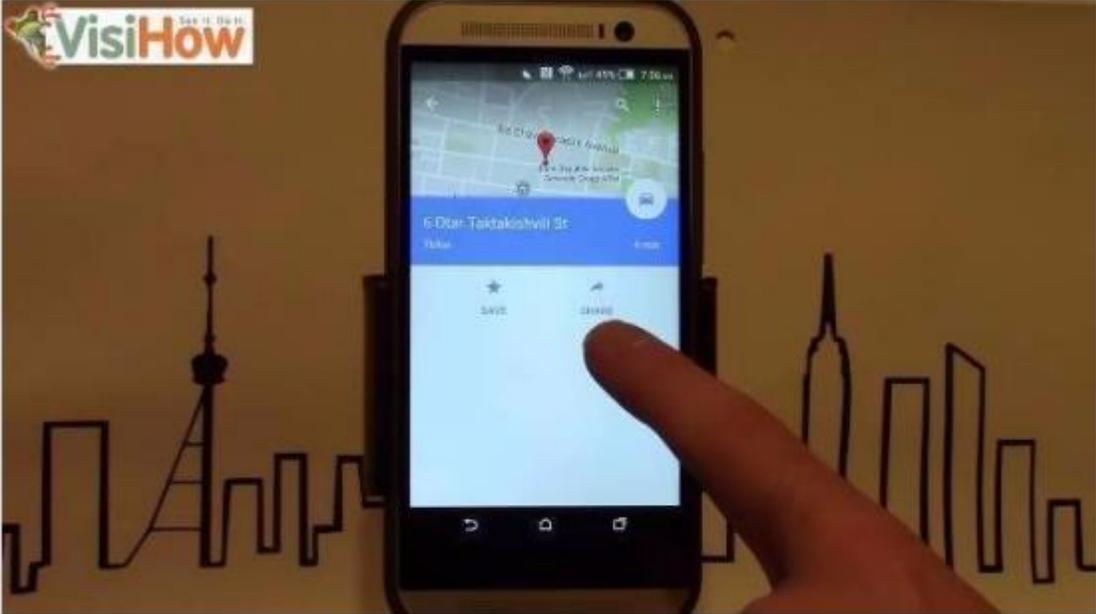
**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC
<p>[28C] receiving respective IP-based responses to the SMS messages, wherein the IP-based responses to the SMS messages include location information of the respective second devices;</p>	<p>HTC infringes directly and/or indirectly by performing, inducing others to perform, and/or contributing to the performance of receiving respective IP-based responses to the SMS messages, wherein the IP-based responses to the SMS messages include location information of the respective second devices. See claim 1[C], which is incorporated herein by reference in its entirety.</p> <p>For example, the HTC accused devices running Maps are configured to receive IP-based communications from the respective second devices that include location information of the second devices.</p> <p><b>Share Locations from Google Maps on HTC One M8</b></p>

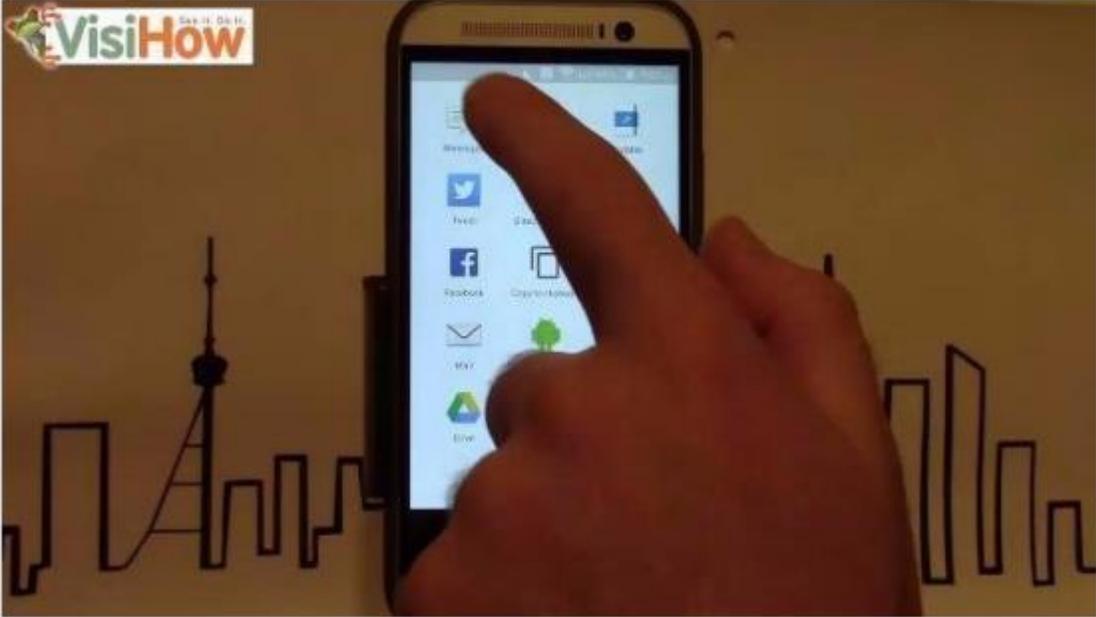
### Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products

US9408055B2	HTC
	<p data-bbox="531 233 827 256"><b>Place a pin on the map.</b></p> <p data-bbox="531 272 1640 380">We can do this anywhere in the entire world, anywhere, on any map that we find in the Google Maps application. We are going to choose a location, press down on the screen, and hold down until we can see that a red pin has been dropped.</p> 

## Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products

US9408055B2	HTC
	<p data-bbox="527 240 1207 310"><b>We have two options:</b> "SAVE" and "SHARE". Press down on the word "SHARE".</p>  <p data-bbox="527 1057 1633 1208"><b>A menu appears with all of the different ways that we can share this location.</b> We can share this location via messages, Bluetooth, Twitter, Facebook, mail, Gmail and Viber. If you have other applications, such as WhatsApp, or any other chat or email applications, you will be able to use them as well and will see them here.</p>

### Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products

US9408055B2	HTC
	<p data-bbox="527 237 856 264"><b>Let's choose "Messages".</b></p> <p data-bbox="527 277 1633 347">This option is located on the top left-hand corner of the screen. All options will have the same basic processes. They will ask us to send the location. Press down on the "Messages" icon.</p> 

## Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products

US9408055B2	HTC
	<p data-bbox="520 228 1260 256"><b>Press the box next to the contact who will be the recipient.</b></p> <p data-bbox="520 266 1549 293">A checkmark will appear in the box. We can select multiple contacts as well if we desire.</p>  <p data-bbox="520 1045 911 1073"><b>Press on the word "DONE (1)".</b></p> <p data-bbox="520 1084 1629 1154">This icon is located in the bottom right-hand corner of the screen. The bracketed number indicates the number of contacts we selected.</p> <p data-bbox="520 1166 1409 1193"><a href="http://visihow.com/Share_Locations_from_Google_Maps_on_HTC_One_M8">http://visihow.com/Share_Locations_from_Google_Maps_on_HTC_One_M8</a></p>



## Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products

US9408055B2	HTC
	<p data-bbox="533 240 884 282"><b>Send your location</b></p> <ol data-bbox="533 305 947 505" style="list-style-type: none"><li>1. Open the Android Messages app .</li><li>2. Open or start a conversation.</li><li>3. Tap Attach .</li><li>4. Tap Location on .</li><li>5. To send your location, tap Send .</li></ol> <p data-bbox="520 526 1535 553"><a href="https://support.google.com/pixelphone/answer/6159880?hl=en&amp;ref_topic=6211804">https://support.google.com/pixelphone/answer/6159880?hl=en&amp;ref_topic=6211804</a></p>  <p data-bbox="520 1328 1514 1356"><a href="https://blog.google/topics/rcs/delivering-rcs-messaging-android-users-worldwide/">https://blog.google/topics/rcs/delivering-rcs-messaging-android-users-worldwide/</a></p>

**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC
	<div data-bbox="533 339 915 378" data-label="Section-Header"> <p>Share a location or place</p> </div> <div data-bbox="554 420 743 444" data-label="Section-Header"> <p>Share your location</p> </div> <div data-bbox="585 472 1104 599" data-label="List-Group"> <ol style="list-style-type: none"> <li>1. On your Android phone or tablet, open the Hangouts app</li> <li>2. Open a conversation.</li> <li>3. Tap Location</li> <li>4. Tap <b>Select this location</b> &gt; <b>Select</b>.</li> </ol> </div> <div data-bbox="554 673 688 699" data-label="Section-Header"> <p>Share a place</p> </div> <div data-bbox="585 725 1104 886" data-label="List-Group"> <ol style="list-style-type: none"> <li>1. On your Android phone or tablet, open the Hangouts app</li> <li>2. Open a conversation.</li> <li>3. Tap Location &gt; Search</li> <li>4. Type in a location or address.</li> <li>5. Tap <b>Select</b>.</li> </ol> </div> <div data-bbox="514 917 1581 1018" data-label="Text"> <p><a href="https://support.google.com/hangouts/answer/3115410?visit_id=1-636271867303650973-2491837168&amp;rd=1&amp;co=GENIE.Platform%3DAndroid&amp;oco=1">https://support.google.com/hangouts/answer/3115410?visit_id=1-636271867303650973-2491837168&amp;rd=1&amp;co=GENIE.Platform%3DAndroid&amp;oco=1</a>  <a href="https://play.google.com/store/apps/details?id=com.google.android.talk&amp;hl=en">https://play.google.com/store/apps/details?id=com.google.android.talk&amp;hl=en</a></p> </div> <div data-bbox="504 1053 1919 1308" data-label="Text"> <p><b>Regarding Google Maps</b>, Google Latitude, Google Plus, Google Hangouts, Google Messages, Google Allo, Google Duo, Google Chrome, and Android Messenger, the Accused Products require a user to join the corresponding network by signing-in to the device with an identifier (e.g., Google Account). When the signed-in user enables Google Maps Share Location and Location Services on the first device, the user shares its location and the first device’s location is sent to a server. When the user enables sharing to one or more contacts (of respective devices) and the one or more contacts enable sharing their location to the user of the first device, the user of the first device receives the locations of the one or more contacts.</p> </div> <div data-bbox="504 1347 1904 1421" data-label="Text"> <p>The first device’s participation in the group is based on receiving the message from the second device, i.e. a message indicating that the second device is sharing its location.</p> </div> <div data-bbox="1346 228 1743 917" data-label="Image"> </div>

**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC
	<p>By participating in the Maps location sharing functionality, the device sends location information to a server (e.g., a network server provided by an ISP such as AT&amp;T and/or a server running Google’s services). The device also receives location information from the server indicating the location of other devices that are sharing location information via Maps.</p> <p><b><u>Further regarding Google Maps</u></b>, Google Latitude, Google Plus, Google Hangouts, Google Messages, Google Allo, Google Duo, Google Chrome, and Android Messenger, the Accused Products alternatively require a user to send a request containing the identifier (e.g., Google Account). When the signed-in user enables Google Maps Share Location and Location Services on the first device, the user shares its location and the first device’s location is sent to a server. When the user sends a message to another contact through Google Maps, Google Messages, and/or another means from within the Google Maps application, the message including location information are sent to a server before transmission to the intended contact. When one or more contacts enable sharing their location to the user of the first device, or alternatively send a message containing location information, or alternatively accept a request to share their location with the first user, the user of the first device receives the locations of the one or more contacts.</p> <p><b><u>Exemplary Support for Google Maps:</u></b></p>

## Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products

US9408055B2	HTC
	<p data-bbox="548 240 968 261">COMPUTER <b>ANDROID</b> IPHONE &amp; IPAD</p> <hr data-bbox="533 289 1566 293"/> <h3 data-bbox="533 347 1024 383">If they have a Google Account</h3> <ol data-bbox="533 406 1419 698" style="list-style-type: none"><li data-bbox="533 406 1220 427">1. If you haven't already, add their Gmail address to your <a href="#">Google Contacts</a>.</li><li data-bbox="533 443 1419 464">2. On your Android phone or tablet, open the Google Maps app  and sign in. <a href="#">Learn how to sign in.</a></li><li data-bbox="533 480 1035 501">3. Tap Menu  &gt; <b>Location sharing</b> &gt; Add People .</li><li data-bbox="533 518 1005 539">4. Choose how long you want to share your location.</li><li data-bbox="533 555 1140 613">5. Tap <b>Select People</b>.<ul data-bbox="569 589 1140 613" style="list-style-type: none"><li data-bbox="569 589 1140 613">• If you're asked about your contacts, give Google Maps access.</li></ul></li><li data-bbox="533 639 884 660">6. Choose who you want to share with.</li><li data-bbox="533 677 663 698">7. Tap <b>Share</b>.</li></ol> <h3 data-bbox="533 756 1110 792">If they don't have a Google Account</h3> <ol data-bbox="533 815 1560 938" style="list-style-type: none"><li data-bbox="533 815 1419 836">1. On your Android phone or tablet, open the Google Maps app  and sign in. <a href="#">Learn how to sign in.</a></li><li data-bbox="533 852 1035 873">2. Tap Menu  &gt; <b>Location sharing</b> &gt; Add People .</li><li data-bbox="533 889 1560 938">3. Tap More  &gt; <b>Copy to clipboard</b>. People with this link can see your location for as long as you choose, up to 72 hours.</li></ol> <h3 data-bbox="533 987 869 1023">Share using another app</h3> <p data-bbox="533 1045 1205 1066">You can also share through messaging apps. Tap More  &gt; select an app.</p> <h3 data-bbox="533 1125 743 1161">Stop sharing</h3> <ol data-bbox="533 1183 1205 1281" style="list-style-type: none"><li data-bbox="533 1183 842 1205">1. Open the Google Maps app .</li><li data-bbox="533 1221 869 1242">2. Tap Menu  &gt; <b>Location sharing</b>.</li><li data-bbox="533 1258 1205 1281">3. Next to the person with whom you want to stop sharing, tap Remove .</li></ol> <p data-bbox="512 1299 1703 1326"><a href="https://support.google.com/maps/answer/7326816?co=GENIE.Platform%3DAndroid&amp;oco=1">https://support.google.com/maps/answer/7326816?co=GENIE.Platform%3DAndroid&amp;oco=1</a></p>

## Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products

US9408055B2	HTC
	<h3>Share your E.T.A</h3> <p>After you start your drive, you can share your destination, estimated arrival time, and where you are on the route.</p> <ol style="list-style-type: none"><li>1. Open the Google Maps app .</li><li>2. Set a driving destination. <a href="#">Learn how to navigate to a place.</a></li><li>3. After you start navigation, tap More  &gt; <b>Share trip progress.</b></li><li>4. Choose a person from the list.</li><li>5. Tap <b>Share.</b></li><li>6. Location Sharing will stop when you reach your destination or stop navigating.</li></ol> <ul style="list-style-type: none"><li>• To stop sharing before you arrive, tap More  &gt; <b>Stop sharing.</b></li></ul> <h3>See where someone is</h3> <p>If someone shares their location with you, you can see them on the map.</p> <ol style="list-style-type: none"><li>1. Open the Google Maps app .</li><li>2. Tap Menu  &gt; <b>Location sharing.</b></li><li>3. Choose someone.</li></ol> <ul style="list-style-type: none"><li>• To see an updated location, tap on a friend's icon &gt; More  &gt; <b>Refresh.</b></li></ul> <h3>Stop seeing someone's location</h3> <ol style="list-style-type: none"><li>1. Open the Google Maps app .</li><li>2. On the map, tap their icon.</li><li>3. At the bottom, tap More .</li><li>4. To temporarily hide someone, tap <b>Hide from map.</b> You can stop hiding them at any time.</li></ol> <p><b>Note:</b> You can stop someone's location from ever appearing on your map. <a href="#">Learn how to block another person's account.</a></p> <p><a href="https://support.google.com/maps/answer/7326816?co=GENIE.Platform%3DAndroid&amp;oco=1">https://support.google.com/maps/answer/7326816?co=GENIE.Platform%3DAndroid&amp;oco=1</a></p>

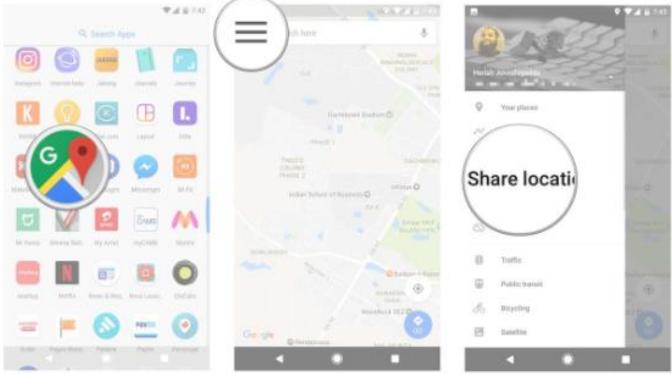
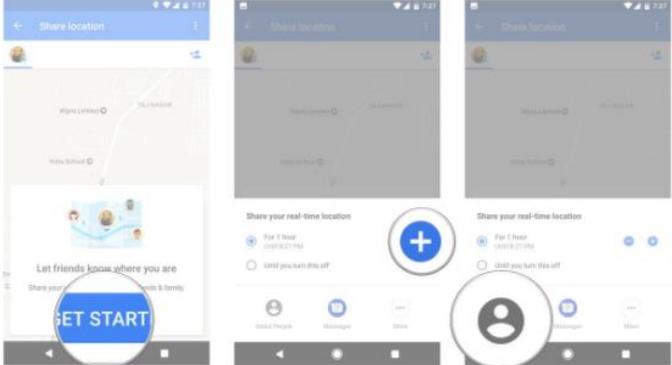
## Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products

US9408055B2	HTC
	<h3 data-bbox="527 240 997 289">Create a list of places</h3> <p data-bbox="527 310 1339 331">In Google Maps, you can create a list of places, like your favorite places or places you want to visit.</p> <p data-bbox="527 402 932 423">COMPUTER <b>ANDROID</b> IPHONE &amp; IPAD</p> <hr data-bbox="527 446 1371 451"/> <h3 data-bbox="527 505 766 537">Make a new list</h3> <ol data-bbox="527 558 1087 716" style="list-style-type: none"><li>1. On your Android phone or tablet, open the Google Maps app .</li><li>2. Tap Menu  &gt; <b>Your places</b> &gt; <b>Saved</b>.</li><li>3. In the bottom right, tap Add .</li><li>4. Enter a name and description.</li><li>5. Tap <b>Save</b>.</li></ol> <h3 data-bbox="527 776 842 808">Save a place to a list</h3> <ol data-bbox="527 829 978 987" style="list-style-type: none"><li>1. Open the Google Maps app .</li><li>2. Search for a place or tap it on the map.</li><li>3. At the bottom, tap the place's name or address.</li><li>4. Tap <b>Save</b>.</li><li>5. Choose a list. To create a new list, tap <b>New list</b> .</li></ol> <h3 data-bbox="527 1047 732 1079">See your lists</h3> <ol data-bbox="527 1101 879 1154" style="list-style-type: none"><li>1. Open the Google Maps app .</li><li>2. Tap Menu  &gt; <b>Your places</b> &gt; <b>Saved</b>.</li></ol> <p data-bbox="512 1166 1898 1230"><a href="https://support.google.com/maps/answer/7280933?hl=en&amp;ref_topic=7301134&amp;co=GENIE.Platform%3DAndroid&amp;oco=1">https://support.google.com/maps/answer/7280933?hl=en&amp;ref_topic=7301134&amp;co=GENIE.Platform%3DAndroid&amp;oco=1</a></p>

## Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products

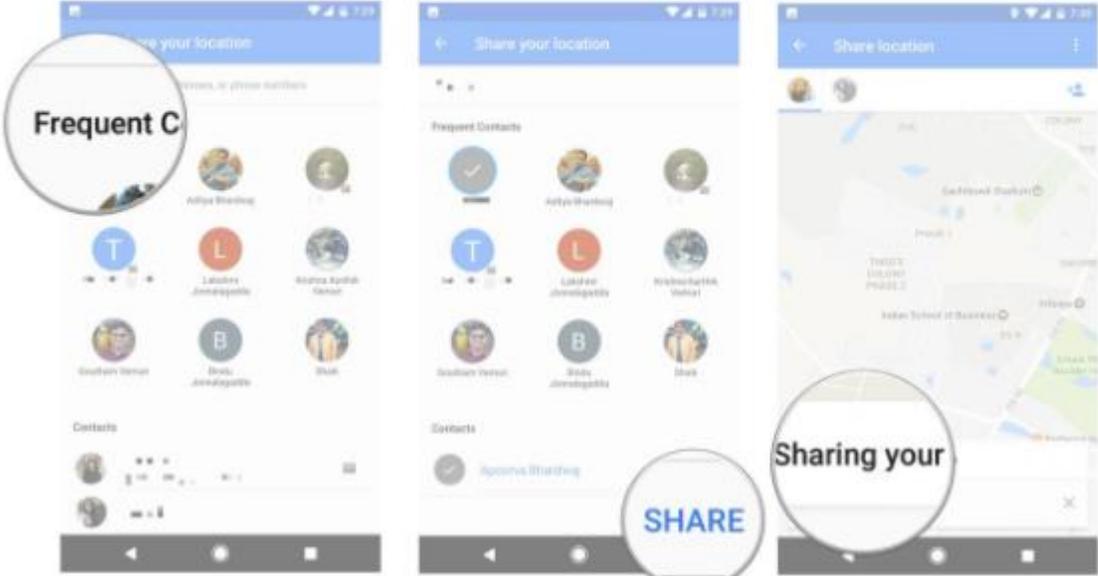
US9408055B2	HTC
	<h3 data-bbox="541 245 877 282">Hide or share lists</h3> <p data-bbox="541 310 909 334"><b>Note:</b> You can't share starred places.</p> <ol data-bbox="554 367 1251 472" style="list-style-type: none"><li data-bbox="554 367 890 391">1. Open the Google Maps app .</li><li data-bbox="554 407 968 431">2. Tap Menu  &gt; <b>Your places</b> &gt; <b>Saved</b>.</li><li data-bbox="554 448 1251 472">3. Next to the list you want to share, tap More  &gt; choose an option:</li></ol> <ul data-bbox="583 488 1682 626" style="list-style-type: none"><li data-bbox="583 488 1440 513">• <b>Hide/Show on your map:</b> Display or hide your saved places when looking at the map.</li><li data-bbox="583 529 1058 553">• <b>Share list:</b> Allow others to see your saved list.</li><li data-bbox="583 570 1682 626">• <b>Sharing options:</b> You can make your list public, private, or shared. To let anyone with the link see your list, tap <b>Shared</b>. To let anyone find and follow your list, tap <b>Public</b>.</li></ul> <h3 data-bbox="541 699 768 737">Follow a list</h3> <p data-bbox="541 764 1728 821">If you follow a list made by someone else, their saved places will show up in Your Places. The places will also appear as suggested locations in Google Maps.</p> <h3 data-bbox="541 878 915 915">Follow a list using a link</h3> <ol data-bbox="554 935 1356 1040" style="list-style-type: none"><li data-bbox="554 935 957 959">1. Tap on the link you received to open it.</li><li data-bbox="554 976 1272 1000">2. Tap <b>Follow</b>. This list will now be added to the group of lists you follow.</li><li data-bbox="554 1016 1356 1040">3. <b>Optional:</b> To unfollow a list someone shared with you, tap the list &gt; <b>Following</b>.</li></ol> <h3 data-bbox="541 1097 926 1135">See lists made by others</h3> <p data-bbox="541 1154 1335 1179">If a user has any Google Maps lists that were made public, you can follow them.</p> <ol data-bbox="554 1211 1136 1317" style="list-style-type: none"><li data-bbox="554 1211 1136 1235">1. Tap on the name of a user whose list you want to follow.</li><li data-bbox="554 1252 674 1276">2. Tap <b>Lists</b>.</li><li data-bbox="554 1292 1136 1317">3. Tap on the list you want to follow &gt; More  &gt; <b>Follow</b>.</li></ol> <p data-bbox="512 1333 1902 1398"><a href="https://support.google.com/maps/answer/7280933?hl=en&amp;ref_topic=7301134&amp;co=GENIE.Platform%3DAndroid&amp;oco=1">https://support.google.com/maps/answer/7280933?hl=en&amp;ref_topic=7301134&amp;co=GENIE.Platform%3DAndroid&amp;oco=1</a></p>

## Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products

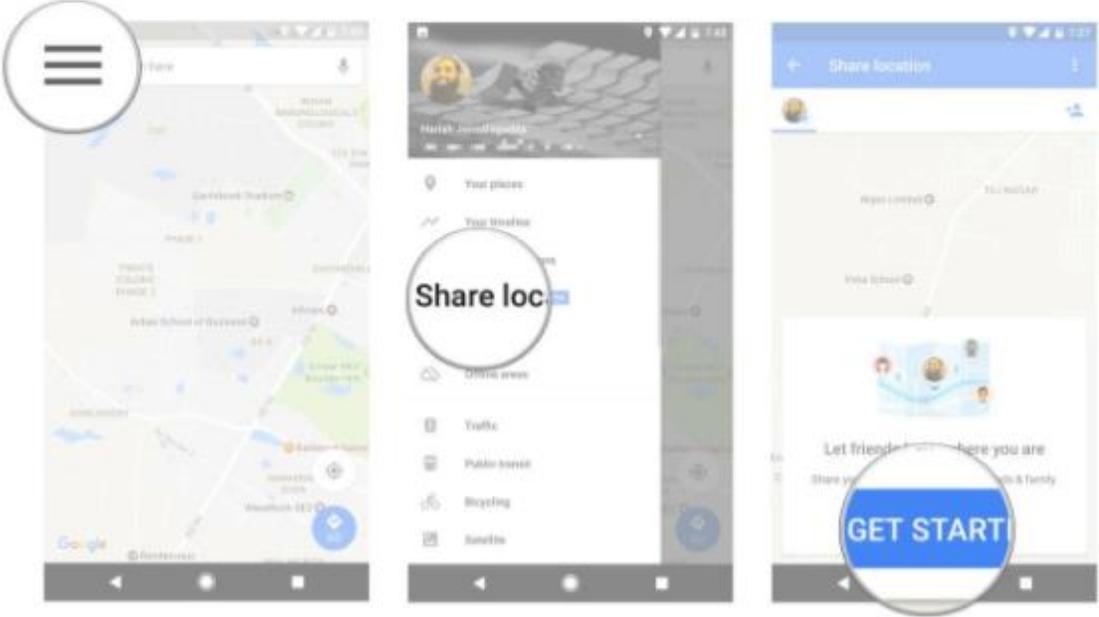
US9408055B2	HTC
	<p data-bbox="520 240 1150 272"><b>How to share your location in Google Maps</b></p> <ol data-bbox="520 302 1134 389" style="list-style-type: none"> <li>1. Open Google Maps from the app drawer or the home screen.</li> <li>2. Tap the hamburger menu (the three horizontal lines) on the top left corner of the screen.</li> <li>3. Select Share location.</li> </ol>  <ol data-bbox="520 828 1165 933" style="list-style-type: none"> <li>4. Tap Get Started.</li> <li>5. Use the + icon to select a time period or select the <b>Until you turn this off</b> setting to share your location indefinitely.</li> <li>6. Tap Select People.</li> </ol>  <p data-bbox="520 1339 1354 1372"><a href="https://www.androidcentral.com/how-share-location-google-maps">https://www.androidcentral.com/how-share-location-google-maps</a></p>



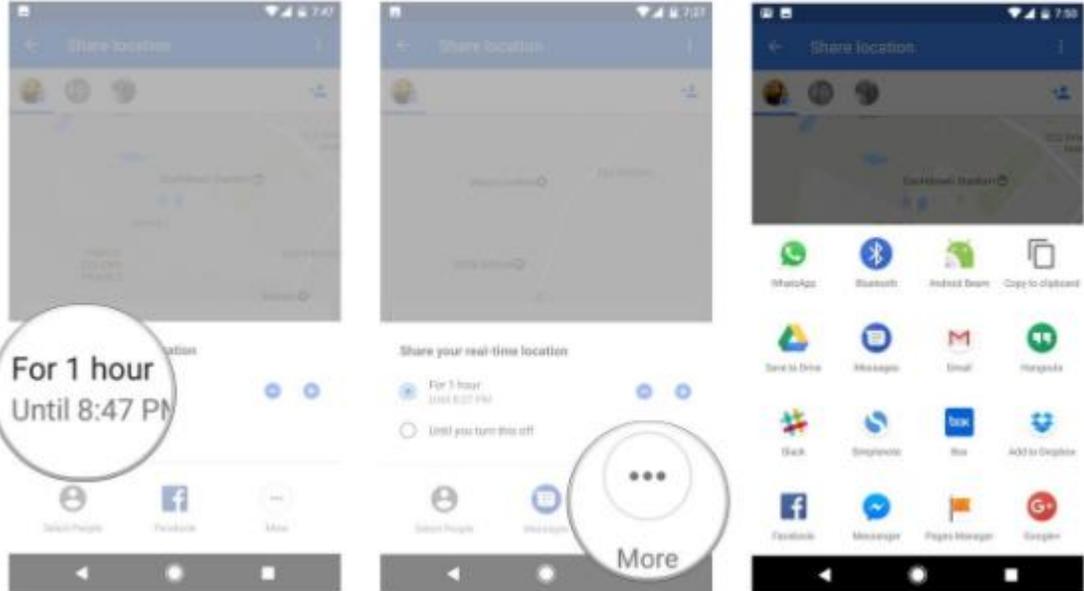
### Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products

US9408055B2	HTC
	<p data-bbox="527 253 1577 310">7. You'll see a list of your frequent contacts at the top, along with a full list of contacts. Pick the contacts by tapping their name.</p> <p data-bbox="527 339 1457 367">8. Once you've selected the contacts you want to share your location to, tap <b>Share</b>.</p> <p data-bbox="527 396 1419 423">9. You'll see a message saying that the selected contact can view your location.</p>  <p data-bbox="510 1101 1356 1128"><a href="https://www.androidcentral.com/how-share-location-google-maps">https://www.androidcentral.com/how-share-location-google-maps</a></p>

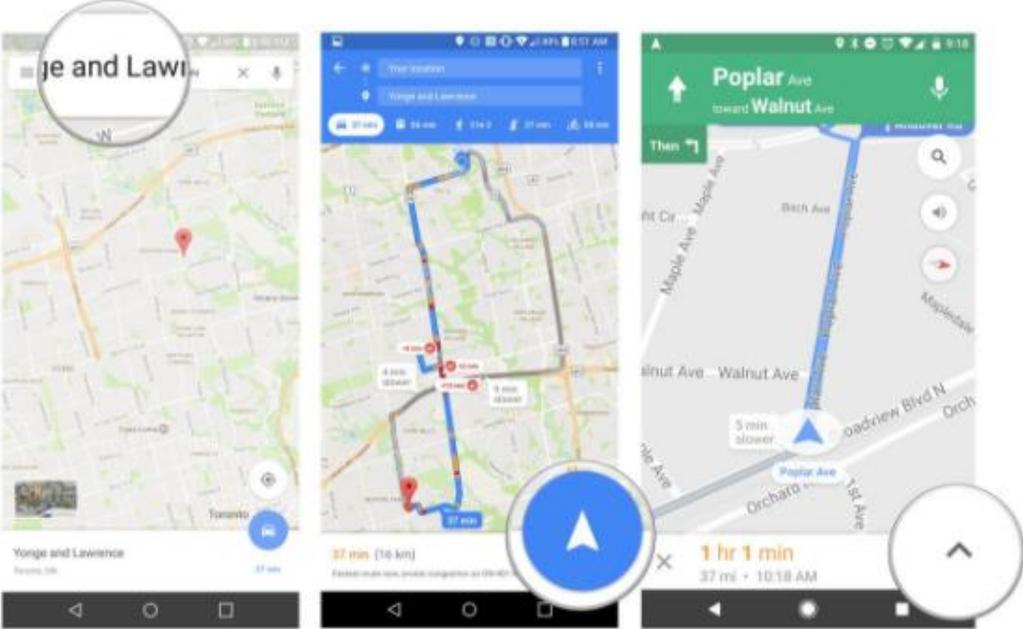
## Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products

US9408055B2	HTC
	<h3 data-bbox="520 245 1255 293">How to create a shareable link</h3> <p data-bbox="520 334 1461 363">You can also create a link and use it to share your location easily. Here's how to do it:</p> <ol data-bbox="520 412 1234 553" style="list-style-type: none"><li data-bbox="520 412 1234 441">1. Tap the hamburger menu on the top left corner of the screen.</li><li data-bbox="520 467 800 496">2. Select Share location.</li><li data-bbox="520 522 737 552">3. Tap Get Started.</li></ol>  <p data-bbox="506 1230 1356 1268"><a href="https://www.androidcentral.com/how-share-location-google-maps">https://www.androidcentral.com/how-share-location-google-maps</a></p>

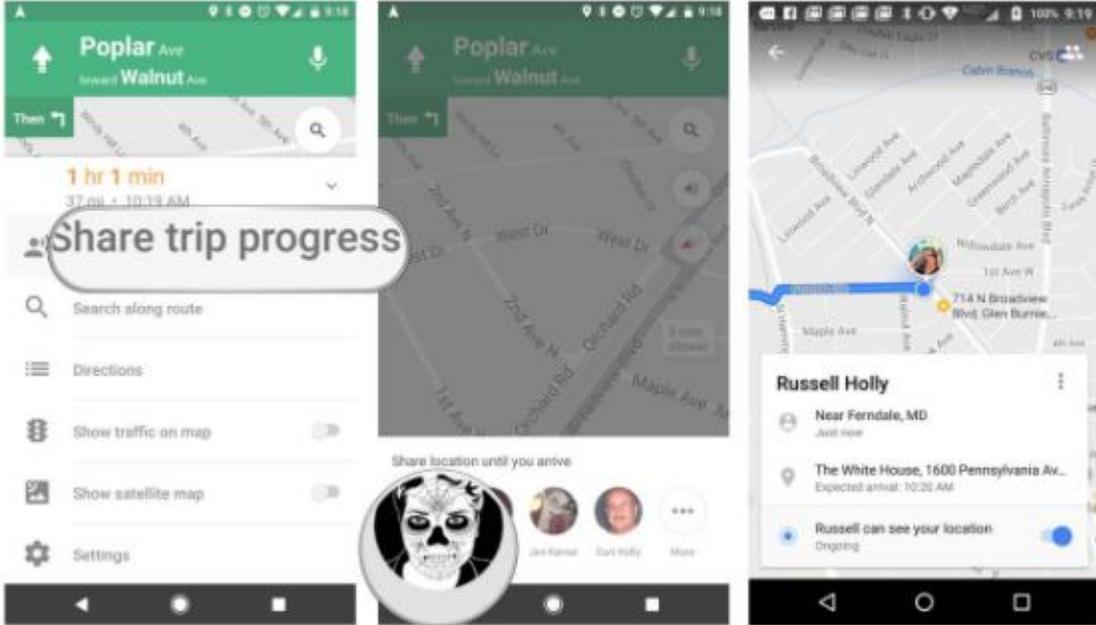
## Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products

US9408055B2	HTC
	<p data-bbox="520 245 1213 272">4. Select the <b>amount of time</b> you want to share your location.</p> <p data-bbox="520 302 680 329">5. Tap <b>More</b>.</p> <p data-bbox="520 358 1633 418">6. Select <b>your app of choice</b> to create and send a <b>unique URL</b> that broadcasts your current location. You can email it, send the link via Messenger, or even tweet it to the intended recipient.</p>  <p data-bbox="506 1084 1360 1117"><a href="https://www.androidcentral.com/how-share-location-google-maps">https://www.androidcentral.com/how-share-location-google-maps</a></p>

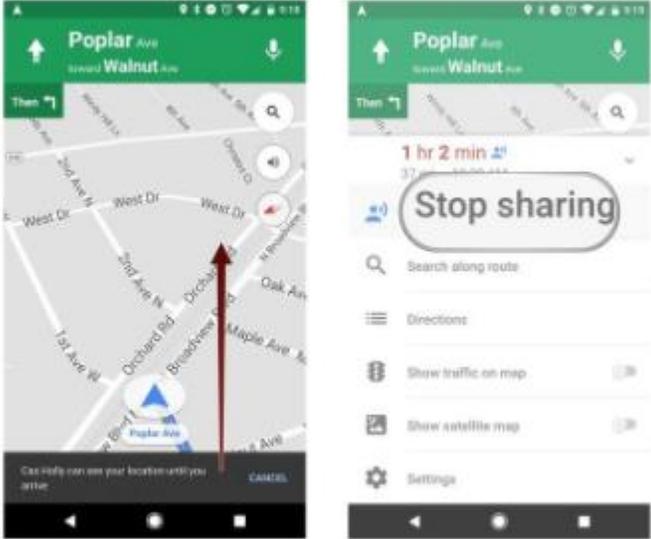
**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC
	<p data-bbox="527 240 1428 337"><b>How to share your navigation directions while you walk, drive or transit</b></p> <p data-bbox="527 375 1556 464">One of the best ways to share your location with someone is during a drive, walk or transit. If you're meeting a friend or family member somewhere, or navigating towards their home, Google Maps lets you share your location with them for the duration of the trip. It's magic!</p> <ol data-bbox="527 513 1396 646" style="list-style-type: none"> <li data-bbox="527 513 976 537">1. In the <b>search bar</b> enter your destination.</li> <li data-bbox="527 565 1396 589">2. Pick your navigation type (drive, transit, walk) and press the blue <b>navigate</b> button.</li> <li data-bbox="527 617 1396 641">3. Tap the arrow next to the <b>time-to-destination</b> number at the bottom of the screen.</li> </ol>  <p data-bbox="512 1328 1356 1359"><a href="https://www.androidcentral.com/how-share-location-google-maps">https://www.androidcentral.com/how-share-location-google-maps</a></p>

### Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products

US9408055B2	HTC
	<p data-bbox="527 240 835 267">4. Tap Share trip progress.</p> <p data-bbox="527 297 1150 324">5. Choose one or more contacts to share trip progress.</p>  <p data-bbox="537 1027 1339 1055">You can also stop sharing your location with someone before a trip ends.</p> <p data-bbox="512 1065 1356 1092"><a href="https://www.androidcentral.com/how-share-location-google-maps">https://www.androidcentral.com/how-share-location-google-maps</a></p>

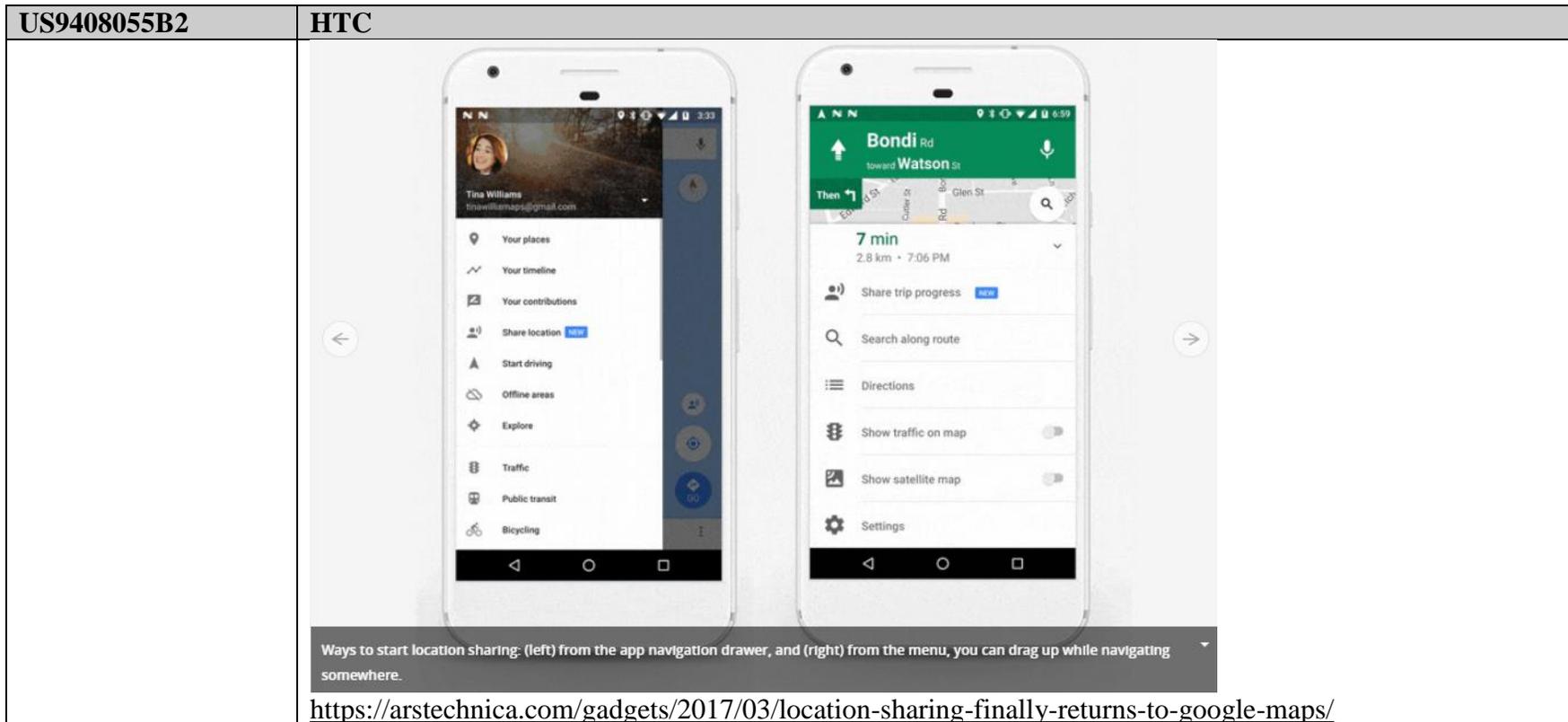
## Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products

US9408055B2	HTC
	<ol style="list-style-type: none"><li data-bbox="533 245 1470 272">1. Tap the arrow next to the <b>time-to-destination</b> number at the bottom of the screen.</li><li data-bbox="533 302 768 329">2. Tap <b>Stop sharing</b>.</li></ol> <div data-bbox="747 383 1398 922"></div> <p data-bbox="541 976 636 1003">That's it!</p> <p data-bbox="541 1045 1612 1073">Are you excited that location sharing is back in Google Maps? How often do you use the feature?</p> <p data-bbox="510 1084 1356 1112"><a href="https://www.androidcentral.com/how-share-location-google-maps">https://www.androidcentral.com/how-share-location-google-maps</a></p> <p data-bbox="510 1122 1419 1149">As shown below, a group may also be defined within Google Contacts.</p>

## Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products

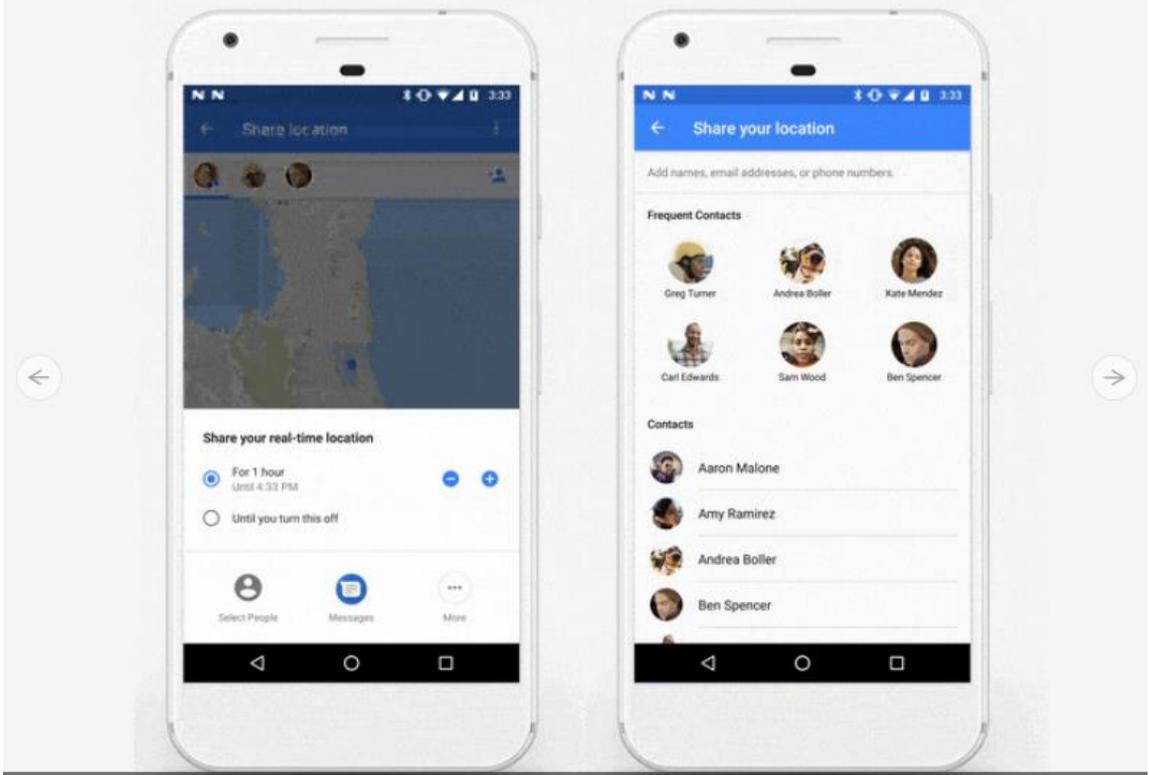
US9408055B2	HTC
	<p data-bbox="541 240 894 277"><b>See your contacts</b></p> <ol data-bbox="552 306 968 378" style="list-style-type: none"><li data-bbox="552 306 968 337">1. Open your device's Contacts app .</li><li data-bbox="552 350 730 378">2. Tap Menu .</li></ol> <ul data-bbox="552 410 1734 605" style="list-style-type: none"><li data-bbox="552 410 1115 438">• <b>See contacts by label:</b> Choose a label from the list.</li><li data-bbox="552 454 1367 482">• <b>See contacts for another account:</b> Tap Down arrow  &gt; pick an account.</li><li data-bbox="552 498 1213 526">• <b>See the contacts for all your accounts:</b> Choose <b>All contacts</b>.</li></ul> <p data-bbox="573 537 1734 565"><b>Tip:</b> If you have multiple contacts with the same information, the information will be grouped into one contact.</p> <li data-bbox="552 581 1360 605">• <b>See your Google Account contacts on the web:</b> Go to <a href="#">Google Contacts</a> .</li> <p data-bbox="512 638 1535 665"><a href="https://support.google.com/android/answer/6118731?hl=en&amp;ref_topic=6118711">https://support.google.com/android/answer/6118731?hl=en&amp;ref_topic=6118711</a></p> <p data-bbox="541 678 894 716"><b>Label your contacts</b></p> <p data-bbox="541 743 982 771">You can group contacts together using labels.</p> <ol data-bbox="552 800 926 898" style="list-style-type: none"><li data-bbox="552 800 926 828">1. Open your device's Contacts app .</li><li data-bbox="552 841 863 868">2. Tap Menu  &gt; <b>Create label</b>.</li><li data-bbox="552 881 873 898">3. Enter a label name and tap <b>Ok</b>.</li></ol> <ul data-bbox="552 930 1713 995" style="list-style-type: none"><li data-bbox="552 930 1234 958">• <b>Add one contact to a label:</b> Tap Add contact  &gt; choose a contact.</li><li data-bbox="552 974 1713 995">• <b>Add multiple contacts to a label:</b> Tap Add contact  &gt; touch and hold a contact &gt; tap the other contacts &gt; tap <b>Add</b>.</li></ul> <p data-bbox="512 1011 1535 1039"><a href="https://support.google.com/android/answer/6118731?hl=en&amp;ref_topic=6118711">https://support.google.com/android/answer/6118731?hl=en&amp;ref_topic=6118711</a></p> <p data-bbox="552 1092 940 1130"><b>Share your contacts</b></p> <ol data-bbox="552 1162 1045 1325" style="list-style-type: none"><li data-bbox="552 1162 978 1190">1. Open your device's Contacts app .</li><li data-bbox="552 1206 842 1234">2. Tap a contact in the list.</li><li data-bbox="552 1250 831 1278">3. Tap More  &gt; <b>Share</b>.</li><li data-bbox="552 1294 1045 1325">4. Choose how you want to share the contact.</li></ol> <p data-bbox="512 1341 1535 1369"><a href="https://support.google.com/android/answer/6118731?hl=en&amp;ref_topic=6118711">https://support.google.com/android/answer/6118731?hl=en&amp;ref_topic=6118711</a></p>

## Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products





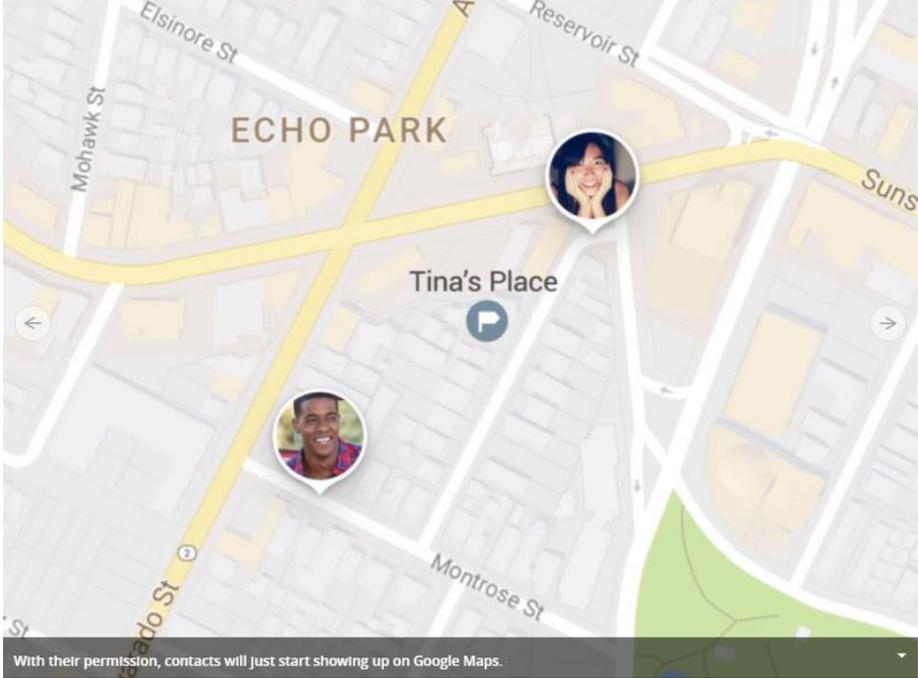
### Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products

US9408055B2	HTC
	 <p data-bbox="512 1024 1661 1057">You can pick how long you want to share your location for, and with whom you want to share it.</p> <p data-bbox="512 1065 1661 1097"><a href="https://arstechnica.com/gadgets/2017/03/location-sharing-finally-returns-to-google-maps/">https://arstechnica.com/gadgets/2017/03/location-sharing-finally-returns-to-google-maps/</a></p>

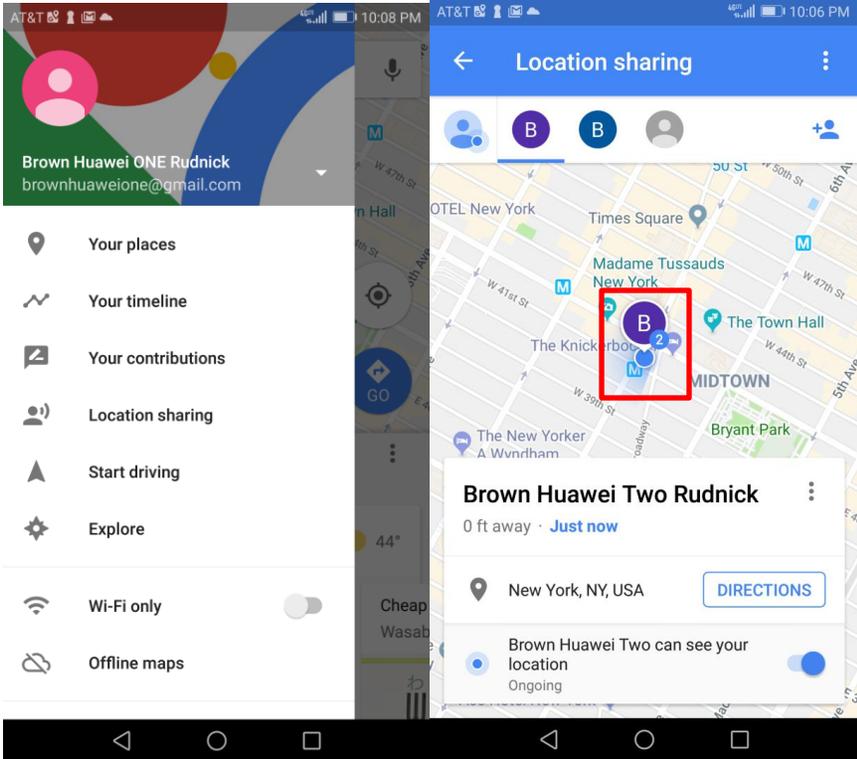
**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC
	 <p data-bbox="514 1144 1176 1177">People you've shared with will get a notification from Google Maps.</p> <p data-bbox="514 1188 1659 1226"><a href="https://arstechnica.com/gadgets/2017/03/location-sharing-finally-returns-to-google-maps/">https://arstechnica.com/gadgets/2017/03/location-sharing-finally-returns-to-google-maps/</a></p>

**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC
	 <p data-bbox="512 883 1430 911">With their permission, contacts will just start showing up on Google Maps.</p> <p data-bbox="512 915 1656 948"><a href="https://arstechnica.com/gadgets/2017/03/location-sharing-finally-returns-to-google-maps/">https://arstechnica.com/gadgets/2017/03/location-sharing-finally-returns-to-google-maps/</a></p> <p data-bbox="512 953 1020 985"><b><u>Exemplary Google Maps Screenshots</u></b></p>

### Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products

US9408055B2	HTC
	 <p data-bbox="1402 329 1797 594">Location information is shared via IP-based communication resulting in map that displays location information</p> <p data-bbox="512 997 856 1027"><b><u>Exemplary Source Code:</u></b></p> <p data-bbox="512 1034 1887 1211">The above functionality is performed at least in part by the following publicly available source code and/or source code that invokes or is invoked by the following source code (or a substantially similar copy compiled and loaded onto the Accused Products by HTC). AGIS reserves the right to amend these contentions to include additional source code as discovery progresses and as additional source code is made available.</p>

## Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products

US9408055B2	HTC
	<h3 data-bbox="527 240 1016 293">Contacts Provider</h3> <p data-bbox="527 329 1472 591">The Contacts Provider is a powerful and flexible Android component that manages the device's central repository of data about people. The Contacts Provider is the source of data you see in the device's contacts application, and you can also access its data in your own application and transfer data between the device and online services. The provider accommodates a wide range of data sources and tries to manage as much data as possible for each person, with the result that its organization is complex. Because of this, the provider's API includes an extensive set of contract classes and interfaces that facilitate both data retrieval and modification.</p> <p data-bbox="527 626 835 646">This guide describes the following:</p> <ul data-bbox="527 678 1373 850" style="list-style-type: none"><li data-bbox="527 678 806 698">• The basic provider structure.</li><li data-bbox="527 727 894 747">• How to retrieve data from the provider.</li><li data-bbox="527 776 863 795">• How to modify data in the provider.</li><li data-bbox="527 824 1373 850">• How to write a sync adapter for synchronizing data from your server to the Contacts Provider.</li></ul> <p data-bbox="512 863 1486 889"><a href="https://developer.android.com/guide/topics/providers/contacts-provider.html">https://developer.android.com/guide/topics/providers/contacts-provider.html</a></p>

**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC
	<p data-bbox="531 240 657 267"><b>Overview</b></p> <p data-bbox="531 297 1608 316">ContactsContract defines an extensible database of contact-related information. Contact information is stored in a three-tier data model:</p> <ul data-bbox="531 342 1713 545" style="list-style-type: none"> <li data-bbox="531 342 1713 394">• A row in the <code>ContactsContract.Data</code> table can store any kind of personal data, such as a phone number or email addresses. The set of data kinds that can be stored in this table is open-ended. There is a predefined set of common kinds, but any application can add its own data kinds.</li> <li data-bbox="531 418 1713 470">• A row in the <code>ContactsContract.RawContacts</code> table represents a set of data describing a person and associated with a single account (for example, one of the user's Gmail accounts).</li> <li data-bbox="531 495 1713 545">• A row in the <code>ContactsContract.Contacts</code> table represents an aggregate of one or more RawContacts presumably describing the same person. When data in or associated with the RawContacts table is changed, the affected aggregate contacts are updated as necessary.</li> </ul> <p data-bbox="531 574 695 594">Other tables include:</p> <ul data-bbox="531 620 1713 899" style="list-style-type: none"> <li data-bbox="531 620 1713 672">• <code>ContactsContract.Groups</code>, which contains information about raw contact groups such as Gmail contact groups. The current API does not support the notion of groups spanning multiple accounts.</li> <li data-bbox="531 696 1325 716">• <code>ContactsContract.StatusUpdates</code>, which contains social status updates including IM availability.</li> <li data-bbox="531 740 1493 760">• <code>ContactsContract.AggregationExceptions</code>, which is used for manual aggregation and disaggregation of raw contacts</li> <li data-bbox="531 784 1325 803">• <code>ContactsContract.Settings</code>, which contains visibility and sync settings for accounts and groups.</li> <li data-bbox="531 828 1346 847">• <code>ContactsContract.SyncState</code>, which contains free-form data maintained on behalf of sync adapters</li> <li data-bbox="531 872 1129 891">• <code>ContactsContract.PhoneLookup</code>, which is used for quick caller-ID lookup</li> </ul> <p data-bbox="510 919 1541 948"><a href="https://developer.android.com/reference/android/provider/ContactsContract.html">https://developer.android.com/reference/android/provider/ContactsContract.html</a></p> <p data-bbox="531 959 615 987"><b>Data</b></p> <p data-bbox="531 1029 1745 1170">As noted previously, the data for a raw contact is stored in a <code>ContactsContract.Data</code> row that is linked to the raw contact's <code>_ID</code> value. This allows a single raw contact to have multiple instances of the same type of data such as email addresses or phone numbers. For example, if "Thomas Higginson" for <code>emilyd@gmail.com</code> (the raw contact row for Thomas Higginson associated with the Google account <code>emilyd@gmail.com</code>) has a home email address of <code>thigg@gmail.com</code> and a work email address of <code>thomas.higginson@gmail.com</code>, the Contacts Provider stores the two email address rows and links them both to the raw contact.</p> <p data-bbox="531 1196 1745 1308">Notice that different types of data are stored in this single table. Display name, phone number, email, postal address, photo, and website detail rows are all found in the <code>ContactsContract.Data</code> table. To help manage this, the <code>ContactsContract.Data</code> table has some columns with descriptive names, and others with generic names. The contents of a descriptive-name column have the same meaning regardless of the type of data in the row, while the contents of a generic-name column have different meanings depending on the type of data.</p> <p data-bbox="510 1326 1488 1356"><a href="https://developer.android.com/guide/topics/providers/contacts-provider.html">https://developer.android.com/guide/topics/providers/contacts-provider.html</a></p>

**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC											
	<table border="1"> <thead> <tr> <th data-bbox="514 240 609 272">Task</th> <th data-bbox="615 240 854 272">Action</th> <th data-bbox="861 240 1188 272">Data</th> <th data-bbox="1194 240 1486 272">MIME type</th> <th data-bbox="1493 240 1745 272">Notes</th> </tr> </thead> <tbody> <tr> <td data-bbox="514 277 609 941">Pick a contact from a list</td> <td data-bbox="615 277 854 941">ACTION_PICK</td> <td data-bbox="861 277 1188 941">                     One of:                     <ul style="list-style-type: none"> <li>• <code>Contacts.CONTENT_URI</code>, which displays a list of contacts.</li> <li>• <code>Phone.CONTENT_URI</code>, which displays a list of phone numbers for a raw contact.</li> <li>• <code>StructuredPostal.CONTENT_URI</code>, which displays a list of postal addresses for a raw contact.</li> <li>• <code>Email.CONTENT_URI</code>, which displays a list of email addresses for a raw contact.</li> </ul> </td> <td data-bbox="1194 277 1486 941">Not used</td> <td data-bbox="1493 277 1745 941">                     Displays a list of raw contacts or a list of data from a raw contact, depending on the content URI type you supply. Call <code>startActivityForResult()</code>, which returns the content URI of the selected row. The form of the URI is the table's content URI with the row's <code>LOOKUP_ID</code> appended to it. The device's contacts app delegates read and write permissions to this content URI for the life of your activity. See the <a href="#">Content Provider Basics</a> guide for more details.                 </td> </tr> </tbody> </table>	Task	Action	Data	MIME type	Notes	Pick a contact from a list	ACTION_PICK	One of: <ul style="list-style-type: none"> <li>• <code>Contacts.CONTENT_URI</code>, which displays a list of contacts.</li> <li>• <code>Phone.CONTENT_URI</code>, which displays a list of phone numbers for a raw contact.</li> <li>• <code>StructuredPostal.CONTENT_URI</code>, which displays a list of postal addresses for a raw contact.</li> <li>• <code>Email.CONTENT_URI</code>, which displays a list of email addresses for a raw contact.</li> </ul>	Not used	Displays a list of raw contacts or a list of data from a raw contact, depending on the content URI type you supply. Call <code>startActivityForResult()</code> , which returns the content URI of the selected row. The form of the URI is the table's content URI with the row's <code>LOOKUP_ID</code> appended to it. The device's contacts app delegates read and write permissions to this content URI for the life of your activity. See the <a href="#">Content Provider Basics</a> guide for more details.	<a href="https://developer.android.com/guide/topics/providers/contacts-provider.html">https://developer.android.com/guide/topics/providers/contacts-provider.html</a>
Task	Action	Data	MIME type	Notes								
Pick a contact from a list	ACTION_PICK	One of: <ul style="list-style-type: none"> <li>• <code>Contacts.CONTENT_URI</code>, which displays a list of contacts.</li> <li>• <code>Phone.CONTENT_URI</code>, which displays a list of phone numbers for a raw contact.</li> <li>• <code>StructuredPostal.CONTENT_URI</code>, which displays a list of postal addresses for a raw contact.</li> <li>• <code>Email.CONTENT_URI</code>, which displays a list of email addresses for a raw contact.</li> </ul>	Not used	Displays a list of raw contacts or a list of data from a raw contact, depending on the content URI type you supply. Call <code>startActivityForResult()</code> , which returns the content URI of the selected row. The form of the URI is the table's content URI with the row's <code>LOOKUP_ID</code> appended to it. The device's contacts app delegates read and write permissions to this content URI for the life of your activity. See the <a href="#">Content Provider Basics</a> guide for more details.								

**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC
	<pre>25  * Parsed form of the intent sent to the Contacts application. 26  */ 27  public class ContactsRequest { 28 29      /** Default mode: browse contacts */ 30      public static final int ACTION_DEFAULT = 10; 31 32      /** Show all contacts */ 33      public static final int ACTION_ALL_CONTACTS = 15; 34 35      /** Show all contacts with phone numbers */ 36      public static final int ACTION_CONTACTS_WITH_PHONES = 17; 37 38      /** Show contents of a specific group */ 39      public static final int ACTION_GROUP = 20; 40 41      /** Show all starred contacts */ 42      public static final int ACTION_STARRED = 30; 43 44      /** Show frequently contacted contacts */ 45      public static final int ACTION_FREQUENT = 40; 46 47      /** Show starred and the frequent */ 48      public static final int ACTION_STREQUENT = 50; 49 50      /** Show all contacts and pick them when clicking */ 51      public static final int ACTION_PICK_CONTACT = 60; 52 53      /** Show all contacts as well as the option to create a new one */ 54      public static final int ACTION_PICK_OR_CREATE_CONTACT = 70; 55 56      /** Show all contacts and pick them for edit when clicking, and allow creating a new contact */ 57      public static final int ACTION_INSERT_OR_EDIT_CONTACT = 80;</pre> <p><a href="https://android.googlesource.com/platform/packages/apps/Contacts/+nougat-mr1-release/src/com/android/contacts/list/ContactsRequest.java">https://android.googlesource.com/platform/packages/apps/Contacts/+nougat-mr1-release/src/com/android/contacts/list/ContactsRequest.java</a></p>



**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC
	<pre>59  /** Show all phone numbers and pick them when clicking */ 60  public static final int ACTION_PICK_PHONE = 90; 61 62  /** Show all postal addresses and pick them when clicking */ 63  public static final int ACTION_PICK_POSTAL = 100; 64 65  /** Show all postal addresses and pick them when clicking */ 66  public static final int ACTION_PICK_EMAIL = 105; 67 68  /** Show all contacts and create a shortcut for the picked contact */ 69  public static final int ACTION_CREATE_SHORTCUT_CONTACT = 110; 70 71  /** Show all phone numbers and create a call shortcut for the picked number */ 72  public static final int ACTION_CREATE_SHORTCUT_CALL = 120; 73 74  /** Show all phone numbers and create an SMS shortcut for the picked number */ 75  public static final int ACTION_CREATE_SHORTCUT_SMS = 130; 76 77  /** Show all contacts and activate the specified one */ 78  public static final int ACTION_VIEW_CONTACT = 140; 79 80  /** Show contacts recommended for joining with a specified target contact */ 81  public static final int ACTION_PICK_JOIN = 150;</pre> <p><a href="https://android.googlesource.com/platform/packages/apps/Contacts+/nougat-mr1-release/src/com/android/contacts/list/ContactsRequest.java">https://android.googlesource.com/platform/packages/apps/Contacts+/nougat-mr1-release/src/com/android/contacts/list/ContactsRequest.java</a></p>

**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC
	<pre> 104  * Displays a list to browse contacts. 105  */ 106  public class PeopleActivity extends ContactsActivity implements 107      View.OnCreateContextMenuListener, 108      View.OnClickListener, 109      ActionBarAdapter.Listener, 110      DialogManager.DialogShowingViewActivity, 111      ContactListFilterController.ContactListFilterListener, 112      ProviderStatusListener, 113      MultiContactDeleteListener, 114      JoinContactsListener { https://android.googlesource.com/platform/packages/apps/Contacts+/nougat-mr1- release/src/com/android/contacts/activities/PeopleActivity.java 145  * Showing a list of Contacts. Also used for showing search results in search mode. 146  */ 147  private MultiSelectContactsListFragment mAllFragment; 148  private ContactTileListFragment mFavoritesFragment; https://android.googlesource.com/platform/packages/apps/Contacts+/nougat-mr1- release/src/com/android/contacts/activities/PeopleActivity.java </pre>

**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC
	<pre>458     private void configureFragments(boolean fromRequest) { 459         if (fromRequest) { 460             ContactListFilter filter = null; 461             int actionCode = mRequest.getActionCode(); 462             boolean searchMode = mRequest.isSearchMode(); 463             final int tabToOpen; 464             switch (actionCode) { 465                 case ContactsRequest.ACTION_ALL_CONTACTS: 466                     filter = ContactListFilter.createFilterWithType( 467                         ContactListFilter.FILTER_TYPE_ALL_ACCOUNTS); 468                     tabToOpen = TabState.ALL; 469                     break; 470                 case ContactsRequest.ACTION_CONTACTS_WITH_PHONES: 471                     filter = ContactListFilter.createFilterWithType( 472                         ContactListFilter.FILTER_TYPE_WITH_PHONE_NUMBERS_ONLY); 473                     tabToOpen = TabState.ALL; 474                     break; 475 476                 case ContactsRequest.ACTION_FREQUENT: 477                 case ContactsRequest.ACTION_STREQUENT: 478                 case ContactsRequest.ACTION_STARRED: 479                     tabToOpen = TabState.FAVORITES; 480                     break; 481                 case ContactsRequest.ACTION_VIEW_CONTACT: 482                     tabToOpen = TabState.ALL; 483                     break; 484                 default: 485                     tabToOpen = -1; 486                     break; 487             } </pre> <p data-bbox="506 1321 1570 1390"><a href="https://android.googlesource.com/platform/packages/apps/Contacts+/nougat-mr1-release/src/com/android/contacts/activities/PeopleActivity.java">https://android.googlesource.com/platform/packages/apps/Contacts+/nougat-mr1-release/src/com/android/contacts/activities/PeopleActivity.java</a></p>

**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC
	<pre>488         if (tabToOpen != -1) { 489             mActionBarAdapter.setCurrentTab(tabToOpen); 490         } 491 492         if (filter != null) { 493             mContactListFilterController.setContactListFilter(filter, false); 494             searchMode = false; 495         } 496 497         if (mRequest.getContactUri() != null) { 498             searchMode = false; 499         } 500 501         mActionBarAdapter.setSearchMode(searchMode); 502         configureContactListFragmentForRequest(); 503     } 504 505     configureContactListFragment(); 506 507     invalidateOptionsMenuIfNeeded(); 508 }</pre> <p data-bbox="506 1019 1570 1084"><a href="https://android.googlesource.com/platform/packages/apps/Contacts+/nougat-mr1-release/src/com/android/contacts/activities/PeopleActivity.java">https://android.googlesource.com/platform/packages/apps/Contacts+/nougat-mr1-release/src/com/android/contacts/activities/PeopleActivity.java</a></p>

**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC
	<pre>35 public class ProfileAndContactsLoader extends CursorLoader { 36 37     private boolean mLoadProfile; 38 39     private String[] mProjection; 40 41     private Uri mExtraUri; 42     private String[] mExtraProjection; 43     private String mExtraSelection; 44     private String[] mExtraSelectionArgs; 45     private boolean mMergeExtraContactsAfterPrimary; 46 47     public ProfileAndContactsLoader(Context context) { 48         super(context); 49     }</pre> <p><a href="https://android.googlesource.com/platform/packages/apps/ContactsCommon/+/nougat-mr1-release/src/com/android/contacts/common/list/ProfileAndContactsLoader.java">https://android.googlesource.com/platform/packages/apps/ContactsCommon/+/nougat-mr1-release/src/com/android/contacts/common/list/ProfileAndContactsLoader.java</a></p>

**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC
	<pre>35 public final class GroupMemberLoader extends CursorLoader { 36 37     public static class GroupEditorQuery { 38         private static final String[] PROJECTION = new String[] { 39             Data.CONTACT_ID,           // 0 40             Data.RAW_CONTACT_ID,      // 1 41             Data.DISPLAY_NAME_PRIMARY, // 2 42             Data.PHOTO_URI,           // 3 43             Data.LOOKUP_KEY,          // 4 44         }; 45 46         public static final int CONTACT_ID           = 0; 47         public static final int RAW_CONTACT_ID       = 1; 48         public static final int CONTACT_DISPLAY_NAME_PRIMARY = 2; 49         public static final int CONTACT_PHOTO_URI    = 3; 50         public static final int CONTACT_LOOKUP_KEY   = 4; 51     } 52 53     public static class GroupDetailQuery { 54         private static final String[] PROJECTION = new String[] { 55             Data.CONTACT_ID,           // 0 56             Data.PHOTO_URI,           // 1 57             Data.LOOKUP_KEY,          // 2 58             Data.DISPLAY_NAME_PRIMARY, // 3 59             Data.CONTACT_PRESENCE,    // 4 60             Data.CONTACT_STATUS,      // 5 61         }; 62 63         public static final int CONTACT_ID           = 0; 64         public static final int CONTACT_PHOTO_URI    = 1; 65         public static final int CONTACT_LOOKUP_KEY   = 2; 66         public static final int CONTACT_DISPLAY_NAME_PRIMARY = 3; 67         public static final int CONTACT_PRESENCE_STATUS = 4; 68         public static final int CONTACT_STATUS      = 5; 69     } 70 71     private final long mGroupId;</pre> <p><a href="https://android.googlesource.com/platform/packages/apps/Contacts/+/nougat-mr1-release/src/com/android/contacts/GroupMemberLoader.java">https://android.googlesource.com/platform/packages/apps/Contacts/+/nougat-mr1-release/src/com/android/contacts/GroupMemberLoader.java</a></p>

**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC
	<pre> 24  * Group loader for the group list that includes details such as the number of contacts per group 25  * and number of groups per account. This list is sorted by account type, account name, where the 26  * group names are in alphabetical order. Note that the list excludes default, favorite, and deleted 27  * groups. 28  */ 29  public final class GroupListLoader extends CursorLoader { 30 31      private final static String[] COLUMNS = new String[] { 32          Groups.ACCOUNT_NAME, 33          Groups.ACCOUNT_TYPE, 34          Groups.DATA_SET, 35          Groups._ID, 36          Groups.TITLE, 37          Groups.SUMMARY_COUNT, 38      }; 39 40      public final static int ACCOUNT_NAME = 0; 41      public final static int ACCOUNT_TYPE = 1; 42      public final static int DATA_SET = 2; 43      public final static int GROUP_ID = 3; 44      public final static int TITLE = 4; 45      public final static int MEMBER_COUNT = 5; 46 47      private static final Uri GROUP_LIST_URI = Groups.CONTENT_SUMMARY_URI; 48 49      public GroupListLoader(Context context) { 50          super(context, GROUP_LIST_URI, COLUMNS, Groups.ACCOUNT_TYPE + " NOT NULL AND " 51              + Groups.ACCOUNT_NAME + " NOT NULL AND " + Groups.AUTO_ADD + "=0 AND " + 52              Groups.FAVORITES + "=0 AND " + Groups.DELETED + "=0", null, 53              Groups.ACCOUNT_TYPE + ", " + Groups.ACCOUNT_NAME + ", " + Groups.DATA_SET + ", " + 54              Groups.TITLE + " COLLATE LOCALIZED ASC"); 55      } 56  } </pre> <p><a href="https://android.googlesource.com/platform/packages/apps/Contacts+/nougat-mr1-release/src/com/android/contacts/GroupListLoader.java">https://android.googlesource.com/platform/packages/apps/Contacts+/nougat-mr1-release/src/com/android/contacts/GroupListLoader.java</a></p>

**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC
	<pre> 24  * Group meta-data loader. Loads all groups or just a single group from the 25  * database (if given a {@link Uri}). 26  */ 27  public final class GroupMetaLoader extends CursorLoader { 28 29      private final static String[] COLUMNS = new String[] { 30          Groups.ACCOUNT_NAME, 31          Groups.ACCOUNT_TYPE, 32          Groups.DATA_SET, 33          Groups._ID, 34          Groups.TITLE, 35          Groups.AUTO_ADD, 36          Groups.FAVORITES, 37          Groups.GROUP_IS_READ_ONLY, 38          Groups.DELETED, 39      }; 40 41      public final static int ACCOUNT_NAME = 0; 42      public final static int ACCOUNT_TYPE = 1; 43      public final static int DATA_SET = 2; 44      public final static int GROUP_ID = 3; 45      public final static int TITLE = 4; 46      public final static int AUTO_ADD = 5; 47      public final static int FAVORITES = 6; 48      public final static int IS_READ_ONLY = 7; 49      public final static int DELETED = 8; 50 51      public GroupMetaLoader(Context context, Uri groupUri) { 52          super(context, ensureIsGroupUri(groupUri), COLUMNS, Groups.ACCOUNT_TYPE + " NOT NULL AND " 53              + Groups.ACCOUNT_NAME + " NOT NULL", null, null); 54      } 55 56      /** 57       * Ensures that this is a valid group URI. If invalid, then an exception is 58       * thrown. Otherwise, the original URI is returned. 59       */ 60      private static Uri ensureIsGroupUri(final Uri groupUri) { 61          // TODO: Fix ContactsProvider2 getType method to resolve the group Uris 62          if (groupUri == null) { 63              throw new IllegalArgumentException("Uri must not be null"); 64          } 65          if (!groupUri.toString().startsWith(Groups.CONTENT_URI.toString())) { 66              throw new IllegalArgumentException("Invalid group Uri: " + groupUri); 67          } 68          return groupUri; 69      } 70  } </pre> <p><a href="https://android.googlesource.com/platform/packages/apps/Contacts/+nougat-mr1-release/src/com/android/contacts/GroupMetaLoader.java">https://android.googlesource.com/platform/packages/apps/Contacts/+nougat-mr1-release/src/com/android/contacts/GroupMetaLoader.java</a></p>



**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC
	<pre> 19  * Meta-data for a contact group. We load all groups associated with the contact's 20  * constituent accounts. 21  */ 22  public final class GroupMetaData { 23      private String mAccountName; 24      private String mAccountType; 25      private String mDataSet; 26      private long mGroupId; 27      private String mTitle; 28      private boolean mDefaultGroup; 29      private boolean mFavorites; 30 31      public GroupMetaData(String accountName, String accountType, String dataSet, long groupId, 32                          String title, boolean defaultGroup, boolean favorites) { 33          this.mAccountName = accountName; 34          this.mAccountType = accountType; 35          this.mDataSet = dataSet; 36          this.mGroupId = groupId; 37          this.mTitle = title; 38          this.mDefaultGroup = defaultGroup; 39          this.mFavorites = favorites; 40      } 41 42      public String getAccountName() { 43          return mAccountName; 44      } 45 46      public String getAccountType() { 47          return mAccountType; 48      } 49 50      public String getDataSet() { 51          return mDataSet; 52      } 53 54      public long getGroupId() { 55          return mGroupId; 56      } 57 58      public String getTitle() { 59          return mTitle; 60      } 61 62      public boolean isDefaultGroup() { 63          return mDefaultGroup; 64      } 65 66      public boolean isFavorites() { 67          return mFavorites; </pre> <p><a href="https://android.googlesource.com/platform/packages/apps/ContactsCommon/+/nougat-mr1-release/src/com/android/contacts/common/GroupMetaData.java">https://android.googlesource.com/platform/packages/apps/ContactsCommon/+/nougat-mr1-release/src/com/android/contacts/common/GroupMetaData.java</a></p>

**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC
	<pre> 44  * Class that sends chat message via SMS. 45  * 46  * The interface emulates a blocking sending similar to making an HTTP request. 47  * It calls the SmsManager to send a (potentially multipart) message and waits 48  * on the sent status on each part. The waiting has a timeout so it won't wait 49  * forever. Once the sent status of all parts received, the call returns. 50  * A successful sending requires success status for all parts. Otherwise, we 51  * pick the highest level of failure as the error for the whole message, which 52  * is used to determine if we need to retry the sending. 53  */ 54  public class SmsSender { 55      private static final String TAG = LogUtil.BUGLE_TAG; 56 57      public static final String EXTRA_PART_ID = "part_id"; 58 59      /* 60       * A map for pending sms messages. The key is the random request UUID. 61       */ 62      private static ConcurrentHashMap&lt;Uri, SendResult&gt; sPendingMessageMap = 63          new ConcurrentHashMap&lt;Uri, SendResult&gt;(); 64 65      private static final Random RANDOM = new Random(); 66 67      // Whether we should send multipart SMS as separate messages 68      private static Boolean sSendMultipartSmsAsSeparateMessages = null; 69 </pre> <p><a href="https://android.googlesource.com/platform/packages/apps/Messaging/+nougat-mr1-release/src/com/android/messaging/sms/SmsSender.java">https://android.googlesource.com/platform/packages/apps/Messaging/+nougat-mr1-release/src/com/android/messaging/sms/SmsSender.java</a></p>

**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC
	<pre> 253 // Actually sending the message using SmsManager 254 private static void sendInternal(final Context context, final int subId, String dest, 255     final ArrayList&lt;String&gt; messages, final String serviceCenter, 256     final boolean requireDeliveryReport, final Uri messageUri) throws SmsException { 257     Assert.notNull(context); 258     final SmsManager smsManager = PhoneUtils.get(subId).getSmsManager(); 259     final int messageCount = messages.size(); 260     final ArrayList&lt;PendingIntent&gt; deliveryIntents = new ArrayList&lt;PendingIntent&gt;(messageCount); 261     final ArrayList&lt;PendingIntent&gt; sentIntents = new ArrayList&lt;PendingIntent&gt;(messageCount); 262     for (int i = 0; i &lt; messageCount; i++) { 263         // Make pending intents different for each message part 264         final int partId = (messageCount &lt;= 1 ? 0 : i + 1); 265         if (requireDeliveryReport &amp;&amp; (i == (messageCount - 1))) { 266             // TODO we only care about the delivery status of the last part 267             // Shall we have better tracking of delivery status of all parts? 268             deliveryIntents.add(PendingIntent.getBroadcast( 269                 context, 270                 partId, 271                 getSendStatusIntent(context, SendStatusReceiver.MESSAGE_DELIVERED_ACTION, 272                     messageUri, partId, subId), 273                 0/*flag*/)); 274         } else { 275             deliveryIntents.add(null); 276         } 277         sentIntents.add(PendingIntent.getBroadcast( 278             context, 279             partId, 280             getSendStatusIntent(context, SendStatusReceiver.MESSAGE_SENT_ACTION, 281                 messageUri, partId, subId), 282             0/*flag*/)); 283     } 284     if (sSendMultipartSmsAsSeparateMessages == null) { 285         sSendMultipartSmsAsSeparateMessages = MmsConfig.get(subId) 286             .getSendMultipartSmsAsSeparateMessages(); 287     } </pre> <p><a href="https://android.googlesource.com/platform/packages/apps/Messaging/+nougat-mr1-release/src/com/android/messaging/sms/SmsSender.java">https://android.googlesource.com/platform/packages/apps/Messaging/+nougat-mr1-release/src/com/android/messaging/sms/SmsSender.java</a></p>

**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC
	<pre> 288     try { 289         if (sSendMultipartSmsAsSeparateMessages) { 290             // If multipart sms is not supported, send them as separate messages 291             for (int i = 0; i &lt; messageCount; i++) { 292                 smsManager.sendTextMessage(dest, 293                     serviceCenter, 294                     messages.get(i), 295                     sentIntents.get(i), 296                     deliveryIntents.get(i)); 297             } 298         } else { 299             smsManager.sendMultipartTextMessage( 300                 dest, serviceCenter, messages, sentIntents, deliveryIntents); 301         } 302     } catch (final Exception e) { 303         throw new SmsException("SmsSender: caught exception in sending " + e); 304     } 305 } </pre> <p><a href="https://android.googlesource.com/platform/packages/apps/Messaging/+nougat-mr1-release/src/com/android/messaging/sms/SmsSender.java">https://android.googlesource.com/platform/packages/apps/Messaging/+nougat-mr1-release/src/com/android/messaging/sms/SmsSender.java</a></p> <pre> 56  * Class that receives incoming SMS messages through android.provider.Telephony.SMS_RECEIVED 57  * 58  * This class serves two purposes: 59  * - Process phone verification SMS messages 60  * - Handle SMS messages when the user has enabled us to be the default SMS app (Pre-KLP) 61  */ 62  public final class SmsReceiver extends BroadcastReceiver { 63      private static final String TAG = LogUtil.BUGLE_TAG; 64 65      private static ArrayList&lt;Pattern&gt; sIgnoreSmsPatterns; 66 </pre> <p><a href="https://android.googlesource.com/platform/packages/apps/Messaging/+nougat-mr1-release/src/com/android/messaging/receiver/SmsReceiver.java">https://android.googlesource.com/platform/packages/apps/Messaging/+nougat-mr1-release/src/com/android/messaging/receiver/SmsReceiver.java</a></p>

**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC
	<pre> 200     public static void deliverSmsMessages(final Context context, final int subId, 201         final int errorCode, final android.telephony.SmsMessage[] messages) { 202         final ContentValues messageValues = 203             MmsUtils.parseReceivedSmsMessage(context, messages, errorCode); 204 205         LogUtil.v(TAG, "SmsReceiver.deliverSmsMessages"); 206 207         final long nowInMillis = System.currentTimeMillis(); 208         final long receivedTimestampMs = MmsUtils.getMessageDate(messages[0], nowInMillis); 209 210         messageValues.put(Sms.Inbox.DATE, receivedTimestampMs); 211         // Default to unread and unseen for us but ReceiveSmsMessageAction will override 212         // seen for the telephony db. 213         messageValues.put(Sms.Inbox.READ, 0); 214         messageValues.put(Sms.Inbox.SEEN, 0); 215         if (OsUtil.isAtLeastL_MR1()) { 216             messageValues.put(Sms.SUBSCRIPTION_ID, subId); 217         } 218 219         if (messages[0].getMessageClass() == android.telephony.SmsMessage.MessageClass.CLASS_0    220             DebugUtils.debugClassZeroSmsEnabled()) { 221             Factory.get().getUIIntents().launchClassZeroActivity(context, messageValues); 222         } else { 223             final ReceiveSmsMessageAction action = new ReceiveSmsMessageAction(messageValues); 224             action.start(); 225         } 226     } </pre> <p><a href="https://android.googlesource.com/platform/packages/apps/Messaging/+nougat-mr1-release/src/com/android/messaging/receiver/SmsReceiver.java">https://android.googlesource.com/platform/packages/apps/Messaging/+nougat-mr1-release/src/com/android/messaging/receiver/SmsReceiver.java</a></p>

**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC
	<pre>228     @Override 229     public void onReceive(final Context context, final Intent intent) { 230         LogUtil.v(TAG, "SmsReceiver.onReceive " + intent); 231         // On KLP+ we only take delivery of SMS messages in SmsDeliverReceiver. 232         if (PhoneUtils.getDefault().isSmsEnabled()) { 233             final String action = intent.getAction(); 234             if (OsUtil.isSecondaryUser() &amp;&amp; 235                 (Telephony.Sms.Intents.SMS_RECEIVED_ACTION.equals(action)    236                  // TODO: update this with the actual constant from Telephony 237                  "android.provider.Telephony.MMS_DOWNLOADED".equals(action))) { 238                 postNewMessageSecondaryUserNotification(); 239             } else if (!OsUtil.isAtLeastKLP()) { 240                 deliverSmsIntent(context, intent); 241             } 242         } 243     }</pre> <p><a href="https://android.googlesource.com/platform/packages/apps/Messaging/+/nougat-mr1-release/src/com/android/messaging/receiver/SmsReceiver.java">https://android.googlesource.com/platform/packages/apps/Messaging/+/nougat-mr1-release/src/com/android/messaging/receiver/SmsReceiver.java</a></p>

**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC
	<pre>52  * Class that sends chat message via MMS. 53  * 54  * The interface emulates a blocking send similar to making an HTTP request. 55  */ 56  public class MmsSender { 57      private static final String TAG = LogUtil.BUGLE_TAG; 58 59      /** 60       * Send an MMS message. 61       * 62       * @param context Context 63       * @param messageUri The unique URI of the message for identifying it during sending 64       * @param sendReq The SendReq PDU of the message 65       * @throws MmsFailureException 66       */ 67      public static void sendMms(final Context context, final int subId, final Uri messageUri, 68                               final SendReq sendReq, final Bundle sentIntentExtras) throws MmsFailureException { 69          sendMms(context, 70                subId, 71                messageUri, 72                null /* locationUri */, 73                sendReq, 74                true /* responseImportant */, 75                sentIntentExtras); 76      }</pre> <p><a href="https://android.googlesource.com/platform/packages/apps/Messaging/+/nougat-mr1-release/src/com/android/messaging/sms/MmsSender.java">https://android.googlesource.com/platform/packages/apps/Messaging/+/nougat-mr1-release/src/com/android/messaging/sms/MmsSender.java</a></p>

**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC
	<pre> 240      * Download an MMS message. 241      * 242      * @param context Context 243      * @param contentLocation The url of the MMS message 244      * @throws MmsFailureException 245      * @throws InvalidHeaderValueException 246      */ 247      public static void downloadMms(final Context context, final int subId, 248          final String contentLocation, Bundle extras) throws MmsFailureException, 249          InvalidHeaderValueException { 250          final Uri requestUri = Uri.parse(contentLocation); 251          final Uri contentUri = MmsFileProvider.buildRawMmsUri(); 252 253          final Intent downloadedIntent = new Intent(SendStatusReceiver.MMS_DOWNLOADED_ACTION, 254              requestUri, 255              context, 256              SendStatusReceiver.class); 257          downloadedIntent.putExtra(SendMessageAction.EXTRA_CONTENT_URI, contentUri); 258          if (extras != null) { 259              downloadedIntent.putExtras(extras); 260          } 261          final PendingIntent downloadedPendingIntent = PendingIntent.getBroadcast( 262              context, 263              0 /*request code*/, 264              downloadedIntent, 265              PendingIntent.FLAG_UPDATE_CURRENT); 266 267          MmsManager.downloadMultimediaMessage(subId, context, contentLocation, contentUri, 268              downloadedPendingIntent); 269      } </pre> <p><a href="https://android.googlesource.com/platform/packages/apps/Messaging/+nougat-mr1-release/src/com/android/messaging/sms/MmsSender.java">https://android.googlesource.com/platform/packages/apps/Messaging/+nougat-mr1-release/src/com/android/messaging/sms/MmsSender.java</a></p>



**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC
	<pre> 97      * Execute an MMS HTTP request, either a POST (sending) or a GET (downloading) 98      * 99      * @param urlString The request URL, for sending it is usually the MMSC, and for downloading 100     *           it is the message URL 101     * @param pdu For POST (sending) only, the PDU to send 102     * @param method HTTP method, POST for sending and GET for downloading 103     * @param isProxySet Is there a proxy for the MMSC 104     * @param proxyHost The proxy host 105     * @param proxyPort The proxy port 106     * @param mmsConfig The MMS config to use 107     * @param userAgent The user agent header value 108     * @param uaProfUrl The UA Prof URL header value 109     * @return The HTTP response body 110     * @throws MmsHttpException For any failures 111     */ 112     public byte[] execute(String urlString, byte[] pdu, String method, boolean isProxySet, 113         String proxyHost, int proxyPort, Bundle mmsConfig, String userAgent, String uaProfUrl) 114         throws MmsHttpException { 115         Log.d(MmsService.TAG, "HTTP: " + method + " " + Utils.redactUrlForNonVerbose(urlString) 116             + (isProxySet ? (" proxy=" + proxyHost + ":" + proxyPort) : "")) 117             + ", PDU size=" + (pdu != null ? pdu.length : 0)); 118         checkMethod(method); 119         HttpURLConnection connection = null; 120         try { 121             Proxy proxy = Proxy.NO_PROXY; 122             if (isProxySet) { 123                 proxy = new Proxy(Proxy.Type.HTTP, new InetSocketAddress(proxyHost, proxyPort)); 124             } 125             final URL url = new URL(urlString); 126             // Now get the connection 127             connection = (HttpURLConnection) url.openConnection(proxy); 128             connection.setDoInput(true); 129             connection.setConnectTimeout( 130                 mmsConfig.getInt(CarrierConfigValuesLoader.CONFIG_HTTP_SOCKET_TIMEOUT, 131                     CarrierConfigValuesLoader.CONFIG_HTTP_SOCKET_TIMEOUT_DEFAULT)); </pre> <p data-bbox="506 1219 1596 1287"><a href="https://android.googlesource.com/platform/packages/apps/Messaging/+nougat-mr1-release/src/android/support/v7/mms/MmsHttpClient.java">https://android.googlesource.com/platform/packages/apps/Messaging/+nougat-mr1-release/src/android/support/v7/mms/MmsHttpClient.java</a></p>

**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC
	<pre> 132 // ----- COMMON HEADERS ----- 133 // Header: Accept 134 connection.setRequestProperty(HEADER_ACCEPT, HEADER_VALUE_ACCEPT); 135 // Header: Accept-Language 136 connection.setRequestProperty( 137     HEADER_ACCEPT_LANGUAGE, getCurrentAcceptLanguage(Locale.getDefault())); 138 // Header: User-Agent 139 Log.i(MmsService.TAG, "HTTP: User-Agent=" + userAgent); 140 connection.setRequestProperty(HEADER_USER_AGENT, userAgent); 141 // Header: x-wap-profile 142 final String uaProfUrlTagName = mmsConfig.getString( 143     CarrierConfigValuesLoader.CONFIG_UA_PROF_TAG_NAME, 144     CarrierConfigValuesLoader.CONFIG_UA_PROF_TAG_NAME_DEFAULT); 145 if (uaProfUrl != null) { 146     Log.i(MmsService.TAG, "HTTP: UaProfUrl=" + uaProfUrl); 147     connection.setRequestProperty(uaProfUrlTagName, uaProfUrl); 148 } 149 // Add extra headers specified by mms_config.xml's httpparams 150 addExtraHeaders(connection, mmsConfig); 151 // Different stuff for GET and POST 152 if (METHOD_POST.equals(method)) { 153     if (pdu == null    pdu.length &lt; 1) { 154         Log.e(MmsService.TAG, "HTTP: empty pdu"); 155         throw new MmsHttpException(0/*statusCode*/, "Sending empty PDU"); 156     } 157     connection.setDoOutput(true); 158     connection.setRequestMethod(METHOD_POST); 159     if (mmsConfig.getBoolean( 160         CarrierConfigValuesLoader.CONFIG_SUPPORT_HTTP_CHARSET_HEADER, 161         CarrierConfigValuesLoader.CONFIG_SUPPORT_HTTP_CHARSET_HEADER_DEFAULT)) { 162         connection.setRequestProperty(HEADER_CONTENT_TYPE, 163             HEADER_VALUE_CONTENT_TYPE_WITH_CHARSET); 164     } else { 165         connection.setRequestProperty(HEADER_CONTENT_TYPE, 166             HEADER_VALUE_CONTENT_TYPE_WITHOUT_CHARSET); </pre> <p><a href="https://android.googlesource.com/platform/packages/apps/Messaging/+/nougat-mr1-release/src/android/support/v7/mms/MmsHttpClient.java">https://android.googlesource.com/platform/packages/apps/Messaging/+/nougat-mr1-release/src/android/support/v7/mms/MmsHttpClient.java</a></p>

**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC
	<pre> 167         } 168         if (Log.isLoggable(MmsService.TAG, Log.VERBOSE)) { 169             logHttpHeaders(connection.getRequestProperties()); 170         } 171         connection.setFixedLengthStreamingMode(pdu.length); 172         // Sending request body 173         final OutputStream out = 174             new BufferedOutputStream(connection.getOutputStream()); 175         out.write(pdu); 176         out.flush(); 177         out.close(); 178     } else if (METHOD_GET.equals(method)) { 179         if (Log.isLoggable(MmsService.TAG, Log.VERBOSE)) { 180             logHttpHeaders(connection.getRequestProperties()); 181         } 182         connection.setRequestMethod(METHOD_GET); 183     } 184     // Get response 185     final int responseCode = connection.getResponseCode(); 186     final String responseMessage = connection.getResponseMessage(); 187     Log.d(MmsService.TAG, "HTTP: " + responseCode + " " + responseMessage); 188     if (Log.isLoggable(MmsService.TAG, Log.VERBOSE)) { 189         logHttpHeaders(connection.getHeaderFields()); 190     } 191     if (responseCode / 100 != 2) { 192         throw new MmsHttpException(responseCode, responseMessage); 193     } 194     final InputStream in = new BufferedInputStream(connection.getInputStream()); 195     final ByteArrayOutputStream byteOut = new ByteArrayOutputStream(); 196     final byte[] buf = new byte[4096]; 197     int count = 0; 198     while ((count = in.read(buf)) &gt; 0) { 199         byteOut.write(buf, 0, count); 200     } 201     in.close(); 202     final byte[] responseBody = byteOut.toByteArray(); 203     Log.d(MmsService.TAG, "HTTP: response size=" 204         + (responseBody != null ? responseBody.length : 0)); 205     return responseBody; </pre> <p><a href="https://android.googlesource.com/platform/packages/apps/Messaging/+nougat-mr1-release/src/android/support/v7/mms/MmsHttpClient.java">https://android.googlesource.com/platform/packages/apps/Messaging/+nougat-mr1-release/src/android/support/v7/mms/MmsHttpClient.java</a></p>

**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC
	<pre> 38  * Request to send an MMS 39  */ 40  class SendRequest extends MmsRequest { 41      // Max send response PDU size in bytes (exceeding this may cause problem with 42      // system intent delivery). 43      private static final int MAX_SEND_RESPONSE_SIZE = 1000 * 1024; 44 45      private byte[] mPduData; 46 47      SendRequest(final String locationUri, final Uri pduUri, final PendingIntent sentIntent) { 48          super(locationUri, pduUri, sentIntent); 49      } 50 51      @Override 52      protected boolean loadRequest(final Context context, final Bundle mmsConfig) { 53          mPduData = readPduFromContentUri( 54              context, 55              mPduUri, 56              mmsConfig.getInt( 57                  CarrierConfigValuesLoader.CONFIG_MAX_MESSAGE_SIZE, 58                  CarrierConfigValuesLoader.CONFIG_MAX_MESSAGE_SIZE_DEFAULT)); 59          return (mPduData != null); 60      } 61 62      @Override 63      protected boolean transferResponse(final Context context, final Intent fillIn, 64          final byte[] response) { 65          // SendConf pdus are always small and can be included in the intent 66          if (response != null &amp;&amp; fillIn != null) { 67              if (response.length &gt; MAX_SEND_RESPONSE_SIZE) { 68                  // If the response PDU is too large, it won't be able to fit in 69                  // the PendingIntent to be transferred via system IPC. 70                  return false; 71              } 72              fillIn.putExtra(SmsManager.EXTRA_MMS_DATA, response); 73          } 74          return true; 75      } </pre>

**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC
	<p><a href="https://android.googlesource.com/platform/packages/apps/Messaging/+/nougat-mr1-release/src/android/support/v7/mms/SendRequest.java">https://android.googlesource.com/platform/packages/apps/Messaging/+/nougat-mr1-release/src/android/support/v7/mms/SendRequest.java</a></p> <pre data-bbox="533 354 1738 397">public static LocationRequest create ()</pre> <p data-bbox="525 423 1029 451">Create a location request with default parameters.</p> <p data-bbox="525 482 1638 542">Default parameters are for a block accuracy, slowly updated location. It can then be adjusted as required by the applications before passing to the <code>FusedLocationProviderApi</code>.</p> <p data-bbox="548 565 630 589"><b>Returns</b></p> <ul data-bbox="554 610 810 634" style="list-style-type: none"> <li>• a new location request</li> </ul> <p><a href="https://developers.google.com/android/reference/com/google/android/gms/location/LocationRequest">https://developers.google.com/android/reference/com/google/android/gms/location/LocationRequest</a></p>

**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC
	<p><code>public static final int PRIORITY_BALANCED_POWER_ACCURACY</code></p> <p>Used with <code>setPriority(int)</code> to request "block" level accuracy.</p> <p>Block level accuracy is considered to be about 100 meter accuracy. Using a coarse accuracy such as this often consumes less power.</p> <p>Constant Value: 102</p> <hr/> <p><code>public static final int PRIORITY_HIGH_ACCURACY</code></p> <p>Used with <code>setPriority(int)</code> to request the most accurate locations available.</p> <p>This will return the finest location available.</p> <p>Constant Value: 100</p> <hr/> <p><code>public static final int PRIORITY_LOW_POWER</code></p> <p>Used with <code>setPriority(int)</code> to request "city" level accuracy.</p> <p>City level accuracy is considered to be about 10km accuracy. Using a coarse accuracy such as this often consumes less power.</p> <p>Constant Value: 104</p> <p><a href="https://developers.google.com/android/reference/com/google/android/gms/location/LocationRequest">https://developers.google.com/android/reference/com/google/android/gms/location/LocationRequest</a></p>

**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC
	<pre data-bbox="533 248 1749 285">public Task&lt;Location&gt; getLastLocation ()</pre> <p data-bbox="527 313 1104 337">Returns the best most recent location currently available.</p> <p data-bbox="527 370 1696 428">If a location is not available, which should happen very rarely, null will be returned. The best accuracy available while respecting the location permissions will be returned.</p> <p data-bbox="527 461 1736 519">This method provides a simplified way to get location. It is particularly well suited for applications that do not require an accurate location and that do not want to maintain extra logic for location updates.</p> <pre data-bbox="533 578 1749 615">public Task&lt;LocationAvailability&gt; getLocationAvailability ()</pre> <p data-bbox="527 643 1692 701">Returns the availability of location data. When <code>isLocationAvailable()</code> returns true, then the location returned by <code>getLastLocation()</code> will be reasonably up to date within the hints specified by the active <code>LocationRequest</code>s.</p> <p data-bbox="527 734 1472 758">If the client isn't connected to Google Play services and the request times out, null is returned.</p> <p data-bbox="527 790 1673 849">Note it's always possible for <code>getLastLocation()</code> to return null even when this method returns true (e.g. location settings were disabled between calls).</p> <p data-bbox="512 865 1898 930"><a href="https://developers.google.com/android/reference/com/google/android/gms/location/FusedLocationProviderClient">https://developers.google.com/android/reference/com/google/android/gms/location/FusedLocationProviderClient</a></p>

**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC						
	<pre data-bbox="527 240 1749 326">public Task&lt;Void&gt; requestLocationUpdates (LocationRequest request,     LocationCallback callback, Looper looper)</pre> <p data-bbox="520 354 1272 378">Requests location updates with a callback on the specified Looper thread.</p> <p data-bbox="520 410 1686 472">This method is suited for the foreground use cases. For background use cases, the <code>PendingIntent</code> version of the method is recommended, see <code>requestLocationUpdates(LocationRequest, PendingIntent)</code>.</p> <p data-bbox="520 505 1371 529">Any previous LocationRequests registered on this LocationListener will be replaced.</p> <p data-bbox="520 561 1686 656">This call will keep the Google Play services connection active, so make sure to call <code>removeLocationUpdates(LocationCallback)</code> when you no longer need it, otherwise you lose the benefits of the automatic connection management.</p> <p data-bbox="520 688 1745 712">Callbacks for <code>LocationCallback</code> will be made on the specified thread, which must already be a prepared looper thread.</p> <p data-bbox="541 737 667 761"><b>Parameters</b></p> <table border="1" data-bbox="520 792 1749 1008"> <tbody> <tr> <td data-bbox="527 800 625 857"><b>request</b></td> <td data-bbox="636 800 1749 857">The location request for the updates.</td> </tr> <tr> <td data-bbox="527 865 625 922"><b>callback</b></td> <td data-bbox="636 865 1749 922">The callback for the location updates.</td> </tr> <tr> <td data-bbox="527 930 625 1003"><b>looper</b></td> <td data-bbox="636 930 1749 1003">The Looper object whose message queue will be used to implement the callback mechanism, or null to make callbacks on the calling thread.</td> </tr> </tbody> </table> <p data-bbox="512 1024 1902 1086"><a href="https://developers.google.com/android/reference/com/google/android/gms/location/FusedLocationProviderClient">https://developers.google.com/android/reference/com/google/android/gms/location/FusedLocationProviderClient</a></p>	<b>request</b>	The location request for the updates.	<b>callback</b>	The callback for the location updates.	<b>looper</b>	The Looper object whose message queue will be used to implement the callback mechanism, or null to make callbacks on the calling thread.
<b>request</b>	The location request for the updates.						
<b>callback</b>	The callback for the location updates.						
<b>looper</b>	The Looper object whose message queue will be used to implement the callback mechanism, or null to make callbacks on the calling thread.						



**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC				
	<p data-bbox="533 245 1738 321"> <code>public Task&lt;Void&gt; requestLocationUpdates (LocationRequest request, PendingIntent callbackIntent)</code> </p> <p data-bbox="533 354 1268 378">Requests location updates with a callback on the specified PendingIntent.</p> <p data-bbox="533 410 1730 540">This method is suited for the background use cases, more specifically for receiving location updates, even when the app has been killed by the system. In order to do so, use a <code>PendingIntent</code> for a started service. For foreground use cases, the <code>LocationCallback</code> version of the method is recommended, see <code>requestLocationUpdates(LocationRequest, LocationCallback, Looper)</code>.</p> <p data-bbox="533 573 1730 630">Any previously registered requests that have the same PendingIntent (as defined by <code>equals(Object)</code>) will be replaced by this request.</p> <p data-bbox="533 662 1730 751">Both <code>LocationResult</code> and <code>LocationAvailability</code> are sent to the given PendingIntent. You can extract data from an Intent using <code>hasResult(Intent)</code>, <code>extractResult(Intent)</code>, <code>hasLocationAvailability(Intent)</code>, and <code>extractLocationAvailability(Intent)</code>.</p> <p data-bbox="533 784 667 808"><b>Parameters</b></p> <table border="1" data-bbox="533 833 1738 971"> <tr> <td data-bbox="533 833 835 906"><code>request</code></td> <td data-bbox="835 833 1738 906">The location request for the updates.</td> </tr> <tr> <td data-bbox="533 906 835 971"><code>callbackIntent</code></td> <td data-bbox="835 906 1738 971">A pending intent to be sent for each location update.</td> </tr> </table> <p data-bbox="533 995 630 1019"><b>Returns</b></p> <ul data-bbox="533 1044 1360 1068" style="list-style-type: none"> <li>• a Task for the call, check <code>isSuccessful()</code> to determine if it was successful.</li> </ul> <p data-bbox="533 1076 1898 1141"> <a href="https://developers.google.com/android/reference/com/google/android/gms/location/FusedLocationProviderClient">https://developers.google.com/android/reference/com/google/android/gms/location/FusedLocationProviderClient</a> </p>	<code>request</code>	The location request for the updates.	<code>callbackIntent</code>	A pending intent to be sent for each location update.
<code>request</code>	The location request for the updates.				
<code>callbackIntent</code>	A pending intent to be sent for each location update.				

## Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products

US9408055B2	HTC						
	<div style="background-color: #f0f0f0; padding: 5px; margin-bottom: 10px;"> <p><code>public void onLocationAvailability (LocationAvailability locationAvailability)</code></p> </div> <p>Called when there is a change in the availability of location data.</p> <p>When <code>isLocationAvailable()</code> returns <code>false</code> you can assume that location will not be returned in <code>onLocationResult(LocationResult)</code> until something changes in the device's settings or environment. Even when <code>isLocationAvailable()</code> returns <code>true</code> the <code>onLocationResult(LocationResult)</code> may not always be called regularly, however the device location is known and both the most recently delivered location and <code>getLastLocation(GoogleApiClient)</code> will be reasonably up to date given the hints specified by the active <code>LocationRequest</code>s.</p> <p><b>Parameters</b></p> <table border="1" style="width: 100%; border-collapse: collapse; margin-top: 10px;"> <tr> <td style="background-color: #444; color: white; padding: 5px;"><code>locationAvailability</code></td> <td style="padding: 5px;">The current status of location availability.</td> </tr> </table> <div style="background-color: #f0f0f0; padding: 5px; margin-top: 10px;"> <p><code>public void onLocationResult (LocationResult result)</code></p> </div> <p>Called when device location information is available.</p> <p>The most recent location returned by <code>getLastLocation()</code> is not guaranteed to be immediately fresh, but will be reasonably up to date given the hints specified by the active <code>LocationRequest</code>s.</p> <p><b>Parameters</b></p> <table border="1" style="width: 100%; border-collapse: collapse; margin-top: 10px;"> <tr> <td style="background-color: #444; color: white; padding: 5px;"><code>result</code></td> <td style="padding: 5px;">The latest location result available.</td> </tr> </table> <p><a href="https://developers.google.com/android/reference/com/google/android/gms/location/LocationCallback">https://developers.google.com/android/reference/com/google/android/gms/location/LocationCallback</a></p> <div style="background-color: #f0f0f0; padding: 5px; margin-top: 10px;"> <p><code>public abstract void onLocationChanged (Location location)</code></p> </div> <p>Called when the location has changed.</p> <p><b>Parameters</b></p> <table border="1" style="width: 100%; border-collapse: collapse; margin-top: 10px;"> <tr> <td style="background-color: #444; color: white; padding: 5px;"><code>location</code></td> <td style="padding: 5px;">The updated location.</td> </tr> </table>	<code>locationAvailability</code>	The current status of location availability.	<code>result</code>	The latest location result available.	<code>location</code>	The updated location.
<code>locationAvailability</code>	The current status of location availability.						
<code>result</code>	The latest location result available.						
<code>location</code>	The updated location.						

**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC		
	<p><a href="https://developers.google.com/android/reference/com/google/android/gms/location/LocationListener">https://developers.google.com/android/reference/com/google/android/gms/location/LocationListener</a></p> <p>Public Constructors</p> <hr/> <p>public <b>MapView</b> (<a href="#">Context</a> context)</p> <hr/> <p>public <b>MapView</b> (<a href="#">Context</a> context, <a href="#">AttributeSet</a> attrs)</p> <hr/> <p>public <b>MapView</b> (<a href="#">Context</a> context, <a href="#">AttributeSet</a> attrs, int defStyleAttr)</p> <hr/> <p>public <b>MapView</b> (<a href="#">Context</a> context, <a href="#">GoogleMapOptions</a> options)</p> <p><a href="https://developers.google.com/android/reference/com/google/android/gms/maps/MapView">https://developers.google.com/android/reference/com/google/android/gms/maps/MapView</a></p> <p>public void <b>getMapAsync</b> (<a href="#">OnMapReadyCallback</a> callback)</p> <p>Returns a non-null instance of the <a href="#">GoogleMap</a> , ready to be used.</p> <p>Note that:</p> <ul style="list-style-type: none"> <li>• This method must be called from the main thread.</li> <li>• The callback will be executed in the main thread.</li> <li>• In the case where Google Play services is not installed on the user's device, the callback will not be triggered until the user installs it.</li> <li>• The <a href="#">GoogleMap</a> object provided by the callback is non-null.</li> </ul> <p><b>Parameters</b></p> <table border="1"> <tr> <td data-bbox="527 1203 695 1268">callback</td> <td data-bbox="701 1203 1738 1268">The callback object that will be triggered when the map is ready to be used.</td> </tr> </table> <hr/> <p>public final void <b>onCreate</b> (<a href="#">Bundle</a> savedInstanceState)</p> <p>You must call this method from the parent Activity/Fragment's corresponding method.</p>	callback	The callback object that will be triggered when the map is ready to be used.
callback	The callback object that will be triggered when the map is ready to be used.		

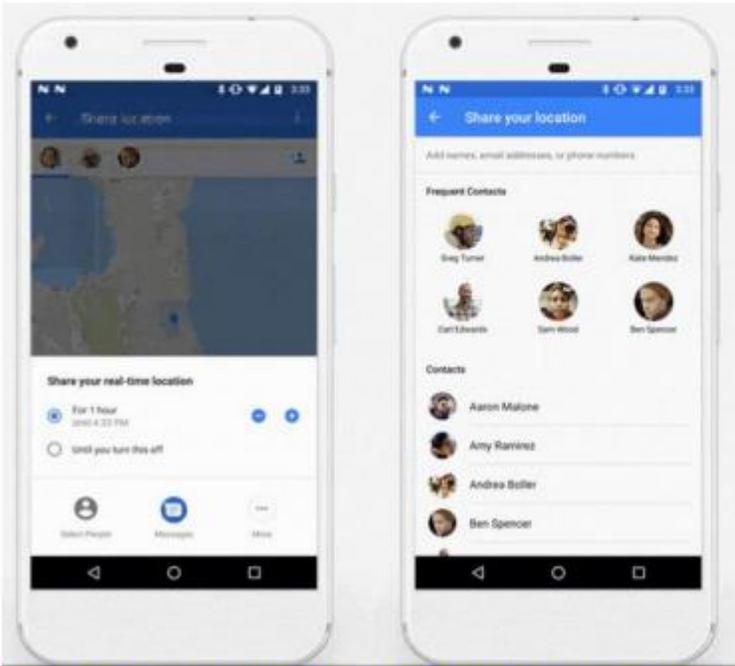
**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC
<p>[28D] transmitting IP-based messages including a location of the first device to the respective second devices;</p>	<p><a href="https://developers.google.com/android/reference/com/google/android/gms/maps/MapView">https://developers.google.com/android/reference/com/google/android/gms/maps/MapView</a></p> <p>HTC infringes directly and/or indirectly by performing, inducing others to perform, and/or contributing to the performance of transmitting IP-based messages including a location of the first device to the respective second devices. See claim 1[D], which is incorporated herein by reference in its entirety.</p> <p>For example, users send their location to a server and receive the location of other devices with whom the location is being shared. To send a location to the network, a user enables location service which enables the device to determine and send its location. If location service is already enabled, the device sends its location to the server as needed by the application (e.g. Google Maps). If location service is not enabled, the application will ask the user to enable location service in order to continue with full functionality, which includes using the device's location. Google Maps applications receive the location of other devices when those devices have location service enabled while using the same respective application. Android Device Manager and Google Maps use the received locations to display those locations on the map, indicating the locations of other devices.</p> <p>See, e.g., location sharing including corresponding code described above with regard to limitation [1C] and [28C].</p> <p>Using Google Maps, a user enables location services to send its location the network, but the user can also choose to share its location, as shown below. Again, each device that participates is able to see the location of the other device using Google Maps' share your location feature. For example:</p>

**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC
	<p data-bbox="527 233 1304 277"><b>HTC One V™ – Google Location Service &amp; GPS</b></p> <p data-bbox="527 321 1717 428">Google Maps lets you track your current location, view real-time traffic situations, and receive detailed directions to your destination. It also provides a search tool where you can locate a place of interest or an address on a vector or aerial map, or view locations in street level.</p> <p data-bbox="527 472 898 500"><b>Turning on Location Services</b></p> <div data-bbox="527 516 1717 954"> </div> <ol data-bbox="527 980 1738 1154" style="list-style-type: none"> <li>1. From the Home Screen, slide the <b>Notifications</b> panel open.</li> <li>2. In the top right corner, tap <b>Settings</b>.</li> <li>3. Tap <b>Location</b>.</li> <li>4. Make your selection by tapping <b>Google's location service</b>, <b>Use GPS satellites</b>, or both. <b>Note:</b> You will need to accept the location consent terms and conditions.</li> </ol>

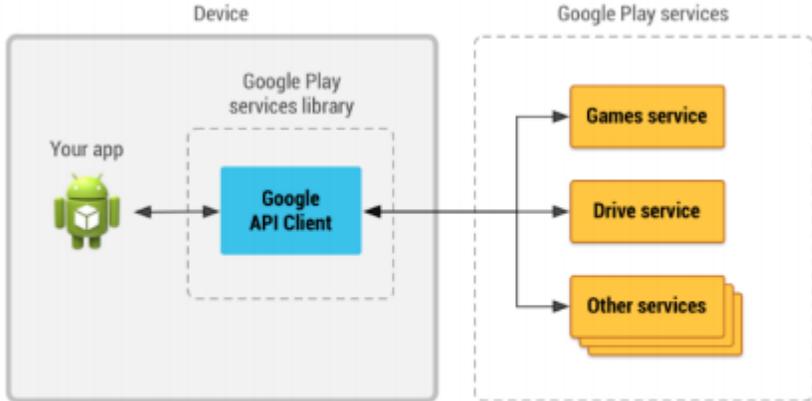
## Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products

US9408055B2	HTC
	<ol style="list-style-type: none"><li>1. If you haven't already, add their Gmail address to your <a href="#">Google Contacts</a>.</li><li>2. On your Android phone or tablet, open the Google Maps app  and sign in. <a href="#">Learn how to sign in.</a></li><li>3. Tap the Menu  &gt; <b>Share location</b> &gt; Add People .</li><li>4. Choose how long you want to share your location.</li><li>5. Tap <b>Select People</b>.<ul style="list-style-type: none"><li>• If you're asked about your contacts, give Google Maps access.</li></ul></li><li>6. Choose who you want to share with.</li><li>7. Tap <b>Share</b>.</li></ol> <p><a href="https://support.google.com/plus/answer/3302509?co=GENIE.Platform%3DAndroid&amp;hl=en">https://support.google.com/plus/answer/3302509?co=GENIE.Platform%3DAndroid&amp;hl=en</a></p>  <p><a href="https://arstechnica.com/gadgets/2017/03/location-sharing-finally-returns-to-google-maps/">https://arstechnica.com/gadgets/2017/03/location-sharing-finally-returns-to-google-maps/</a></p>

**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC
	 <p><a href="https://arstechnica.com/gadgets/2017/03/location-sharing-finally-returns-to-google-maps/">https://arstechnica.com/gadgets/2017/03/location-sharing-finally-returns-to-google-maps/</a></p> <p>Below are exemplary methods used by Google applications to obtain, send, and receive locations.</p>

**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC
	<p>The Google Play services Location API</p> <p>The Google Play services <a href="#">Location API</a> is the preferred method for adding location awareness to your Android application. It includes functionality that lets you:</p> <ul style="list-style-type: none"> <li>• Determine the device location.</li> <li>• Listen for location changes.</li> <li>• Determine the mode of transportation, if the device is moving.</li> <li>• Create and monitor predefined geographical regions, known as geofences.</li> </ul> <p>The location APIs make it easy for you to build power efficient, location-aware applications. Like the Google Maps Android API, the Location API is distributed as part of the Google Play services SDK. For more information on the Location API, please refer to the Android training class <a href="#">Making Your App Location Aware</a> or the <a href="#">Location API Reference</a>. Code examples are included as part of the Google Play services SDK.</p> <p><a href="https://developers.google.com/maps/documentation/android-api/location">https://developers.google.com/maps/documentation/android-api/location</a></p>  <p>The diagram illustrates the architecture of the Google API Client. On the left, a box labeled 'Device' contains 'Your app' (represented by an Android robot icon) and the 'Google Play services library'. Inside the library is the 'Google API Client'. On the right, a dashed box labeled 'Google Play services' contains three service boxes: 'Games service', 'Drive service', and 'Other services'. Bidirectional arrows connect 'Your app' to the 'Google API Client', and the 'Google API Client' to each of the three service boxes.</p> <p>Figure 1: An illustration showing how the Google API Client provides an interface for connecting and making calls to any of the available Google Play services such as Google Play Games and Google Drive.</p> <p><a href="https://developers.google.com/android/guides/api-client#Starting">https://developers.google.com/android/guides/api-client#Starting</a></p>



**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC
	<p data-bbox="527 233 919 261"><b>Get the Last Known Location</b></p> <p data-bbox="527 289 1545 386">Once you have connected to Google Play services and the location services API, you can get the last known location of a user's device. When your app is connected to these you can use the fused location provider's <code>getLastLocation()</code> method to retrieve the device location. The precision of the location returned by this call is determined by the permission setting you put in your app manifest, as described in the <a href="#">Specify App Permissions</a> section of this document.</p> <p data-bbox="527 407 1545 480">To request the last known location, call the <code>getLastLocation()</code> method, passing it your instance of the <code>GoogleApiClient</code> object. Do this in the <code>onConnected()</code> callback provided by Google API Client, which is called when the client is ready. The following code snippet illustrates the request and a simple handling of the response:</p> <pre data-bbox="527 493 1545 781"> public class MainActivity extends ActionBarActivity implements     ConnectionCallbacks, OnConnectionFailedListener {     ...     @Override     public void onConnected(Bundle connectionHint) {         mLastLocation = LocationServices.FusedLocationApi.getLastLocation(             mGoogleApiClient);         if (mLastLocation != null) {             mLatitudeText.setText(String.valueOf(mLastLocation.getLatitude()));             mLongitudeText.setText(String.valueOf(mLastLocation.getLongitude()));         }     } } </pre> <p data-bbox="527 797 1545 846">The <code>getLastLocation()</code> method returns a <code>Location</code> object from which you can retrieve the latitude and longitude coordinates of a geographic location. The location object returned may be null in rare cases when the location is not available.</p> <p data-bbox="527 850 1289 878"><a href="https://developer.android.com/training/location/retrieve-current.html">https://developer.android.com/training/location/retrieve-current.html</a></p>

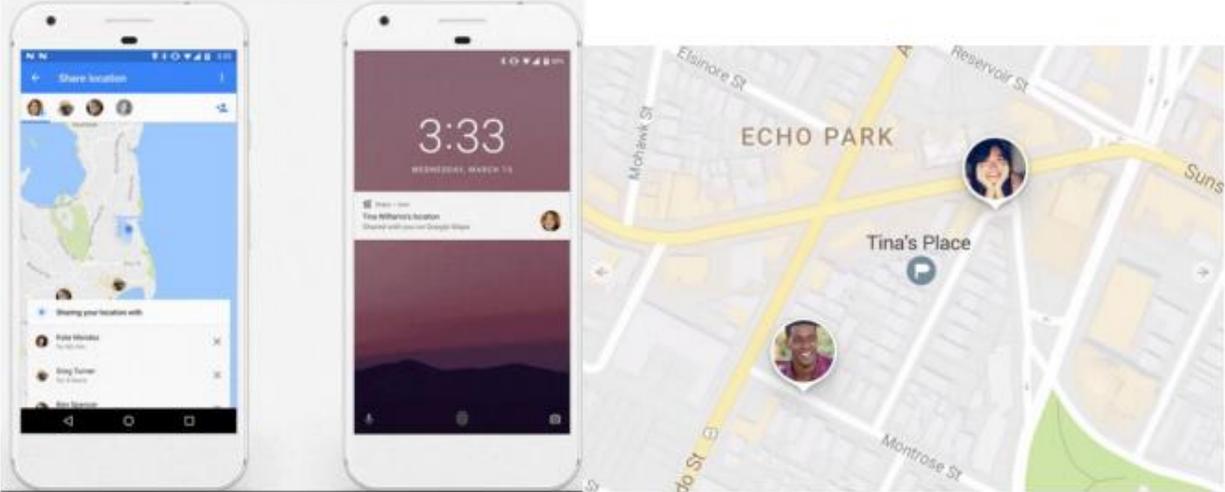
**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC
	<p>Determining the user's current location</p> <hr/> <p>The Geolocation API offers a simple, "one-shot" method to obtain the user's location: <code>getCurrentPosition()</code> . A call to this method asynchronously reports on the user's current location.</p> <pre> window.onload = function() {   var startPos;   var geoSuccess = function(position) {     startPos = position;     document.getElementById('startLat').innerHTML = startPos.coords.latitude;     document.getElementById('startLon').innerHTML = startPos.coords.longitude;   };   navigator.geolocation.getCurrentPosition(geoSuccess); }; </pre> <p>If this is the first time that an application on this domain has requested permissions, the browser typically checks for user consent. Depending on the browser, there may also be preferences to always allow—or disallow—permission lookups, in which case the confirmation process is bypassed.</p> <p>Depending on the location device your browser is using, the position object might actually contain a lot more than just latitude and longitude; for example, it might include an altitude or a direction. You can't tell what extra information that location system uses until it actually returns the data.</p> <p><a href="https://developers.google.com/web/fundamentals/native-hardware/user-location/">https://developers.google.com/web/fundamentals/native-hardware/user-location/</a></p>
<p>[28E] presenting, via an interactive display of the first device, an interactive map and a plurality of user selectable symbols corresponding to the plurality of second devices, wherein the symbols are positioned on the map at respective positions</p>	<p>HTC infringes directly and/or indirectly by performing, inducing others to perform, and/or contributing to the performance of presenting, via an interactive display of the first device, an interactive map and a plurality of user selectable symbols corresponding to the plurality of second devices, wherein the symbols are positioned on the map at respective positions corresponding to the respective locations of the second devices. See claim 1[E], which is incorporated herein by reference in its entirety.</p> <p>For example, the Accused Products use Android Device Manager, and Google Maps to display an interface with a map and symbols representing devices.</p> <p>Using Android Device Manager, the user is presented with a map that appears to be based on or imported from Google Maps. The map is interactive because the user can pan, zoom, and make selections, among other interactivities. Depending on the number of devices linked to the Google Account, Android Device</p>

**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC
<p>corresponding to the respective locations of the second devices;</p>	<p>Manager places symbols on the map and in a list, where those symbols represent the linked devices and their positions. The symbols are selectable by touching the display. The map is georeferenced, at least because its contents, including the symbols, are placed and spaced according to geographical coordinates.</p> <p><b><u>Regarding Google Maps</u></b>, Google Latitude, Google Plus, Google Hangouts, Google Messages, Google Allo, Google Duo, Google Chrome, and Android Messenger, the Accused Products display, to the user on the display of the first device, a map with one or more symbols corresponding to one or more second users (or second devices corresponding to the second users). The map is interactive because the user may control the display of the map, e.g., pan, zoom, and/or effect change to the map in an otherwise interactive manner. The map is georeferenced for at least the reason that one or more symbols are associated with spatial locations, i.e., coordinates. The symbols are user-selectable because a user may touch the display to select the user or device associated with the symbol. For example, the user may input a touch selection directly on or near the portion of the display corresponding to the symbol’s coordinates on the map to effect a selection of the user or device.</p> <p><b><u>Exemplary Support for Google Maps:</u></b>                      Using Google Maps and its location sharing feature, the user is presented with a map that is interactive because the user can pan, zoom, and make selections, among other interactivities. Depending on how many other devices or Google Accounts are sharing their locations, Google Maps places symbols on the map and in a list, where those symbols represent the linked devices and their positions. The symbols are selectable by touching the display. The map is georeferenced, at least because its contents, including the symbols, are placed and spaced according to geographical coordinates.</p>

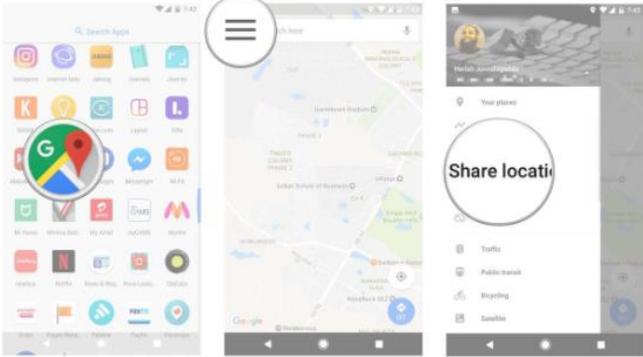
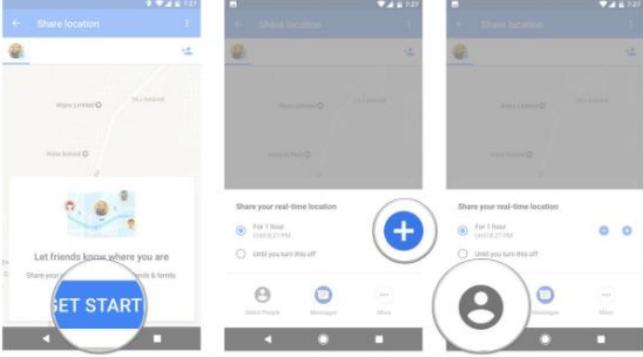
## Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products

US9408055B2	HTC
	 <p><a href="https://arstechnica.com/gadgets/2017/03/location-sharing-finally-returns-to-google-maps/">https://arstechnica.com/gadgets/2017/03/location-sharing-finally-returns-to-google-maps/</a></p> <h3>See where someone is</h3> <p>If someone shares their location with you, you can see them on the map.</p> <ol style="list-style-type: none"><li>1. Open the Google Maps app</li><li>2. Tap the Menu ≡ &gt; <b>Share location</b>.</li><li>3. Choose someone.</li></ol> <p>• To see an updated location, tap on a friend's icon &gt; More ≡ &gt; <b>Refresh</b>.</p> <h3>Stop seeing someone's location</h3> <ol style="list-style-type: none"><li>1. Open the Google Maps app</li><li>2. On the map, tap their icon.</li><li>3. At the bottom, tap More ≡</li><li>4. To temporarily hide someone, tap <b>Hide from map</b>. You can stop hiding them at any time.</li></ol> <p><b>Note:</b> You can stop someone's location from ever appearing on your map. <a href="#">Learn how to block another person's account.</a></p> <p><a href="https://support.google.com/maps/answer/7326816?hl=en&amp;ref_topic=3092425&amp;co=GENIE.Platform%3DAndroid&amp;oc=1">https://support.google.com/maps/answer/7326816?hl=en&amp;ref_topic=3092425&amp;co=GENIE.Platform%3DAndroid&amp;oc=1</a></p>

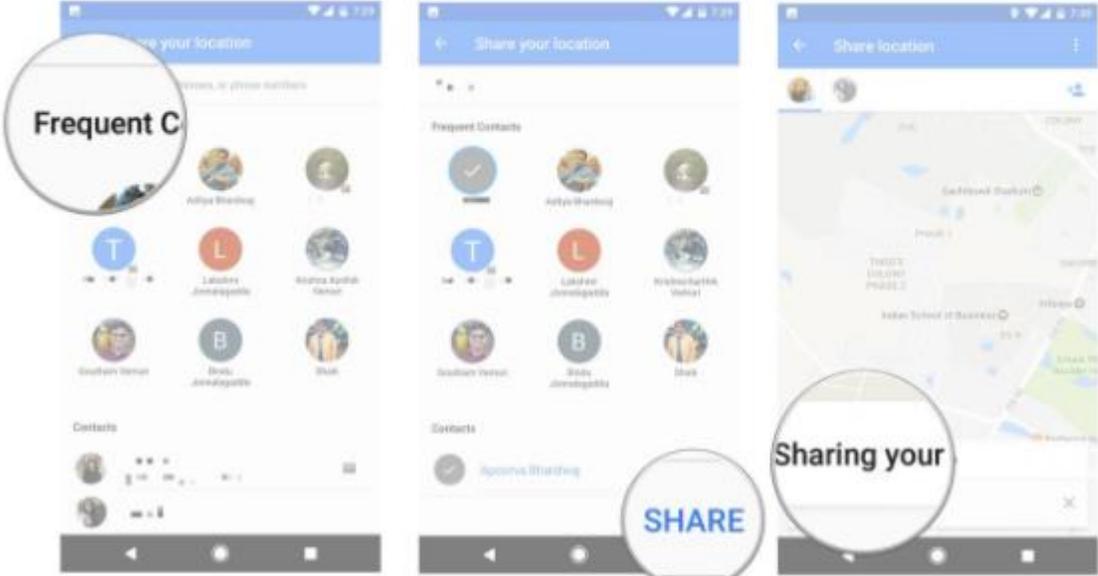
## Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products

US9408055B2	HTC
	<h3>Share your E.T.A</h3> <p>After you start your drive, you can share your destination, estimated arrival time, and where you are on the route.</p> <ol style="list-style-type: none"><li>1. Open the Google Maps app .</li><li>2. Set a driving destination. <a href="#">Learn how to navigate to a place.</a></li><li>3. After you start navigation, tap More ^ &gt; <b>Share trip progress.</b></li><li>4. Choose a person from the list.</li><li>5. Tap <b>Share.</b></li><li>6. Location Sharing will stop when you reach your destination or stop navigating.</li></ol> <ul style="list-style-type: none"><li>• To stop sharing before you arrive, tap More ^ &gt; <b>Stop sharing.</b></li></ul> <h3>See where someone is</h3> <p>If someone shares their location with you, you can see them on the map.</p> <ol style="list-style-type: none"><li>1. Open the Google Maps app .</li><li>2. Tap Menu ≡ &gt; <b>Location sharing.</b></li><li>3. Choose someone.</li></ol> <ul style="list-style-type: none"><li>• To see an updated location, tap on a friend's icon &gt; More ⋮ &gt; <b>Refresh.</b></li></ul> <h3>Stop seeing someone's location</h3> <ol style="list-style-type: none"><li>1. Open the Google Maps app .</li><li>2. On the map, tap their icon.</li><li>3. At the bottom, tap More ^ .</li><li>4. To temporarily hide someone, tap <b>Hide from map.</b> You can stop hiding them at any time.</li></ol> <p><b>Note:</b> You can stop someone's location from ever appearing on your map. Learn how to <a href="#">block another person's account.</a></p> <p><a href="https://support.google.com/maps/answer/7326816?co=GENIE.Platform%3DAndroid&amp;oco=1">https://support.google.com/maps/answer/7326816?co=GENIE.Platform%3DAndroid&amp;oco=1</a></p>

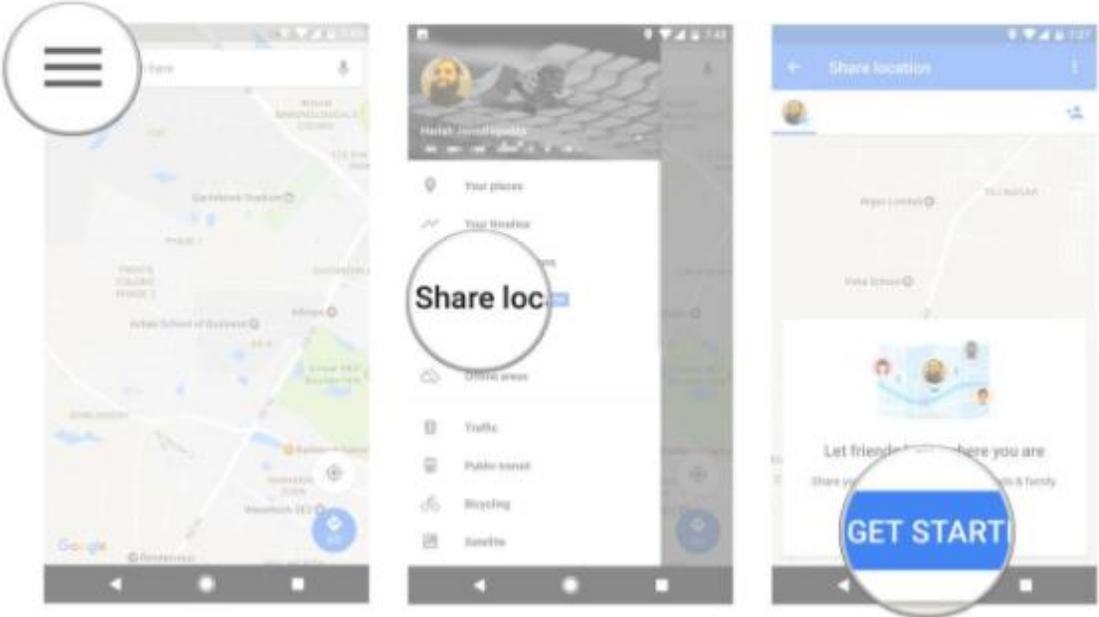
## Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products

US9408055B2	HTC
	<p data-bbox="520 240 1129 272"><b>How to share your location in Google Maps</b></p> <ol data-bbox="520 298 1129 386" style="list-style-type: none"><li>1. Open Google Maps from the app drawer or the home screen.</li><li>2. Tap the <b>hamburger menu</b> (the three horizontal lines) on the top left corner of the screen.</li><li>3. Select <b>Share location</b>.</li></ol>  <p data-bbox="520 808 1129 906"><b>4. Tap Get Started.</b> <b>5. Use the + icon to select a time period or select the <i>Until you turn this off</i> setting to share your location indefinitely.</b> <b>6. Tap Select People.</b></p>  <p data-bbox="520 1299 1360 1328"><a href="https://www.androidcentral.com/how-share-location-google-maps">https://www.androidcentral.com/how-share-location-google-maps</a></p>

## Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products

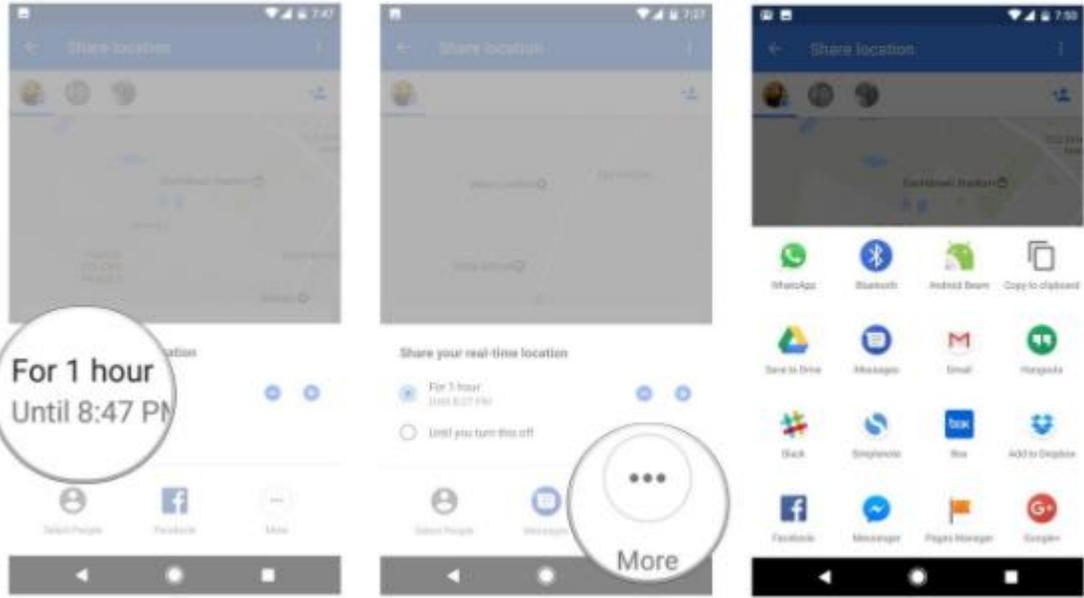
US9408055B2	HTC
	<p data-bbox="527 253 1577 310">7. You'll see a list of your frequent contacts at the top, along with a full list of contacts. Pick the contacts by tapping their name.</p> <p data-bbox="527 339 1457 367">8. Once you've selected the contacts you want to share your location to, tap <b>Share</b>.</p> <p data-bbox="527 396 1419 423">9. You'll see a message saying that the selected contact can view your location.</p>  <p data-bbox="506 1101 1356 1133"><a href="https://www.androidcentral.com/how-share-location-google-maps">https://www.androidcentral.com/how-share-location-google-maps</a></p>

## Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products

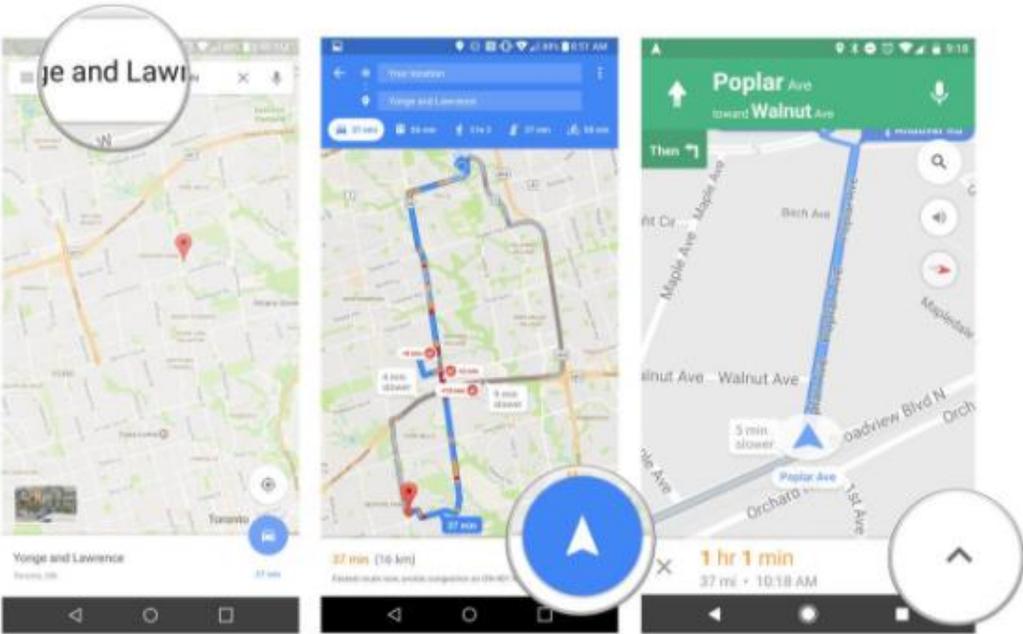
US9408055B2	HTC
	<h3 data-bbox="520 245 1255 289">How to create a shareable link</h3> <p data-bbox="520 331 1461 358">You can also create a link and use it to share your location easily. Here's how to do it:</p> <ol data-bbox="520 410 1234 548" style="list-style-type: none"><li data-bbox="520 410 1234 438">1. Tap the hamburger menu on the top left corner of the screen.</li><li data-bbox="520 467 800 495">2. Select Share location.</li><li data-bbox="520 524 737 552">3. Tap Get Started.</li></ol>  <p data-bbox="506 1230 1356 1263"><a href="https://www.androidcentral.com/how-share-location-google-maps">https://www.androidcentral.com/how-share-location-google-maps</a></p>



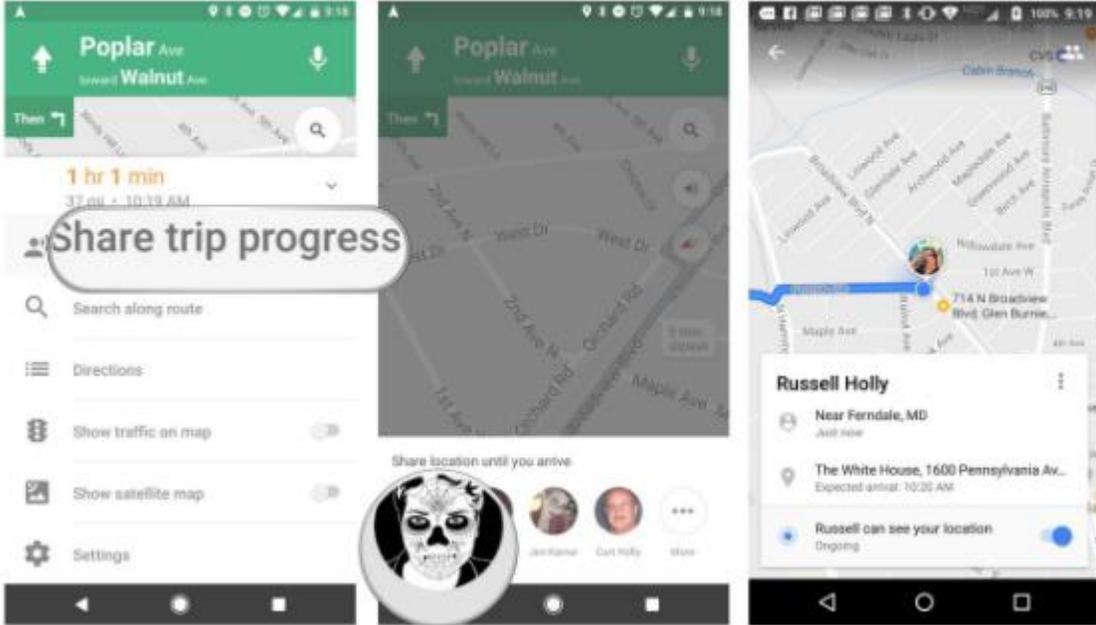
### Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products

US9408055B2	HTC
	<p data-bbox="520 245 1213 272">4. Select the <b>amount of time</b> you want to share your location.</p> <p data-bbox="520 302 680 329">5. Tap <b>More</b>.</p> <p data-bbox="520 358 1633 418">6. Select <b>your app of choice</b> to create and send a <b>unique URL</b> that broadcasts your current location. You can email it, send the link via Messenger, or even tweet it to the intended recipient.</p>  <p data-bbox="506 1081 1358 1117"><a href="https://www.androidcentral.com/how-share-location-google-maps">https://www.androidcentral.com/how-share-location-google-maps</a></p>

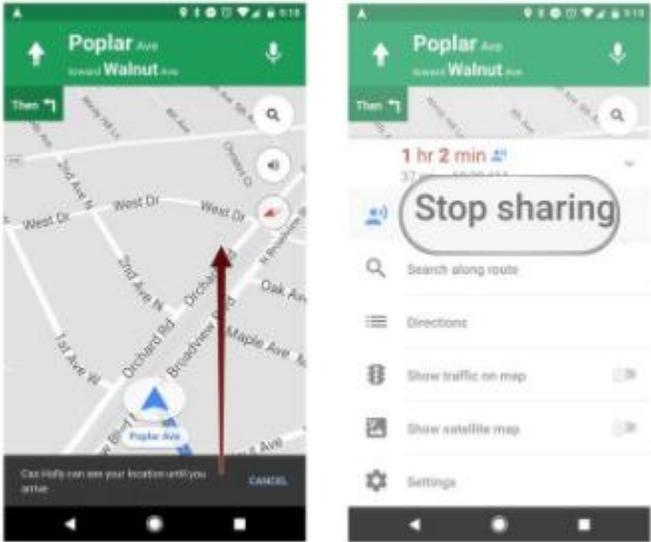
**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC
	<p data-bbox="527 240 1428 337"><b>How to share your navigation directions while you walk, drive or transit</b></p> <p data-bbox="527 375 1556 464">One of the best ways to share your location with someone is during a drive, walk or transit. If you're meeting a friend or family member somewhere, or navigating towards their home, Google Maps lets you share your location with them for the duration of the trip. It's magic!</p> <ol data-bbox="527 513 1394 643" style="list-style-type: none"> <li>1. In the <b>search bar</b> enter your destination.</li> <li>2. Pick your navigation type (drive, transit, walk) and press the blue <b>navigate</b> button.</li> <li>3. Tap the arrow next to the <b>time-to-destination</b> number at the bottom of the screen.</li> </ol>  <p data-bbox="512 1328 1356 1359"><a href="https://www.androidcentral.com/how-share-location-google-maps">https://www.androidcentral.com/how-share-location-google-maps</a></p>

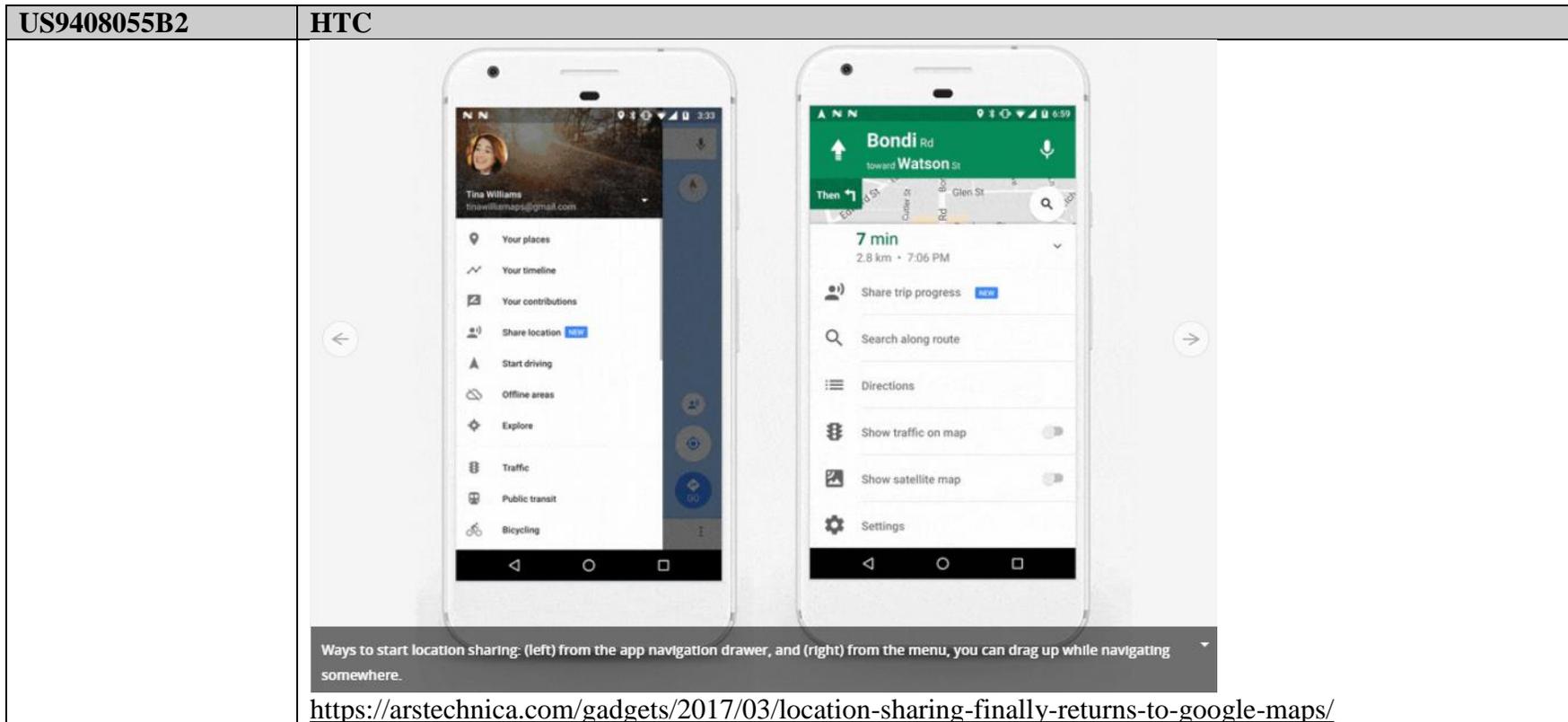
### Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products

US9408055B2	HTC
	<p data-bbox="527 240 835 266">4. Tap Share trip progress.</p> <p data-bbox="527 297 1150 323">5. Choose one or more contacts to share trip progress.</p>  <p data-bbox="527 1027 1360 1097">You can also stop sharing your location with someone before a trip ends. <a href="https://www.androidcentral.com/how-share-location-google-maps">https://www.androidcentral.com/how-share-location-google-maps</a></p>

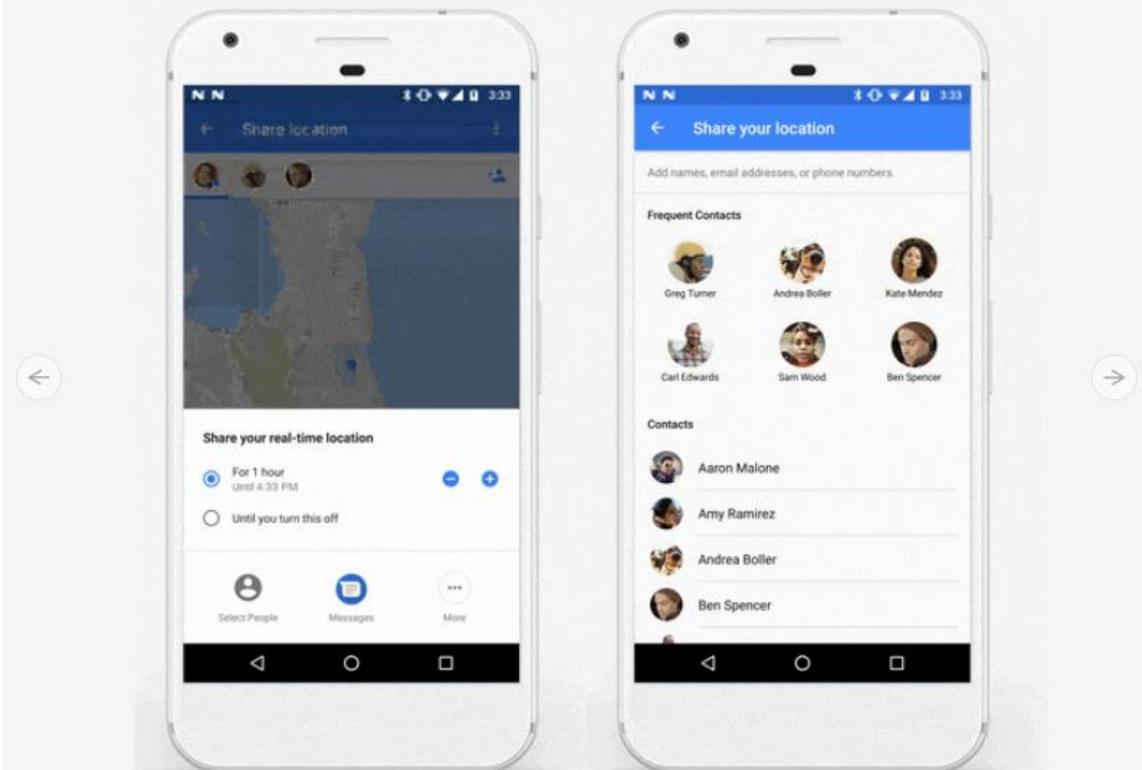
## Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products

US9408055B2	HTC
	<ol style="list-style-type: none"><li data-bbox="533 245 1472 272">1. Tap the arrow next to the <b>time-to-destination</b> number at the bottom of the screen.</li><li data-bbox="533 302 768 329">2. Tap Stop sharing.</li></ol> <div data-bbox="747 380 1398 922"></div> <p data-bbox="541 976 638 1003">That's it!</p> <p data-bbox="541 1045 1612 1073">Are you excited that location sharing is back in Google Maps? How often do you use the feature?</p> <p data-bbox="512 1084 1356 1112"><a href="https://www.androidcentral.com/how-share-location-google-maps">https://www.androidcentral.com/how-share-location-google-maps</a></p>

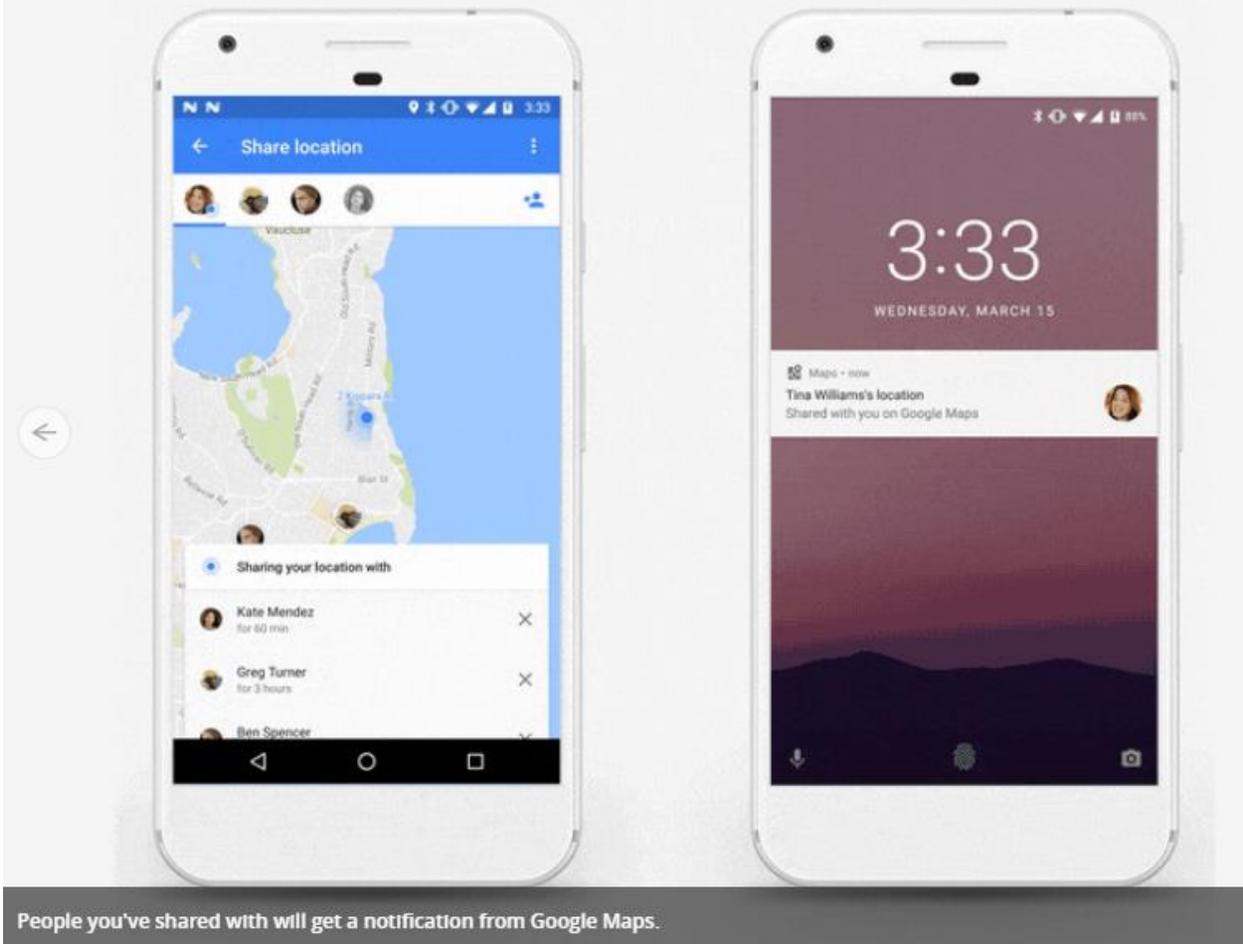
## Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products



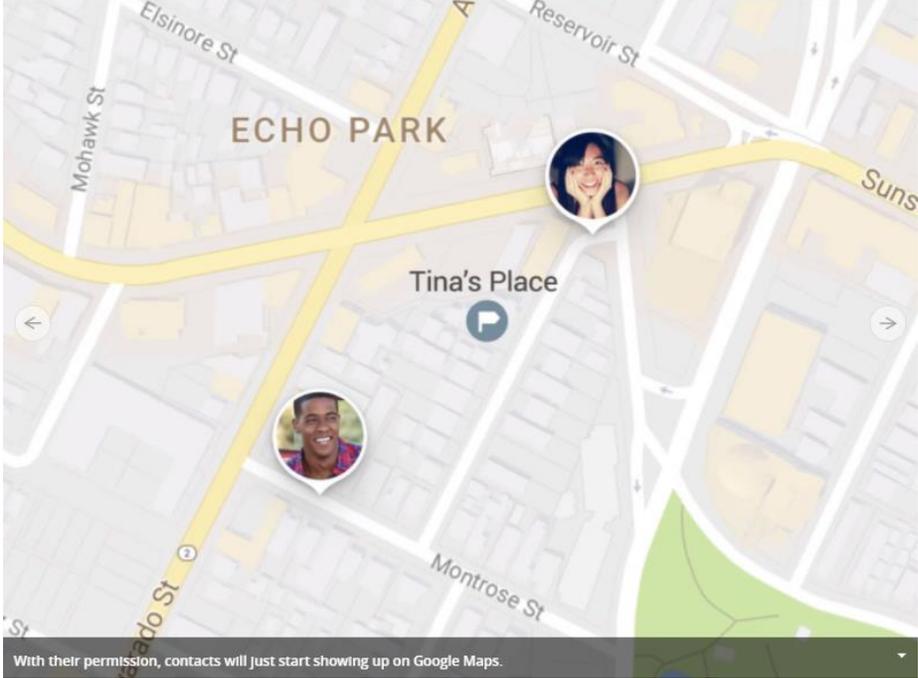
### Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products

US9408055B2	HTC
	 <p data-bbox="520 1027 1646 1052">You can pick how long you want to share your location for, and with whom you want to share it.</p> <p data-bbox="512 1065 1654 1092"><a href="https://arstechnica.com/gadgets/2017/03/location-sharing-finally-returns-to-google-maps/">https://arstechnica.com/gadgets/2017/03/location-sharing-finally-returns-to-google-maps/</a></p>

**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

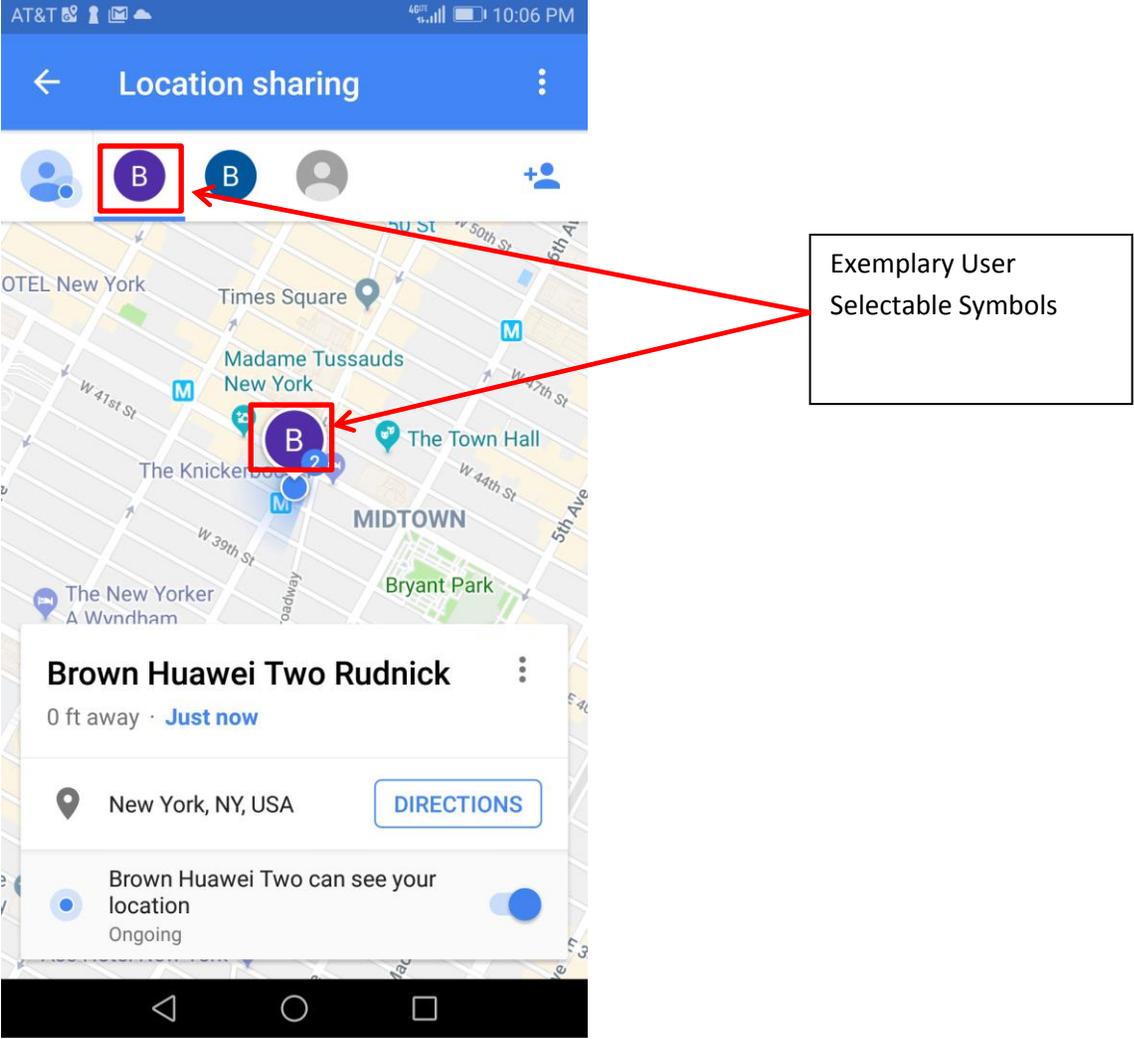
US9408055B2	HTC
	 <p data-bbox="512 1143 1171 1170">People you've shared with will get a notification from Google Maps.</p> <p data-bbox="512 1187 1656 1219"><a href="https://arstechnica.com/gadgets/2017/03/location-sharing-finally-returns-to-google-maps/">https://arstechnica.com/gadgets/2017/03/location-sharing-finally-returns-to-google-maps/</a></p>

**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC
	 <p data-bbox="512 883 1016 906">With their permission, contacts will just start showing up on Google Maps.</p> <p data-bbox="512 915 1656 946"><a href="https://arstechnica.com/gadgets/2017/03/location-sharing-finally-returns-to-google-maps/">https://arstechnica.com/gadgets/2017/03/location-sharing-finally-returns-to-google-maps/</a></p> <p data-bbox="512 987 1031 1018"><b><u>Exemplary Google Maps Screenshots:</u></b></p>



### Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products

US9408055B2	HTC
	 <p>The screenshot shows an Android phone's 'Location sharing' screen. At the top, there's a blue header with a back arrow, the text 'Location sharing', and a menu icon. Below the header, there are three user icons: a blue circle with a white 'B', a grey circle with a white 'B', and a grey person icon. The first 'B' icon is highlighted with a red box. Below the icons is a map of Midtown Manhattan, New York. A second 'B' icon is also highlighted with a red box on the map. A callout box with a black border and white background contains the text 'Exemplary User Selectable Symbols'. Two red arrows point from this box to the two 'B' icons. Below the map, there's a card for 'Brown Huawei Two Rudnick' showing '0 ft away · Just now', the location 'New York, NY, USA', a 'DIRECTIONS' button, and a toggle switch for 'Brown Huawei Two can see your location' which is currently turned on.</p> <p><b>Exemplary Source Code:</b> The above functionality is performed at least in part by the following publicly available source code and/or</p>

**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC
	<p>source code that invokes or is invoked by the following source code (or a substantially similar copy compiled and loaded onto the Accused Products by HTC). AGIS reserves the right to supplement these contentions with additional source code as discovery progresses and as additional source code is made available.</p> <pre data-bbox="533 428 1738 472">public static LocationRequest create ()</pre> <p>Create a location request with default parameters.</p> <p>Default parameters are for a block accuracy, slowly updated location. It can then be adjusted as required by the applications before passing to the <code>FusedLocationProviderApi</code>.</p> <p><b>Returns</b></p> <ul data-bbox="554 683 814 711" style="list-style-type: none"> <li>• a new location request</li> </ul> <p><a data-bbox="512 724 1793 751" href="https://developers.google.com/android/reference/com/google/android/gms/location/LocationRequest">https://developers.google.com/android/reference/com/google/android/gms/location/LocationRequest</a></p>

**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC
	<p><code>public static final int PRIORITY_BALANCED_POWER_ACCURACY</code></p> <p>Used with <code>setPriority(int)</code> to request "block" level accuracy.</p> <p>Block level accuracy is considered to be about 100 meter accuracy. Using a coarse accuracy such as this often consumes less power.</p> <p>Constant Value: 102</p> <hr/> <p><code>public static final int PRIORITY_HIGH_ACCURACY</code></p> <p>Used with <code>setPriority(int)</code> to request the most accurate locations available.</p> <p>This will return the finest location available.</p> <p>Constant Value: 100</p> <hr/> <p><code>public static final int PRIORITY_LOW_POWER</code></p> <p>Used with <code>setPriority(int)</code> to request "city" level accuracy.</p> <p>City level accuracy is considered to be about 10km accuracy. Using a coarse accuracy such as this often consumes less power.</p> <p>Constant Value: 104</p> <p><a href="https://developers.google.com/android/reference/com/google/android/gms/location/LocationRequest">https://developers.google.com/android/reference/com/google/android/gms/location/LocationRequest</a></p>

## Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products

US9408055B2	HTC
	<pre>public Task&lt;Location&gt; getLastLocation ()</pre> <p>Returns the best most recent location currently available.</p> <p>If a location is not available, which should happen very rarely, null will be returned. The best accuracy available while respecting the location permissions will be returned.</p> <p>This method provides a simplified way to get location. It is particularly well suited for applications that do not require an accurate location and that do not want to maintain extra logic for location updates.</p> <pre>public Task&lt;LocationAvailability&gt; getLocationAvailability ()</pre> <p>Returns the availability of location data. When <code>isLocationAvailable()</code> returns true, then the location returned by <code>getLastLocation()</code> will be reasonably up to date within the hints specified by the active <code>LocationRequest</code>s.</p> <p>If the client isn't connected to Google Play services and the request times out, null is returned.</p> <p>Note it's always possible for <code>getLastLocation()</code> to return null even when this method returns true (e.g. location settings were disabled between calls).</p> <p><a href="https://developers.google.com/android/reference/com/google/android/gms/location/FusedLocationProviderClient">https://developers.google.com/android/reference/com/google/android/gms/location/FusedLocationProviderClient</a></p>

**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC						
	<p data-bbox="527 245 1749 326"> <code>public Task&lt;Void&gt; requestLocationUpdates (LocationRequest request, LocationCallback callback, Looper looper)</code> </p> <p data-bbox="527 354 1272 378">Requests location updates with a callback on the specified Looper thread.</p> <p data-bbox="527 410 1686 472">This method is suited for the foreground use cases. For background use cases, the <code>PendingIntent</code> version of the method is recommended, see <code>requestLocationUpdates(LocationRequest, PendingIntent)</code>.</p> <p data-bbox="527 505 1371 529">Any previous LocationRequests registered on this LocationListener will be replaced.</p> <p data-bbox="527 561 1686 656">This call will keep the Google Play services connection active, so make sure to call <code>removeLocationUpdates(LocationCallback)</code> when you no longer need it, otherwise you lose the benefits of the automatic connection management.</p> <p data-bbox="527 688 1745 712">Callbacks for <code>LocationCallback</code> will be made on the specified thread, which must already be a prepared looper thread.</p> <p data-bbox="527 737 667 761"><b>Parameters</b></p> <table border="1" data-bbox="527 789 1749 1008"> <tbody> <tr> <td data-bbox="527 797 625 854"><b>request</b></td> <td data-bbox="632 797 1749 854">The location request for the updates.</td> </tr> <tr> <td data-bbox="527 862 625 919"><b>callback</b></td> <td data-bbox="632 862 1749 919">The callback for the location updates.</td> </tr> <tr> <td data-bbox="527 927 625 1000"><b>looper</b></td> <td data-bbox="632 927 1749 1000">The Looper object whose message queue will be used to implement the callback mechanism, or null to make callbacks on the calling thread.</td> </tr> </tbody> </table> <p data-bbox="527 1024 1902 1089"> <a href="https://developers.google.com/android/reference/com/google/android/gms/location/FusedLocationProviderClient">https://developers.google.com/android/reference/com/google/android/gms/location/FusedLocationProviderClient</a> </p>	<b>request</b>	The location request for the updates.	<b>callback</b>	The callback for the location updates.	<b>looper</b>	The Looper object whose message queue will be used to implement the callback mechanism, or null to make callbacks on the calling thread.
<b>request</b>	The location request for the updates.						
<b>callback</b>	The callback for the location updates.						
<b>looper</b>	The Looper object whose message queue will be used to implement the callback mechanism, or null to make callbacks on the calling thread.						

## Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products

US9408055B2	HTC				
	<pre>public Task&lt;Void&gt; requestLocationUpdates (LocationRequest request, PendingIntent callbackIntent)</pre> <p>Requests location updates with a callback on the specified PendingIntent.</p> <p>This method is suited for the background use cases, more specifically for receiving location updates, even when the app has been killed by the system. In order to do so, use a <code>PendingIntent</code> for a started service. For foreground use cases, the <code>LocationCallback</code> version of the method is recommended, see <code>requestLocationUpdates(LocationRequest, LocationCallback, Looper)</code>.</p> <p>Any previously registered requests that have the same PendingIntent (as defined by <code>equals(Object)</code>) will be replaced by this request.</p> <p>Both <code>LocationResult</code> and <code>LocationAvailability</code> are sent to the given PendingIntent. You can extract data from an Intent using <code>hasResult(Intent)</code>, <code>extractResult(Intent)</code>, <code>hasLocationAvailability(Intent)</code>, and <code>extractLocationAvailability(Intent)</code>.</p> <p><b>Parameters</b></p> <table border="1"> <tr> <td><code>request</code></td> <td>The location request for the updates.</td> </tr> <tr> <td><code>callbackIntent</code></td> <td>A pending intent to be sent for each location update.</td> </tr> </table> <p><b>Returns</b></p> <ul style="list-style-type: none"> <li>a Task for the call, check <code>isSuccessful()</code> to determine if it was successful.</li> </ul> <p><a href="https://developers.google.com/android/reference/com/google/android/gms/location/FusedLocationProviderClient">https://developers.google.com/android/reference/com/google/android/gms/location/FusedLocationProviderClient</a></p>	<code>request</code>	The location request for the updates.	<code>callbackIntent</code>	A pending intent to be sent for each location update.
<code>request</code>	The location request for the updates.				
<code>callbackIntent</code>	A pending intent to be sent for each location update.				

**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC						
	<p><code>public void onLocationAvailability (LocationAvailability locationAvailability)</code></p> <p>Called when there is a change in the availability of location data.</p> <p>When <code>isLocationAvailable()</code> returns <code>false</code> you can assume that location will not be returned in <code>onLocationResult(LocationResult)</code> until something changes in the device's settings or environment. Even when <code>isLocationAvailable()</code> returns <code>true</code> the <code>onLocationResult(LocationResult)</code> may not always be called regularly, however the device location is known and both the most recently delivered location and <code>getLastLocation(GoogleApiClient)</code> will be reasonably up to date given the hints specified by the active <code>LocationRequest</code>s.</p> <p><b>Parameters</b></p> <table border="1" data-bbox="527 639 1738 704"> <tr> <td><code>locationAvailability</code></td> <td>The current status of location availability.</td> </tr> </table> <p><code>public void onLocationResult (LocationResult result)</code></p> <p>Called when device location information is available.</p> <p>The most recent location returned by <code>getLastLocation()</code> is not guaranteed to be immediately fresh, but will be reasonably up to date given the hints specified by the active <code>LocationRequest</code>s.</p> <p><b>Parameters</b></p> <table border="1" data-bbox="527 1019 1738 1084"> <tr> <td><code>result</code></td> <td>The latest location result available.</td> </tr> </table> <p><a href="https://developers.google.com/android/reference/com/google/android/gms/location/LocationCallback">https://developers.google.com/android/reference/com/google/android/gms/location/LocationCallback</a></p> <p><code>public abstract void onLocationChanged (Location location)</code></p> <p>Called when the location has changed.</p> <p><b>Parameters</b></p> <table border="1" data-bbox="527 1318 1738 1383"> <tr> <td><code>location</code></td> <td>The updated location.</td> </tr> </table>	<code>locationAvailability</code>	The current status of location availability.	<code>result</code>	The latest location result available.	<code>location</code>	The updated location.
<code>locationAvailability</code>	The current status of location availability.						
<code>result</code>	The latest location result available.						
<code>location</code>	The updated location.						

**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

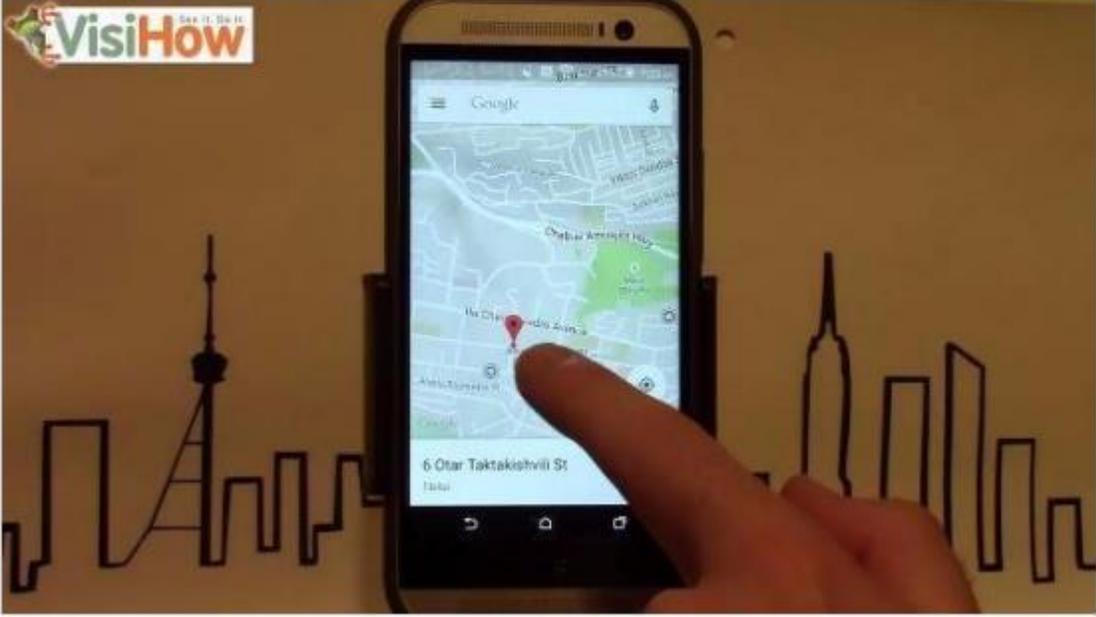
US9408055B2	HTC
	<p data-bbox="512 233 1797 264"><a href="https://developers.google.com/android/reference/com/google/android/gms/location/LocationListener">https://developers.google.com/android/reference/com/google/android/gms/location/LocationListener</a></p> <p data-bbox="512 318 814 349">Public Constructors</p> <hr data-bbox="512 358 1743 362"/> <p data-bbox="512 412 1743 451">public <b>MapView</b> (<a href="#">Context</a> context)</p> <p data-bbox="512 505 1743 544">public <b>MapView</b> (<a href="#">Context</a> context, <a href="#">AttributeSet</a> attrs)</p> <p data-bbox="512 597 1743 636">public <b>MapView</b> (<a href="#">Context</a> context, <a href="#">AttributeSet</a> attrs, int defStyle)</p> <p data-bbox="512 690 1743 729">public <b>MapView</b> (<a href="#">Context</a> context, <a href="#">GoogleMapOptions</a> options)</p> <p data-bbox="512 750 1671 781"><a href="https://developers.google.com/android/reference/com/google/android/gms/maps/MapView">https://developers.google.com/android/reference/com/google/android/gms/maps/MapView</a></p>



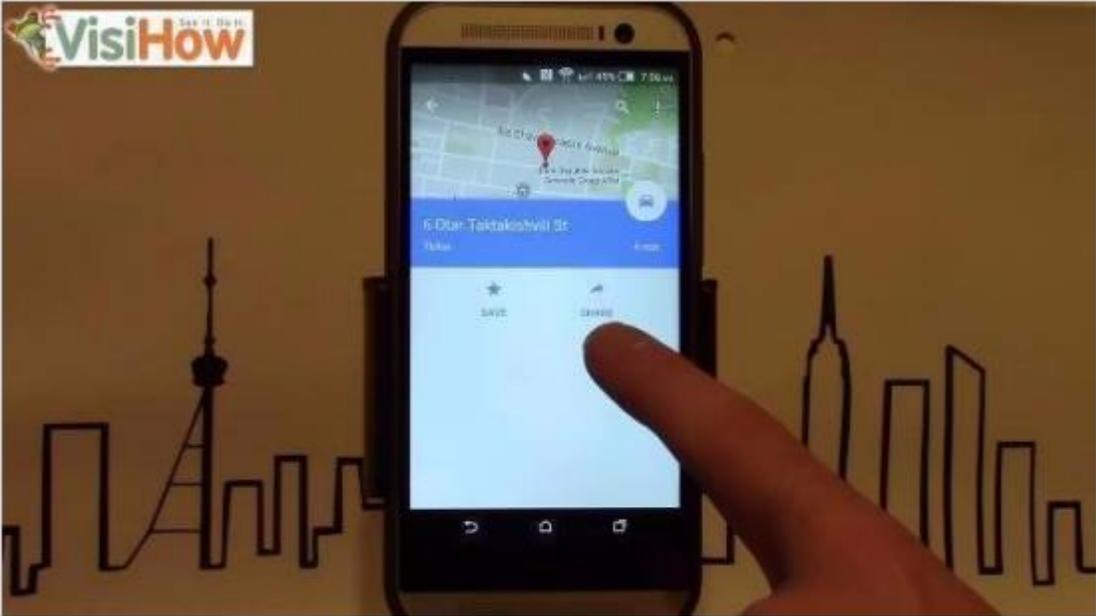
**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC		
	<p><code>public void <b>getMapAsync</b> (<a href="#">OnMapReadyCallback</a> callback)</code></p> <p>Returns a non-null instance of the <a href="#">GoogleMap</a>, ready to be used.</p> <p>Note that:</p> <ul style="list-style-type: none"> <li>• This method must be called from the main thread.</li> <li>• The callback will be executed in the main thread.</li> <li>• In the case where Google Play services is not installed on the user's device, the callback will not be triggered until the user installs it.</li> <li>• The <a href="#">GoogleMap</a> object provided by the callback is non-null.</li> </ul> <p><b>Parameters</b></p> <table border="1" data-bbox="527 688 1738 753"> <tr> <td><code>callback</code></td> <td>The callback object that will be triggered when the map is ready to be used.</td> </tr> </table> <p><code>public final void <b>onCreate</b> (<a href="#">Bundle</a> savedInstanceState)</code></p> <p>You must call this method from the parent Activity/Fragment's corresponding method.</p> <p><a href="https://developers.google.com/android/reference/com/google/android/gms/maps/MapView">https://developers.google.com/android/reference/com/google/android/gms/maps/MapView</a></p>	<code>callback</code>	The callback object that will be triggered when the map is ready to be used.
<code>callback</code>	The callback object that will be triggered when the map is ready to be used.		
<p>[28F] identifying user interaction with the interactive display selecting one or more of the user-selectable symbols corresponding to one or more of the second devices and user interaction with the display specifying an action and, based thereon, sending data</p>	<p>HTC infringes directly and/or indirectly by performing, inducing others to perform, and/or contributing to the performance of identifying user interaction with the interactive display selecting one or more of the user-selectable symbols corresponding to one or more of the second devices and user interaction with the display specifying an action and, based thereon, sending data to the one or more second devices. See claim 1[F], which is incorporated herein by reference in its entirety.</p> <p><b>Share Locations from Google Maps on HTC One M8</b></p>		

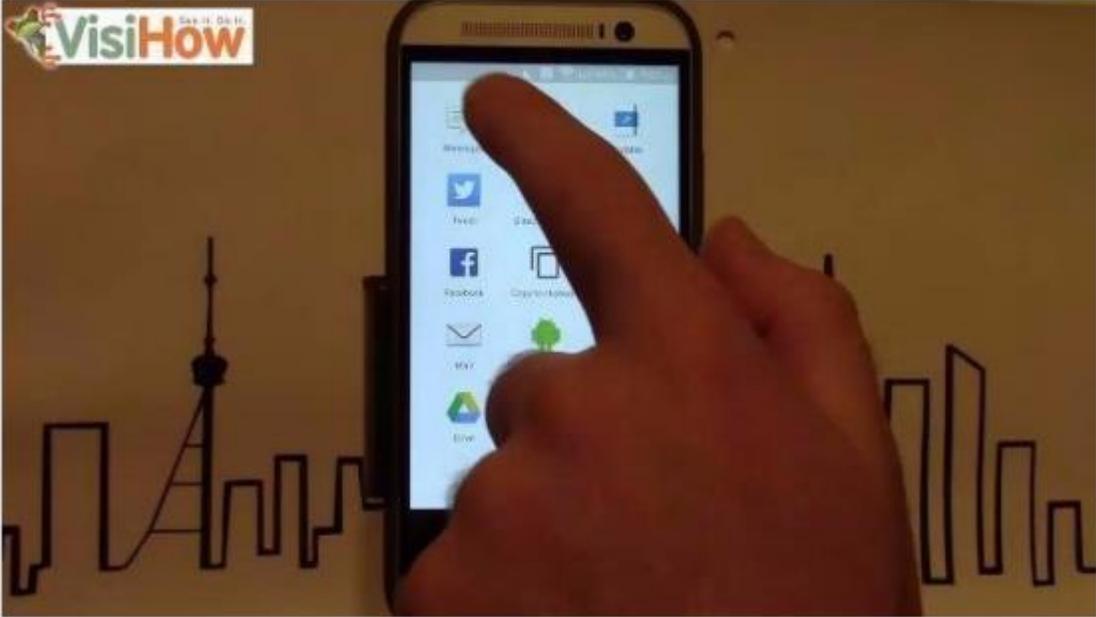
### Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products

US9408055B2	HTC
to the one or more second devices;	<p data-bbox="531 233 827 261"><b>Place a pin on the map.</b></p> <p data-bbox="531 272 1640 383">We can do this anywhere in the entire world, anywhere, on any map that we find in the Google Maps application. We are going to choose a location, press down on the screen, and hold down until we can see that a red pin has been dropped.</p> 

## Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products

US9408055B2	HTC
	<p data-bbox="527 240 1207 305"><b>We have two options:</b> "SAVE" and "SHARE". Press down on the word "SHARE".</p>  <p data-bbox="527 1057 1633 1208"><b>A menu appears with all of the different ways that we can share this location.</b> We can share this location via messages, Bluetooth, Twitter, Facebook, mail, Gmail and Viber. If you have other applications, such as WhatsApp, or any other chat or email applications, you will be able to use them as well and will see them here.</p>

## Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products

US9408055B2	HTC
	<p data-bbox="527 240 856 264"><b>Let's choose "Messages".</b></p> <p data-bbox="527 280 1633 345">This option is located on the top left-hand corner of the screen. All options will have the same basic processes. They will ask us to send the location. Press down on the "Messages" icon.</p> 

## Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products

US9408055B2	HTC
	<p data-bbox="520 233 1260 256">Press the box next to the contact who will be the recipient.</p> <p data-bbox="520 269 1549 292">A checkmark will appear in the box. We can select multiple contacts as well if we desire.</p>  <p data-bbox="520 1045 911 1068"><b>Press on the word "DONE (1)".</b></p> <p data-bbox="520 1084 1629 1153">This icon is located in the bottom right-hand corner of the screen. The bracketed number indicates the number of contacts we selected.</p> <p data-bbox="520 1166 1407 1188"><a href="http://visihow.com/Share_Locations_from_Google_Maps_on_HTC_One_M8">http://visihow.com/Share_Locations_from_Google_Maps_on_HTC_One_M8</a></p> <p data-bbox="512 1243 1908 1422"><b>Regarding Google Maps</b>, Google Latitude, Google Plus, Google Hangouts, Google Messages, and Android Messenger, the Accused Products are configured to allow a user of the first device to interact with the display, to select a user, contact, or device, and to select an action to be performed, such as: sending a message, initiating a call, initiating a data conference, sharing a location, stop sharing a location, block a user from location sharing, sending a location, requesting a location, or sending other data. When an action is</p>

**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC
	<p>specified, data is sent from the first device to the second device via a server.</p> <p><b><u>Exemplary Support for Google Maps:</u></b> Using Google Maps, a user may choose a symbol and send data to that device. For example, a user who is already sharing her location with another user can stop sharing by making a selection resulting in the second device no longer displaying the first device's location. Additionally, a user can share an ETA message with another user or send another user a link in a message to share her location. Additionally, a user who is sharing a location until she arrives can make a selection to stop her location from showing on the second device.</p>

## Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products

US9408055B2	HTC
	<p data-bbox="548 240 968 261">COMPUTER <b>ANDROID</b> IPHONE &amp; IPAD</p> <hr data-bbox="533 289 1566 293"/> <h3 data-bbox="533 347 1024 383">If they have a Google Account</h3> <ol data-bbox="533 402 1419 695" style="list-style-type: none"><li data-bbox="533 402 1220 423">1. If you haven't already, add their Gmail address to your <a href="#">Google Contacts</a>.</li><li data-bbox="533 440 1419 461">2. On your Android phone or tablet, open the Google Maps app  and sign in. <a href="#">Learn how to sign in.</a></li><li data-bbox="533 477 1031 498">3. Tap Menu ≡ &gt; <b>Location sharing</b> &gt; Add People +.</li><li data-bbox="533 514 1003 535">4. Choose how long you want to share your location.</li><li data-bbox="533 552 1140 610">5. Tap <b>Select People</b>.<ul data-bbox="569 586 1140 610" style="list-style-type: none"><li data-bbox="569 586 1140 610">• If you're asked about your contacts, give Google Maps access.</li></ul></li><li data-bbox="533 634 884 656">6. Choose who you want to share with.</li><li data-bbox="533 672 663 693">7. Tap <b>Share</b>.</li></ol> <h3 data-bbox="533 753 1108 789">If they don't have a Google Account</h3> <ol data-bbox="533 813 1560 935" style="list-style-type: none"><li data-bbox="533 813 1419 834">1. On your Android phone or tablet, open the Google Maps app  and sign in. <a href="#">Learn how to sign in.</a></li><li data-bbox="533 850 1031 872">2. Tap Menu ≡ &gt; <b>Location sharing</b> &gt; Add People +.</li><li data-bbox="533 888 1560 935">3. Tap More ... &gt; <b>Copy to clipboard</b>. People with this link can see your location for as long as you choose, up to 72 hours.</li></ol> <h3 data-bbox="533 987 869 1023">Share using another app</h3> <p data-bbox="533 1040 1205 1062">You can also share through messaging apps. Tap More ... &gt; select an app.</p> <h3 data-bbox="533 1122 743 1157">Stop sharing</h3> <ol data-bbox="533 1182 1205 1276" style="list-style-type: none"><li data-bbox="533 1182 842 1203">1. Open the Google Maps app .</li><li data-bbox="533 1219 869 1240">2. Tap Menu ≡ &gt; <b>Location sharing</b>.</li><li data-bbox="533 1256 1205 1278">3. Next to the person with whom you want to stop sharing, tap Remove ✕.</li></ol> <p data-bbox="512 1295 1703 1328"><a href="https://support.google.com/maps/answer/7326816?co=GENIE.Platform%3DAndroid&amp;oco=1">https://support.google.com/maps/answer/7326816?co=GENIE.Platform%3DAndroid&amp;oco=1</a></p>

## Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products

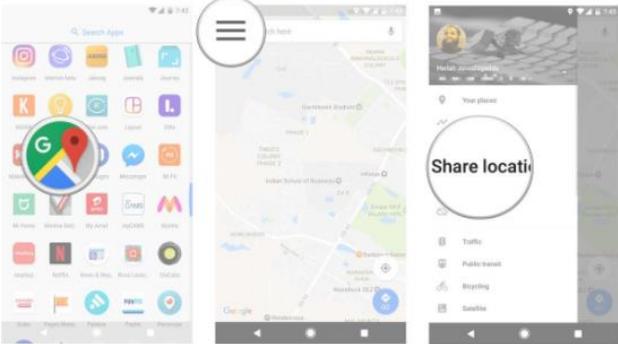
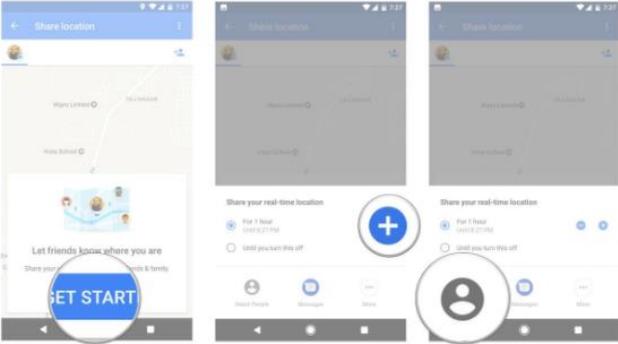
US9408055B2	HTC
	<h3>Share your E.T.A</h3> <p>After you start your drive, you can share your destination, estimated arrival time, and where you are on the route.</p> <ol style="list-style-type: none"><li>1. Open the Google Maps app .</li><li>2. Set a driving destination. <a href="#">Learn how to navigate to a place.</a></li><li>3. After you start navigation, tap More  &gt; <b>Share trip progress.</b></li><li>4. Choose a person from the list.</li><li>5. Tap <b>Share.</b></li><li>6. Location Sharing will stop when you reach your destination or stop navigating.</li></ol> <ul style="list-style-type: none"><li>• To stop sharing before you arrive, tap More  &gt; <b>Stop sharing.</b></li></ul> <h3>See where someone is</h3> <p>If someone shares their location with you, you can see them on the map.</p> <ol style="list-style-type: none"><li>1. Open the Google Maps app .</li><li>2. Tap Menu  &gt; <b>Location sharing.</b></li><li>3. Choose someone.</li></ol> <ul style="list-style-type: none"><li>• To see an updated location, tap on a friend's icon &gt; More  &gt; <b>Refresh.</b></li></ul> <h3>Stop seeing someone's location</h3> <ol style="list-style-type: none"><li>1. Open the Google Maps app .</li><li>2. On the map, tap their icon.</li><li>3. At the bottom, tap More .</li><li>4. To temporarily hide someone, tap <b>Hide from map.</b> You can stop hiding them at any time.</li></ol> <p><b>Note:</b> You can stop someone's location from ever appearing on your map. Learn how to <a href="#">block another person's account.</a></p> <p><a href="https://support.google.com/maps/answer/7326816?co=GENIE.Platform%3DAndroid&amp;oco=1">https://support.google.com/maps/answer/7326816?co=GENIE.Platform%3DAndroid&amp;oco=1</a></p>



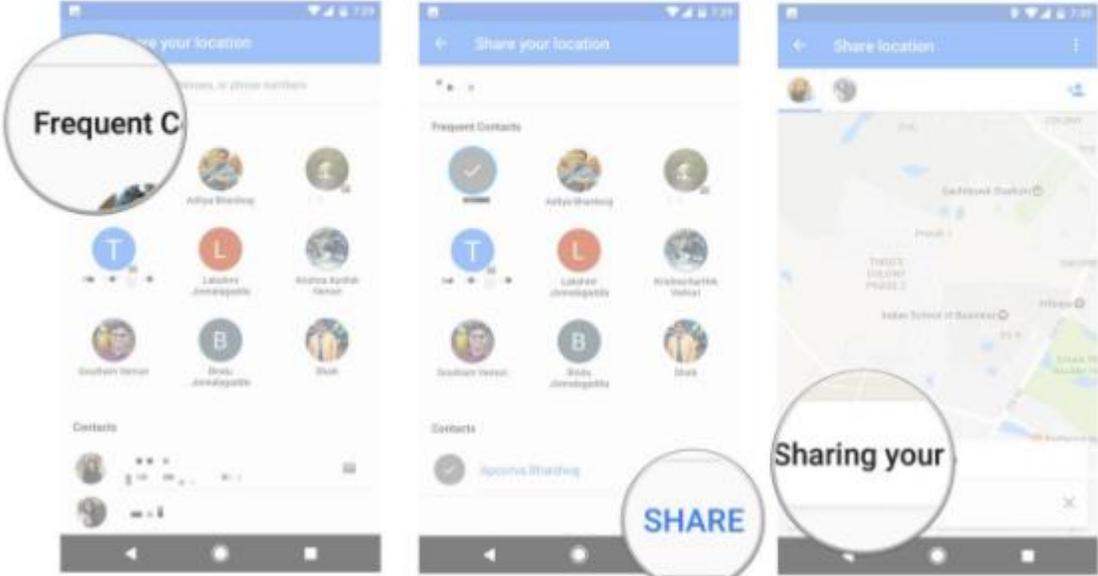
## Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products

US9408055B2	HTC
	<h3 data-bbox="541 280 877 318">Hide or share lists</h3> <p data-bbox="541 347 909 370"><b>Note:</b> You can't share starred places.</p> <ol data-bbox="554 402 1682 667" style="list-style-type: none"><li data-bbox="554 402 890 425">1. Open the Google Maps app .</li><li data-bbox="554 443 968 466">2. Tap Menu  &gt; <b>Your places</b> &gt; <b>Saved</b>.</li><li data-bbox="554 483 1682 667">3. Next to the list you want to share, tap More  &gt; choose an option:<ul data-bbox="583 524 1682 667" style="list-style-type: none"><li data-bbox="583 524 1444 547">• <b>Hide/Show on your map:</b> Display or hide your saved places when looking at the map.</li><li data-bbox="583 565 1058 587">• <b>Share list:</b> Allow others to see your saved list.</li><li data-bbox="583 605 1682 667">• <b>Sharing options:</b> You can make your list public, private, or shared. To let anyone with the link see your list, tap <b>Shared</b>. To let anyone find and follow your list, tap <b>Public</b>.</li></ul></li></ol> <h3 data-bbox="541 735 768 773">Follow a list</h3> <p data-bbox="541 802 1728 857">If you follow a list made by someone else, their saved places will show up in Your Places. The places will also appear as suggested locations in Google Maps.</p> <h3 data-bbox="541 914 915 951">Follow a list using a link</h3> <ol data-bbox="554 976 1356 1079" style="list-style-type: none"><li data-bbox="554 976 961 998">1. Tap on the link you received to open it.</li><li data-bbox="554 1016 1272 1039">2. Tap <b>Follow</b>. This list will now be added to the group of lists you follow.</li><li data-bbox="554 1057 1356 1079">3. <b>Optional:</b> To unfollow a list someone shared with you, tap the list &gt; <b>Following</b>.</li></ol> <h3 data-bbox="541 1133 926 1170">See lists made by others</h3> <p data-bbox="541 1195 1335 1218">If a user has any Google Maps lists that were made public, you can follow them.</p> <ol data-bbox="554 1252 1136 1356" style="list-style-type: none"><li data-bbox="554 1252 1136 1274">1. Tap on the name of a user whose list you want to follow.</li><li data-bbox="554 1292 678 1315">2. Tap <b>Lists</b>.</li><li data-bbox="554 1333 1136 1356">3. Tap on the list you want to follow &gt; More  &gt; <b>Follow</b>.</li></ol> <p data-bbox="510 1373 1902 1401"><a href="https://support.google.com/maps/answer/7280933?hl=en&amp;ref_topic=7301134&amp;co=GENIE.Platform%3DAn">https://support.google.com/maps/answer/7280933?hl=en&amp;ref_topic=7301134&amp;co=GENIE.Platform%3DAn</a></p>

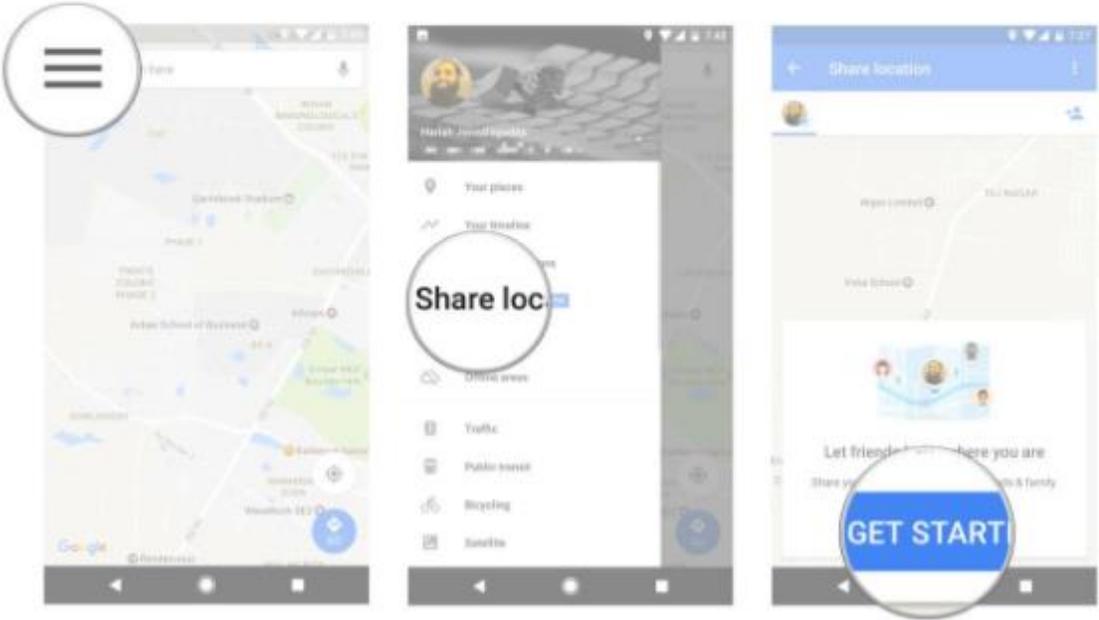
# Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products

US9408055B2	HTC
	<p data-bbox="514 233 682 261">droid&amp;oco=1</p> <h3 data-bbox="514 310 1102 342">How to share your location in Google Maps</h3> <ol data-bbox="514 367 1087 451" style="list-style-type: none"><li>1. Open Google Maps from the app drawer or the home screen.</li><li>2. Tap the <b>hamburger menu</b> (the three horizontal lines) on the top left corner of the screen.</li><li>3. Select <b>Share location</b>.</li></ol>  <ol data-bbox="514 854 1117 954" style="list-style-type: none"><li>4. Tap <b>Get Started</b>.</li><li>5. Use the <b>+</b> icon to select a time period or select the <b>Until you turn this off</b> setting to share your location indefinitely.</li><li>6. Tap <b>Select People</b>.</li></ol>  <p data-bbox="514 1333 1354 1365"><a href="https://www.androidcentral.com/how-share-location-google-maps">https://www.androidcentral.com/how-share-location-google-maps</a></p>

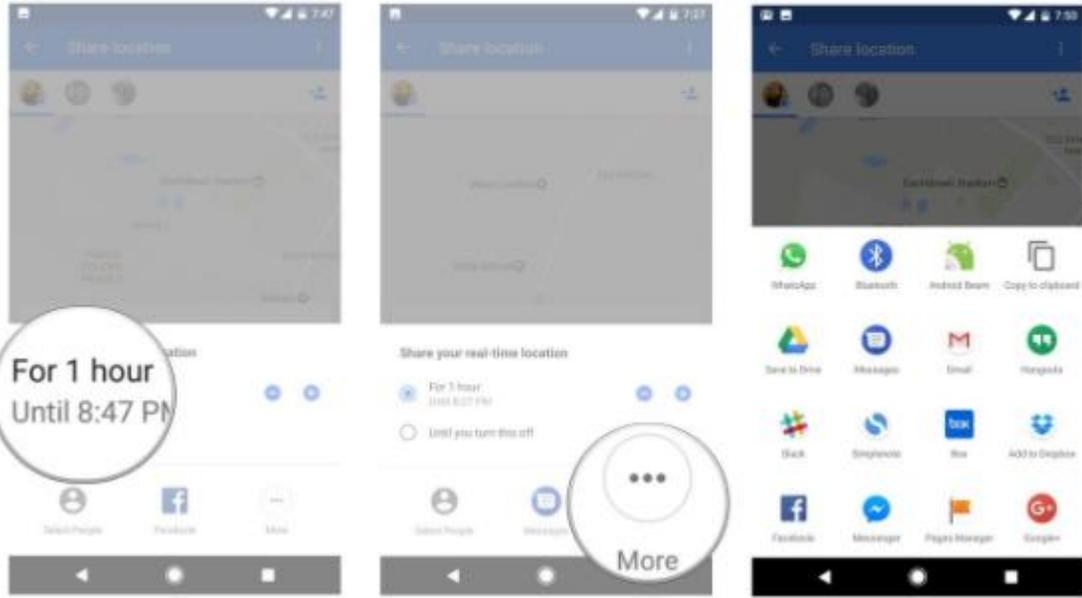
## Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products

US9408055B2	HTC
	<p data-bbox="527 253 1577 310">7. You'll see a list of your frequent contacts at the top, along with a full list of contacts. Pick the contacts by tapping their name.</p> <p data-bbox="527 339 1457 367">8. Once you've selected the contacts you want to share your location to, tap <b>Share</b>.</p> <p data-bbox="527 396 1419 423">9. You'll see a message saying that the selected contact can view your location.</p>  <p data-bbox="510 1101 1356 1128"><a href="https://www.androidcentral.com/how-share-location-google-maps">https://www.androidcentral.com/how-share-location-google-maps</a></p>

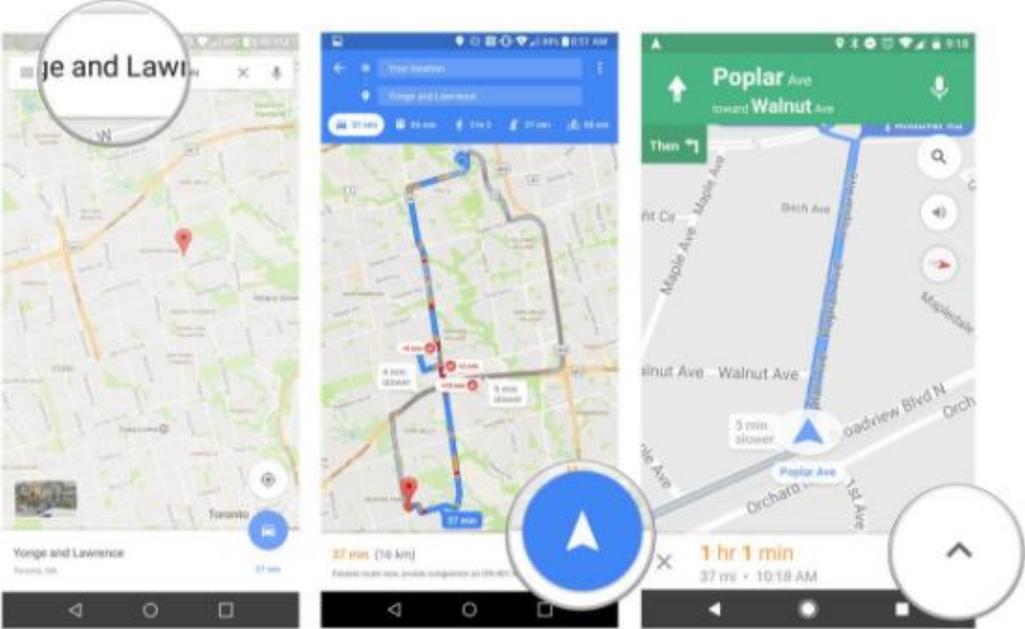
## Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products

US9408055B2	HTC
	<h3 data-bbox="520 245 1255 289">How to create a shareable link</h3> <p data-bbox="520 334 1461 362">You can also create a link and use it to share your location easily. Here's how to do it:</p> <ol data-bbox="520 412 1234 553" style="list-style-type: none"><li data-bbox="520 412 1234 440">1. Tap the hamburger menu on the top left corner of the screen.</li><li data-bbox="520 467 800 495">2. Select Share location.</li><li data-bbox="520 522 737 550">3. Tap Get Started.</li></ol>  <p data-bbox="506 1230 1356 1263"><a href="https://www.androidcentral.com/how-share-location-google-maps">https://www.androidcentral.com/how-share-location-google-maps</a></p>

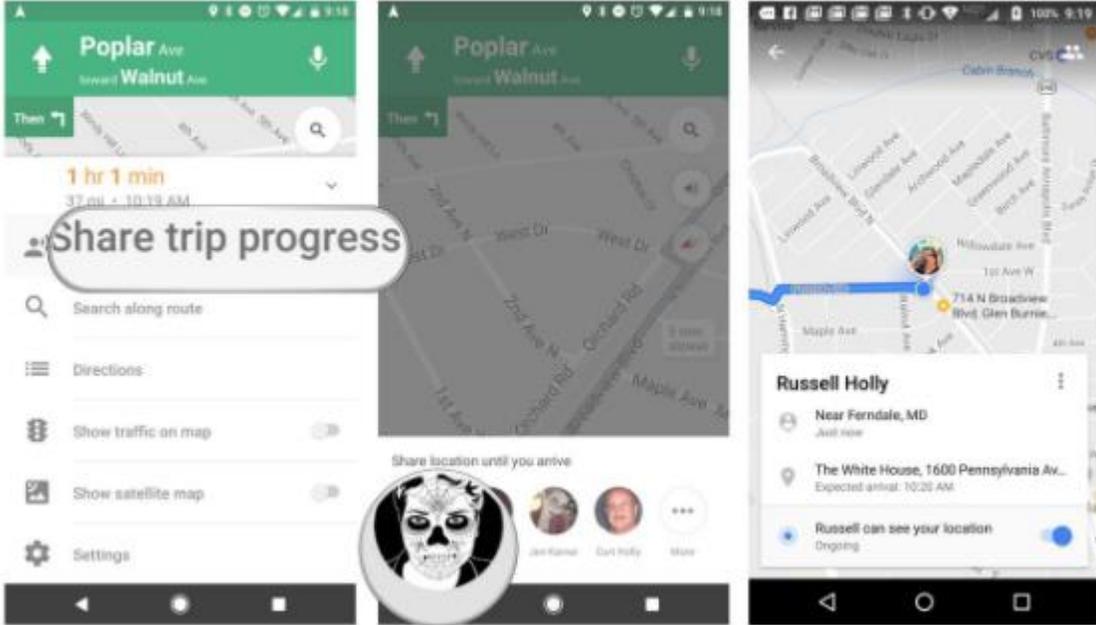
**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC
	<p data-bbox="520 245 1213 272">4. Select the <b>amount of time</b> you want to share your location.</p> <p data-bbox="520 302 680 329">5. Tap <b>More</b>.</p> <p data-bbox="520 358 1633 418">6. Select <b>your app of choice</b> to create and send a <b>unique URL</b> that broadcasts your current location. You can email it, send the link via Messenger, or even tweet it to the intended recipient.</p>  <p data-bbox="506 1084 1356 1117"><a href="https://www.androidcentral.com/how-share-location-google-maps">https://www.androidcentral.com/how-share-location-google-maps</a></p>

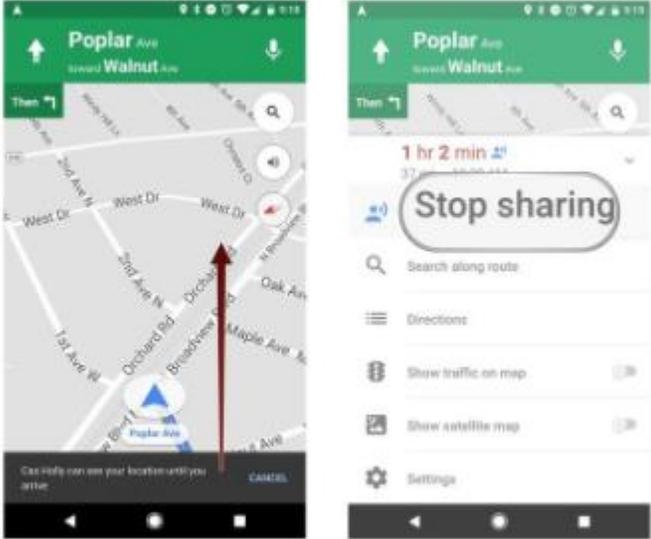
**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC
	<h2 data-bbox="527 240 1423 337">How to share your navigation directions while you walk, drive or transit</h2> <p data-bbox="527 375 1556 464">One of the best ways to share your location with someone is during a drive, walk or transit. If you're meeting a friend or family member somewhere, or navigating towards their home, Google Maps lets you share your location with them for the duration of the trip. It's magic!</p> <ol data-bbox="527 513 1394 643" style="list-style-type: none"><li>1. In the <b>search bar</b> enter your destination.</li><li>2. Pick your navigation type (drive, transit, walk) and press the blue <b>navigate</b> button.</li><li>3. Tap the arrow next to the <b>time-to-destination</b> number at the bottom of the screen.</li></ol>  <p data-bbox="512 1328 1356 1360"><a href="https://www.androidcentral.com/how-share-location-google-maps">https://www.androidcentral.com/how-share-location-google-maps</a></p>

**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

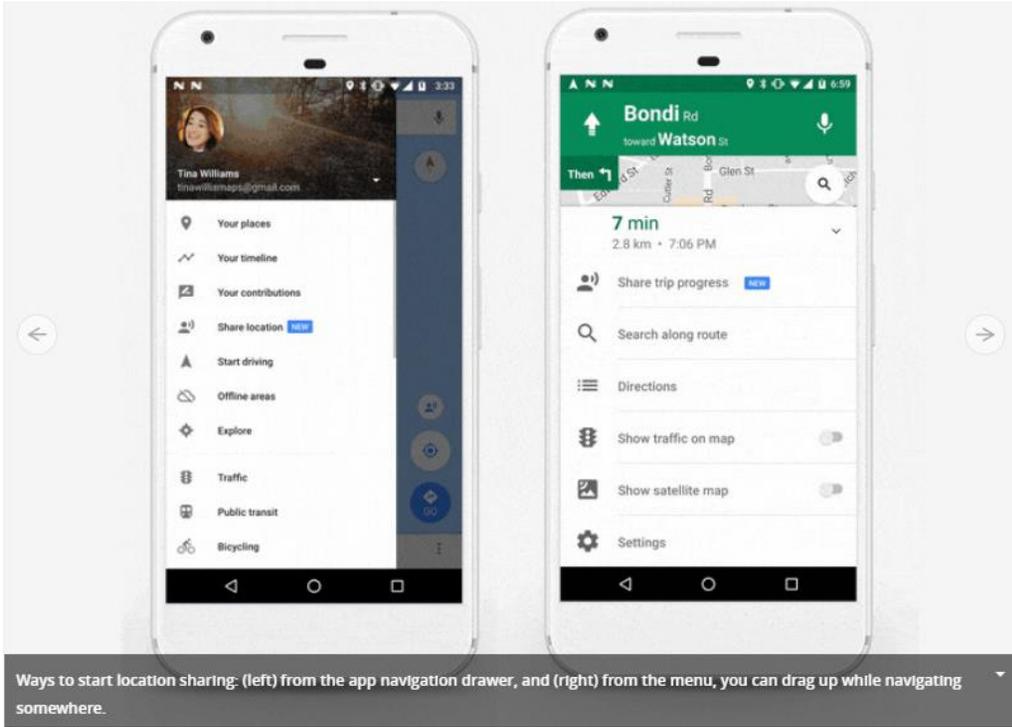
US9408055B2	HTC
	<p data-bbox="527 277 835 305">4. Tap Share trip progress.</p> <p data-bbox="527 334 1150 362">5. Choose one or more contacts to share trip progress.</p>  <p data-bbox="527 1065 1339 1092">You can also stop sharing your location with someone before a trip ends.</p> <p data-bbox="512 1101 1356 1128"><a href="https://www.androidcentral.com/how-share-location-google-maps">https://www.androidcentral.com/how-share-location-google-maps</a></p>

**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

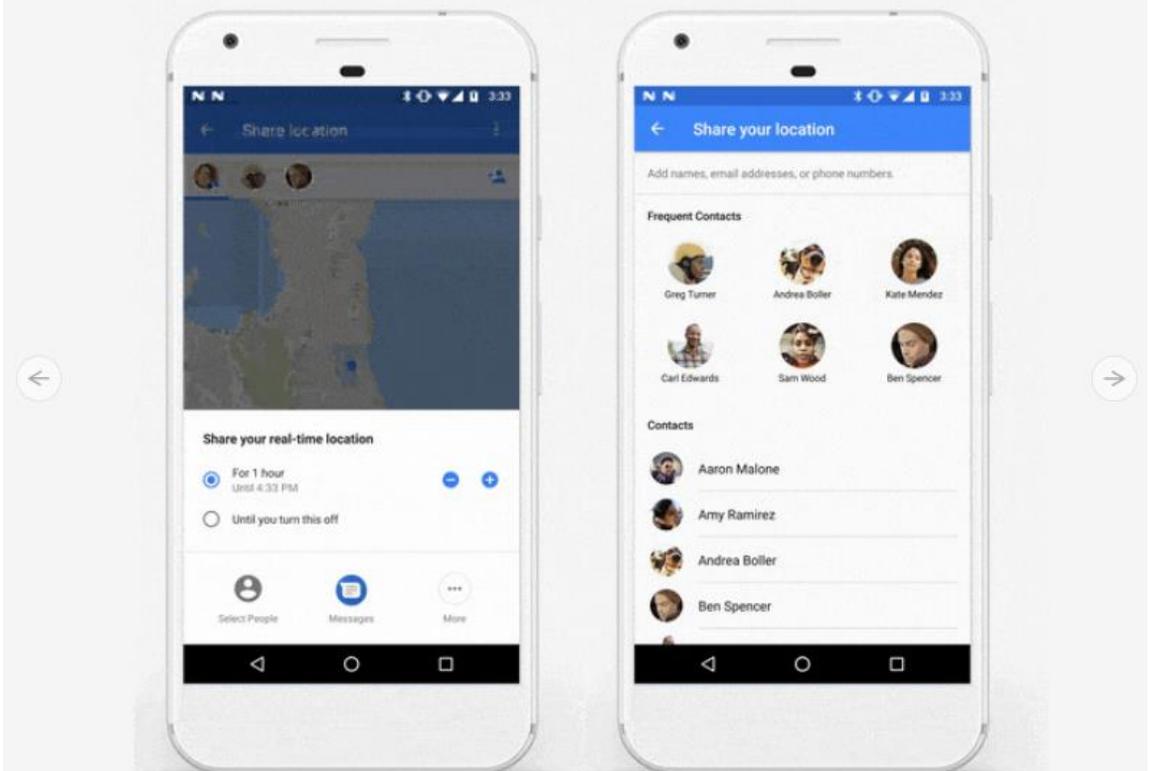
US9408055B2	HTC
	<ol style="list-style-type: none"><li data-bbox="533 245 1465 272">1. Tap the arrow next to the <b>time-to-destination</b> number at the bottom of the screen.</li><li data-bbox="533 302 768 329">2. Tap <b>Stop sharing</b>.</li></ol> <div data-bbox="747 383 1398 922"></div> <p data-bbox="541 976 638 1003">That's it!</p> <p data-bbox="541 1045 1608 1073">Are you excited that location sharing is back in Google Maps? How often do you use the feature?</p> <p data-bbox="512 1084 1356 1112"><a href="https://www.androidcentral.com/how-share-location-google-maps">https://www.androidcentral.com/how-share-location-google-maps</a></p> <p data-bbox="512 1195 1419 1222">As shown below, a group may also be defined within Google Contacts.</p>



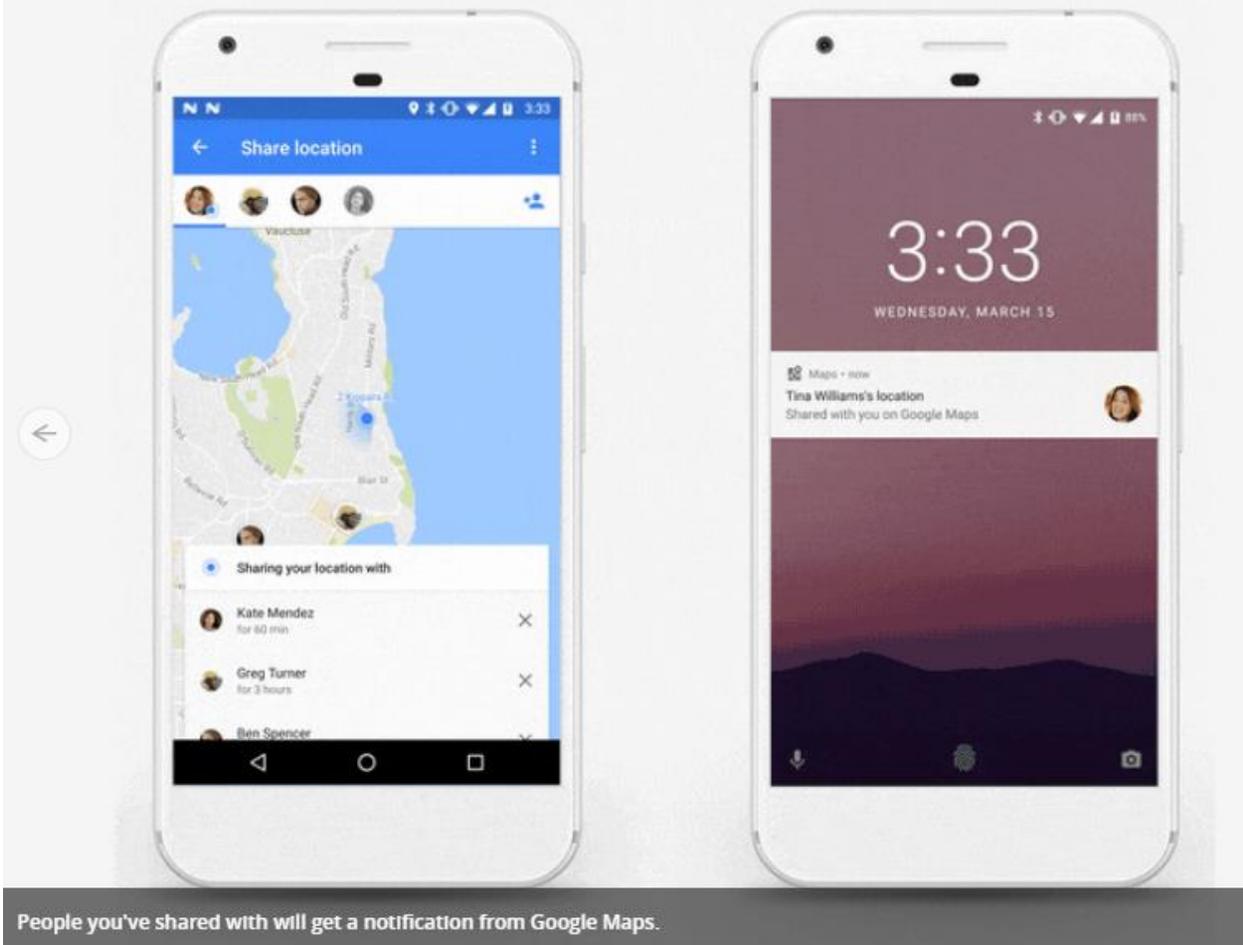
# Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products

US9408055B2	HTC
	<h2 data-bbox="548 245 940 289">Share your contacts</h2> <ol data-bbox="562 318 1045 477" style="list-style-type: none"><li>1. Open your device's Contacts app .</li><li>2. Tap a contact in the list.</li><li>3. Tap More  &gt; <b>Share</b>.</li><li>4. Choose how you want to share the contact.</li></ol> <p data-bbox="512 496 1535 529"><a href="https://support.google.com/android/answer/6118731?hl=en&amp;ref_topic=6118711">https://support.google.com/android/answer/6118731?hl=en&amp;ref_topic=6118711</a></p>  <p data-bbox="520 1230 1516 1284">Ways to start location sharing: (left) from the app navigation drawer, and (right) from the menu, you can drag up while navigating somewhere.</p> <p data-bbox="512 1295 1656 1325"><a href="https://arstechnica.com/gadgets/2017/03/location-sharing-finally-returns-to-google-maps/">https://arstechnica.com/gadgets/2017/03/location-sharing-finally-returns-to-google-maps/</a></p>

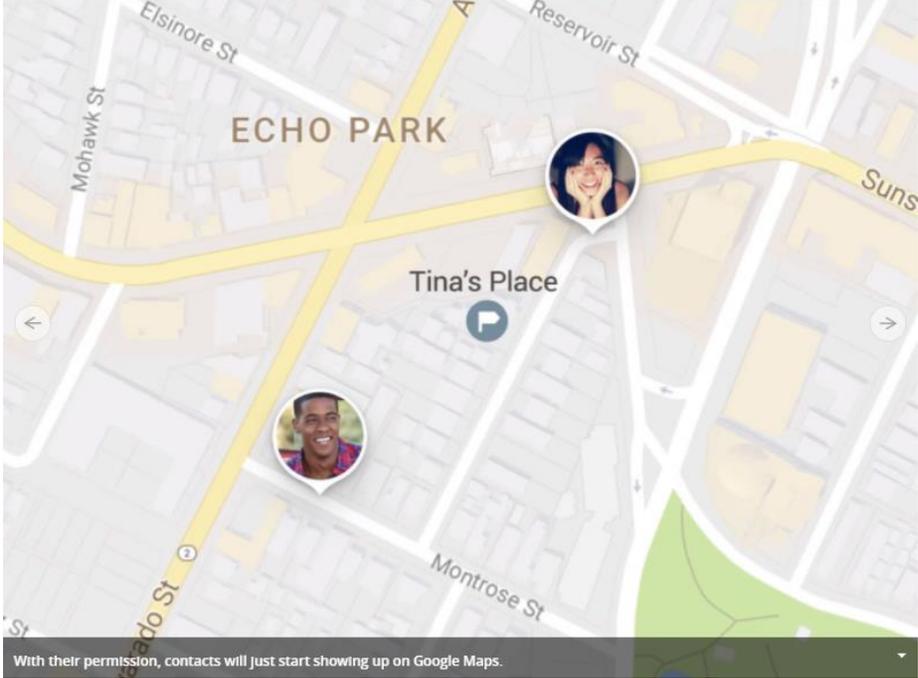
**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC
	 <p data-bbox="512 1008 1661 1057">You can pick how long you want to share your location for, and with whom you want to share it.</p> <p data-bbox="512 1057 1661 1096"><a href="https://arstechnica.com/gadgets/2017/03/location-sharing-finally-returns-to-google-maps/">https://arstechnica.com/gadgets/2017/03/location-sharing-finally-returns-to-google-maps/</a></p>

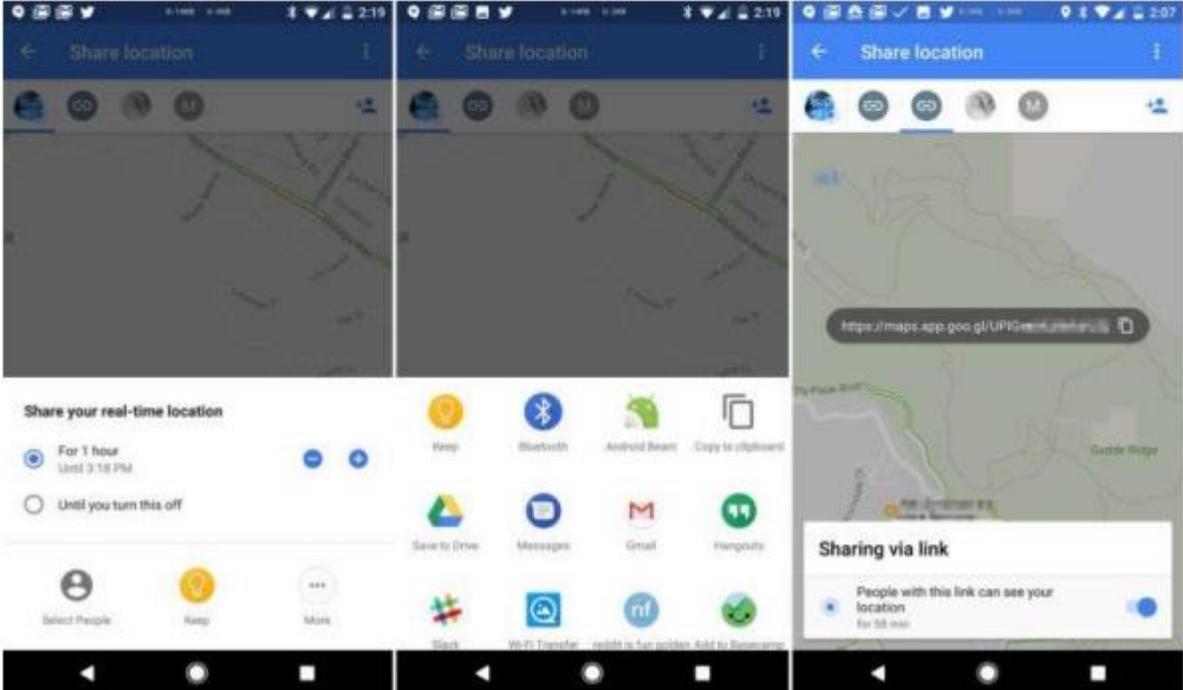
**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC
	 <p data-bbox="512 1143 1176 1170">People you've shared with will get a notification from Google Maps.</p> <p data-bbox="512 1187 1656 1219"><a href="https://arstechnica.com/gadgets/2017/03/location-sharing-finally-returns-to-google-maps/">https://arstechnica.com/gadgets/2017/03/location-sharing-finally-returns-to-google-maps/</a></p>

**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC
	 <p data-bbox="512 883 1430 911">With their permission, contacts will just start showing up on Google Maps.</p> <p data-bbox="512 915 1656 948"><a href="https://arstechnica.com/gadgets/2017/03/location-sharing-finally-returns-to-google-maps/">https://arstechnica.com/gadgets/2017/03/location-sharing-finally-returns-to-google-maps/</a></p> <h3 data-bbox="512 1024 779 1068">Stop sharing</h3> <ol data-bbox="512 1094 1352 1219" style="list-style-type: none"><li>1. Open the Google Maps app 📍.</li><li>2. Tap the Menu ☰ &gt; Share location.</li><li>3. Next to the person with whom you want to stop sharing, tap Remove ✕.</li></ol>

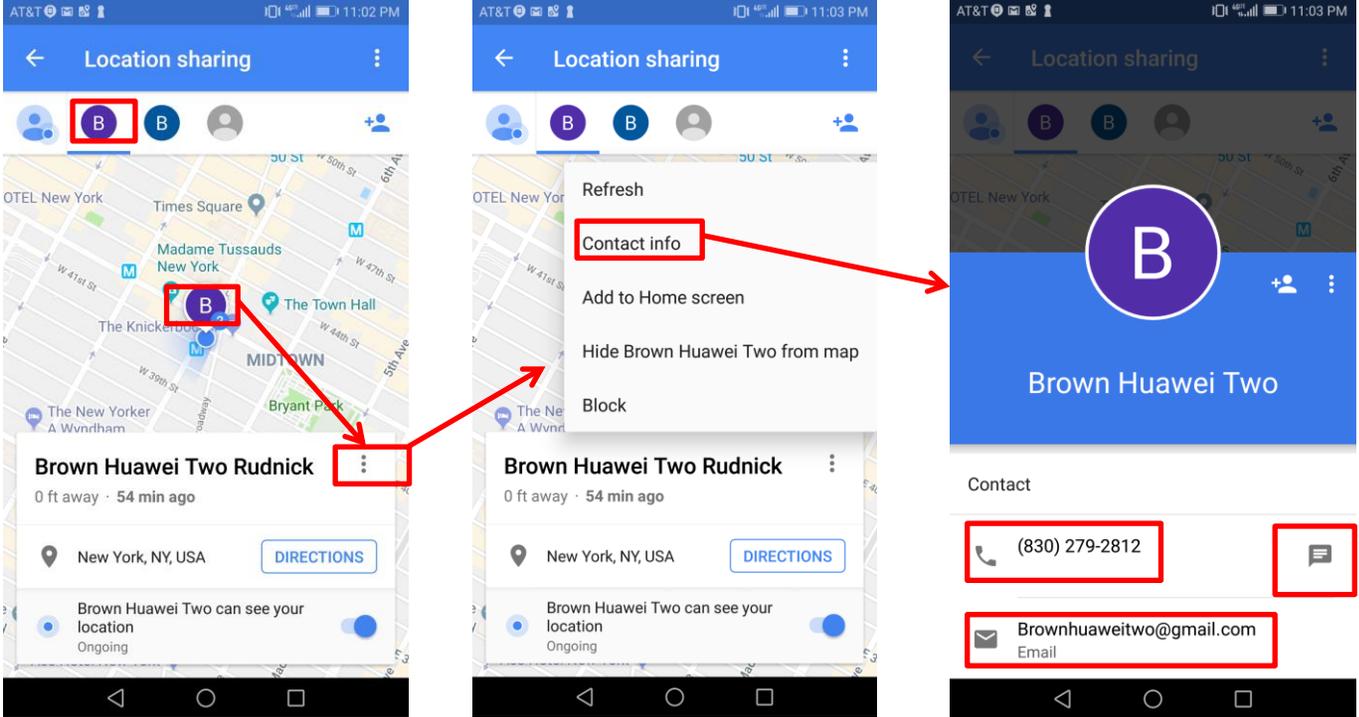
# Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products

US9408055B2	HTC
	<h2 data-bbox="537 232 856 272">Share your E.T.A</h2> <p data-bbox="537 302 1703 326">After you start your drive, you can share your destination, estimated arrival time, and where you are on the route.</p> <ol data-bbox="548 358 1388 602" style="list-style-type: none"><li>1. Open the Google Maps app .</li><li>2. Set a driving destination. <a href="#">Learn how to navigate to a place.</a></li><li>3. After you start navigation, tap <b>More</b>  <b>&gt; Share trip progress.</b></li><li>4. Choose a person from the list.</li><li>5. Tap <b>Share.</b></li><li>6. Location Sharing will stop when you reach your destination or stop navigating.</li></ol> <p data-bbox="537 634 1230 659">• To stop sharing before you arrive, tap <b>More</b>  <b>&gt; Stop sharing.</b></p> 

**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC
	 <p><b><u>Exemplary Maps Screenshots:</u></b></p>

**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC
	 <p><b>Exemplary Source Code:</b></p> <p>The above functionality is performed at least in part by the following publicly available source code and/or source code that invokes or is invoked by the following source code (or a substantially similar copy compiled and loaded onto the Accused Products by HTC). AGIS reserves the right to supplement these contentions with additional source code as discovery progresses and as additional source code is made available. AGIS reserves the right to supplement these contentions with additional source code as discovery progresses and as additional source code is made available.</p>

**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC
	<pre>44 * Class that sends chat message via SMS. 45 * 46 * The interface emulates a blocking sending similar to making an HTTP request. 47 * It calls the SmsManager to send a (potentially multipart) message and waits 48 * on the sent status on each part. The waiting has a timeout so it won't wait 49 * forever. Once the sent status of all parts received, the call returns. 50 * A successful sending requires success status for all parts. Otherwise, we 51 * pick the highest level of failure as the error for the whole message, which 52 * is used to determine if we need to retry the sending. 53 */ 54 public class SmsSender { 55     private static final String TAG = LogUtil.BUGLE_TAG; 56 57     public static final String EXTRA_PART_ID = "part_id"; 58 59     /* 60     * A map for pending sms messages. The key is the random request UUID. 61     */ 62     private static ConcurrentHashMap&lt;Uri, SendResult&gt; sPendingMessageMap = 63         new ConcurrentHashMap&lt;Uri, SendResult&gt;(); 64 65     private static final Random RANDOM = new Random(); 66 67     // Whether we should send multipart SMS as separate messages 68     private static Boolean sSendMultipartSmsAsSeparateMessages = null; 69</pre> <p><a href="https://android.googlesource.com/platform/packages/apps/Messaging/+nougat-mr1-release/src/com/android/messaging/sms/SmsSender.java">https://android.googlesource.com/platform/packages/apps/Messaging/+nougat-mr1-release/src/com/android/messaging/sms/SmsSender.java</a></p>



**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC
	<pre>253 // Actually sending the message using SmsManager 254 private static void sendInternal(final Context context, final int subId, String dest, 255     final ArrayList&lt;String&gt; messages, final String serviceCenter, 256     final boolean requireDeliveryReport, final Uri messageUri) throws SmsException { 257     Assert.notNull(context); 258     final SmsManager smsManager = PhoneUtils.get(subId).getSmsManager(); 259     final int messageCount = messages.size(); 260     final ArrayList&lt;PendingIntent&gt; deliveryIntents = new ArrayList&lt;PendingIntent&gt;(messageCount); 261     final ArrayList&lt;PendingIntent&gt; sentIntents = new ArrayList&lt;PendingIntent&gt;(messageCount); 262     for (int i = 0; i &lt; messageCount; i++) { 263         // Make pending intents different for each message part 264         final int partId = (messageCount &lt;= 1 ? 0 : i + 1); 265         if (requireDeliveryReport &amp;&amp; (i == (messageCount - 1))) { 266             // TODO we only care about the delivery status of the last part 267             // Shall we have better tracking of delivery status of all parts? 268             deliveryIntents.add(PendingIntent.getBroadcast( 269                 context, 270                 partId, 271                 getSendStatusIntent(context, SendStatusReceiver.MESSAGE_DELIVERED_ACTION, 272                     messageUri, partId, subId), 273                 0/*flag*/)); 274         } else { 275             deliveryIntents.add(null); 276         } 277         sentIntents.add(PendingIntent.getBroadcast( 278             context, 279             partId, 280             getSendStatusIntent(context, SendStatusReceiver.MESSAGE_SENT_ACTION, 281                 messageUri, partId, subId), 282             0/*flag*/)); 283     } 284     if (sSendMultipartSmsAsSeparateMessages == null) { 285         sSendMultipartSmsAsSeparateMessages = MmsConfig.get(subId) 286             .getSendMultipartSmsAsSeparateMessages(); 287     }</pre> <p><a href="https://android.googlesource.com/platform/packages/apps/Messaging/+nougat-mr1-release/src/com/android/messaging/sms/SmsSender.java">https://android.googlesource.com/platform/packages/apps/Messaging/+nougat-mr1-release/src/com/android/messaging/sms/SmsSender.java</a></p>

**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC
	<pre>288     try { 289         if (sSendMultipartSmsAsSeparateMessages) { 290             // If multipart sms is not supported, send them as separate messages 291             for (int i = 0; i &lt; messageCount; i++) { 292                 smsManager.sendTextMessage(dest, 293                     serviceCenter, 294                     messages.get(i), 295                     sentIntents.get(i), 296                     deliveryIntents.get(i)); 297             } 298         } else { 299             smsManager.sendMultipartTextMessage( 300                 dest, serviceCenter, messages, sentIntents, deliveryIntents); 301         } 302     } catch (final Exception e) { 303         throw new SmsException("SmsSender: caught exception in sending " + e); 304     } 305 }</pre> <p><a href="https://android.googlesource.com/platform/packages/apps/Messaging/+nougat-mr1-release/src/com/android/messaging/sms/SmsSender.java">https://android.googlesource.com/platform/packages/apps/Messaging/+nougat-mr1-release/src/com/android/messaging/sms/SmsSender.java</a></p> <pre>56  * Class that receives incoming SMS messages through android.provider.Telephony.SMS_RECEIVED 57  * 58  * This class serves two purposes: 59  * - Process phone verification SMS messages 60  * - Handle SMS messages when the user has enabled us to be the default SMS app (Pre-KLP) 61  */ 62  public final class SmsReceiver extends BroadcastReceiver { 63      private static final String TAG = LogUtil.BUGLE_TAG; 64 65      private static ArrayList&lt;Pattern&gt; sIgnoreSmsPatterns; 66  }</pre> <p><a href="https://android.googlesource.com/platform/packages/apps/Messaging/+nougat-mr1-release/src/com/android/messaging/receiver/SmsReceiver.java">https://android.googlesource.com/platform/packages/apps/Messaging/+nougat-mr1-release/src/com/android/messaging/receiver/SmsReceiver.java</a></p>

**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC
	<pre>200 public static void deliverSmsMessages(final Context context, final int subId, 201     final int errorCode, final android.telephony.SmsMessage[] messages) { 202     final ContentValues messageValues = 203         MmsUtils.parseReceivedSmsMessage(context, messages, errorCode); 204 205     LogUtil.v(TAG, "SmsReceiver.deliverSmsMessages"); 206 207     final long nowInMillis = System.currentTimeMillis(); 208     final long receivedTimestampMs = MmsUtils.getMessageDate(messages[0], nowInMillis); 209 210     messageValues.put(Sms.Inbox.DATE, receivedTimestampMs); 211     // Default to unread and unseen for us but ReceiveSmsMessageAction will override 212     // seen for the telephony db. 213     messageValues.put(Sms.Inbox.READ, 0); 214     messageValues.put(Sms.Inbox.SEEN, 0); 215     if (OsUtil.isAtLeastL_MR1()) { 216         messageValues.put(Sms.SUBSCRIPTION_ID, subId); 217     } 218 219     if (messages[0].getMessageClass() == android.telephony.SmsMessage.MessageClass.CLASS_0    220         DebugUtils.debugClassZeroSmsEnabled()) { 221         Factory.get().getUIIntents().launchClassZeroActivity(context, messageValues); 222     } else { 223         final ReceiveSmsMessageAction action = new ReceiveSmsMessageAction(messageValues); 224         action.start(); 225     } 226 }</pre> <p><a href="https://android.googlesource.com/platform/packages/apps/Messaging/+nougat-mr1-release/src/com/android/messaging/receiver/SmsReceiver.java">https://android.googlesource.com/platform/packages/apps/Messaging/+nougat-mr1-release/src/com/android/messaging/receiver/SmsReceiver.java</a></p>

**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC
	<pre>228     @Override 229     public void onReceive(final Context context, final Intent intent) { 230         LogUtil.v(TAG, "SmsReceiver.onReceive " + intent); 231         // On KLP+ we only take delivery of SMS messages in SmsDeliverReceiver. 232         if (PhoneUtils.getDefault().isSmsEnabled()) { 233             final String action = intent.getAction(); 234             if (OsUtil.isSecondaryUser() &amp;&amp; 235                 (Telephony.Sms.Intents.SMS_RECEIVED_ACTION.equals(action)    236                  // TODO: update this with the actual constant from Telephony 237                  "android.provider.Telephony.MMS_DOWNLOADED".equals(action))) { 238                 postNewMessageSecondaryUserNotification(); 239             } else if (!OsUtil.isAtLeastKLP()) { 240                 deliverSmsIntent(context, intent); 241             } 242         } 243     }</pre> <p><a href="https://android.googlesource.com/platform/packages/apps/Messaging/+/nougat-mr1-release/src/com/android/messaging/receiver/SmsReceiver.java">https://android.googlesource.com/platform/packages/apps/Messaging/+/nougat-mr1-release/src/com/android/messaging/receiver/SmsReceiver.java</a></p>

**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC
	<pre>52  * Class that sends chat message via MMS. 53  * 54  * The interface emulates a blocking send similar to making an HTTP request. 55  */ 56  public class MmsSender { 57      private static final String TAG = LogUtil.BUGLE_TAG; 58 59      /** 60       * Send an MMS message. 61       * 62       * @param context Context 63       * @param messageUri The unique URI of the message for identifying it during sending 64       * @param sendReq The SendReq PDU of the message 65       * @throws MmsFailureException 66       */ 67      public static void sendMms(final Context context, final int subId, final Uri messageUri, 68                               final SendReq sendReq, final Bundle sentIntentExtras) throws MmsFailureException { 69          sendMms(context, 70                subId, 71                messageUri, 72                null /* locationUri */, 73                sendReq, 74                true /* responseImportant */, 75                sentIntentExtras); 76      }</pre> <p><a href="https://android.googlesource.com/platform/packages/apps/Messaging/+/nougat-mr1-release/src/com/android/messaging/sms/MmsSender.java">https://android.googlesource.com/platform/packages/apps/Messaging/+/nougat-mr1-release/src/com/android/messaging/sms/MmsSender.java</a></p>

**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC
	<pre>240     * Download an MMS message. 241     * 242     * @param context Context 243     * @param contentLocation The url of the MMS message 244     * @throws MmsFailureException 245     * @throws InvalidHeaderValueException 246     */ 247     public static void downloadMms(final Context context, final int subId, 248         final String contentLocation, Bundle extras) throws MmsFailureException, 249         InvalidHeaderValueException { 250         final Uri requestUri = Uri.parse(contentLocation); 251         final Uri contentUri = MmsFileProvider.buildRawMmsUri(); 252 253         final Intent downloadedIntent = new Intent(SendStatusReceiver.MMS_DOWNLOADED_ACTION, 254             requestUri, 255             context, 256             SendStatusReceiver.class); 257         downloadedIntent.putExtra(SendMessageAction.EXTRA_CONTENT_URI, contentUri); 258         if (extras != null) { 259             downloadedIntent.putExtras(extras); 260         } 261         final PendingIntent downloadedPendingIntent = PendingIntent.getBroadcast( 262             context, 263             0 /*request code*/, 264             downloadedIntent, 265             PendingIntent.FLAG_UPDATE_CURRENT); 266 267         MmsManager.downloadMultimediaMessage(subId, context, contentLocation, contentUri, 268             downloadedPendingIntent); 269     }</pre> <p><a href="https://android.googlesource.com/platform/packages/apps/Messaging/+nougat-mr1-release/src/com/android/messaging/sms/MmsSender.java">https://android.googlesource.com/platform/packages/apps/Messaging/+nougat-mr1-release/src/com/android/messaging/sms/MmsSender.java</a></p>

**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC
	<pre> 97      * Execute an MMS HTTP request, either a POST (sending) or a GET (downloading) 98      * 99      * @param urlString The request URL, for sending it is usually the MMSC, and for downloading 100     *           it is the message URL 101     * @param pdu For POST (sending) only, the PDU to send 102     * @param method HTTP method, POST for sending and GET for downloading 103     * @param isProxySet Is there a proxy for the MMSC 104     * @param proxyHost The proxy host 105     * @param proxyPort The proxy port 106     * @param mmsConfig The MMS config to use 107     * @param userAgent The user agent header value 108     * @param uaProfUrl The UA Prof URL header value 109     * @return The HTTP response body 110     * @throws MmsHttpException For any failures 111     */ 112     public byte[] execute(String urlString, byte[] pdu, String method, boolean isProxySet, 113                          String proxyHost, int proxyPort, Bundle mmsConfig, String userAgent, String uaProfUrl) 114         throws MmsHttpException { 115         Log.d(MmsService.TAG, "HTTP: " + method + " " + Utils.redactUrlForNonVerbose(urlString) 116             + (isProxySet ? (" proxy=" + proxyHost + ":" + proxyPort) : "")) 117             + ", PDU size=" + (pdu != null ? pdu.length : 0)); 118         checkMethod(method); 119         HttpURLConnection connection = null; 120         try { 121             Proxy proxy = Proxy.NO_PROXY; 122             if (isProxySet) { 123                 proxy = new Proxy(Proxy.Type.HTTP, new InetSocketAddress(proxyHost, proxyPort)); 124             } 125             final URL url = new URL(urlString); 126             // Now get the connection 127             connection = (HttpURLConnection) url.openConnection(proxy); 128             connection.setDoInput(true); 129             connection.setConnectTimeout( 130                 mmsConfig.getInt(CarrierConfigValuesLoader.CONFIG_HTTP_SOCKET_TIMEOUT, 131                     CarrierConfigValuesLoader.CONFIG_HTTP_SOCKET_TIMEOUT_DEFAULT)); </pre> <p data-bbox="506 1219 1596 1287"> <a href="https://android.googlesource.com/platform/packages/apps/Messaging/+/nougat-mr1-release/src/android/support/v7/mms/MmsHttpClient.java">https://android.googlesource.com/platform/packages/apps/Messaging/+/nougat-mr1-release/src/android/support/v7/mms/MmsHttpClient.java</a> </p>

**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC
	<pre> 132 // ----- COMMON HEADERS ----- 133 // Header: Accept 134 connection.setRequestProperty(HEADER_ACCEPT, HEADER_VALUE_ACCEPT); 135 // Header: Accept-Language 136 connection.setRequestProperty( 137     HEADER_ACCEPT_LANGUAGE, getCurrentAcceptLanguage(Locale.getDefault())); 138 // Header: User-Agent 139 Log.i(MmsService.TAG, "HTTP: User-Agent=" + userAgent); 140 connection.setRequestProperty(HEADER_USER_AGENT, userAgent); 141 // Header: x-wap-profile 142 final String uaProfUrlTagName = mmsConfig.getString( 143     CarrierConfigValuesLoader.CONFIG_UA_PROF_TAG_NAME, 144     CarrierConfigValuesLoader.CONFIG_UA_PROF_TAG_NAME_DEFAULT); 145 if (uaProfUrl != null) { 146     Log.i(MmsService.TAG, "HTTP: UaProfUrl=" + uaProfUrl); 147     connection.setRequestProperty(uaProfUrlTagName, uaProfUrl); 148 } 149 // Add extra headers specified by mms_config.xml's httpparams 150 addExtraHeaders(connection, mmsConfig); 151 // Different stuff for GET and POST 152 if (METHOD_POST.equals(method)) { 153     if (pdu == null    pdu.length &lt; 1) { 154         Log.e(MmsService.TAG, "HTTP: empty pdu"); 155         throw new MmsHttpException(0/*statusCode*/, "Sending empty PDU"); 156     } 157     connection.setDoOutput(true); 158     connection.setRequestMethod(METHOD_POST); 159     if (mmsConfig.getBoolean( 160         CarrierConfigValuesLoader.CONFIG_SUPPORT_HTTP_CHARSET_HEADER, 161         CarrierConfigValuesLoader.CONFIG_SUPPORT_HTTP_CHARSET_HEADER_DEFAULT)) { 162         connection.setRequestProperty(HEADER_CONTENT_TYPE, 163             HEADER_VALUE_CONTENT_TYPE_WITH_CHARSET); 164     } else { 165         connection.setRequestProperty(HEADER_CONTENT_TYPE, 166             HEADER_VALUE_CONTENT_TYPE_WITHOUT_CHARSET); </pre> <p><a href="https://android.googlesource.com/platform/packages/apps/Messaging/+nougat-mr1-release/src/android/support/v7/mms/MmsHttpClient.java">https://android.googlesource.com/platform/packages/apps/Messaging/+nougat-mr1-release/src/android/support/v7/mms/MmsHttpClient.java</a></p>



**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC
	<pre>167     } 168     if (Log.isLoggable(MmsService.TAG, Log.VERBOSE)) { 169         logHttpHeaders(connection.getRequestProperties()); 170     } 171     connection.setFixedLengthStreamingMode(pdu.length); 172     // Sending request body 173     final OutputStream out = 174         new BufferedOutputStream(connection.getOutputStream()); 175     out.write(pdu); 176     out.flush(); 177     out.close(); 178 } else if (METHOD_GET.equals(method)) { 179     if (Log.isLoggable(MmsService.TAG, Log.VERBOSE)) { 180         logHttpHeaders(connection.getRequestProperties()); 181     } 182     connection.setRequestMethod(METHOD_GET); 183 } 184 // Get response 185 final int responseCode = connection.getResponseCode(); 186 final String responseMessage = connection.getResponseMessage(); 187 Log.d(MmsService.TAG, "HTTP: " + responseCode + " " + responseMessage); 188 if (Log.isLoggable(MmsService.TAG, Log.VERBOSE)) { 189     logHttpHeaders(connection.getHeaderFields()); 190 } 191 if (responseCode / 100 != 2) { 192     throw new MmsHttpException(responseCode, responseMessage); 193 } 194 final InputStream in = new BufferedInputStream(connection.getInputStream()); 195 final ByteArrayOutputStream byteOut = new ByteArrayOutputStream(); 196 final byte[] buf = new byte[4096]; 197 int count = 0; 198 while ((count = in.read(buf)) &gt; 0) { 199     byteOut.write(buf, 0, count); 200 } 201 in.close(); 202 final byte[] responseBody = byteOut.toByteArray(); 203 Log.d(MmsService.TAG, "HTTP: response size=" 204     + (responseBody != null ? responseBody.length : 0)); 205 return responseBody;</pre> <p><a href="https://android.goesource.com/platform/packages/apps/Messaging/+nougat-mr1-release/src/android/support/v7/mms/MmsHttpClient.java">https://android.goesource.com/platform/packages/apps/Messaging/+nougat-mr1-release/src/android/support/v7/mms/MmsHttpClient.java</a></p>

**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC
	<pre>38  * Request to send an MMS 39  */ 40  class SendRequest extends MmsRequest { 41      // Max send response PDU size in bytes (exceeding this may cause problem with 42      // system intent delivery). 43      private static final int MAX_SEND_RESPONSE_SIZE = 1000 * 1024; 44 45      private byte[] mPduData; 46 47      SendRequest(final String locationUri, final Uri pduUri, final PendingIntent sentIntent) { 48          super(locationUri, pduUri, sentIntent); 49      } 50 51      @Override 52      protected boolean loadRequest(final Context context, final Bundle mmsConfig) { 53          mPduData = readPduFromContentUri( 54              context, 55              mPduUri, 56              mmsConfig.getInt( 57                  CarrierConfigValuesLoader.CONFIG_MAX_MESSAGE_SIZE, 58                  CarrierConfigValuesLoader.CONFIG_MAX_MESSAGE_SIZE_DEFAULT)); 59          return (mPduData != null); 60      } 61 62      @Override 63      protected boolean transferResponse(final Context context, final Intent fillIn, 64          final byte[] response) { 65          // SendConf pdus are always small and can be included in the intent 66          if (response != null &amp;&amp; fillIn != null) { 67              if (response.length &gt; MAX_SEND_RESPONSE_SIZE) { 68                  // If the response PDU is too large, it won't be able to fit in 69                  // the PendingIntent to be transferred via system IPC. 70                  return false; 71              } 72              fillIn.putExtra(SmsManager.EXTRA_MMS_DATA, response); 73          } 74          return true; 75      } }</pre>

**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC
	<p data-bbox="512 233 1596 302"><a href="https://android.googlesource.com/platform/packages/apps/Messaging/+/nougat-mr1-release/src/android/support/v7/mms/SendRequest.java">https://android.googlesource.com/platform/packages/apps/Messaging/+/nougat-mr1-release/src/android/support/v7/mms/SendRequest.java</a></p> <pre data-bbox="533 354 1738 399">public static LocationRequest create ()</pre> <p data-bbox="512 423 1031 451">Create a location request with default parameters.</p> <p data-bbox="512 480 1640 542">Default parameters are for a block accuracy, slowly updated location. It can then be adjusted as required by the applications before passing to the <a href="#">FusedLocationProviderApi</a>.</p> <p data-bbox="533 565 632 592"><b>Returns</b></p> <ul data-bbox="554 610 814 638" style="list-style-type: none"><li>• a new location request</li></ul> <p data-bbox="512 651 1797 682"><a href="https://developers.google.com/android/reference/com/google/android/gms/location/LocationRequest">https://developers.google.com/android/reference/com/google/android/gms/location/LocationRequest</a></p>

**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC
	<p><code>public static final int PRIORITY_BALANCED_POWER_ACCURACY</code></p> <p>Used with <code>setPriority(int)</code> to request "block" level accuracy.</p> <p>Block level accuracy is considered to be about 100 meter accuracy. Using a coarse accuracy such as this often consumes less power.</p> <p>Constant Value: 102</p> <hr/> <p><code>public static final int PRIORITY_HIGH_ACCURACY</code></p> <p>Used with <code>setPriority(int)</code> to request the most accurate locations available.</p> <p>This will return the finest location available.</p> <p>Constant Value: 100</p> <hr/> <p><code>public static final int PRIORITY_LOW_POWER</code></p> <p>Used with <code>setPriority(int)</code> to request "city" level accuracy.</p> <p>City level accuracy is considered to be about 10km accuracy. Using a coarse accuracy such as this often consumes less power.</p> <p>Constant Value: 104</p> <p><a href="https://developers.google.com/android/reference/com/google/android/gms/location/LocationRequest">https://developers.google.com/android/reference/com/google/android/gms/location/LocationRequest</a></p>

**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC
	<pre data-bbox="533 248 1749 285">public Task&lt;Location&gt; getLastLocation ()</pre> <p data-bbox="525 313 1104 337">Returns the best most recent location currently available.</p> <p data-bbox="525 370 1696 430">If a location is not available, which should happen very rarely, null will be returned. The best accuracy available while respecting the location permissions will be returned.</p> <p data-bbox="525 462 1736 522">This method provides a simplified way to get location. It is particularly well suited for applications that do not require an accurate location and that do not want to maintain extra logic for location updates.</p> <pre data-bbox="533 578 1749 615">public Task&lt;LocationAvailability&gt; getLocationAvailability ()</pre> <p data-bbox="525 646 1692 706">Returns the availability of location data. When <code>isLocationAvailable()</code> returns true, then the location returned by <code>getLastLocation()</code> will be reasonably up to date within the hints specified by the active <code>LocationRequest</code>s.</p> <p data-bbox="525 738 1472 763">If the client isn't connected to Google Play services and the request times out, null is returned.</p> <p data-bbox="525 795 1673 855">Note it's always possible for <code>getLastLocation()</code> to return null even when this method returns true (e.g. location settings were disabled between calls).</p> <p data-bbox="510 865 1900 930"><a href="https://developers.google.com/android/reference/com/google/android/gms/location/FusedLocationProviderClient">https://developers.google.com/android/reference/com/google/android/gms/location/FusedLocationProviderClient</a></p>

**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC						
	<p><code>public Task&lt;Void&gt; requestLocationUpdates (LocationRequest request, LocationCallback callback, Looper looper)</code></p> <p>Requests location updates with a callback on the specified Looper thread.</p> <p>This method is suited for the foreground use cases. For background use cases, the <code>PendingIntent</code> version of the method is recommended, see <code>requestLocationUpdates(LocationRequest, PendingIntent)</code>.</p> <p>Any previous LocationRequests registered on this LocationListener will be replaced.</p> <p>This call will keep the Google Play services connection active, so make sure to call <code>removeLocationUpdates(LocationCallback)</code> when you no longer need it, otherwise you lose the benefits of the automatic connection management.</p> <p>Callbacks for <code>LocationCallback</code> will be made on the specified thread, which must already be a prepared looper thread.</p> <p><b>Parameters</b></p> <table border="1"> <tr> <td><b>request</b></td> <td>The location request for the updates.</td> </tr> <tr> <td><b>callback</b></td> <td>The callback for the location updates.</td> </tr> <tr> <td><b>looper</b></td> <td>The Looper object whose message queue will be used to implement the callback mechanism, or null to make callbacks on the calling thread.</td> </tr> </table> <p><a href="https://developers.google.com/android/reference/com/google/android/gms/location/FusedLocationProviderClient">https://developers.google.com/android/reference/com/google/android/gms/location/FusedLocationProviderClient</a></p>	<b>request</b>	The location request for the updates.	<b>callback</b>	The callback for the location updates.	<b>looper</b>	The Looper object whose message queue will be used to implement the callback mechanism, or null to make callbacks on the calling thread.
<b>request</b>	The location request for the updates.						
<b>callback</b>	The callback for the location updates.						
<b>looper</b>	The Looper object whose message queue will be used to implement the callback mechanism, or null to make callbacks on the calling thread.						

**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC				
	<pre data-bbox="533 240 1740 321">public Task&lt;Void&gt; requestLocationUpdates (LocationRequest request, PendingIntent callbackIntent)</pre> <p data-bbox="525 354 1268 378">Requests location updates with a callback on the specified PendingIntent.</p> <p data-bbox="525 410 1732 540">This method is suited for the background use cases, more specifically for receiving location updates, even when the app has been killed by the system. In order to do so, use a <code>PendingIntent</code> for a started service. For foreground use cases, the <code>LocationCallback</code> version of the method is recommended, see <code>requestLocationUpdates(LocationRequest, LocationCallback, Looper)</code>.</p> <p data-bbox="525 573 1724 630">Any previously registered requests that have the same PendingIntent (as defined by <code>equals(Object)</code>) will be replaced by this request.</p> <p data-bbox="525 662 1728 751">Both <code>LocationResult</code> and <code>LocationAvailability</code> are sent to the given PendingIntent. You can extract data from an Intent using <code>hasResult(Intent)</code>, <code>extractResult(Intent)</code>, <code>hasLocationAvailability(Intent)</code>, and <code>extractLocationAvailability(Intent)</code>.</p> <p data-bbox="543 784 667 808"><b>Parameters</b></p> <table border="1" data-bbox="525 833 1740 971"> <tbody> <tr> <td data-bbox="525 833 835 906"><code>request</code></td> <td data-bbox="835 833 1740 906">The location request for the updates.</td> </tr> <tr> <td data-bbox="525 906 835 971"><code>callbackIntent</code></td> <td data-bbox="835 906 1740 971">A pending intent to be sent for each location update.</td> </tr> </tbody> </table> <p data-bbox="543 995 630 1019"><b>Returns</b></p> <ul data-bbox="554 1044 1358 1068" style="list-style-type: none"> <li>• a Task for the call, check <code>isSuccessful()</code> to determine if it was successful.</li> </ul> <p data-bbox="512 1076 1898 1141"><a href="https://developers.google.com/android/reference/com/google/android/gms/location/FusedLocationProviderClient">https://developers.google.com/android/reference/com/google/android/gms/location/FusedLocationProviderClient</a></p>	<code>request</code>	The location request for the updates.	<code>callbackIntent</code>	A pending intent to be sent for each location update.
<code>request</code>	The location request for the updates.				
<code>callbackIntent</code>	A pending intent to be sent for each location update.				

**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC						
	<p data-bbox="533 245 1738 277"><code>public void onLocationAvailability (LocationAvailability locationAvailability)</code></p> <p data-bbox="533 310 1171 334">Called when there is a change in the availability of location data.</p> <p data-bbox="533 367 1738 561">When <code>isLocationAvailable()</code> returns <code>false</code> you can assume that location will not be returned in <code>onLocationResult(LocationResult)</code> until something changes in the device's settings or environment. Even when <code>isLocationAvailable()</code> returns <code>true</code> the <code>onLocationResult(LocationResult)</code> may not always be called regularly, however the device location is known and both the most recently delivered location and <code>getLastLocation(GoogleApiClient)</code> will be reasonably up to date given the hints specified by the active <code>LocationRequest</code>s.</p> <p data-bbox="533 586 667 610"><b>Parameters</b></p> <table border="1" data-bbox="533 643 1738 708"> <tr> <td data-bbox="533 651 961 708"><code>locationAvailability</code></td> <td data-bbox="968 651 1738 708">The current status of location availability.</td> </tr> </table> <p data-bbox="533 756 1738 789"><code>public void onLocationResult (LocationResult result)</code></p> <p data-bbox="533 821 1052 846">Called when device location information is available.</p> <p data-bbox="533 878 1661 943">The most recent location returned by <code>getLastLocation()</code> is not guaranteed to be immediately fresh, but will be reasonably up to date given the hints specified by the active <code>LocationRequest</code>s.</p> <p data-bbox="533 967 667 992"><b>Parameters</b></p> <table border="1" data-bbox="533 1024 1738 1089"> <tr> <td data-bbox="533 1032 768 1089"><code>result</code></td> <td data-bbox="774 1032 1738 1089">The latest location result available.</td> </tr> </table> <p data-bbox="512 1105 1801 1130"><a href="https://developers.google.com/android/reference/com/google/android/gms/location/LocationCallback">https://developers.google.com/android/reference/com/google/android/gms/location/LocationCallback</a></p> <p data-bbox="533 1146 1738 1179"><code>public abstract void onLocationChanged (Location location)</code></p> <p data-bbox="533 1211 915 1235">Called when the location has changed.</p> <p data-bbox="533 1260 667 1284"><b>Parameters</b></p> <table border="1" data-bbox="533 1317 1738 1382"> <tr> <td data-bbox="533 1325 926 1382"><code>location</code></td> <td data-bbox="932 1325 1738 1382">The updated location.</td> </tr> </table>	<code>locationAvailability</code>	The current status of location availability.	<code>result</code>	The latest location result available.	<code>location</code>	The updated location.
<code>locationAvailability</code>	The current status of location availability.						
<code>result</code>	The latest location result available.						
<code>location</code>	The updated location.						



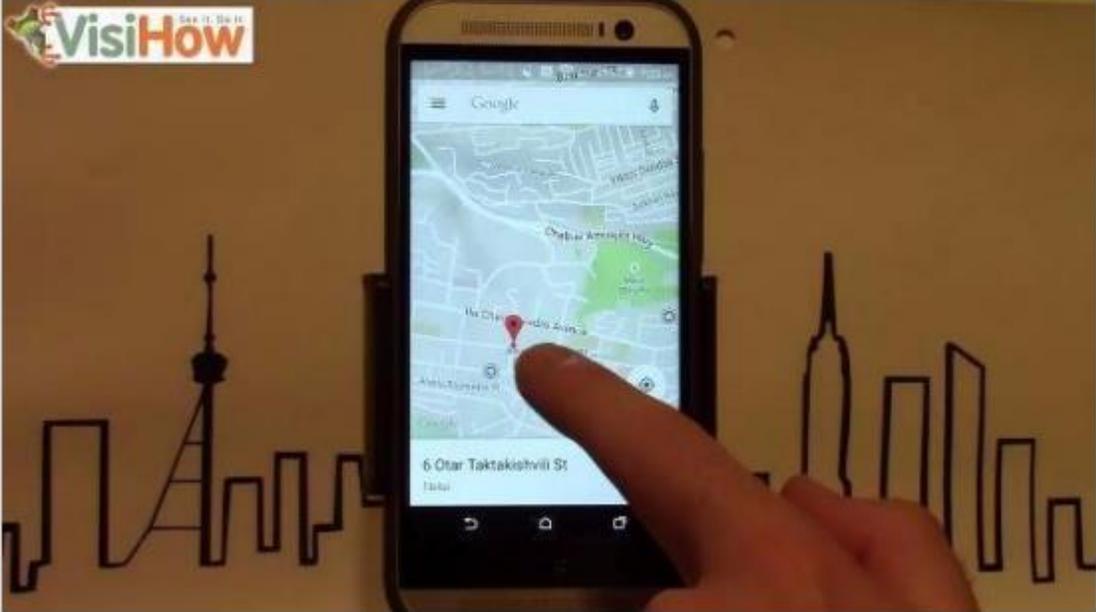
**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC
	<p data-bbox="512 233 1797 264"><a href="https://developers.google.com/android/reference/com/google/android/gms/location/LocationListener">https://developers.google.com/android/reference/com/google/android/gms/location/LocationListener</a></p> <p data-bbox="512 315 814 345">Public Constructors</p> <hr data-bbox="512 358 1743 362"/> <p data-bbox="512 412 926 443">public <b>MapView</b> (<a href="#">Context</a> context)</p> <p data-bbox="512 505 1129 535">public <b>MapView</b> (<a href="#">Context</a> context, <a href="#">AttributeSet</a> attrs)</p> <p data-bbox="512 597 1268 628">public <b>MapView</b> (<a href="#">Context</a> context, <a href="#">AttributeSet</a> attrs, int defStyle)</p> <p data-bbox="512 690 1247 721">public <b>MapView</b> (<a href="#">Context</a> context, <a href="#">GoogleMapOptions</a> options)</p> <p data-bbox="512 748 1671 779"><a href="https://developers.google.com/android/reference/com/google/android/gms/maps/MapView">https://developers.google.com/android/reference/com/google/android/gms/maps/MapView</a></p>

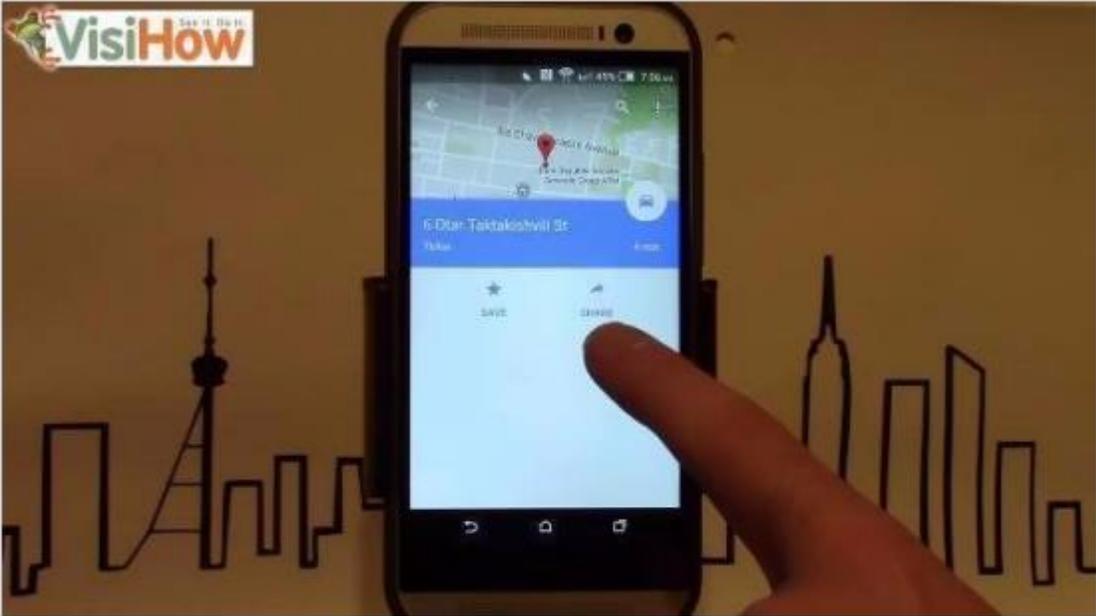
**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC		
	<p><code>public void <b>getMapAsync</b> (<a href="#">OnMapReadyCallback</a> callback)</code></p> <p>Returns a non-null instance of the <a href="#">GoogleMap</a>, ready to be used.</p> <p>Note that:</p> <ul style="list-style-type: none"> <li>• This method must be called from the main thread.</li> <li>• The callback will be executed in the main thread.</li> <li>• In the case where Google Play services is not installed on the user's device, the callback will not be triggered until the user installs it.</li> <li>• The <a href="#">GoogleMap</a> object provided by the callback is non-null.</li> </ul> <p><b>Parameters</b></p> <table border="1" data-bbox="527 688 1738 753"> <tr> <td><code>callback</code></td> <td>The callback object that will be triggered when the map is ready to be used.</td> </tr> </table> <p><code>public final void <b>onCreate</b> (<a href="#">Bundle</a> savedInstanceState)</code></p> <p>You must call this method from the parent Activity/Fragment's corresponding method.</p> <p><a href="https://developers.google.com/android/reference/com/google/android/gms/maps/MapView">https://developers.google.com/android/reference/com/google/android/gms/maps/MapView</a></p>	<code>callback</code>	The callback object that will be triggered when the map is ready to be used.
<code>callback</code>	The callback object that will be triggered when the map is ready to be used.		
<p>[28G] receiving user input via user interaction with the interactive display of the first device, the user input specifying a location and a symbol corresponding to an entity other than the first device and the second devices;</p>	<p>HTC infringes directly and/or indirectly by performing, inducing others to perform, and/or contributing to the performance of receiving user input via user interaction with the interactive display of the first device, the user input specifying a location and a symbol corresponding to an entity other than the first device and the second devices. See claim 1[G], which is incorporated herein by reference in its entirety.</p> <p><b>Share Locations from Google Maps on HTC One M8</b></p>		

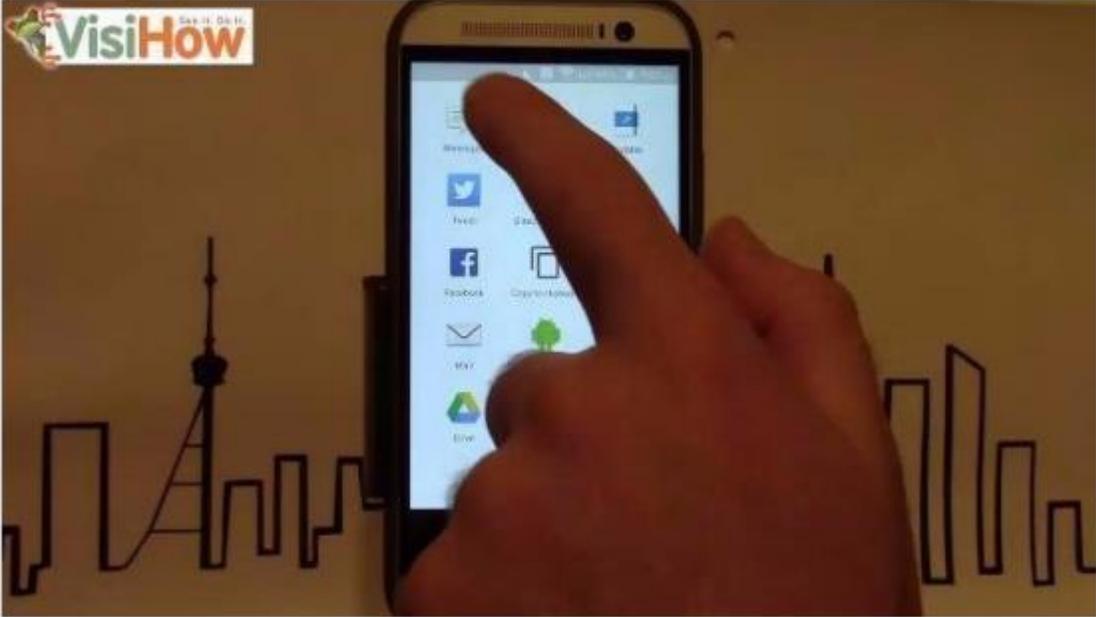
**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC
	<p data-bbox="531 232 829 256"><b>Place a pin on the map.</b></p> <p data-bbox="531 272 1640 378">We can do this anywhere in the entire world, anywhere, on any map that we find in the Google Maps application. We are going to choose a location, press down on the screen, and hold down until we can see that a red pin has been dropped.</p> 

**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC
	<p data-bbox="527 240 1207 305"><b>We have two options:</b> "SAVE" and "SHARE". Press down on the word "SHARE".</p>  <p data-bbox="527 1057 1633 1208"><b>A menu appears with all of the different ways that we can share this location.</b> We can share this location via messages, Bluetooth, Twitter, Facebook, mail, Gmail and Viber. If you have other applications, such as WhatsApp, or any other chat or email applications, you will be able to use them as well and will see them here.</p>

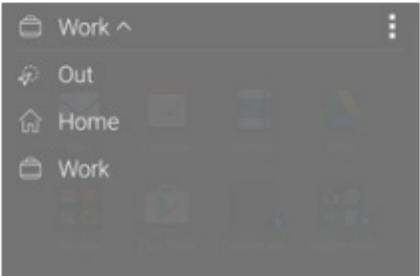
**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC
	<p data-bbox="527 237 856 264"><b>Let's choose "Messages".</b></p> <p data-bbox="527 277 1633 347">This option is located on the top left-hand corner of the screen. All options will have the same basic processes. They will ask us to send the location. Press down on the "Messages" icon.</p> 

**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC
	<p data-bbox="520 233 1260 256"><b>Press the box next to the contact who will be the recipient.</b></p> <p data-bbox="520 266 1549 289">A checkmark will appear in the box. We can select multiple contacts as well if we desire.</p>  <p data-bbox="520 1045 911 1068"><b>Press on the word "DONE (1)".</b></p> <p data-bbox="520 1084 1625 1156">This icon is located in the bottom right-hand corner of the screen. The bracketed number indicates the number of contacts we selected.</p> <p data-bbox="520 1166 1407 1188"><a href="http://visihow.com/Share_Locations_from_Google_Maps_on_HTC_One_M8">http://visihow.com/Share_Locations_from_Google_Maps_on_HTC_One_M8</a></p>

## Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products

US9408055B2	HTC
	<h3 data-bbox="516 261 1115 310">Manually switching locations</h3> <p data-bbox="516 358 1413 467">The HTC Sense Home widget automatically changes locations based on where you are. You can also manually change the location in the HTC Sense Home widget.</p> <p data-bbox="516 505 1402 613">For the HTC Sense Home widget to change locations automatically, you need to make sure that location services is turned on. See <a href="#">Turning location services on or off</a>.</p> <ol data-bbox="583 651 1392 776" style="list-style-type: none"><li>1. On your Home screen, slide right or left until you see the HTC Sense Home widget.</li><li>2. Tap , and then tap the location you want.</li></ol>  <p data-bbox="527 1097 1409 1125">HTC One (M8) - Manually switching locations - Support   HTC United States</p>

**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC
	<h2 data-bbox="552 269 1331 318">Setting your home and work locations</h2> <p data-bbox="552 367 1415 435">In the HTC Sense Home widget, set your home and work locations based on your address, Wi-Fi network, or both.</p> <p data-bbox="552 475 1425 621">You can associate multiple addresses and Wi-Fi networks to each of these locations. Using your set addresses or Wi-Fi networks, the HTC Sense Home widget will be able to determine where you are and display the appropriate apps.</p> <ol data-bbox="621 662 1457 1146" style="list-style-type: none"><li data-bbox="621 662 1423 727">1. On the Home screen, swipe right or left until you see the HTC Sense Home widget.</li><li data-bbox="621 751 972 784">2. Tap  &gt;  &gt; <b>Set locations</b>.</li><li data-bbox="621 816 1062 849">3. Choose the location you want to set.</li><li data-bbox="621 865 1457 1092">4. Tap  and do one of the following:<ul data-bbox="701 930 1457 1092" style="list-style-type: none"><li data-bbox="701 930 1444 995">• Tap <b>Address</b> and then enter your street address or select it on the map.</li><li data-bbox="701 1027 1457 1092">• Tap <b>Wi-Fi network</b> and select one or more Wi-Fi networks you want to associate with the location.</li></ul></li><li data-bbox="621 1117 1457 1146">5. When you've finished setting your home and work locations, press .</li></ol> <p data-bbox="520 1182 1503 1214">HTC One (M8) - Setting your home and work locations - Support   HTC United States</p>



**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC
	<p data-bbox="533 233 1304 272"><b>HTC One V™ – Google Location Service &amp; GPS</b></p> <p data-bbox="533 318 1717 423">Google Maps lets you track your current location, view real-time traffic situations, and receive detailed directions to your destination. It also provides a search tool where you can locate a place of interest or an address on a vector or aerial map, or view locations in street level.</p> <p data-bbox="533 467 905 496"><b>Turning on Location Services</b></p> <div data-bbox="533 509 1717 943"> </div> <ol data-bbox="533 976 1717 1146" style="list-style-type: none"> <li>1. From the Home Screen, slide the <b>Notifications</b> panel open.</li> <li>2. In the top right corner, tap <b>Settings</b>.</li> <li>3. Tap <b>Location</b>.</li> <li>4. Make your selection by tapping <b>Google's location service</b>, <b>Use GPS satellites</b>, or both. <b>Note:</b> You will need to accept the location consent terms and conditions.</li> </ol> <p data-bbox="512 1203 1906 1414"><b>Regarding Google Maps,</b> Google Latitude, Google Plus, Google Hangouts, Google Messages, and Android Messenger, the Accused Products are configured to allow a user of the first device to interact with the display, to select a user, contact, or device, and to select an action to be performed, such as: sending a message, initiating a call, initiating a data conference, sharing a location, stop sharing a location, block a user from location sharing, sending a location, requesting a location, or sending other data. When an action is specified, data is sent from the first device to the second device via a server. In an example, using Google</p>

**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC
	<p>Maps, a user can interact with the display to specify a location that does not correspond to the first or second devices. Alternatively, a user can share an ETA which includes a route, where the route is not the location of any of the first or second devices.</p> <p>Selection with Markers: <a href="https://developers.google.com/maps/documentation/android-api/marker">https://developers.google.com/maps/documentation/android-api/marker</a></p> <p>Queries with GeoTagging database: <a href="https://developers.google.com/android/reference/com/google/android/gms/location/places/GeoDataClient">https://developers.google.com/android/reference/com/google/android/gms/location/places/GeoDataClient</a></p>

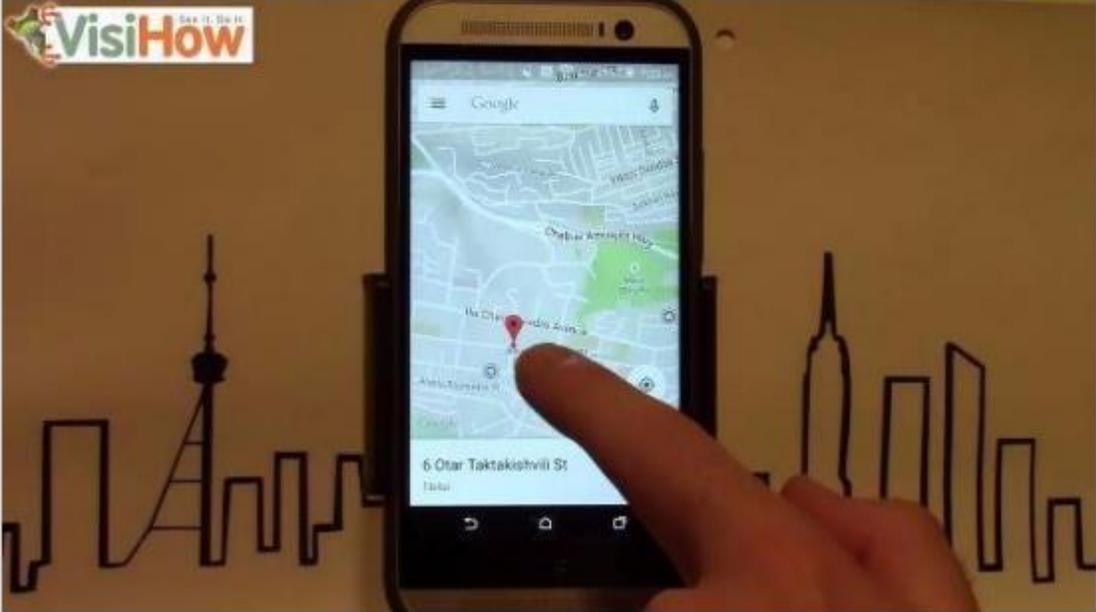
## Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products

US9408055B2	HTC
	<h3 data-bbox="541 245 1255 293">Embed a map or share a location</h3> <p data-bbox="541 313 1516 391">On your computer, you can embed a map, Street View image, driving directions, or search into your website or blog. On your computer, phone, or tablet, you can share a location or map with others over email, Google+, Facebook, Twitter, or text.</p> <p data-bbox="554 467 947 488"><a href="#">ANDROID</a> <a href="#">COMPUTER</a> <a href="#">IPHONE &amp; IPAD</a></p> <hr data-bbox="541 509 1528 513"/> <h3 data-bbox="541 565 848 591">Share a map or location</h3> <ol data-bbox="554 613 1230 773" style="list-style-type: none"><li>1. Open the Google Maps app .</li><li>2. Search for a place. Or find a place on the map then touch and hold to drop a pin.</li><li>3. At the bottom, tap the place's name or address.</li><li>4. Tap Share . If you don't see this, tap More  &gt; <b>Share</b>.</li><li>5. Select an app. It'll send a link that shows the place in Google Maps.</li></ol> <p data-bbox="520 805 1640 831"><a href="https://support.google.com/maps/answer/144361?co=GENIE.Platform%3DAndroid&amp;hl=en">https://support.google.com/maps/answer/144361?co=GENIE.Platform%3DAndroid&amp;hl=en</a></p> <h3 data-bbox="541 898 737 924">Share your E.T.A</h3> <p data-bbox="541 946 1461 967">After you start your drive, you can share your destination, estimated arrival time, and where you are on the route.</p> <ol data-bbox="546 987 1209 1170" style="list-style-type: none"><li>1. Open the Google Maps app .</li><li>2. Set a driving destination. <a href="#">Learn how to navigate to a place.</a></li><li>3. After you start navigation, tap <b>More</b>  &gt; <b>Share trip progress</b>.</li><li>4. Choose a person from the list.</li><li>5. Tap <b>Share</b>.</li><li>6. Location Sharing will stop when you reach your destination or stop navigating.</li></ol> <p data-bbox="541 1190 1083 1211">• To stop sharing before you arrive, tap <b>More</b>  &gt; <b>Stop sharing</b>.</p> <p data-bbox="510 1219 1696 1245"><a href="https://support.google.com/maps/answer/7326816?co=GENIE.Platform%3DAndroid&amp;hl=en">https://support.google.com/maps/answer/7326816?co=GENIE.Platform%3DAndroid&amp;hl=en</a></p> <p data-bbox="510 1292 1520 1318">Markers (adding location information to the link associated with the database):</p> <div data-bbox="510 1360 1915 1396" style="background-color: black; height: 22px;"></div>

**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC
	<pre>static final LatLng PERTH = new LatLng(-31.90, 115.86); Marker perth = mMap.addMarker(new MarkerOptions()     .position(PERTH)     .draggable(true));</pre>
<p>[28H] and based on the user input, adding the user-specified symbol to the interactive display at a position on the interactive map corresponding to the user-specified location, and transmitting the user-specified symbol and location to the second devices for addition of the user-specified symbol to respective interactive displays of the second devices at respective positions on respective interactive maps corresponding to the user-specified location.</p>	<p>HTC infringes directly and/or indirectly by performing, inducing others to perform, and/or contributing to the performance of [] based on the user input, adding the user-specified symbol to the interactive display at a position on the interactive map corresponding to the user-specified location, and transmitting the user-specified symbol and location to the second devices for addition of the user-specified symbol to respective interactive displays of the second devices at respective positions on respective interactive maps corresponding to the user-specified location. See claim 1[H], which is incorporated herein by reference in its entirety.</p> <p>A user can interact with the display to specify a location that does not correspond to the first or second devices. A user can drop a symbol pin on the specified location. A user can then share that location and transmit the location to one or more second devices using Android Messages, Google Hangouts, or another application.</p> <p>Alternatively, a user can share an ETA which includes a route, where the route is not the location of any of the first or second devices. Again, this route can be shared with users over Android Messages, Google Hangouts, or another application.</p> <p><b>Share Locations from Google Maps on HTC One M8</b></p>

**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC
	<p data-bbox="531 233 827 256"><b>Place a pin on the map.</b></p> <p data-bbox="531 272 1640 380">We can do this anywhere in the entire world, anywhere, on any map that we find in the Google Maps application. We are going to choose a location, press down on the screen, and hold down until we can see that a red pin has been dropped.</p>  <p data-bbox="512 1141 743 1164"><b>Placing a Marker:</b></p> <p data-bbox="512 1179 1430 1201"><a href="https://developers.google.com/maps/documentation/android-api/marker">https://developers.google.com/maps/documentation/android-api/marker</a></p> <p data-bbox="512 1252 1083 1274">based on queries with GeoTagging database:</p> <p data-bbox="512 1289 1860 1312"><a href="https://developers.google.com/android/reference/com/google/android/gms/location/places/GeoDataClient">https://developers.google.com/android/reference/com/google/android/gms/location/places/GeoDataClient</a></p>

**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC
	<p><b>Embed a map or share a location</b></p> <p>On your computer, you can embed a map, Street View image, driving directions, or search into your website or blog. On your computer, phone, or tablet, you can share a location or map with others over email, Google+, Facebook, Twitter, or text.</p> <p><b>ANDROID</b> COMPUTER IPHONE &amp; IPAD</p> <hr/> <p><b>Share a map or location</b></p> <ol style="list-style-type: none"> <li>1. Open the Google Maps app .</li> <li>2. Search for a place. Or find a place on the map then touch and hold to drop a pin.</li> <li>3. At the bottom, tap the place's name or address.</li> <li>4. Tap Share . If you don't see this, tap More  &gt; <b>Share</b>.</li> <li>5. Select an app. It'll send a link that shows the place in Google Maps.</li> </ol> <p><a href="https://support.google.com/maps/answer/144361?co=GENIE.Platform%3DAndroid&amp;hl=en">https://support.google.com/maps/answer/144361?co=GENIE.Platform%3DAndroid&amp;hl=en</a></p> <p><b>Share your E.T.A</b></p> <p>After you start your drive, you can share your destination, estimated arrival time, and where you are on the route.</p> <ol style="list-style-type: none"> <li>1. Open the Google Maps app .</li> <li>2. Set a driving destination. <a href="#">Learn how to navigate to a place.</a></li> <li>3. After you start navigation, tap <b>More</b>  &gt; <b>Share trip progress</b>.</li> <li>4. Choose a person from the list.</li> <li>5. Tap <b>Share</b>.</li> <li>6. Location Sharing will stop when you reach your destination or stop navigating.</li> </ol> <ul style="list-style-type: none"> <li>• To stop sharing before you arrive, tap <b>More</b>  &gt; <b>Stop sharing</b>.</li> </ul>
<p>29[A]. The system of claim 28 wherein the operations further</p>	<p>HTC infringes directly and/or indirectly by performing, inducing others to perform, and/or contributing to the performance of wherein the operations further comprise: presenting another symbol on the interactive map corresponding to a fixed location and associated with a telephone number.</p>

**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC
comprise: presenting another symbol on the interactive map corresponding to a fixed location and associated with a telephone number;	See claims 2[A] and 28, which are incorporated herein by reference in their entirety.
[29B] and receiving user selection of the other symbol and, based thereon, initiating a telephone call to the telephone number associated with the symbol.	HTC infringes directly and/or indirectly by performing, inducing others to perform, and/or contributing to the performance of [ ] and receiving user selection of the other symbol and, based thereon, initiating a telephone call to the telephone number associated with the symbol. See claims 2[B] and 28, which are incorporated by reference in their entirety.
30. The system of claim 28 wherein the data comprises a text message, an image, a video, or a command to cause the second devices corresponding to the selected symbols to convert text to speech.	HTC infringes directly and/or indirectly by performing, inducing others to perform, and/or contributing to the performance of wherein the data comprises a text message, an image, a video, or a command to cause the second devices corresponding to the selected symbols to convert text to speech. See claims 3 and 28, which are incorporated by reference in their entirety.
31[A]. The system of claim 28 wherein: the SMS messages include an Internet Protocol (IP) address of the first device;	HTC infringes directly and/or indirectly by performing, inducing others to perform, and/or contributing to the performance of wherein: the SMS messages include an Internet Protocol (IP) address of the first device. See claims 4[A] and 28, which are incorporated by reference in their entirety.
[31B] and the IP-based	HTC infringes directly and/or indirectly by performing, inducing others to perform, and/or contributing to

**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC
responses include respective IP addresses of the second devices.	the performance of [] the IP-based responses include respective IP addresses of the second devices. See claims 4[B] and 28, which are incorporated by reference in their entirety.
32. The system of claim 28 wherein the operations further comprise: transmitting location information including an updated location of the first device to the second devices based on displacement of the first device by at least a predetermined distance relative to a previous location of the first device, passage of at least a predetermined time interval since transmitting information including a location of the first device, or a combination of the displacement of the first device and the passage of time.	HTC infringes directly and/or indirectly by performing, inducing others to perform, and/or contributing to the performance of wherein the operations further comprise: transmitting location information including an updated location of the first device to the second devices based on displacement of the first device by at least a predetermined distance relative to a previous location of the first device, passage of at least a predetermined time interval since transmitting information including a location of the first device, or a combination of the displacement of the first device and the passage of time. See claims 5 and 28, which are incorporated by reference in their entirety.
33[A]. The system of claim 28 wherein the operations further	HTC infringes directly and/or indirectly by performing, inducing others to perform, and/or contributing to the performance of wherein the operations further comprise: receiving second user selection of one or more of the symbols corresponding to one or more of the second devices. See claims 6[A] and 28, which are



**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC
comprise: receiving second user selection of one or more of the symbols corresponding to one or more of the second devices;	incorporated by reference in their entirety.
[33B] and receiving user input assigning the one or more second devices corresponding to the second selected one or more symbols to a sub-net.	HTC infringes directly and/or indirectly by performing, inducing others to perform, and/or contributing to the performance of [] receiving user input assigning the one or more second devices corresponding to the second selected one or more symbols to a sub-net. See claims 6[B] and 28, which are incorporated by reference in their entirety.
34[A]. The system of claim 33 wherein the operations further comprise: receiving user selection of the sub-net;	HTC infringes directly and/or indirectly by performing, inducing others to perform, and/or contributing to the performance of wherein the operations further comprise: receiving user selection of the sub-net. See claims 7[A], 33 and 28, which are incorporated by reference in their entirety.
[34B] and establishing a conference among the one or more second devices of the sub-net for sharing voice, text, photographs, or video communications.	HTC infringes directly and/or indirectly by performing, inducing others to perform, and/or contributing to the performance of [] establishing a conference among the one or more second devices of the sub-net for sharing voice, text, photographs, or video communications. See claims 7[B], 33, and 28, which are incorporated by reference in their entirety.
35. The system of claim 28, wherein the first device is a	HTC infringes directly and/or indirectly by performing, inducing others to perform, and/or contributing to the performance of wherein the first device is a cellular phone or a personal digital assistant (PDA). See claims 8 and 28, which are incorporated by reference in their entirety.

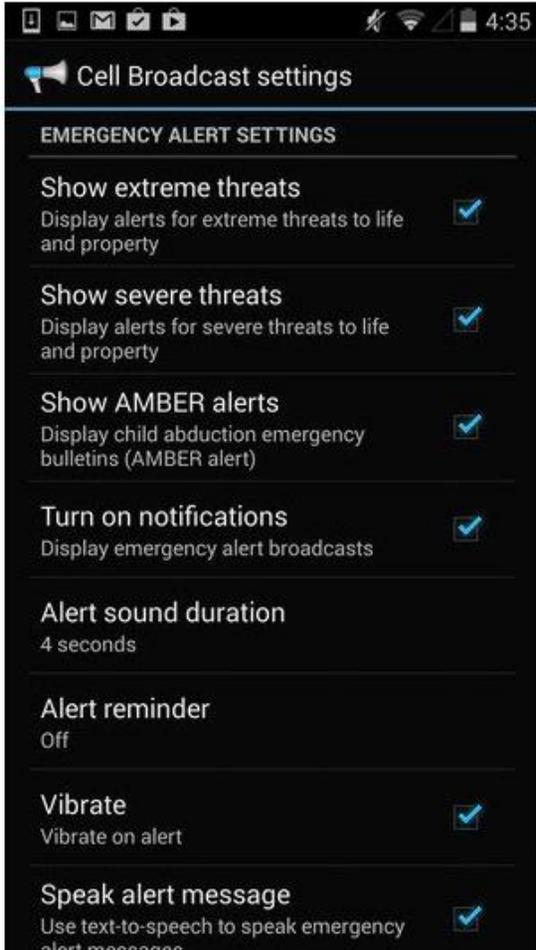
**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC
cellular phone or a personal digital assistant (PDA).	
36. The system of claim 28, wherein the operations further comprise: identifying second user interaction with the interactive display selecting at least one of the user-selectable symbols corresponding to at least one of the second devices and user interaction with the display specifying an action and, based thereon, initiating a phone call or phone conference with the at least one second device.	HTC infringes directly and/or indirectly by performing, inducing others to perform, and/or contributing to the performance of wherein the operations further comprise: identifying second user interaction with the interactive display selecting at least one of the user-selectable symbols corresponding to at least one of the second devices and user interaction with the display specifying an action and, based thereon, initiating a phone call or phone conference with the at least one second device. See claims 9 and 28, which are incorporated by reference in their entirety.
37. The system of claim 30, wherein the video comprises a video clip or a video transmission.	HTC infringes directly and/or indirectly by performing, inducing others to perform, and/or contributing to the performance of wherein the video comprises a video clip or a video transmission. See claims 10, 30, and 28, which are incorporated by reference in their entirety.
38[A]. The system of claim 33, wherein the operations further comprise: receiving	HTC infringes directly and/or indirectly by performing, inducing others to perform, and/or contributing to the performance of wherein the operations further comprise: receiving user selection of the sub-net. See claims 11[A], 33, and 28, which are incorporated by reference in their entirety.

**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC
user selection of the sub-net;	
[38B] and causing the one or more second devices of the sub-net to place a call, make a verbal announcement, convert text to speech, vibrate, or increase sound levels.	HTC infringes directly and/or indirectly by performing, inducing others to perform, and/or contributing to the performance of [] causing the one or more second devices of the sub-net to place a call, make a verbal announcement, convert text to speech, vibrate, or increase sound levels. See claims 11[B], 33, and 28, which are incorporated by reference in their entirety.
39. The system of claim 28, wherein the data sent to the one or more second devices causes at least one of the second devices to play an audio message announcing an emergency.	<p>HTC infringes directly and/or indirectly by performing, inducing others to perform, and/or contributing to the performance of wherein the data sent to the one or more second devices causes at least one of the second devices to play an audio message announcing an emergency. See claim 28, which is incorporated herein by reference in its entirety.</p> <p>A user can send an audio message or a video message that includes to announce an emergency to the one or more second devices.</p>

**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC
	 <p>The screenshot shows the 'Cell Broadcast settings' screen on an HTC device. Under the 'EMERGENCY ALERT SETTINGS' section, the following options are checked: 'Show extreme threats', 'Show severe threats', 'Show AMBER alerts', 'Turn on notifications', 'Vibrate', and 'Speak alert message'. The 'Alert sound duration' is set to 4 seconds and 'Alert reminder' is set to Off.</p> <p><a href="https://www.greenbot.com/article/2689993/how-to-change-settings-for-emergency-alerts-on-android-phones.html">https://www.greenbot.com/article/2689993/how-to-change-settings-for-emergency-alerts-on-android-phones.html</a></p>
<p>40. The system of claim 28, wherein the data sent to the one or more second devices</p>	<p>HTC infringes directly and/or indirectly by performing, inducing others to perform, and/or contributing to the performance of wherein the data sent to the one or more second devices causes at least one of the second devices to place a phone call to the first device. See claims 23[A] and 28, which are incorporated by reference in their entirety.</p>

**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC
causes at least one of the second devices to place a phone call to the first device.	
41[P] A non-transitory storage device having instructions stored thereon that, when executed by a first device, cause the first device to perform operations comprising:	<p>HTC infringes directly and/or indirectly by performing, inducing others to perform, and/or contributing to the performance of this this non-transitory storage device having instructions stored thereon that, when executed by a first device, cause the first device to perform operations as set forth below. See claims 1[P] and 28[P], which are incorporated herein by reference in their entirety.</p> <p>The Accused Products meet the claim limitations by providing device-location tracking features such as those features described below. For example, the Accused Products meet the claim limitations because they are pre-installed with Android mobile operating systems containing code for providing device-location tracking features as provided in the claims limitations herein. For example, the Accused Products run applications and/or software that run within the Android mobile operating system and that use components of the Android mobile operating system to provide device-location tracking features. Upon information and belief, in addition to the components and features of the Android mobile operating system itself, the following applications and/or software run within the Android mobile operating system and use components of the Android mobile operating system to provide device-location tracking features: Google Latitude, Google Plus, Google Hangouts (including Allo and Duo), Google Maps, Google Chrome, Google Messages, and Android Messenger.</p> <p><b><u>Google Maps Share Location</u></b></p> <p>Share Location is currently included as a standard feature on the Accused Devices operating as a feature of Google Maps. Google Maps is a pre-installed software application in Android OS. The Accused Devices have included the Share Location functionalities since 2009 as part of Google Latitude, which was an opt-in feature for Google Maps on Android OS-based mobile devices, such as the Accused Products. Share Location functionalities were briefly shifted from Latitude for Google Maps to Google Plus and Google Hangouts, until reappearing as a standard feature in Google Maps. Upon information and belief, the Share Location method also uses and/or works in conjunction with functionalities associated with Google Maps, Google Messages, Android Messenger, Location Access, and other features, which are pre-installed on the</p>

**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC
	<p>Accused Products. For the purposes of these contentions, AGIS sets forth Google Maps' Share Location feature of the Accused Products as representative of this exemplary software. AGIS reserves the right to supplement these contentions to the extent that defendant requires additional information in accordance with P.R. 3-1 and for any other reason.</p> <p><i>See, e.g.,</i> <a href="https://techcrunch.com/2017/03/22/google-maps-now-lets-you-share-your-location-with-friends-and-family-for-a-specific-period-of-time/">https://techcrunch.com/2017/03/22/google-maps-now-lets-you-share-your-location-with-friends-and-family-for-a-specific-period-of-time/</a>; <a href="https://googleblog.blogspot.com/2009/02/see-where-your-friends-are-with-google.html">https://googleblog.blogspot.com/2009/02/see-where-your-friends-are-with-google.html</a>; <a href="https://googleblog.blogspot.jp/2013/07/a-new-google-maps-app-for-smartphones.html">https://googleblog.blogspot.jp/2013/07/a-new-google-maps-app-for-smartphones.html</a>; <a href="http://googleplusproject.blogspot.com/2013/05/google-for-android-42.html">http://googleplusproject.blogspot.com/2013/05/google-for-android-42.html</a>; <a href="https://googleblog.blogspot.com/2013/10/google-hangouts-and-photos-save-some.html">https://googleblog.blogspot.com/2013/10/google-hangouts-and-photos-save-some.html</a></p> <p><b>Control within reach, even when your device isn't</b></p> <p>One of the biggest security risks you're likely to face is simply losing your phone. To help in these times of need, we're launching <a href="#">Find My Device</a> as part of Google Play Protect. With Find My Device you can locate, ring, lock and erase your Android devices—phones, tablets, and even watches. This feature is built in and enabled on all devices; visit <a href="http://android.com/find">android.com/find</a> or check out <a href="#">the app</a>.</p> <p><i>See, e.g.,</i> <a href="https://www.blog.google/products/android/google-play-protect/">https://www.blog.google/products/android/google-play-protect/</a></p>

**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC
	<p data-bbox="537 245 1312 345"><b>Find your device using Android Device Manager</b></p> <p data-bbox="537 363 1491 410">If you've lost a device, you can use Android Device Manager to find its approximate location on a map and when it was last used. When Android Device Manager locates your device, that device will get a notification.</p> <p data-bbox="537 435 1482 508"><b>Before you can use Android Device Manager to locate your device:</b> Your device's <a href="#">location access</a> need to be turned on <a href="#">↗</a> and be signed in to your Google Account. Android Device Manager won't work for devices that are turned off or that don't have a mobile data or Wi-Fi connection.</p> <p data-bbox="537 532 1461 553"><b>Tip:</b> If you've linked your phone to Google, you can locate or ring it by searching for <a href="#">find my phone</a> on <a href="#">google.com</a> <a href="#">↗</a>.</p> <p data-bbox="525 578 1192 609"><b><a href="https://support.google.com/pixelphone/answer/6160491">https://support.google.com/pixelphone/answer/6160491</a></b></p> <p data-bbox="525 633 913 664"><b>Link your phone to Google</b></p> <p data-bbox="525 680 1201 716">You can connect your Android phone to Google, which lets you send information from your computer to your phone. For example, you can send directions you searched for on your computer to Google Maps on your phone.</p> <p data-bbox="525 740 739 761"><b>Link your Android phone</b></p> <p data-bbox="525 786 739 807"><b>Step 1: Update the Google app</b></p> <p data-bbox="525 812 884 849">1. On your phone, go to the <a href="#">Google app page on the Play Store</a>. 2. Tap <b>Update</b>.</p> <p data-bbox="525 873 722 894"><b>Step 2: Turn on Google Now</b></p> <p data-bbox="525 899 852 979">1. On your phone, open the Google app . 2. At the top left, tap Menu  &gt; <b>Settings</b> &gt; <b>Now cards</b>. 3. Turn on <b>Show cards</b>. 4. Turn on <b>Show notifications</b>.</p> <p data-bbox="525 1003 772 1024"><b>Step 3: Turn on Web &amp; App Activity</b></p> <p data-bbox="525 1029 743 1066">1. Visit the <a href="#">Account History page</a>. 2. Make sure the switch is on (green).</p> <p data-bbox="525 1091 743 1112"><b>Step 4: Sign in to your browser</b></p> <p data-bbox="525 1117 978 1240">1. On your phone, open the Google app . 2. At the top left, tap the Menu . 3. At the top left, you'll see the email address you use for the Google app. 4. Visit <a href="http://www.google.com">www.google.com</a> <a href="#">↗</a> on your computer. 5. If you aren't signed in already, click <b>Sign in</b> in the top right corner of the page. 6. Sign in using the Google Account you use for the Google app.</p> <p data-bbox="525 1265 802 1286"><b>Step 5: Send information to your phone</b></p> <p data-bbox="525 1291 1209 1344">1. Do one of the searches below, like <b>note to self</b>, or <b>send directions to my phone</b>. 2. If a box doesn't pop up with the option to send information to your phone, try refreshing the page. If you just turned on Google Now, it may take a few minutes for the box to show up</p> <p data-bbox="512 1352 1218 1383"><b><a href="https://support.google.com/websearch/answer/6128427">https://support.google.com/websearch/answer/6128427</a></b></p>

**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC
	<p data-bbox="520 237 1010 261"><b>What you can do once your phone is linked</b></p> <hr/> <p data-bbox="537 293 667 315"><a href="#">Find my phone</a> </p> <p data-bbox="562 334 1041 355">You can get the current location of your phone if you can't find it.</p> <ol data-bbox="573 371 1388 469" style="list-style-type: none"><li>1. On your computer's browser, search on <a href="http://www.google.com">www.google.com</a>  for <b>find my phone</b>.</li><li>2. If your phone is turned on and connected to the Internet, you'll see your phone's location.</li><li>3. If your phone's location is unavailable, you can still make it ring for 5 minutes on full volume by clicking <b>Ring</b>. You can stop the ringing from your phone when you find it.</li></ol> <p data-bbox="562 487 1388 529"><b>Tip:</b> You can also find your missing phone using the <a href="#">Android Device manager</a>  which lets you find your device or remotely ring, lock, or erase it.</p> <hr/> <p data-bbox="537 586 785 607"><a href="#">Send directions to my phone</a> </p> <p data-bbox="562 626 1388 669">Once you've looked up directions on your computer, you can send them to your phone so you have them on your trip.</p> <ol data-bbox="573 686 1314 789" style="list-style-type: none"><li>1. On your computer's browser, search on <a href="http://www.google.com">www.google.com</a>  for <b>send directions to my phone</b>.</li><li>2. Enter in your destination.</li><li>3. Click <b>Send directions to your phone</b>.</li><li>4. You'll get a notification on your phone. Tap to navigate to your destination using Google Maps.</li></ol> <hr/> <p data-bbox="537 846 753 867"><a href="#">Send a note to my phone</a> </p> <ol data-bbox="573 886 1367 989" style="list-style-type: none"><li>1. On your computer's browser, search on <a href="http://www.google.com">www.google.com</a>  for <b>send a note to my phone</b>.</li><li>2. Type your note in the box.</li><li>3. Click <b>Send note to your phone</b>.</li><li>4. You'll get a notification on your phone with your note that you can either save to one of your apps or copy.</li></ol> <hr/> <p data-bbox="537 1045 651 1066"><a href="#">Set an alarm</a> </p> <ol data-bbox="573 1086 1178 1188" style="list-style-type: none"><li>1. On your computer's browser, search on <a href="http://www.google.com">www.google.com</a>  for <b>set an alarm</b>.</li><li>2. Choose the time you want the alarm to go off.</li><li>3. Click <b>Set an alarm on your phone</b>.</li><li>4. An alarm will now be set on your phone's Clock app.</li></ol> <hr/> <p data-bbox="537 1245 667 1266"><a href="#">Set a reminder</a> </p> <ol data-bbox="573 1286 1335 1362" style="list-style-type: none"><li>1. On your computer's browser, search on <a href="http://www.google.com">www.google.com</a>  for <b>set an reminder</b>.</li><li>2. Type what you want to be reminded about, and either when or where you want the reminder to go off.</li><li>3. Click <b>Remind me on my devices</b>.</li></ol> <hr/> <p data-bbox="512 1369 1220 1398"><a href="https://support.google.com/websearch/answer/6128427">https://support.google.com/websearch/answer/6128427</a></p>



**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC
	<p data-bbox="548 233 1325 277"><b>Share your location using Google Maps</b></p> <p data-bbox="548 297 1465 342">You can't share your location in Google+ anymore. If you used to share your location in Google+ and want to keep sharing it, you'll need to share it again in Google Maps.</p> <p data-bbox="520 358 1633 386"><a href="https://support.google.com/plus/answer/3302509?hl=en&amp;co=GENIE.Platform%3DAndroid&amp;oco=1">https://support.google.com/plus/answer/3302509?hl=en&amp;co=GENIE.Platform%3DAndroid&amp;oco=1</a></p> <p data-bbox="533 402 625 427"><b>Location</b></p> <p data-bbox="533 443 1409 537">Turn on location service, your phone determines your approximate location using Wi-Fi and mobile networks. When you select this option, you're asked whether you consent to allowing Google to use your location when providing these services.</p> <ul data-bbox="562 548 1367 760" style="list-style-type: none"><li>• <b>Mode</b> – Sets the how your current location information is determined.</li><li>• <b>Recent Location Request</b> – Displays applications and services that have recently requested your location information.</li><li>• <b>Camera</b> – Checkmark to tag photos or videos with their locations.</li><li>• <b>Google Location History</b> – Allows you to view and manage your Google location history.</li></ul> <p data-bbox="533 776 716 800"><b>Accounts &amp; sync</b></p> <p data-bbox="533 816 1402 943">Use the Accounts &amp; sync settings menu to add, remove, and manage your Google and other supported accounts. You also use these settings to control how and whether all applications send, receive, and sync data on their own schedules and whether all applications can synchronize user data automatically.</p> <p data-bbox="533 959 1373 1053">Gmail™, Calendar, and other applications may also have their own settings to control how they synchronize data; see the sections on those applications for details. Touch <b>Add account</b> to add new account.</p> <p data-bbox="533 1057 653 1073">-</p>

**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC
	 <p data-bbox="520 852 1486 880"><a href="http://googlemobile.blogspot.com/2009/02/locate-your-friends-in-real-time-with.html">http://googlemobile.blogspot.com/2009/02/locate-your-friends-in-real-time-with.html</a></p> <p data-bbox="520 894 1871 964">Google's location-sharing feature also appeared in Google+, Google Trust Contacts, and Google Hangouts services until its current integration in Google Maps.</p> <p data-bbox="520 1003 1892 1182">HTC makes, uses, sells, and otherwise provides this first device by making, using, selling, and importing Android devices such as HTC mobile devices, HTC tablets, and HTC Smartwatches as well as by providing its servers or using third party servers (e.g., Google servers) for use with Android devices to enable features such as Maps. Below are example HTC Android devices that perform each step of this method as set forth below.</p>

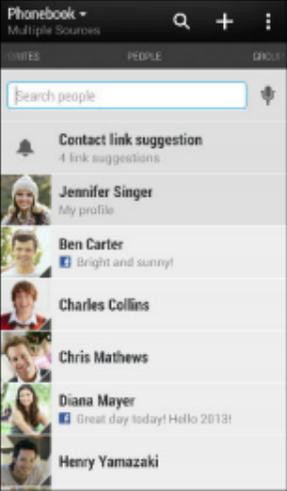
# Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products

US9408055B2	HTC
	<p>Sort by: <a href="#">Popularity</a> <a href="#">Date</a> <a href="#">Price</a></p> <p>HTC LG Samsung Motorola Fly Sony-Ericsson Apple Nokia Mobiado Vertu BenQ-Siemens Sagem Alcatel Philips</p> <p><a href="#">All brands</a></p> <div data-bbox="814 298 1050 479"><p><b>HTC Desire 816</b></p><p>Mobile phone 2014 year Touchscreen: 720 x 1280 5.5 inch. Android 4.4</p></div> <div data-bbox="535 576 777 755"><p><b>HTC One M8</b></p><p>Mobile phone 2014 year Touchscreen: 1080 x 1920 5 inch. Android 4.4</p></div> <div data-bbox="814 576 1050 755"><p><b>HTC Desire 300</b></p><p>Mobile phone 2013 year Touchscreen: 480 x 800 4.3 inch. Android 4.2</p></div> <div data-bbox="535 852 777 1031"><p><b>HTC Desire 601</b></p><p>Mobile phone 2013 year Touchscreen: 540 x 960 4.5 inch. Android 4.4</p></div> <div data-bbox="814 852 1050 1031"><p><b>HTC Desire 700</b></p><p>Mobile phone 2013 year Touchscreen: 540 x 960 5 inch. Android 4.2</p></div> <div data-bbox="535 1128 777 1234"><p><b>HTC Desire 400 Dual Sim</b></p><p>Mobile phone 2013 year Touchscreen: 480 x 800 4.3 inch. Android 4.1</p></div> <div data-bbox="814 1128 1050 1234"><p><b>HTC One Max</b></p><p>Mobile phone 2013 year Touchscreen: 1080 x 1920 5.9 inch. Android 4.3</p></div> <p><a href="https://mob.org/phone/htc/page_3/sort_date_down/">https://mob.org/phone/htc/page_3/sort_date_down/</a></p>

**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

<b>US9408055B2</b>	<b>HTC</b>
41[A ] obtaining contact information of a plurality of second devices, wherein the contact information comprises respective telephone numbers of the second devices;	<p>HTC infringes directly and/or indirectly by performing, inducing others to perform, and/or contributing to the performance of obtaining contact information of a plurality of second devices, wherein the contact information comprises respective telephone numbers of the second devices. See claims 1[A] and 28[A], which are incorporated herein by reference in their entirety.</p> <p>For example, the Accused products include a contacts app to access contact information for second users using respective second devices.</p>

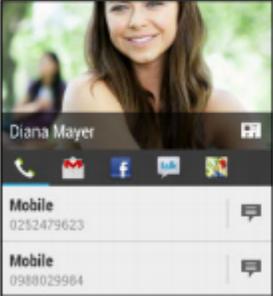
**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC
	<h2 data-bbox="562 233 934 272">Your contacts list</h2> <p data-bbox="562 310 1560 402">The Contacts app lists all contacts you've stored on HTC One and from online accounts you're logged in to. Use the Contacts app to easily manage communications with people that matter to you.</p> <ol data-bbox="590 431 909 456" style="list-style-type: none"><li data-bbox="590 431 909 456">1. Open the Contacts app.</li></ol>  <p data-bbox="590 1000 982 1024">2. On your contacts list, you can:</p> <ul data-bbox="653 1049 1560 1292" style="list-style-type: none"><li data-bbox="653 1049 1287 1073">▪ View your profile and edit your contact information.</li><li data-bbox="653 1084 1087 1109">▪ Create, edit, find, or send contacts.</li><li data-bbox="653 1120 909 1144">▪ See status updates.</li><li data-bbox="653 1156 1497 1180">▪ Tap a contact photo to find ways to quickly connect with the contact.</li><li data-bbox="653 1192 1461 1216">▪ See a notification icon when a contact has sent you new messages.</li><li data-bbox="653 1227 1560 1292">▪ Check out who's online in Google Talk™. Online status icons are displayed if you're signed in to Google Talk.</li></ul> <p data-bbox="520 1328 1560 1385"> To sort your contacts by their first or last name, tap  &gt; Settings &gt; Sort contact list.</p>

**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC
	<p data-bbox="529 245 926 277"><b>Filtering your contacts list</b></p> <p data-bbox="529 302 1575 329">When your contacts list gets long, you can choose which contact accounts to show.</p> <ol data-bbox="562 358 1444 488" style="list-style-type: none"><li>1. On the Contacts tab, tap ▼.</li><li>2. Choose the accounts that contain the contacts you want to display.</li><li>3. Press &lt;.</li></ol> <p data-bbox="529 529 758 561"><b>Finding people</b></p> <p data-bbox="529 586 1516 646">Search for contacts stored on HTC One, your company directory if you have an Exchange ActiveSync account, or social networks you've signed into.</p> <ol data-bbox="562 678 1606 1097" style="list-style-type: none"><li>1. Open the Contacts app.</li><li>2. On the Contacts tab, you can:<ul data-bbox="625 781 1606 1097" style="list-style-type: none"><li>▪ Find people in your contacts list. Tap the Search people box, and then enter the first few letters of the contact name.</li><li>▪ Find people on your company directory. Tap the Search people box, enter the first few letters of the contact name, and then tap Search contacts in your Company Directory.</li><li>▪ Search for people you know on your social networks. Tap ☰ &gt; Settings &gt; Find people you know on, and then select the social networks you're signed in to. The Contacts app then uploads your contacts to the selected social networks to help you find friends.</li></ul></li></ol> <p data-bbox="529 1138 1591 1235">) Aside from searching for a contact by name, you can search using a contact's email address or company name. On the Contacts tab, tap ☰ &gt; Settings &gt; Search contacts by, and then choose a search criteria.</p>

**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC
	<p data-bbox="569 240 1283 282"><b>Getting in touch with a contact</b></p> <ol data-bbox="600 329 1625 440" style="list-style-type: none"><li data-bbox="600 329 940 354">1. Open the Contacts app.</li><li data-bbox="600 378 1625 440">2. Tap a contact's photo (not the name), and then choose how you want to get in touch with that contact.</li></ol>  <p data-bbox="516 789 1646 883"> For more ways of getting in touch with your contact, tap an icon below the contact photo.</p> <p data-bbox="516 927 1856 992">In other examples, the Accused products run Android Messages and Google Hangouts which both access contact information for second users using respective second devices.</p>

**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC
	<h2 data-bbox="527 240 989 289">Contacts Provider</h2> <p data-bbox="527 318 1415 545">The Contacts Provider is a powerful and flexible Android component that manages the device's central repository of data about people. The Contacts Provider is the source of data you see in the device's contacts application, and you can also access its data in your own application and transfer data between the device and online services. The provider accommodates a wide range of data sources and tries to manage as much data as possible for each person, with the result that its organization is complex. Because of this, the provider's API includes an extensive set of contract classes and interfaces that facilitate both data retrieval and modification.</p> <p data-bbox="527 574 816 594">This guide describes the following:</p> <ul data-bbox="527 618 1325 773" style="list-style-type: none"><li data-bbox="527 618 789 638">• The basic provider structure.</li><li data-bbox="527 662 873 682">• How to retrieve data from the provider.</li><li data-bbox="527 706 842 725">• How to modify data in the provider.</li><li data-bbox="527 750 1325 769">• How to write a sync adapter for synchronizing data from your server to the Contacts Provider.</li></ul> <p data-bbox="512 786 1488 813"><a href="https://developer.android.com/guide/topics/providers/contacts-provider.html">https://developer.android.com/guide/topics/providers/contacts-provider.html</a></p>



**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC
	<p><b>Overview</b></p> <p>ContactsContract defines an extensible database of contact-related information. Contact information is stored in a three-tier data model:</p> <ul style="list-style-type: none"> <li>• A row in the <code>ContactsContract.Data</code> table can store any kind of personal data, such as a phone number or email addresses. The set of data kinds that can be stored in this table is open-ended. There is a predefined set of common kinds, but any application can add its own data kinds.</li> <li>• A row in the <code>ContactsContract.RawContacts</code> table represents a set of data describing a person and associated with a single account (for example, one of the user's Gmail accounts).</li> <li>• A row in the <code>ContactsContract.Contacts</code> table represents an aggregate of one or more RawContacts presumably describing the same person. When data in or associated with the RawContacts table is changed, the affected aggregate contacts are updated as necessary.</li> </ul> <p>Other tables include:</p> <ul style="list-style-type: none"> <li>• <code>ContactsContract.Groups</code>, which contains information about raw contact groups such as Gmail contact groups. The current API does not support the notion of groups spanning multiple accounts.</li> <li>• <code>ContactsContract.StatusUpdates</code>, which contains social status updates including IM availability.</li> <li>• <code>ContactsContract.AggregationExceptions</code>, which is used for manual aggregation and disaggregation of raw contacts</li> <li>• <code>ContactsContract.Settings</code>, which contains visibility and sync settings for accounts and groups.</li> <li>• <code>ContactsContract.SyncState</code>, which contains free-form data maintained on behalf of sync adapters</li> <li>• <code>ContactsContract.PhoneLookup</code>, which is used for quick caller-ID lookup</li> </ul> <p><a href="https://developer.android.com/reference/android/provider/ContactsContract.html">https://developer.android.com/reference/android/provider/ContactsContract.html</a></p> <p><b>Data</b></p> <p>As noted previously, the data for a raw contact is stored in a <code>ContactsContract.Data</code> row that is linked to the raw contact's <code>_ID</code> value. This allows a single raw contact to have multiple instances of the same type of data such as email addresses or phone numbers. For example, if "Thomas Higginson" for <code>emilyd@gmail.com</code> (the raw contact row for Thomas Higginson associated with the Google account <code>emilyd@gmail.com</code>) has a home email address of <code>thigg@gmail.com</code> and a work email address of <code>thomas.higginson@gmail.com</code>, the Contacts Provider stores the two email address rows and links them both to the raw contact.</p> <p>Notice that different types of data are stored in this single table. Display name, phone number, email, postal address, photo, and website detail rows are all found in the <code>ContactsContract.Data</code> table. To help manage this, the <code>ContactsContract.Data</code> table has some columns with descriptive names, and others with generic names. The contents of a descriptive-name column have the same meaning regardless of the type of data in the row, while the contents of a generic-name column have different meanings depending on the type of data.</p> <p><a href="https://developer.android.com/guide/topics/providers/contacts-provider.html">https://developer.android.com/guide/topics/providers/contacts-provider.html</a></p>

**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC											
	<table border="1"> <thead> <tr> <th data-bbox="514 240 609 272">Task</th> <th data-bbox="615 240 856 272">Action</th> <th data-bbox="863 240 1188 272">Data</th> <th data-bbox="1194 240 1486 272">MIME type</th> <th data-bbox="1493 240 1745 272">Notes</th> </tr> </thead> <tbody> <tr> <td data-bbox="514 277 609 946">Pick a contact from a list</td> <td data-bbox="615 277 856 946">ACTION_PICK</td> <td data-bbox="863 277 1188 946">                     One of:                     <ul style="list-style-type: none"> <li>• <code>Contacts.CONTENT_URI</code>, which displays a list of contacts.</li> <li>• <code>Phone.CONTENT_URI</code>, which displays a list of phone numbers for a raw contact.</li> <li>• <code>StructuredPostal.CONTENT_URI</code>, which displays a list of postal addresses for a raw contact.</li> <li>• <code>Email.CONTENT_URI</code>, which displays a list of email addresses for a raw contact.</li> </ul> </td> <td data-bbox="1194 277 1486 946">Not used</td> <td data-bbox="1493 277 1745 946">                     Displays a list of raw contacts or a list of data from a raw contact, depending on the content URI type you supply. Call <code>startActivityForResult()</code>, which returns the content URI of the selected row. The form of the URI is the table's content URI with the row's <code>LOOKUP_ID</code> appended to it. The device's contacts app delegates read and write permissions to this content URI for the life of your activity. See the <a href="#">Content Provider Basics</a> guide for more details.                 </td> </tr> </tbody> </table>	Task	Action	Data	MIME type	Notes	Pick a contact from a list	ACTION_PICK	One of: <ul style="list-style-type: none"> <li>• <code>Contacts.CONTENT_URI</code>, which displays a list of contacts.</li> <li>• <code>Phone.CONTENT_URI</code>, which displays a list of phone numbers for a raw contact.</li> <li>• <code>StructuredPostal.CONTENT_URI</code>, which displays a list of postal addresses for a raw contact.</li> <li>• <code>Email.CONTENT_URI</code>, which displays a list of email addresses for a raw contact.</li> </ul>	Not used	Displays a list of raw contacts or a list of data from a raw contact, depending on the content URI type you supply. Call <code>startActivityForResult()</code> , which returns the content URI of the selected row. The form of the URI is the table's content URI with the row's <code>LOOKUP_ID</code> appended to it. The device's contacts app delegates read and write permissions to this content URI for the life of your activity. See the <a href="#">Content Provider Basics</a> guide for more details.	<a href="https://developer.android.com/guide/topics/providers/contacts-provider.html">https://developer.android.com/guide/topics/providers/contacts-provider.html</a>
Task	Action	Data	MIME type	Notes								
Pick a contact from a list	ACTION_PICK	One of: <ul style="list-style-type: none"> <li>• <code>Contacts.CONTENT_URI</code>, which displays a list of contacts.</li> <li>• <code>Phone.CONTENT_URI</code>, which displays a list of phone numbers for a raw contact.</li> <li>• <code>StructuredPostal.CONTENT_URI</code>, which displays a list of postal addresses for a raw contact.</li> <li>• <code>Email.CONTENT_URI</code>, which displays a list of email addresses for a raw contact.</li> </ul>	Not used	Displays a list of raw contacts or a list of data from a raw contact, depending on the content URI type you supply. Call <code>startActivityForResult()</code> , which returns the content URI of the selected row. The form of the URI is the table's content URI with the row's <code>LOOKUP_ID</code> appended to it. The device's contacts app delegates read and write permissions to this content URI for the life of your activity. See the <a href="#">Content Provider Basics</a> guide for more details.								

**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC
	<pre>25  * Parsed form of the intent sent to the Contacts application. 26  */ 27  public class ContactsRequest { 28 29      /** Default mode: browse contacts */ 30      public static final int ACTION_DEFAULT = 10; 31 32      /** Show all contacts */ 33      public static final int ACTION_ALL_CONTACTS = 15; 34 35      /** Show all contacts with phone numbers */ 36      public static final int ACTION_CONTACTS_WITH_PHONES = 17; 37 38      /** Show contents of a specific group */ 39      public static final int ACTION_GROUP = 20; 40 41      /** Show all starred contacts */ 42      public static final int ACTION_STARRED = 30; 43 44      /** Show frequently contacted contacts */ 45      public static final int ACTION_FREQUENT = 40; 46 47      /** Show starred and the frequent */ 48      public static final int ACTION_STREQUENT = 50; 49 50      /** Show all contacts and pick them when clicking */ 51      public static final int ACTION_PICK_CONTACT = 60; 52 53      /** Show all contacts as well as the option to create a new one */ 54      public static final int ACTION_PICK_OR_CREATE_CONTACT = 70; 55 56      /** Show all contacts and pick them for edit when clicking, and allow creating a new contact */ 57      public static final int ACTION_INSERT_OR_EDIT_CONTACT = 80;</pre> <p><a href="https://android.googlesource.com/platform/packages/apps/Contacts/+nougat-mr1-release/src/com/android/contacts/list/ContactsRequest.java">https://android.googlesource.com/platform/packages/apps/Contacts/+nougat-mr1-release/src/com/android/contacts/list/ContactsRequest.java</a></p>

**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC
	<pre>59  /** Show all phone numbers and pick them when clicking */ 60  public static final int ACTION_PICK_PHONE = 90; 61 62  /** Show all postal addresses and pick them when clicking */ 63  public static final int ACTION_PICK_POSTAL = 100; 64 65  /** Show all postal addresses and pick them when clicking */ 66  public static final int ACTION_PICK_EMAIL = 105; 67 68  /** Show all contacts and create a shortcut for the picked contact */ 69  public static final int ACTION_CREATE_SHORTCUT_CONTACT = 110; 70 71  /** Show all phone numbers and create a call shortcut for the picked number */ 72  public static final int ACTION_CREATE_SHORTCUT_CALL = 120; 73 74  /** Show all phone numbers and create an SMS shortcut for the picked number */ 75  public static final int ACTION_CREATE_SHORTCUT_SMS = 130; 76 77  /** Show all contacts and activate the specified one */ 78  public static final int ACTION_VIEW_CONTACT = 140; 79 80  /** Show contacts recommended for joining with a specified target contact */ 81  public static final int ACTION_PICK_JOIN = 150;</pre> <p><a href="https://android.googlesource.com/platform/packages/apps/Contacts+/nougat-mr1-release/src/com/android/contacts/list/ContactsRequest.java">https://android.googlesource.com/platform/packages/apps/Contacts+/nougat-mr1-release/src/com/android/contacts/list/ContactsRequest.java</a></p>

**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC
	<pre> 104      * Displays a list to browse contacts. 105      */ 106      public class PeopleActivity extends ContactsActivity implements 107          View.OnCreateContextMenuListener, 108          View.OnClickListener, 109          ActionBarAdapter.Listener, 110          DialogManager.DialogShowingViewActivity, 111          ContactListFilterController.ContactListFilterListener, 112          ProviderStatusListener, 113          MultiContactDeleteListener, 114          JoinContactsListener { </pre> <p data-bbox="506 659 1566 727"><a href="https://android.googlesource.com/platform/packages/apps/Contacts+/nougat-mr1-release/src/com/android/contacts/activities/PeopleActivity.java">https://android.googlesource.com/platform/packages/apps/Contacts+/nougat-mr1-release/src/com/android/contacts/activities/PeopleActivity.java</a></p> <pre> 145      * Showing a list of Contacts. Also used for showing search results in search mode. 146      */ 147      private MultiSelectContactsListFragment mAllFragment; 148      private ContactTileListFragment mFavoritesFragment; </pre> <p data-bbox="506 870 1566 937"><a href="https://android.googlesource.com/platform/packages/apps/Contacts+/nougat-mr1-release/src/com/android/contacts/activities/PeopleActivity.java">https://android.googlesource.com/platform/packages/apps/Contacts+/nougat-mr1-release/src/com/android/contacts/activities/PeopleActivity.java</a></p>

**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC
	<pre>458     private void configureFragments(boolean fromRequest) { 459         if (fromRequest) { 460             ContactListFilter filter = null; 461             int actionCode = mRequest.getActionCode(); 462             boolean searchMode = mRequest.isSearchMode(); 463             final int tabToOpen; 464             switch (actionCode) { 465                 case ContactsRequest.ACTION_ALL_CONTACTS: 466                     filter = ContactListFilter.createFilterWithType( 467                         ContactListFilter.FILTER_TYPE_ALL_ACCOUNTS); 468                     tabToOpen = TabState.ALL; 469                     break; 470                 case ContactsRequest.ACTION_CONTACTS_WITH_PHONES: 471                     filter = ContactListFilter.createFilterWithType( 472                         ContactListFilter.FILTER_TYPE_WITH_PHONE_NUMBERS_ONLY); 473                     tabToOpen = TabState.ALL; 474                     break; 475 476                 case ContactsRequest.ACTION_FREQUENT: 477                 case ContactsRequest.ACTION_STREQUENT: 478                 case ContactsRequest.ACTION_STARRED: 479                     tabToOpen = TabState.FAVORITES; 480                     break; 481                 case ContactsRequest.ACTION_VIEW_CONTACT: 482                     tabToOpen = TabState.ALL; 483                     break; 484                 default: 485                     tabToOpen = -1; 486                     break; 487             } </pre> <p data-bbox="506 1321 1566 1390"><a href="https://android.googlesource.com/platform/packages/apps/Contacts+/nougat-mr1-release/src/com/android/contacts/activities/PeopleActivity.java">https://android.googlesource.com/platform/packages/apps/Contacts+/nougat-mr1-release/src/com/android/contacts/activities/PeopleActivity.java</a></p>

**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC
	<pre>488         if (tabToOpen != -1) { 489             mActionBarAdapter.setCurrentTab(tabToOpen); 490         } 491 492         if (filter != null) { 493             mContactListFilterController.setContactListFilter(filter, false); 494             searchMode = false; 495         } 496 497         if (mRequest.getContactUri() != null) { 498             searchMode = false; 499         } 500 501         mActionBarAdapter.setSearchMode(searchMode); 502         configureContactListFragmentForRequest(); 503     } 504 505     configureContactListFragment(); 506 507     invalidateOptionsMenuIfNeeded(); 508 }</pre> <p><a href="https://android.googlesource.com/platform/packages/apps/Contacts+/nougat-mr1-release/src/com/android/contacts/activities/PeopleActivity.java">https://android.googlesource.com/platform/packages/apps/Contacts+/nougat-mr1-release/src/com/android/contacts/activities/PeopleActivity.java</a></p>

**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC
	<pre>35 public class ProfileAndContactsLoader extends CursorLoader { 36 37     private boolean mLoadProfile; 38 39     private String[] mProjection; 40 41     private Uri mExtraUri; 42     private String[] mExtraProjection; 43     private String mExtraSelection; 44     private String[] mExtraSelectionArgs; 45     private boolean mMergeExtraContactsAfterPrimary; 46 47     public ProfileAndContactsLoader(Context context) { 48         super(context); 49     }</pre> <p><a href="https://android.googlesource.com/platform/packages/apps/ContactsCommon/+/nougat-mr1-release/src/com/android/contacts/common/list/ProfileAndContactsLoader.java">https://android.googlesource.com/platform/packages/apps/ContactsCommon/+/nougat-mr1-release/src/com/android/contacts/common/list/ProfileAndContactsLoader.java</a></p>



**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC
	<pre>35 public final class GroupMemberLoader extends CursorLoader { 36 37     public static class GroupEditorQuery { 38         private static final String[] PROJECTION = new String[] { 39             Data.CONTACT_ID,           // 0 40             Data.RAW_CONTACT_ID,      // 1 41             Data.DISPLAY_NAME_PRIMARY, // 2 42             Data.PHOTO_URI,           // 3 43             Data.LOOKUP_KEY,          // 4 44         }; 45 46         public static final int CONTACT_ID           = 0; 47         public static final int RAW_CONTACT_ID      = 1; 48         public static final int CONTACT_DISPLAY_NAME_PRIMARY = 2; 49         public static final int CONTACT_PHOTO_URI   = 3; 50         public static final int CONTACT_LOOKUP_KEY  = 4; 51     } 52 53     public static class GroupDetailQuery { 54         private static final String[] PROJECTION = new String[] { 55             Data.CONTACT_ID,           // 0 56             Data.PHOTO_URI,           // 1 57             Data.LOOKUP_KEY,          // 2 58             Data.DISPLAY_NAME_PRIMARY, // 3 59             Data.CONTACT_PRESENCE,    // 4 60             Data.CONTACT_STATUS,      // 5 61         }; 62 63         public static final int CONTACT_ID           = 0; 64         public static final int CONTACT_PHOTO_URI   = 1; 65         public static final int CONTACT_LOOKUP_KEY  = 2; 66         public static final int CONTACT_DISPLAY_NAME_PRIMARY = 3; 67         public static final int CONTACT_PRESENCE_STATUS = 4; 68         public static final int CONTACT_STATUS      = 5; 69     } 70 71     private final long mGroupId;</pre> <p><a href="https://android.googlesource.com/platform/packages/apps/Contacts/+/nougat-mr1-release/src/com/android/contacts/GroupMemberLoader.java">https://android.googlesource.com/platform/packages/apps/Contacts/+/nougat-mr1-release/src/com/android/contacts/GroupMemberLoader.java</a></p>

**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC
	<pre>24 * Group loader for the group list that includes details such as the number of contacts per group 25 * and number of groups per account. This list is sorted by account type, account name, where the 26 * group names are in alphabetical order. Note that the list excludes default, favorite, and deleted 27 * groups. 28 */ 29 public final class GroupListLoader extends CursorLoader { 30 31     private final static String[] COLUMNS = new String[] { 32         Groups.ACCOUNT_NAME, 33         Groups.ACCOUNT_TYPE, 34         Groups.DATA_SET, 35         Groups._ID, 36         Groups.TITLE, 37         Groups.SUMMARY_COUNT, 38     }; 39 40     public final static int ACCOUNT_NAME = 0; 41     public final static int ACCOUNT_TYPE = 1; 42     public final static int DATA_SET = 2; 43     public final static int GROUP_ID = 3; 44     public final static int TITLE = 4; 45     public final static int MEMBER_COUNT = 5; 46 47     private static final Uri GROUP_LIST_URI = Groups.CONTENT_SUMMARY_URI; 48 49     public GroupListLoader(Context context) { 50         super(context, GROUP_LIST_URI, COLUMNS, Groups.ACCOUNT_TYPE + " NOT NULL AND " 51             + Groups.ACCOUNT_NAME + " NOT NULL AND " + Groups.AUTO_ADD + "=0 AND " + 52             Groups.FAVORITES + "=0 AND " + Groups.DELETED + "=0", null, 53             Groups.ACCOUNT_TYPE + ", " + Groups.ACCOUNT_NAME + ", " + Groups.DATA_SET + ", " + 54             Groups.TITLE + " COLLATE LOCALIZED ASC"); 55     } 56 }</pre> <p><a href="https://android.googlesource.com/platform/packages/apps/Contacts+/nougat-mr1-release/src/com/android/contacts/GroupListLoader.java">https://android.googlesource.com/platform/packages/apps/Contacts+/nougat-mr1-release/src/com/android/contacts/GroupListLoader.java</a></p>

**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC
	<pre>24  * Group meta-data loader. Loads all groups or just a single group from the 25  * database (if given a {@link Uri}). 26  */ 27  public final class GroupMetaLoader extends CursorLoader { 28 29      private final static String[] COLUMNS = new String[] { 30          Groups.ACCOUNT_NAME, 31          Groups.ACCOUNT_TYPE, 32          Groups.DATA_SET, 33          Groups._ID, 34          Groups.TITLE, 35          Groups.AUTO_ADD, 36          Groups.FAVORITES, 37          Groups.GROUP_IS_READ_ONLY, 38          Groups.DELETED, 39      }; 40 41      public final static int ACCOUNT_NAME = 0; 42      public final static int ACCOUNT_TYPE = 1; 43      public final static int DATA_SET = 2; 44      public final static int GROUP_ID = 3; 45      public final static int TITLE = 4; 46      public final static int AUTO_ADD = 5; 47      public final static int FAVORITES = 6; 48      public final static int IS_READ_ONLY = 7; 49      public final static int DELETED = 8; 50 51      public GroupMetaLoader(Context context, Uri groupUri) { 52          super(context, ensureIsGroupUri(groupUri), COLUMNS, Groups.ACCOUNT_TYPE + " NOT NULL AND " 53              + Groups.ACCOUNT_NAME + " NOT NULL", null, null); 54      } 55 56      /** 57       * Ensures that this is a valid group URI. If invalid, then an exception is 58       * thrown. Otherwise, the original URI is returned. 59       */ 60      private static Uri ensureIsGroupUri(final Uri groupUri) { 61          // TODO: Fix ContactsProvider2 getType method to resolve the group Uris 62          if (groupUri == null) { 63              throw new IllegalArgumentException("Uri must not be null"); 64          } 65          if (!groupUri.toString().startsWith(Groups.CONTENT_URI.toString())) { 66              throw new IllegalArgumentException("Invalid group Uri: " + groupUri); 67          } 68          return groupUri; 69      } 70  }</pre> <p><a href="https://android.googlesource.com/platform/packages/apps/Contacts/+nougat-mr1-release/src/com/android/contacts/GroupMetaLoader.java">https://android.googlesource.com/platform/packages/apps/Contacts/+nougat-mr1-release/src/com/android/contacts/GroupMetaLoader.java</a></p>

**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC
	<pre>19  * Meta-data for a contact group. We load all groups associated with the contact's 20  * constituent accounts. 21  */ 22  public final class GroupMetaData { 23      private String mAccountName; 24      private String mAccountType; 25      private String mDataSet; 26      private long mGroupId; 27      private String mTitle; 28      private boolean mDefaultGroup; 29      private boolean mFavorites; 30 31      public GroupMetaData(String accountName, String accountType, String dataSet, long groupId, 32                          String title, boolean defaultGroup, boolean favorites) { 33          this.mAccountName = accountName; 34          this.mAccountType = accountType; 35          this.mDataSet = dataSet; 36          this.mGroupId = groupId; 37          this.mTitle = title; 38          this.mDefaultGroup = defaultGroup; 39          this.mFavorites = favorites; 40      } 41 42      public String getAccountName() { 43          return mAccountName; 44      } 45 46      public String getAccountType() { 47          return mAccountType; 48      } 49 50      public String getDataSet() { 51          return mDataSet; 52      } 53 54      public long getGroupId() { 55          return mGroupId; 56      } 57 58      public String getTitle() { 59          return mTitle; 60      } 61 62      public boolean isDefaultGroup() { 63          return mDefaultGroup; 64      } 65 66      public boolean isFavorites() { 67          return mFavorites; 68      } </pre> <p><a href="https://android.googlesource.com/platform/packages/apps/ContactsCommon/+/nougat-mr1-release/src/com/android/contacts/common/GroupMetaData.java">https://android.googlesource.com/platform/packages/apps/ContactsCommon/+/nougat-mr1-release/src/com/android/contacts/common/GroupMetaData.java</a></p>

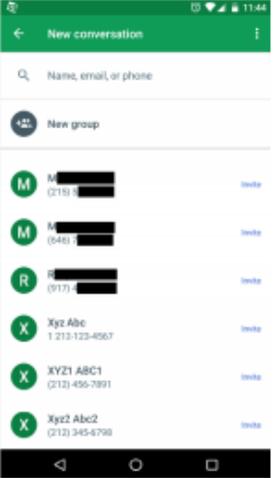
**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC
	<p data-bbox="541 277 1297 370"><b>Send &amp; receive text messages in Android Messages</b></p> <p data-bbox="541 386 1178 407">You can send and receive text messages with friends and contacts on Android Messages.</p> <p data-bbox="531 423 846 456"><b>Start a conversation</b></p> <ol data-bbox="541 480 1528 638" style="list-style-type: none"><li>1. Open the Android Messages app .</li><li>2. Tap Compose .</li><li>3. In "To," enter the names, phone numbers, or email addresses that you'd like to message. You can also pick from your top contacts or your whole contact list.</li><li>4. Tap Next .</li></ol> <p data-bbox="516 659 1472 691"><a href="https://support.google.com/nexus/answer/6080324?hl=en&amp;ref_topic=6080329">https://support.google.com/nexus/answer/6080324?hl=en&amp;ref_topic=6080329</a></p> <p data-bbox="541 781 894 821"><b>See your contacts</b></p> <ol data-bbox="552 846 968 919" style="list-style-type: none"><li>1. Open your device's Contacts app .</li><li>2. Tap Menu .</li></ol> <ul data-bbox="548 951 1738 1146" style="list-style-type: none"><li>• <b>See contacts by label:</b> Choose a label from the list.</li><li>• <b>See contacts for another account:</b> Tap Down arrow  &gt; pick an account.</li><li>• <b>See the contacts for all your accounts:</b> Choose <b>All contacts</b>. <b>Tip:</b> If you have multiple contacts with the same information, the information will be grouped into one contact.</li><li>• <b>See your Google Account contacts on the web:</b> Go to <a href="#">Google Contacts</a> .</li></ul> <p data-bbox="516 1179 1535 1211"><a href="https://support.google.com/android/answer/6118731?hl=en&amp;ref_topic=6118711">https://support.google.com/android/answer/6118731?hl=en&amp;ref_topic=6118711</a></p>

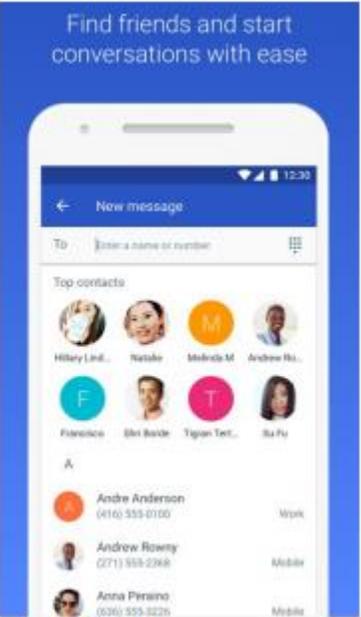
**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC
	<p data-bbox="541 240 894 277"><b>Label your contacts</b></p> <p data-bbox="541 305 982 326">You can group contacts together using labels.</p> <ol data-bbox="541 358 926 461" style="list-style-type: none"><li data-bbox="541 358 926 380">1. Open your device's Contacts app .</li><li data-bbox="541 396 863 417">2. Tap Menu  &gt; <b>Create label</b>.</li><li data-bbox="541 433 863 454">3. Enter a label name and tap <b>Ok</b>.</li></ol> <ul data-bbox="541 493 1717 553" style="list-style-type: none"><li data-bbox="541 493 1234 514">• <b>Add one contact to a label:</b> Tap Add contact  &gt; choose a contact.</li><li data-bbox="541 531 1717 552">• <b>Add multiple contacts to a label:</b> Tap Add contact  &gt; touch and hold a contact &gt; tap the other contacts &gt; tap <b>Add</b>.</li></ul> <p data-bbox="512 570 1535 591"><a href="https://support.google.com/android/answer/6118731?hl=en&amp;ref_topic=6118711">https://support.google.com/android/answer/6118731?hl=en&amp;ref_topic=6118711</a></p> <p data-bbox="541 656 940 693"><b>Share your contacts</b></p> <ol data-bbox="541 725 1045 883" style="list-style-type: none"><li data-bbox="541 725 974 747">1. Open your device's Contacts app .</li><li data-bbox="541 769 842 790">2. Tap a contact in the list.</li><li data-bbox="541 813 831 834">3. Tap More  &gt; <b>Share</b>.</li><li data-bbox="541 857 1045 878">4. Choose how you want to share the contact.</li></ol> <p data-bbox="512 899 1535 920"><a href="https://support.google.com/android/answer/6118731?hl=en&amp;ref_topic=6118711">https://support.google.com/android/answer/6118731?hl=en&amp;ref_topic=6118711</a></p>

# Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products

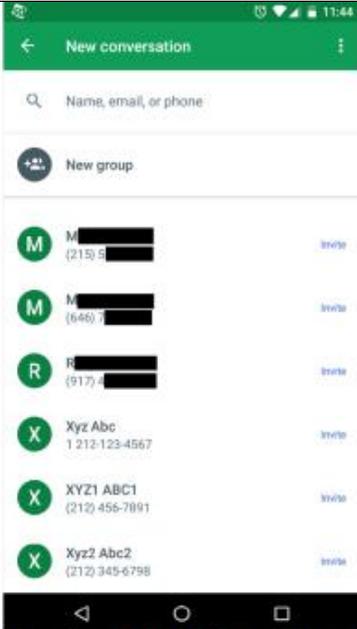
US9408055B2	HTC
	<p><b>Start a Hangout</b></p> <p>You can send and receive messages with one person or multiple people.</p> <p>COMPUTER <b>ANDROID</b> IPHONE &amp; IPAD</p> <p><b>Start a conversation</b></p> <ol style="list-style-type: none"><li>1. On your Android phone or tablet, open the Hangouts app .</li><li>2. At the bottom right, tap Add  &gt; New Conversation .</li><li>3. Type and select a person's name.</li><li>4. Enter your message. You can also add emojis, photos, your location, or a sticker.</li><li>5. Tap Send .</li></ol>  <p><a href="https://support.google.com/hangouts/answer/3115553?hl=en&amp;ref_topic=6386410&amp;co=GENIE.Platform%3DAndroid&amp;oco=0">https://support.google.com/hangouts/answer/3115553?hl=en&amp;ref_topic=6386410&amp;co=GENIE.Platform%3DAndroid&amp;oco=0</a></p> <p><b>Contact someone</b></p> <p>You can call, email, or send text messages to your contacts.</p> <ol style="list-style-type: none"><li>1. Open your device's Contacts app .</li><li>2. Tap a contact in the list.</li><li>3. Choose an option:<ul style="list-style-type: none"><li>• Call </li><li>• Email </li><li>• New message </li></ul></li></ol> <p><a href="https://support.google.com/nexus/answer/6118731?hl=en&amp;ref_topic=6118711">https://support.google.com/nexus/answer/6118731?hl=en&amp;ref_topic=6118711</a></p>

**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC
	<p data-bbox="527 688 762 716"><b>Start a conversation</b></p> <ol data-bbox="527 732 1272 862" style="list-style-type: none"><li>1. Open the Android Messages app</li><li>2. Tap Compose</li><li>3. In "To," enter the names, phone numbers, or email addresses that you'd like to message. You can also pick from your top contacts or your whole contact list.</li><li>4. Tap Next</li></ol> <p data-bbox="516 878 1612 943"><a href="https://support.google.com/nexus/answer/6080324?hl=en&amp;ref_topic=6080329">https://support.google.com/nexus/answer/6080324?hl=en&amp;ref_topic=6080329</a> <a href="https://play.google.com/store/apps/details?id=com.google.android.apps.messaging&amp;hl=en">https://play.google.com/store/apps/details?id=com.google.android.apps.messaging&amp;hl=en</a></p> 



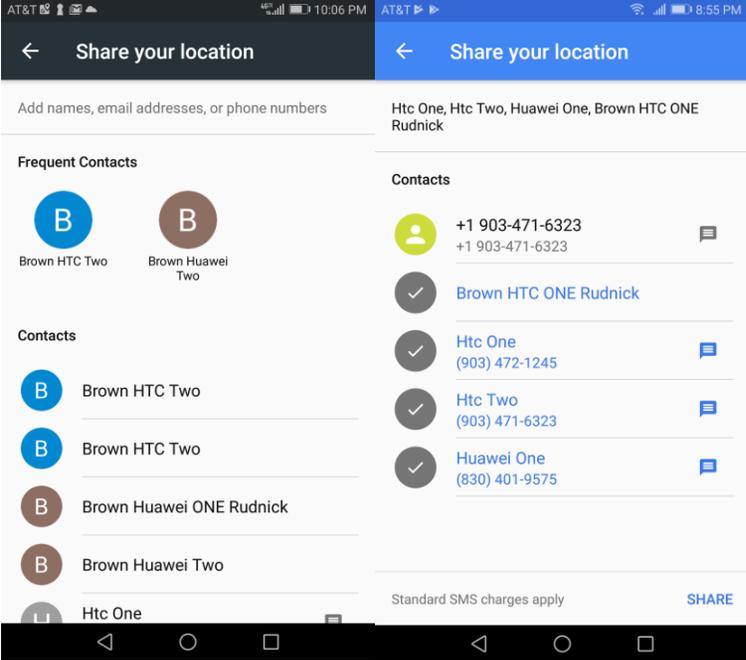
**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC
	<p data-bbox="533 618 842 651"><b>Start a conversation</b></p> <ol data-bbox="533 672 1218 829" style="list-style-type: none"><li>1. On your Android phone or tablet, open the Hangouts app .</li><li>2. At the bottom right, tap Add  &gt; New Conversation .</li><li>3. Type and select a person's name.</li><li>4. Enter your message. You can also add emojis, photos, your location, or a sticker.</li><li>5. Tap Send .</li></ol>  <p data-bbox="520 862 1751 919"><a href="https://support.google.com/hangouts/answer/3115553?hl=en&amp;ref_topic=6386410&amp;co=GENIE.Platform%3DAndroid&amp;oco=0">https://support.google.com/hangouts/answer/3115553?hl=en&amp;ref_topic=6386410&amp;co=GENIE.Platform%3DAndroid&amp;oco=0</a></p> <ol data-bbox="533 980 1352 1138" style="list-style-type: none"><li>1. Open the Hangouts app .</li><li>2. At the bottom, tap Add  &gt; <b>New conversation</b> &gt; <b>New group</b>.</li><li>3. Enter and select the names, phone numbers, or email addresses of people in your group.</li><li>4. Tap Done .</li></ol> <p data-bbox="520 1149 1751 1206"><a href="https://support.google.com/hangouts/answer/3111943?hl=en&amp;ref_topic=6386410&amp;co=GENIE.Platform%3DAndroid&amp;oco=1">https://support.google.com/hangouts/answer/3111943?hl=en&amp;ref_topic=6386410&amp;co=GENIE.Platform%3DAndroid&amp;oco=1</a></p>

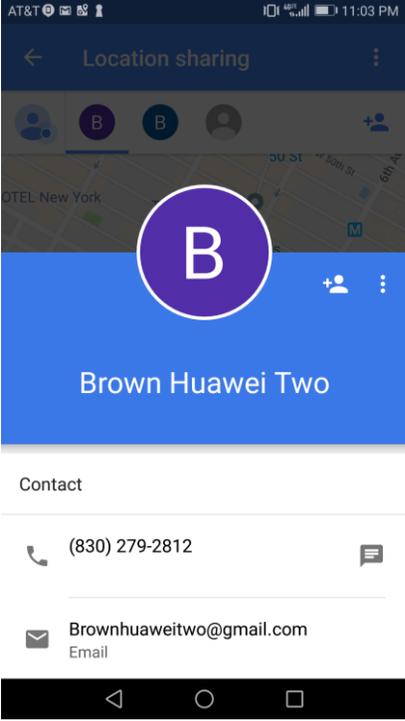
**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC
	<p data-bbox="541 235 903 276"><b>Contact someone</b></p> <p data-bbox="541 305 1192 332">You can call, email, or send text messages to your contacts.</p> <ol data-bbox="554 365 982 625" style="list-style-type: none"><li data-bbox="554 365 982 397">1. Open your device's Contacts app .</li><li data-bbox="554 414 840 446">2. Tap a contact in the list.</li><li data-bbox="554 462 787 495">3. Choose an option:<ul data-bbox="583 503 808 625" style="list-style-type: none"><li data-bbox="583 503 682 535">• Call </li><li data-bbox="583 552 703 584">• Email </li><li data-bbox="583 600 808 625">• New message </li></ul></li></ol> <p data-bbox="520 641 1470 673"><a href="https://support.google.com/nexus/answer/6118731?hl=en&amp;ref_topic=6118711">https://support.google.com/nexus/answer/6118731?hl=en&amp;ref_topic=6118711</a></p> <p data-bbox="512 706 1029 738"><b><u>Exemplary Google Maps Screenshots:</u></b></p>

**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC
	 <p>For example, the Accused Products include software that obtains contact information including the phone numbers . Furthermore, these phone calls can merge multiple parties into a conference call.</p>

**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC
	
<p>[41B] facilitating initiation of Internet Protocol (IP) based communication between the first device and the respective second devices by using the respective telephone numbers to send, from</p>	<p>HTC infringes directly and/or indirectly by performing, inducing others to perform, and/or contributing to the performance of facilitating initiation of Internet Protocol (IP) based communication between the first device and the respective second devices by using the respective telephone numbers to send, from the first device to the second devices, respective Short Message Service (SMS) messages including a telephone number of the first device and information usable by the respective second device to send IP-based communication to the first device. See claims 1[B] and 28[B], which are incorporated herein by reference in their entirety.</p> <p>For example, the Accused Products utilize SMS-based messages to initiate IP communication between participants of Maps location sharing. For example, both Android Messages and Hangouts, in conjunction</p>

**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC
the first device to the second devices, respective Short Message Service (SMS) messages including a telephone number of the first device and information usable by the respective second device to send IP-based communication to the first device;	<p>with Maps, utilize SMS messages, including group messages from one device to several devices, to send an SMS message, with additional information, to a contact.</p> <p>D2 Technologies Showcases its mCUE IP Communications Interface over WiMAX on HTC's EVO 4G Android Smartphone</p> <p><small>Amsterdam, Netherlands (WiMAX Forum Global Congress) and Santa Barbara, CA – June 14, 2010 —D2 Technologies, the market leader in embedded IP communications software platforms, today announced that it is holding private demonstrations of its mCUE® converged communications client for mobile devices and handsets on the HTC EVO™ 4G smartphone on Thursday, June 17 at the WiMAX Forum® Global Congress in Amsterdam. D2's mCUE on the HTC EVO, the first 4G phone to be introduced in the United States, was configured and installed in less than a week – clearly illustrating how OEMs and ODMs can more rapidly develop Android™-based devices by choosing to incorporate the converged presence-based communications user interface (CUI).</small></p> <p><a href="http://www.d2tech.com/press-releases-year.html?Y=2010">http://www.d2tech.com/press-releases-year.html?Y=2010</a></p>

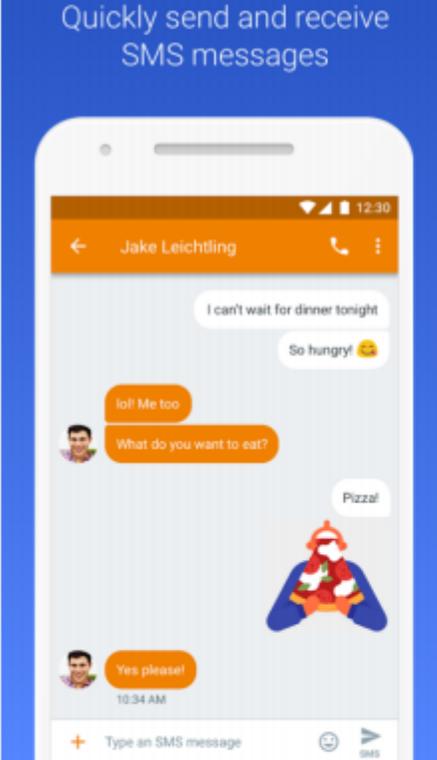
**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC
	<p data-bbox="527 233 1289 302">Siemens Enterprise Communications and HTC Simplify Device Choice for Mobile UC   </p> <p data-bbox="527 331 1035 350">Reston, VA and Frankfurt, Germany, Feb 19, 2013</p> <p data-bbox="527 380 1451 464"><b>Enterasys' Mobile IAM and MDM connect™ BYOD solutions now support AirWatch to onboard and manage mobile devices and applications</b></p> <p data-bbox="527 483 1514 643">In an ongoing effort to support today's increasingly mobile workforce, Siemens Enterprise Communications and HTC Corporation today announced a strategic global partnership to make it easier for enterprises to embrace mobile unified communications (UC) on HTC enterprise-enabled devices. This partnership makes it even simpler for enterprises to embrace a BYOD strategy for mobile UC, since HTC's popular consumer Android smartphones now fully support Siemens Enterprise Communications' OpenScape Mobile and OpenScape Web Collaboration solutions.</p> <p data-bbox="527 695 625 714"><b>Key Facts</b></p> <ul data-bbox="527 734 1514 961" style="list-style-type: none"> <li>▪ Siemens Enterprise Communications OpenScape Mobile and OpenScape Web Collaboration solutions will be validated on select HTC devices to increase users' confidence that their chosen device will work seamlessly with their mobile communication tools</li> <li>▪ Siemens Enterprise Communications customers will have a simplified process to secure validated HTCPro devices supporting Siemens Enterprise Communications solutions</li> <li>▪ Siemens Enterprise Communications and HTC will collaborate to simplify deployment of mobile UC through joint marketing and fulfillment efforts</li> <li>▪ This collaboration will take place through HTCPro, a program that provides mobile solutions for companies and their employees and ensures that HTC's entire portfolio is business-ready</li> </ul> <p data-bbox="527 1003 1514 1068">Siemens Enterprise Communications OpenScape Mobile and OpenScape Web Collaboration solutions will be validated on select HTC devices to increase users' confidence that their chosen device will work seamlessly with their mobile communication tools.</p> <p data-bbox="527 1084 1713 1114"><a href="http://www.unify.com/us/news/E853B94A-F94F-4ADA-98DA-8C80BB965953/?isarchive=1">http://www.unify.com/us/news/E853B94A-F94F-4ADA-98DA-8C80BB965953/?isarchive=1</a></p>

**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC
	<p>Android Messages makes it easy to communicate with anyone by using SMS, MMS, and more. Stay in touch with friends and family, send group texts, and share your favorite pictures, videos, audio messages.</p> <p>· <b>Enhanced features:</b> On supported carriers, you can send messages over Wi-Fi or your data network, see when friends have read your message, and more.</p> <p><a href="https://play.google.com/store/apps/details?id=com.google.android.apps.messaging&amp;hl=en">https://play.google.com/store/apps/details?id=com.google.android.apps.messaging&amp;hl=en</a></p>

**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC
	 <p><a href="https://play.google.com/store/apps/details?id=com.google.android.apps.messaging&amp;hl=en">https://play.google.com/store/apps/details?id=com.google.android.apps.messaging&amp;hl=en</a></p> <h3>Get started with Hangouts</h3> <p>You can use Hangouts to:</p> <ul style="list-style-type: none"><li>• Start a chat conversation or video call.</li><li>• Make phone calls using Wi-Fi or data.</li><li>• Send text messages with your <a href="#">Google Voice</a> or <a href="#">Project Fi</a> phone number.</li></ul> <p>Hangouts sync automatically across devices. If you start a Hangout on your computer, you can continue your chat on another device, like your phone.</p> <p><a href="https://support.google.com/hangouts/answer/2944865?hl=en&amp;ref_topic=6386410">https://support.google.com/hangouts/answer/2944865?hl=en&amp;ref_topic=6386410</a></p>



## Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products

US9408055B2	HTC
	<p data-bbox="541 266 835 305"><b>Start a Hangout</b></p> <p data-bbox="541 321 1054 341">You can send and receive messages with one person or multiple people.</p> <p data-bbox="554 402 890 418">COMPUTER <b>ANDROID</b> IPHONE &amp; IPAD</p> <hr/> <p data-bbox="541 553 806 581"><b>Start a conversation</b></p> <ol data-bbox="548 602 1136 737" style="list-style-type: none"><li>1. On your Android phone or tablet, open the Hangouts app .</li><li>2. At the bottom right, tap Add  &gt; New Conversation .</li><li>3. Type and select a person's name.</li><li>4. Enter your message. You can also add emojis, photos, your location, or a sticker.</li><li>5. Tap Send .</li></ol> <p data-bbox="527 753 1745 813"><a href="https://support.google.com/hangouts/answer/3115553?hl=en&amp;ref_topic=6386410&amp;co=GENIE.Platform%3DAndroid&amp;oco=0">https://support.google.com/hangouts/answer/3115553?hl=en&amp;ref_topic=6386410&amp;co=GENIE.Platform%3DAndroid&amp;oco=0</a></p> <ul data-bbox="562 862 1562 1013" style="list-style-type: none"><li>• Connect your Google Voice account for phone calling, SMS texting, and voicemail integration.</li><li>• Keep in touch with contacts across Android, iOS, and the web, and sync chats across all your devices.</li><li>• Message contacts anytime, even if they're offline.</li></ul> <p data-bbox="527 1045 1398 1073"><a href="https://play.google.com/store/apps/details?id=com.google.android.talk&amp;hl=en">https://play.google.com/store/apps/details?id=com.google.android.talk&amp;hl=en</a></p> <ol data-bbox="548 1084 1493 1344" style="list-style-type: none"><li>1. Open the Hangouts app .</li><li>2. At the bottom right, tap Add .</li><li>3. Choose <b>New SMS</b>.</li><li>4. Type the name or phone number. If you're traveling, use the "+" sign and country code when texting.</li><li>5. Tap the number or contact.</li><li>6. Tap Continue .</li><li>7. Type your message and tap Send .</li></ol> <p data-bbox="527 1360 1255 1388"><a href="https://support.google.com/hangouts/answer/3441321?hl=en">https://support.google.com/hangouts/answer/3441321?hl=en</a></p>

**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC
	<p data-bbox="512 233 911 264"><b><u>Google Maps Share Location</u></b></p> <p data-bbox="512 306 1902 959">Google Latitude, Google Plus, Google Hangouts, Google Messages, and Android Messenger, Google Duo, Google Allo, and Google Chrome, the Accused Products require a user to join the corresponding network by signing-in to the device with an identifier (e.g., Google Account). The sign-in process takes place within the Google Maps software on the Accused Product or by navigating to maps.google.com within the Google Chrome browser on the Accused Product. Alternatively, the sign-in process may partially or completely take place using credentials already provided when the user associates a Google Account with the Accused Product, e.g., during initial setup of the Accused Product. Subject to discovery, one or more additional or substitute identifiers may correspond to the group. The sign-in process involves a user entering its Google Account and additional authentication data on the interface of the Accused Product and sending a message containing the Google Account and additional authentication data over a network to members of a group. The group comprises the multiple identifiers, individuals, profiles, and/or devices associated with the group. Further regarding Google Maps, Google Latitude, Google Plus, Google Hangouts, Google Messages, and Android Messenger, Google Duo, Google Allo, and Google Chrome, the Accused Products alternatively require a user to send a request containing the identifier (e.g., Google Account). Subject to discovery, additional identifiers may be assigned or used to correspond to the group. The request may be an invitation or message that associates a Google Account with one or more Google Accounts for the purposes of sharing locations within the group. The group comprises the multiple identifiers, individuals, profiles, and/or devices associated with the group</p> <p data-bbox="512 1002 1029 1032"><b><u>Exemplary Support for Google Maps:</u></b></p>

# Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products

US9408055B2	HTC
	<p>COMPUTER   <b>ANDROID</b>   IPHONE &amp; IPAD</p> <hr/> <h3>If they have a Google Account</h3> <ol style="list-style-type: none"><li>1. If you haven't already, add their Gmail address to your <a href="#">Google Contacts</a>.</li><li>2. On your Android phone or tablet, open the Google Maps app  and sign in. <a href="#">Learn how to sign in.</a></li><li>3. Tap Menu ≡ &gt; <b>Location sharing</b> &gt; Add People .</li><li>4. Choose how long you want to share your location.</li><li>5. Tap <b>Select People</b>.<ul style="list-style-type: none"><li>• If you're asked about your contacts, give Google Maps access.</li></ul></li><li>6. Choose who you want to share with.</li><li>7. Tap <b>Share</b>.</li></ol> <h3>If they don't have a Google Account</h3> <ol style="list-style-type: none"><li>1. On your Android phone or tablet, open the Google Maps app  and sign in. <a href="#">Learn how to sign in.</a></li><li>2. Tap Menu ≡ &gt; <b>Location sharing</b> &gt; Add People .</li><li>3. Tap More ... &gt; <b>Copy to clipboard</b>. People with this link can see your location for as long as you choose, up to 72 hours.</li></ol> <h3>Share using another app</h3> <p>You can also share through messaging apps. Tap More ... &gt; select an app.</p> <h3>Stop sharing</h3> <ol style="list-style-type: none"><li>1. Open the Google Maps app .</li><li>2. Tap Menu ≡ &gt; <b>Location sharing</b>.</li><li>3. Next to the person with whom you want to stop sharing, tap Remove ✕.</li></ol> <p><a href="https://support.google.com/maps/answer/7326816?co=GENIE.Platform%3DAndroid&amp;oco=1">https://support.google.com/maps/answer/7326816?co=GENIE.Platform%3DAndroid&amp;oco=1</a></p>

## Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products

US9408055B2	HTC
	<h3>Share your E.T.A</h3> <p>After you start your drive, you can share your destination, estimated arrival time, and where you are on the route.</p> <ol style="list-style-type: none"><li>1. Open the Google Maps app .</li><li>2. Set a driving destination. <a href="#">Learn how to navigate to a place.</a></li><li>3. After you start navigation, tap More ^ &gt; <b>Share trip progress.</b></li><li>4. Choose a person from the list.</li><li>5. Tap <b>Share.</b></li><li>6. Location Sharing will stop when you reach your destination or stop navigating.</li></ol> <ul style="list-style-type: none"><li>• To stop sharing before you arrive, tap More ^ &gt; <b>Stop sharing.</b></li></ul> <h3>See where someone is</h3> <p>If someone shares their location with you, you can see them on the map.</p> <ol style="list-style-type: none"><li>1. Open the Google Maps app .</li><li>2. Tap Menu ≡ &gt; <b>Location sharing.</b></li><li>3. Choose someone.</li></ol> <ul style="list-style-type: none"><li>• To see an updated location, tap on a friend's icon &gt; More ⋮ &gt; <b>Refresh.</b></li></ul> <h3>Stop seeing someone's location</h3> <ol style="list-style-type: none"><li>1. Open the Google Maps app .</li><li>2. On the map, tap their icon.</li><li>3. At the bottom, tap More ^ .</li><li>4. To temporarily hide someone, tap <b>Hide from map.</b> You can stop hiding them at any time.</li></ol> <p><b>Note:</b> You can stop someone's location from ever appearing on your map. <a href="#">Learn how to block another person's account.</a></p> <p><a href="https://support.google.com/maps/answer/7326816?co=GENIE.Platform%3DAndroid&amp;oco=1">https://support.google.com/maps/answer/7326816?co=GENIE.Platform%3DAndroid&amp;oco=1</a></p>

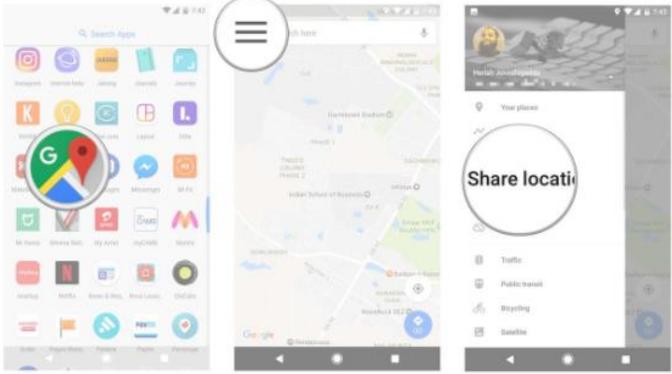
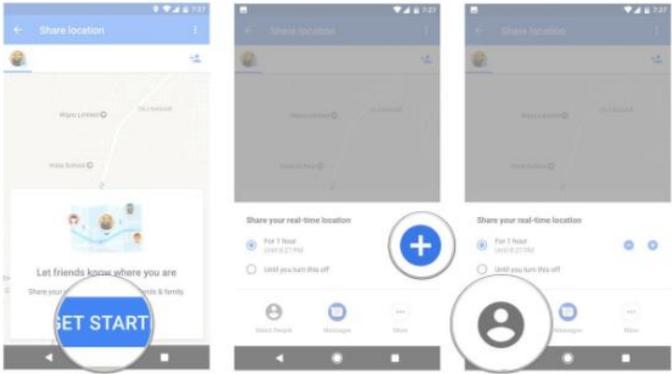
## Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products

US9408055B2	HTC
	<h3>Create a list of places</h3> <p>In Google Maps, you can create a list of places, like your favorite places or places you want to visit.</p> <p>COMPUTER <b>ANDROID</b> IPHONE &amp; IPAD</p> <hr/> <h3>Make a new list</h3> <ol style="list-style-type: none"><li>1. On your Android phone or tablet, open the Google Maps app .</li><li>2. Tap Menu  &gt; <b>Your places</b> &gt; <b>Saved</b>.</li><li>3. In the bottom right, tap Add .</li><li>4. Enter a name and description.</li><li>5. Tap <b>Save</b>.</li></ol> <h3>Save a place to a list</h3> <ol style="list-style-type: none"><li>1. Open the Google Maps app .</li><li>2. Search for a place or tap it on the map.</li><li>3. At the bottom, tap the place's name or address.</li><li>4. Tap <b>Save</b>.</li><li>5. Choose a list. To create a new list, tap <b>New list</b> .</li></ol> <h3>See your lists</h3> <ol style="list-style-type: none"><li>1. Open the Google Maps app .</li><li>2. Tap Menu  &gt; <b>Your places</b> &gt; <b>Saved</b>.</li></ol> <p><a href="https://support.google.com/maps/answer/7280933?hl=en&amp;ref_topic=7301134&amp;co=GENIE.Platform%3DAndroid&amp;oco=1">https://support.google.com/maps/answer/7280933?hl=en&amp;ref_topic=7301134&amp;co=GENIE.Platform%3DAndroid&amp;oco=1</a></p>

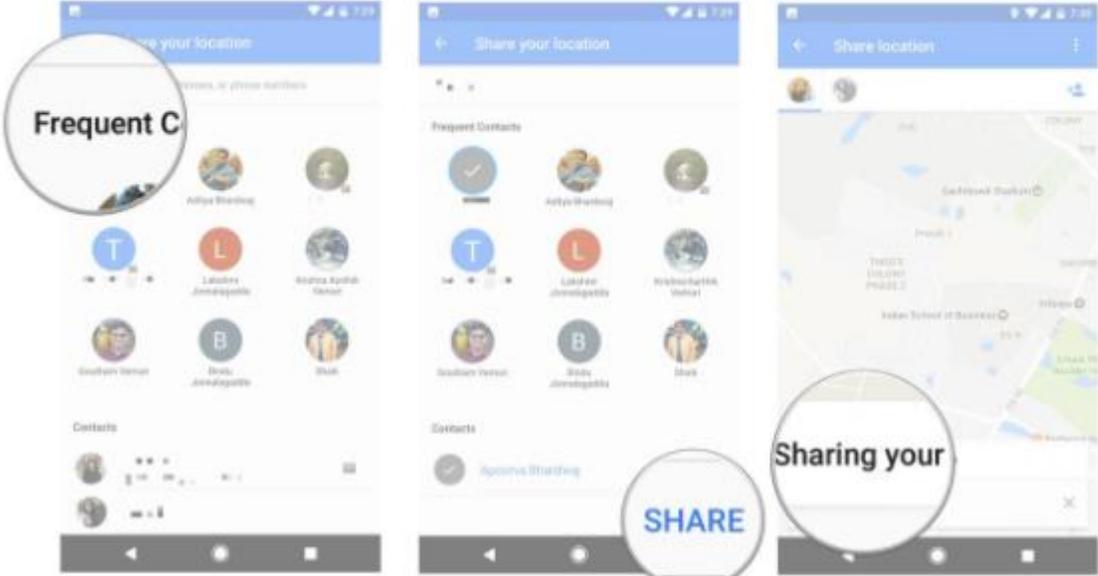
## Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products

US9408055B2	HTC
	<h3 data-bbox="541 245 877 282">Hide or share lists</h3> <p data-bbox="541 313 907 337"><b>Note:</b> You can't share starred places.</p> <ol data-bbox="554 367 1251 475" style="list-style-type: none"><li data-bbox="554 367 890 391">1. Open the Google Maps app .</li><li data-bbox="554 407 968 431">2. Tap Menu  &gt; <b>Your places</b> &gt; <b>Saved</b>.</li><li data-bbox="554 448 1251 475">3. Next to the list you want to share, tap More  &gt; choose an option:</li></ol> <ul data-bbox="583 492 1682 634" style="list-style-type: none"><li data-bbox="583 492 1440 516">• <b>Hide/Show on your map:</b> Display or hide your saved places when looking at the map.</li><li data-bbox="583 532 1058 557">• <b>Share list:</b> Allow others to see your saved list.</li><li data-bbox="583 573 1682 634">• <b>Sharing options:</b> You can make your list public, private, or shared. To let anyone with the link see your list, tap <b>Shared</b>. To let anyone find and follow your list, tap <b>Public</b>.</li></ul> <h3 data-bbox="541 699 764 737">Follow a list</h3> <p data-bbox="541 768 1728 824">If you follow a list made by someone else, their saved places will show up in Your Places. The places will also appear as suggested locations in Google Maps.</p> <h3 data-bbox="541 878 915 915">Follow a list using a link</h3> <ol data-bbox="554 938 1352 1047" style="list-style-type: none"><li data-bbox="554 938 957 963">1. Tap on the link you received to open it.</li><li data-bbox="554 979 1272 1003">2. Tap <b>Follow</b>. This list will now be added to the group of lists you follow.</li><li data-bbox="554 1019 1352 1047">3. <b>Optional:</b> To unfollow a list someone shared with you, tap the list &gt; <b>Following</b>.</li></ol> <h3 data-bbox="541 1101 924 1138">See lists made by others</h3> <p data-bbox="541 1161 1335 1185">If a user has any Google Maps lists that were made public, you can follow them.</p> <ol data-bbox="554 1216 1136 1325" style="list-style-type: none"><li data-bbox="554 1216 1136 1240">1. Tap on the name of a user whose list you want to follow.</li><li data-bbox="554 1256 680 1281">2. Tap <b>Lists</b>.</li><li data-bbox="554 1297 1136 1325">3. Tap on the list you want to follow &gt; More  &gt; <b>Follow</b>.</li></ol> <p data-bbox="512 1341 1898 1401"><a href="https://support.google.com/maps/answer/7280933?hl=en&amp;ref_topic=7301134&amp;co=GENIE.Platform%3DAndroid&amp;oco=1">https://support.google.com/maps/answer/7280933?hl=en&amp;ref_topic=7301134&amp;co=GENIE.Platform%3DAndroid&amp;oco=1</a></p>

**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

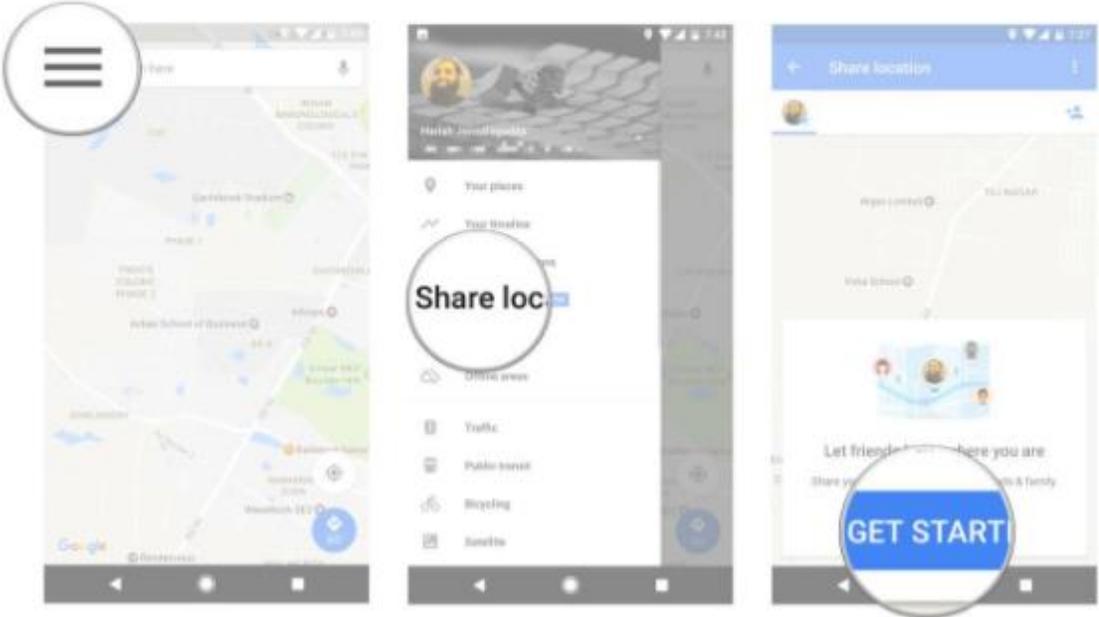
US9408055B2	HTC
	<p data-bbox="520 240 1150 272"><b>How to share your location in Google Maps</b></p> <ol data-bbox="520 300 1134 389" style="list-style-type: none"> <li>1. Open Google Maps from the app drawer or the home screen.</li> <li>2. Tap the hamburger menu (the three horizontal lines) on the top left corner of the screen.</li> <li>3. Select Share location.</li> </ol>  <ol data-bbox="520 824 1165 933" style="list-style-type: none"> <li>4. Tap Get Started.</li> <li>5. Use the + icon to select a time period or select the <b>Until you turn this off</b> setting to share your location indefinitely.</li> <li>6. Tap Select People.</li> </ol>  <p data-bbox="520 1339 1354 1372"><a href="https://www.androidcentral.com/how-share-location-google-maps">https://www.androidcentral.com/how-share-location-google-maps</a></p>

**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

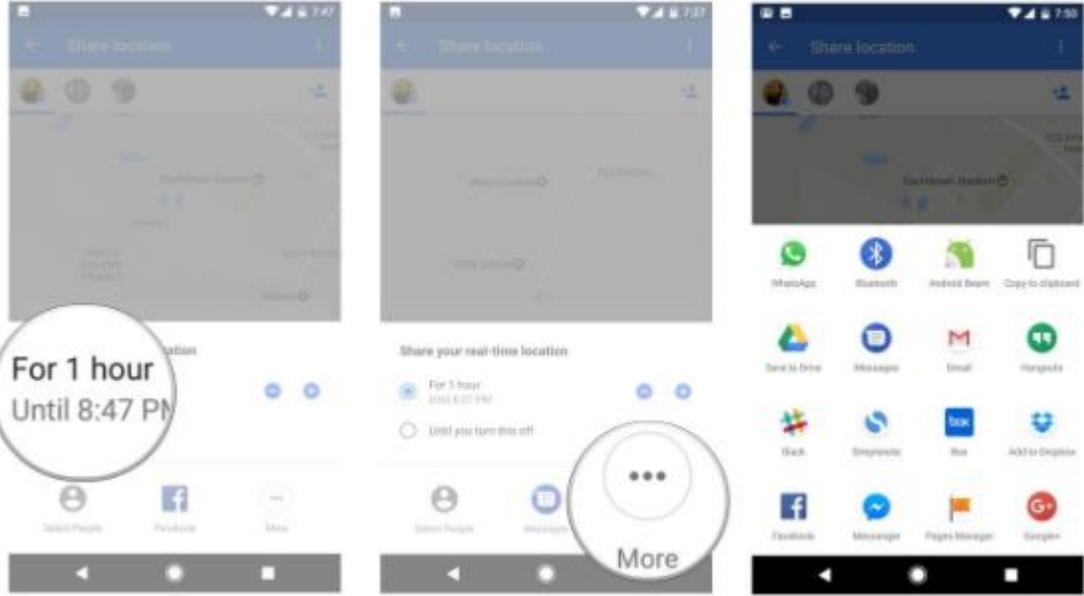
US9408055B2	HTC
	<p data-bbox="527 253 1577 310">7. You'll see a list of your frequent contacts at the top, along with a full list of contacts. Pick the contacts by tapping their name.</p> <p data-bbox="527 339 1457 367">8. Once you've selected the contacts you want to share your location to, tap <b>Share</b>.</p> <p data-bbox="527 396 1419 423">9. You'll see a message saying that the selected contact can view your location.</p>  <p data-bbox="510 1065 1356 1092"><a href="https://www.androidcentral.com/how-share-location-google-maps">https://www.androidcentral.com/how-share-location-google-maps</a></p>



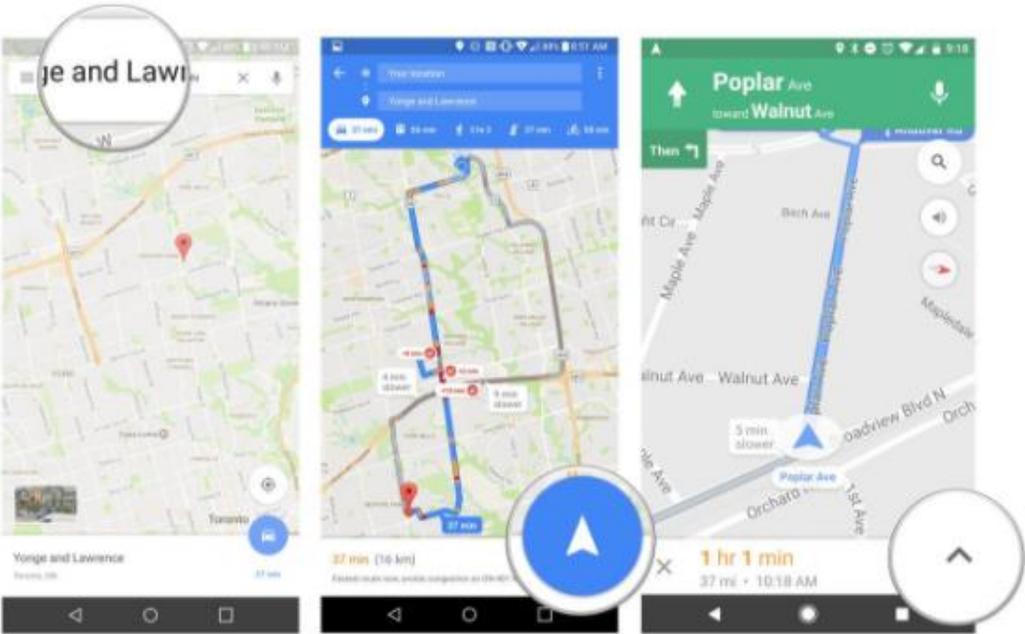
## Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products

US9408055B2	HTC
	<h3 data-bbox="527 245 1255 289">How to create a shareable link</h3> <p data-bbox="527 334 1461 358">You can also create a link and use it to share your location easily. Here's how to do it:</p> <ol data-bbox="527 412 1234 548" style="list-style-type: none"><li data-bbox="527 412 1234 440">1. Tap the hamburger menu on the top left corner of the screen.</li><li data-bbox="527 467 800 495">2. Select Share location.</li><li data-bbox="527 522 737 550">3. Tap Get Started.</li></ol>  <p data-bbox="512 1232 1356 1263"><a href="https://www.androidcentral.com/how-share-location-google-maps">https://www.androidcentral.com/how-share-location-google-maps</a></p>

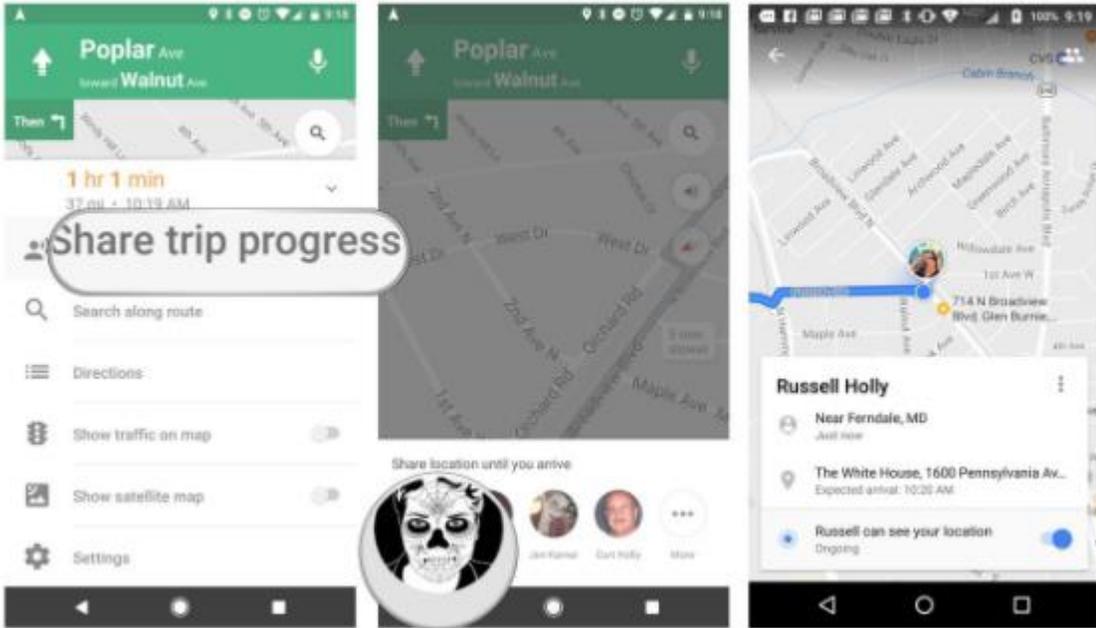
**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC
	<p data-bbox="520 245 1213 272">4. Select the <b>amount of time</b> you want to share your location.</p> <p data-bbox="520 302 680 329">5. Tap <b>More</b>.</p> <p data-bbox="520 358 1633 418">6. Select <b>your app of choice</b> to create and send a <b>unique URL</b> that broadcasts your current location. You can email it, send the link via Messenger, or even tweet it to the intended recipient.</p>  <p data-bbox="506 1084 1360 1117"><a href="https://www.androidcentral.com/how-share-location-google-maps">https://www.androidcentral.com/how-share-location-google-maps</a></p>

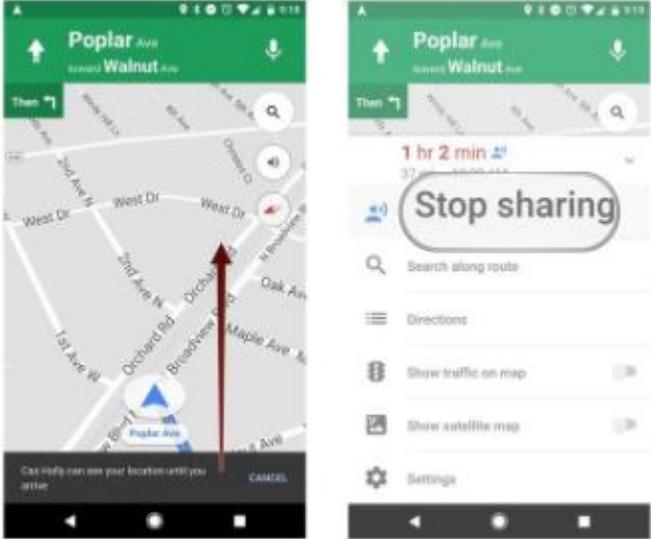
**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC
	<h2 data-bbox="527 240 1423 337">How to share your navigation directions while you walk, drive or transit</h2> <p data-bbox="527 375 1556 464">One of the best ways to share your location with someone is during a drive, walk or transit. If you're meeting a friend or family member somewhere, or navigating towards their home, Google Maps lets you share your location with them for the duration of the trip. It's magic!</p> <ol data-bbox="527 513 1394 643" style="list-style-type: none"><li>1. In the <b>search bar</b> enter your destination.</li><li>2. Pick your navigation type (drive, transit, walk) and press the blue <b>navigate</b> button.</li><li>3. Tap the arrow next to the <b>time-to-destination</b> number at the bottom of the screen.</li></ol>  <p data-bbox="512 1328 1356 1360"><a href="https://www.androidcentral.com/how-share-location-google-maps">https://www.androidcentral.com/how-share-location-google-maps</a></p>

**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC
	<p data-bbox="527 272 840 305">4. Tap Share trip progress.</p> <p data-bbox="527 329 1150 362">5. Choose one or more contacts to share trip progress.</p>  <p data-bbox="527 1062 1344 1094">You can also stop sharing your location with someone before a trip ends.</p> <p data-bbox="510 1101 1358 1133"><a href="https://www.androidcentral.com/how-share-location-google-maps">https://www.androidcentral.com/how-share-location-google-maps</a></p>

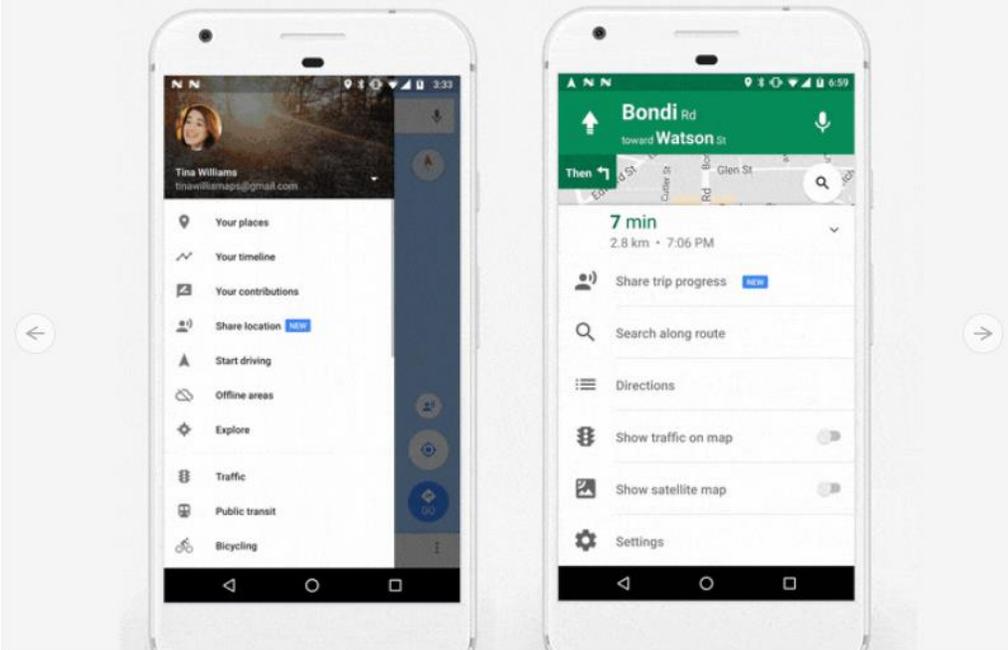
**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC
	<ol style="list-style-type: none"><li data-bbox="533 245 1465 272">1. Tap the arrow next to the <b>time-to-destination</b> number at the bottom of the screen.</li><li data-bbox="533 302 768 329">2. Tap Stop sharing.</li></ol> <div data-bbox="747 383 1398 922"></div> <p data-bbox="541 976 638 1003">That's it!</p> <p data-bbox="541 1045 1608 1073">Are you excited that location sharing is back in Google Maps? How often do you use the feature?</p> <p data-bbox="512 1084 1356 1112"><a href="https://www.androidcentral.com/how-share-location-google-maps">https://www.androidcentral.com/how-share-location-google-maps</a></p> <p data-bbox="512 1195 1419 1222">As shown below, a group may also be defined within Google Contacts.</p>

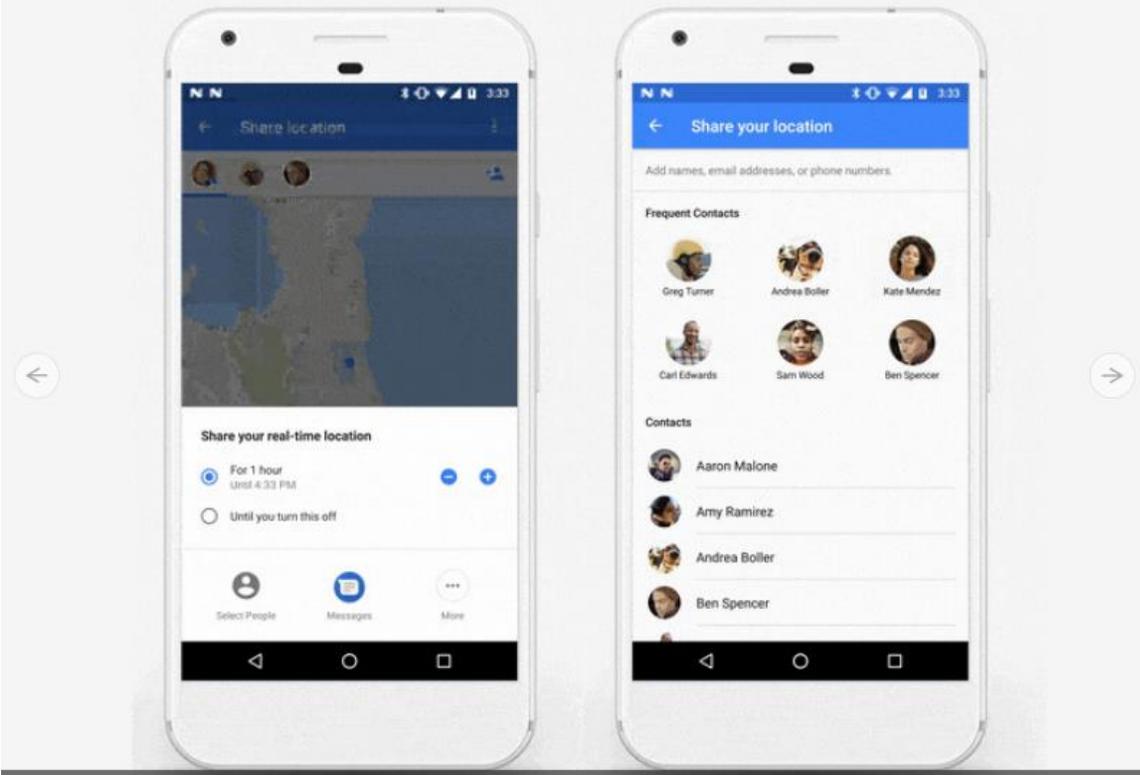
## Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products

US9408055B2	HTC
	<h3 data-bbox="541 235 892 277">See your contacts</h3> <ol data-bbox="552 305 968 375" style="list-style-type: none"><li data-bbox="552 305 968 334">1. Open your device's Contacts app .</li><li data-bbox="552 350 730 375">2. Tap Menu .</li></ol> <ul data-bbox="552 407 1738 602" style="list-style-type: none"><li data-bbox="552 407 1115 431">• <b>See contacts by label:</b> Choose a label from the list.</li><li data-bbox="552 451 1367 475">• <b>See contacts for another account:</b> Tap Down arrow  &gt; pick an account.</li><li data-bbox="552 495 1213 519">• <b>See the contacts for all your accounts:</b> Choose <b>All contacts</b>.</li></ul> <p data-bbox="569 532 1738 557"><b>Tip:</b> If you have multiple contacts with the same information, the information will be grouped into one contact.</p> <ul data-bbox="552 576 1360 602" style="list-style-type: none"><li data-bbox="552 576 1360 602">• <b>See your Google Account contacts on the web:</b> Go to <a href="#">Google Contacts</a> .</li></ul> <p data-bbox="510 634 1535 659"><a href="https://support.google.com/android/answer/6118731?hl=en&amp;ref_topic=6118711">https://support.google.com/android/answer/6118731?hl=en&amp;ref_topic=6118711</a></p> <h3 data-bbox="541 711 892 753">Label your contacts</h3> <p data-bbox="541 776 982 800">You can group contacts together using labels.</p> <ol data-bbox="552 829 930 935" style="list-style-type: none"><li data-bbox="552 829 930 854">1. Open your device's Contacts app .</li><li data-bbox="552 870 863 894">2. Tap Menu  &gt; <b>Create label</b>.</li><li data-bbox="552 911 873 935">3. Enter a label name and tap <b>Ok</b>.</li></ol> <ul data-bbox="552 967 1717 1024" style="list-style-type: none"><li data-bbox="552 967 1234 992">• <b>Add one contact to a label:</b> Tap Add contact  &gt; choose a contact.</li><li data-bbox="552 1008 1717 1024">• <b>Add multiple contacts to a label:</b> Tap Add contact  &gt; touch and hold a contact &gt; tap the other contacts &gt; tap <b>Add</b>.</li></ul> <p data-bbox="510 1040 1535 1065"><a href="https://support.google.com/android/answer/6118731?hl=en&amp;ref_topic=6118711">https://support.google.com/android/answer/6118731?hl=en&amp;ref_topic=6118711</a></p> <h3 data-bbox="552 1130 940 1172">Share your contacts</h3> <ol data-bbox="552 1195 1045 1357" style="list-style-type: none"><li data-bbox="552 1195 978 1219">1. Open your device's Contacts app .</li><li data-bbox="552 1235 842 1260">2. Tap a contact in the list.</li><li data-bbox="552 1276 831 1300">3. Tap More  &gt; <b>Share</b>.</li><li data-bbox="552 1317 1045 1357">4. Choose how you want to share the contact.</li></ol> <p data-bbox="510 1373 1535 1398"><a href="https://support.google.com/android/answer/6118731?hl=en&amp;ref_topic=6118711">https://support.google.com/android/answer/6118731?hl=en&amp;ref_topic=6118711</a></p>

**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

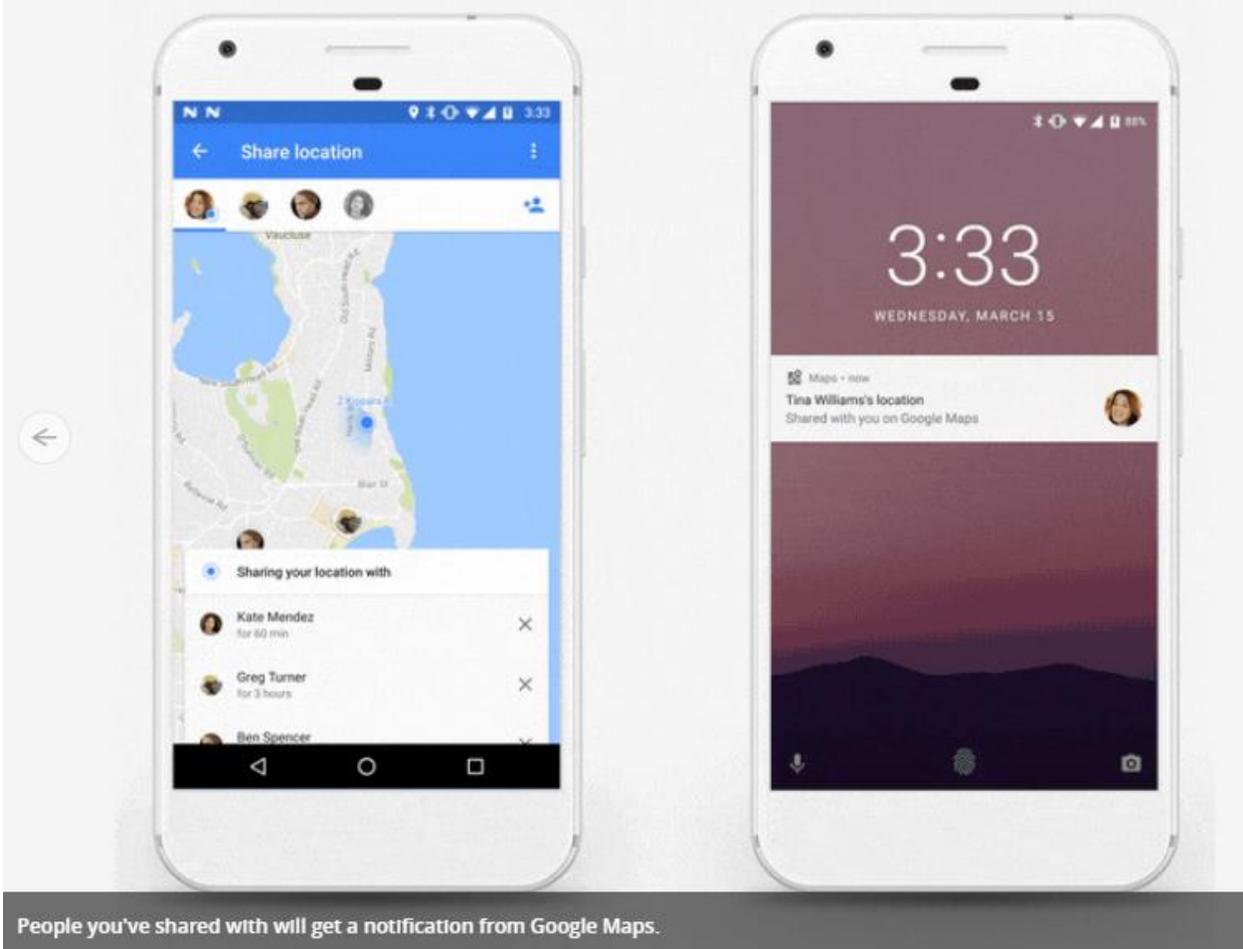
US9408055B2	HTC
	 <p data-bbox="514 933 1522 990">Ways to start location sharing: (left) from the app navigation drawer, and (right) from the menu, you can drag up while navigating somewhere.</p> <p data-bbox="514 998 1522 1023"><a href="https://arstechnica.com/gadgets/2017/03/location-sharing-finally-returns-to-google-maps/">https://arstechnica.com/gadgets/2017/03/location-sharing-finally-returns-to-google-maps/</a></p>

**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

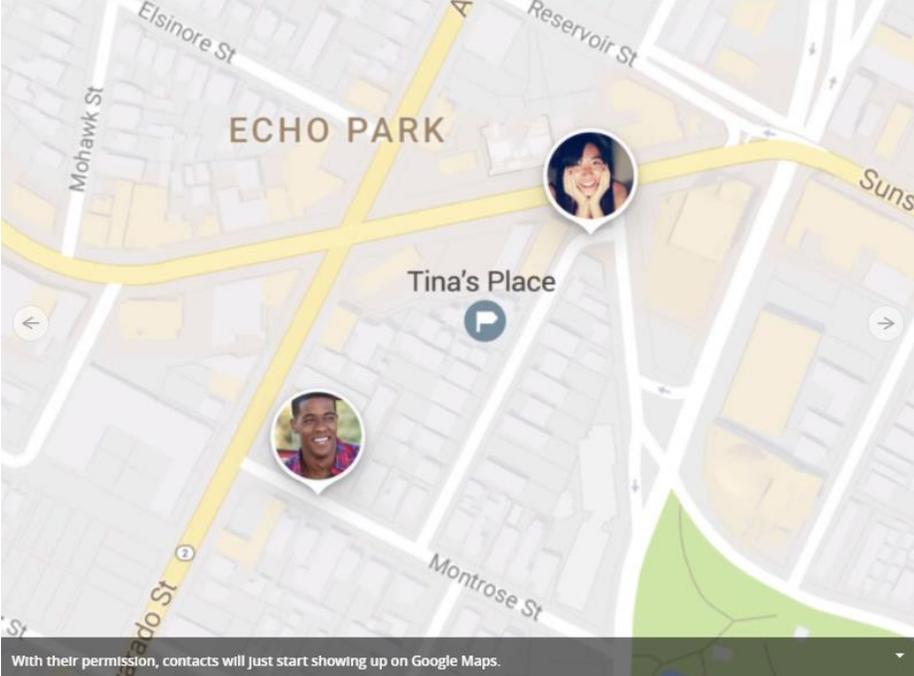
US9408055B2	HTC
	 <p data-bbox="514 1023 1654 1063">You can pick how long you want to share your location for, and with whom you want to share it.</p> <p data-bbox="514 1063 1654 1091"><a href="https://arstechnica.com/gadgets/2017/03/location-sharing-finally-returns-to-google-maps/">https://arstechnica.com/gadgets/2017/03/location-sharing-finally-returns-to-google-maps/</a></p>



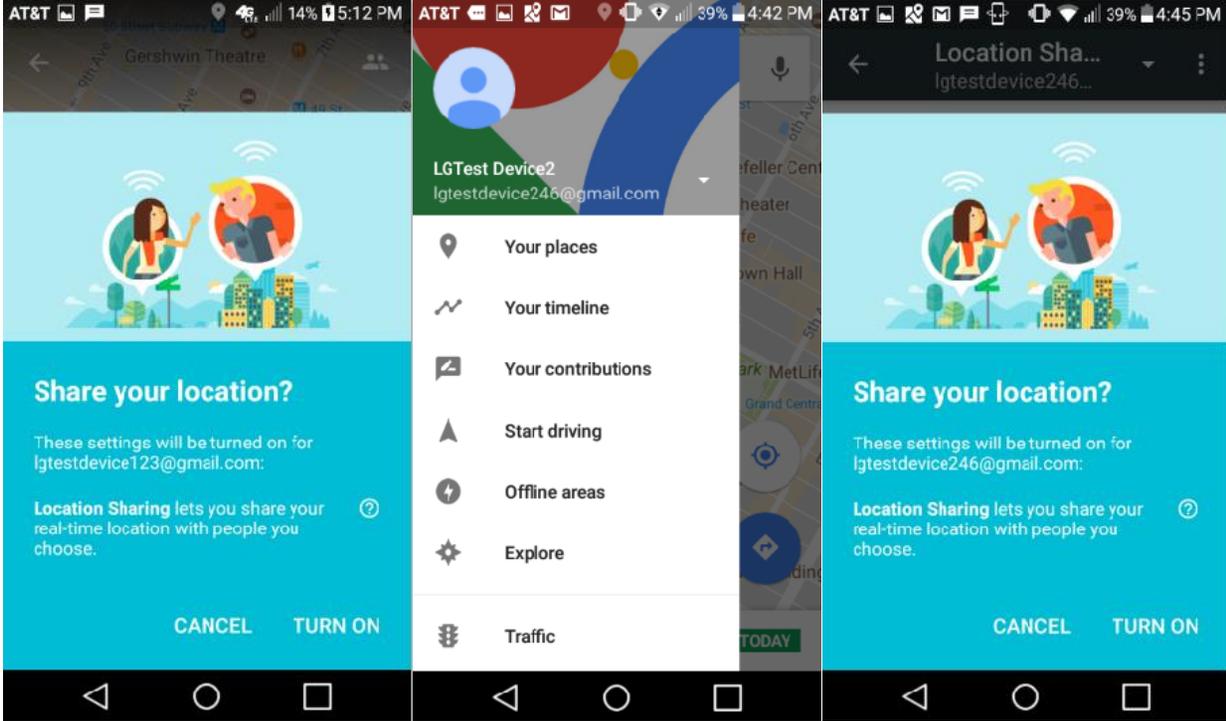
**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC
	 <p data-bbox="512 1143 1176 1170">People you've shared with will get a notification from Google Maps.</p> <p data-bbox="512 1187 1656 1219"><a href="https://arstechnica.com/gadgets/2017/03/location-sharing-finally-returns-to-google-maps/">https://arstechnica.com/gadgets/2017/03/location-sharing-finally-returns-to-google-maps/</a></p>

**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC
	 <p data-bbox="514 885 1428 909">With their permission, contacts will just start showing up on Google Maps.</p> <p data-bbox="514 917 1659 950"><a href="https://arstechnica.com/gadgets/2017/03/location-sharing-finally-returns-to-google-maps/">https://arstechnica.com/gadgets/2017/03/location-sharing-finally-returns-to-google-maps/</a></p> <p data-bbox="514 982 1018 1015"><b><u>Exemplary Google Maps Screenshots</u></b></p>

**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC
	 <p>The image displays three screenshots of an HTC mobile application interface. The first screenshot (left) shows a 'Share your location?' dialog box with a blue background and a light blue header. It features an illustration of two people in speech bubbles. The text reads: 'These settings will be turned on for lgtestdevice123@gmail.com: Location Sharing lets you share your real-time location with people you choose.' At the bottom are 'CANCEL' and 'TURN ON' buttons. The second screenshot (middle) shows a similar dialog box but with a white menu overlay. The menu items are: 'Your places', 'Your timeline', 'Your contributions', 'Start driving', 'Offline areas', 'Explore', and 'Traffic'. The third screenshot (right) shows the 'Share your location?' dialog box for 'lgtestdevice246@gmail.com' with the same 'CANCEL' and 'TURN ON' buttons. All screenshots show an AT&amp;T carrier and various status icons at the top.</p> <p>Exemplary Source Code: The above functionality is performed at least in part by the following publicly available source code and/or source code that invokes or is invoked by the following source code (or a substantially similar copy compiled and loaded onto the Accused Products by HTC): AGIS reserves the right to amend these contentions to include additional source code as discovery progresses and as additional source code is made available.</p>

**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC
	<h2 data-bbox="527 240 1016 293">Contacts Provider</h2> <p data-bbox="527 329 1472 591">The Contacts Provider is a powerful and flexible Android component that manages the device's central repository of data about people. The Contacts Provider is the source of data you see in the device's contacts application, and you can also access its data in your own application and transfer data between the device and online services. The provider accommodates a wide range of data sources and tries to manage as much data as possible for each person, with the result that its organization is complex. Because of this, the provider's API includes an extensive set of contract classes and interfaces that facilitate both data retrieval and modification.</p> <p data-bbox="527 626 835 646">This guide describes the following:</p> <ul data-bbox="527 675 1373 850" style="list-style-type: none"><li data-bbox="527 675 806 695">• The basic provider structure.</li><li data-bbox="527 724 894 743">• How to retrieve data from the provider.</li><li data-bbox="527 773 863 792">• How to modify data in the provider.</li><li data-bbox="527 821 1373 841">• How to write a sync adapter for synchronizing data from your server to the Contacts Provider.</li></ul> <p data-bbox="512 863 1486 889"><a href="https://developer.android.com/guide/topics/providers/contacts-provider.html">https://developer.android.com/guide/topics/providers/contacts-provider.html</a></p>

**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC
	<p><b>Overview</b></p> <p>ContactsContract defines an extensible database of contact-related information. Contact information is stored in a three-tier data model:</p> <ul style="list-style-type: none"> <li>• A row in the <code>ContactsContract.Data</code> table can store any kind of personal data, such as a phone number or email addresses. The set of data kinds that can be stored in this table is open-ended. There is a predefined set of common kinds, but any application can add its own data kinds.</li> <li>• A row in the <code>ContactsContract.RawContacts</code> table represents a set of data describing a person and associated with a single account (for example, one of the user's Gmail accounts).</li> <li>• A row in the <code>ContactsContract.Contacts</code> table represents an aggregate of one or more RawContacts presumably describing the same person. When data in or associated with the RawContacts table is changed, the affected aggregate contacts are updated as necessary.</li> </ul> <p>Other tables include:</p> <ul style="list-style-type: none"> <li>• <code>ContactsContract.Groups</code>, which contains information about raw contact groups such as Gmail contact groups. The current API does not support the notion of groups spanning multiple accounts.</li> <li>• <code>ContactsContract.StatusUpdates</code>, which contains social status updates including IM availability.</li> <li>• <code>ContactsContract.AggregationExceptions</code>, which is used for manual aggregation and disaggregation of raw contacts</li> <li>• <code>ContactsContract.Settings</code>, which contains visibility and sync settings for accounts and groups.</li> <li>• <code>ContactsContract.SyncState</code>, which contains free-form data maintained on behalf of sync adapters</li> <li>• <code>ContactsContract.PhoneLookup</code>, which is used for quick caller-ID lookup</li> </ul> <p><a href="https://developer.android.com/reference/android/provider/ContactsContract.html">https://developer.android.com/reference/android/provider/ContactsContract.html</a></p> <p><b>Data</b></p> <p>As noted previously, the data for a raw contact is stored in a <code>ContactsContract.Data</code> row that is linked to the raw contact's <code>_id</code> value. This allows a single raw contact to have multiple instances of the same type of data such as email addresses or phone numbers. For example, if "Thomas Higginson" for <code>emilyd@gmail.com</code> (the raw contact row for Thomas Higginson associated with the Google account <code>emilyd@gmail.com</code>) has a home email address of <code>thigg@gmail.com</code> and a work email address of <code>thomas.higginson@gmail.com</code>, the Contacts Provider stores the two email address rows and links them both to the raw contact.</p> <p>Notice that different types of data are stored in this single table. Display name, phone number, email, postal address, photo, and website detail rows are all found in the <code>ContactsContract.Data</code> table. To help manage this, the <code>ContactsContract.Data</code> table has some columns with descriptive names, and others with generic names. The contents of a descriptive-name column have the same meaning regardless of the type of data in the row, while the contents of a generic-name column have different meanings depending on the type of data.</p> <p><a href="https://developer.android.com/guide/topics/providers/contacts-provider.html">https://developer.android.com/guide/topics/providers/contacts-provider.html</a></p>

**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC														
	<table border="1"> <thead> <tr> <th>Task</th> <th>Action</th> <th>Data</th> <th>MIME type</th> <th>Notes</th> </tr> </thead> <tbody> <tr> <td>Pick a contact from a list</td> <td><a href="#">ACTION_PICK</a></td> <td>                     One of:                     <ul style="list-style-type: none"> <li><a href="#">Contacts.CONTENT_URI</a>, which displays a list of contacts.</li> <li><a href="#">Phone.CONTENT_URI</a>, which displays a list of phone numbers for a raw contact.</li> <li><a href="#">StructuredPostal.CONTENT_URI</a>, which displays a list of postal addresses for a raw contact.</li> <li><a href="#">Email.CONTENT_URI</a>, which displays a list of email addresses for a raw contact.</li> </ul> </td> <td>Not used</td> <td>                     Displays a list of raw contacts or a list of data from a raw contact, depending on the content URI type you supply. Call <a href="#">startActivityForResult()</a>, which returns the content URI of the selected row. The form of the URI is the table's content URI with the row's <a href="#">LOOKUP_ID</a> appended to it. The device's contacts app delegates read and write permissions to this content URI for the life of your activity. See the <a href="#">Content Provider Basics</a> guide for more details.                 </td> </tr> </tbody> </table>	Task	Action	Data	MIME type	Notes	Pick a contact from a list	<a href="#">ACTION_PICK</a>	One of: <ul style="list-style-type: none"> <li><a href="#">Contacts.CONTENT_URI</a>, which displays a list of contacts.</li> <li><a href="#">Phone.CONTENT_URI</a>, which displays a list of phone numbers for a raw contact.</li> <li><a href="#">StructuredPostal.CONTENT_URI</a>, which displays a list of postal addresses for a raw contact.</li> <li><a href="#">Email.CONTENT_URI</a>, which displays a list of email addresses for a raw contact.</li> </ul>	Not used	Displays a list of raw contacts or a list of data from a raw contact, depending on the content URI type you supply. Call <a href="#">startActivityForResult()</a> , which returns the content URI of the selected row. The form of the URI is the table's content URI with the row's <a href="#">LOOKUP_ID</a> appended to it. The device's contacts app delegates read and write permissions to this content URI for the life of your activity. See the <a href="#">Content Provider Basics</a> guide for more details.				
Task	Action	Data	MIME type	Notes											
Pick a contact from a list	<a href="#">ACTION_PICK</a>	One of: <ul style="list-style-type: none"> <li><a href="#">Contacts.CONTENT_URI</a>, which displays a list of contacts.</li> <li><a href="#">Phone.CONTENT_URI</a>, which displays a list of phone numbers for a raw contact.</li> <li><a href="#">StructuredPostal.CONTENT_URI</a>, which displays a list of postal addresses for a raw contact.</li> <li><a href="#">Email.CONTENT_URI</a>, which displays a list of email addresses for a raw contact.</li> </ul>	Not used	Displays a list of raw contacts or a list of data from a raw contact, depending on the content URI type you supply. Call <a href="#">startActivityForResult()</a> , which returns the content URI of the selected row. The form of the URI is the table's content URI with the row's <a href="#">LOOKUP_ID</a> appended to it. The device's contacts app delegates read and write permissions to this content URI for the life of your activity. See the <a href="#">Content Provider Basics</a> guide for more details.											
<a href="https://developer.android.com/guide/topics/providers/contacts-provider.html">https://developer.android.com/guide/topics/providers/contacts-provider.html</a>															

**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC
	<pre>25  * Parsed form of the intent sent to the Contacts application. 26  */ 27  public class ContactsRequest { 28 29      /** Default mode: browse contacts */ 30      public static final int ACTION_DEFAULT = 10; 31 32      /** Show all contacts */ 33      public static final int ACTION_ALL_CONTACTS = 15; 34 35      /** Show all contacts with phone numbers */ 36      public static final int ACTION_CONTACTS_WITH_PHONES = 17; 37 38      /** Show contents of a specific group */ 39      public static final int ACTION_GROUP = 20; 40 41      /** Show all starred contacts */ 42      public static final int ACTION_STARRED = 30; 43 44      /** Show frequently contacted contacts */ 45      public static final int ACTION_FREQUENT = 40; 46 47      /** Show starred and the frequent */ 48      public static final int ACTION_STREQUENT = 50; 49 50      /** Show all contacts and pick them when clicking */ 51      public static final int ACTION_PICK_CONTACT = 60; 52 53      /** Show all contacts as well as the option to create a new one */ 54      public static final int ACTION_PICK_OR_CREATE_CONTACT = 70; 55 56      /** Show all contacts and pick them for edit when clicking, and allow creating a new contact */ 57      public static final int ACTION_INSERT_OR_EDIT_CONTACT = 80;</pre> <p><a href="https://android.googlesource.com/platform/packages/apps/Contacts/+nougat-mr1-release/src/com/android/contacts/list/ContactsRequest.java">https://android.googlesource.com/platform/packages/apps/Contacts/+nougat-mr1-release/src/com/android/contacts/list/ContactsRequest.java</a></p>

**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC
	<pre>59  /** Show all phone numbers and pick them when clicking */ 60  public static final int ACTION_PICK_PHONE = 90; 61 62  /** Show all postal addresses and pick them when clicking */ 63  public static final int ACTION_PICK_POSTAL = 100; 64 65  /** Show all postal addresses and pick them when clicking */ 66  public static final int ACTION_PICK_EMAIL = 105; 67 68  /** Show all contacts and create a shortcut for the picked contact */ 69  public static final int ACTION_CREATE_SHORTCUT_CONTACT = 110; 70 71  /** Show all phone numbers and create a call shortcut for the picked number */ 72  public static final int ACTION_CREATE_SHORTCUT_CALL = 120; 73 74  /** Show all phone numbers and create an SMS shortcut for the picked number */ 75  public static final int ACTION_CREATE_SHORTCUT_SMS = 130; 76 77  /** Show all contacts and activate the specified one */ 78  public static final int ACTION_VIEW_CONTACT = 140; 79 80  /** Show contacts recommended for joining with a specified target contact */ 81  public static final int ACTION_PICK_JOIN = 150;</pre> <p><a href="https://android.googlesource.com/platform/packages/apps/Contacts+/nougat-mr1-release/src/com/android/contacts/list/ContactsRequest.java">https://android.googlesource.com/platform/packages/apps/Contacts+/nougat-mr1-release/src/com/android/contacts/list/ContactsRequest.java</a></p>



**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC
	<pre> 104      * Displays a list to browse contacts. 105      */ 106      public class PeopleActivity extends ContactsActivity implements 107          View.OnCreateContextMenuListener, 108          View.OnClickListener, 109          ActionBarAdapter.Listener, 110          DialogManager.DialogShowingViewActivity, 111          ContactListFilterController.ContactListFilterListener, 112          ProviderStatusListener, 113          MultiContactDeleteListener, 114          JoinContactsListener { </pre> <p data-bbox="506 662 1570 727"><a href="https://android.googlesource.com/platform/packages/apps/Contacts/+/nougat-mr1-release/src/com/android/contacts/activities/PeopleActivity.java">https://android.googlesource.com/platform/packages/apps/Contacts/+/nougat-mr1-release/src/com/android/contacts/activities/PeopleActivity.java</a></p> <pre> 145      * Showing a list of Contacts. Also used for showing search results in search mode. 146      */ 147      private MultiSelectContactsListFragment mAllFragment; 148      private ContactTileListFragment mFavoritesFragment; </pre> <p data-bbox="506 873 1570 937"><a href="https://android.googlesource.com/platform/packages/apps/Contacts/+/nougat-mr1-release/src/com/android/contacts/activities/PeopleActivity.java">https://android.googlesource.com/platform/packages/apps/Contacts/+/nougat-mr1-release/src/com/android/contacts/activities/PeopleActivity.java</a></p>

**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC
	<pre>458     private void configureFragments(boolean fromRequest) { 459         if (fromRequest) { 460             ContactListFilter filter = null; 461             int actionCode = mRequest.getActionCode(); 462             boolean searchMode = mRequest.isSearchMode(); 463             final int tabToOpen; 464             switch (actionCode) { 465                 case ContactsRequest.ACTION_ALL_CONTACTS: 466                     filter = ContactListFilter.createFilterWithType( 467                         ContactListFilter.FILTER_TYPE_ALL_ACCOUNTS); 468                     tabToOpen = TabState.ALL; 469                     break; 470                 case ContactsRequest.ACTION_CONTACTS_WITH_PHONES: 471                     filter = ContactListFilter.createFilterWithType( 472                         ContactListFilter.FILTER_TYPE_WITH_PHONE_NUMBERS_ONLY); 473                     tabToOpen = TabState.ALL; 474                     break; 475 476                 case ContactsRequest.ACTION_FREQUENT: 477                 case ContactsRequest.ACTION_STREQUENT: 478                 case ContactsRequest.ACTION_STARRED: 479                     tabToOpen = TabState.FAVORITES; 480                     break; 481                 case ContactsRequest.ACTION_VIEW_CONTACT: 482                     tabToOpen = TabState.ALL; 483                     break; 484                 default: 485                     tabToOpen = -1; 486                     break; 487             } </pre> <p data-bbox="506 1321 1566 1390"><a href="https://android.googlesource.com/platform/packages/apps/Contacts+/nougat-mr1-release/src/com/android/contacts/activities/PeopleActivity.java">https://android.googlesource.com/platform/packages/apps/Contacts+/nougat-mr1-release/src/com/android/contacts/activities/PeopleActivity.java</a></p>

**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC
	<pre>488         if (tabToOpen != -1) { 489             mActionBarAdapter.setCurrentTab(tabToOpen); 490         } 491 492         if (filter != null) { 493             mContactListFilterController.setContactListFilter(filter, false); 494             searchMode = false; 495         } 496 497         if (mRequest.getContactUri() != null) { 498             searchMode = false; 499         } 500 501         mActionBarAdapter.setSearchMode(searchMode); 502         configureContactListFragmentForRequest(); 503     } 504 505     configureContactListFragment(); 506 507     invalidateOptionsMenuIfNeeded(); 508 }</pre> <p data-bbox="506 1019 1570 1084"><a href="https://android.googlesource.com/platform/packages/apps/Contacts+/nougat-mr1-release/src/com/android/contacts/activities/PeopleActivity.java">https://android.googlesource.com/platform/packages/apps/Contacts+/nougat-mr1-release/src/com/android/contacts/activities/PeopleActivity.java</a></p>

**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC
	<pre>35 public class ProfileAndContactsLoader extends CursorLoader { 36 37     private boolean mLoadProfile; 38 39     private String[] mProjection; 40 41     private Uri mExtraUri; 42     private String[] mExtraProjection; 43     private String mExtraSelection; 44     private String[] mExtraSelectionArgs; 45     private boolean mMergeExtraContactsAfterPrimary; 46 47     public ProfileAndContactsLoader(Context context) { 48         super(context); 49     }</pre> <p><a href="https://android.googlesource.com/platform/packages/apps/ContactsCommon/+/nougat-mr1-release/src/com/android/contacts/common/list/ProfileAndContactsLoader.java">https://android.googlesource.com/platform/packages/apps/ContactsCommon/+/nougat-mr1-release/src/com/android/contacts/common/list/ProfileAndContactsLoader.java</a></p>

**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC
	<pre>35 public final class GroupMemberLoader extends CursorLoader { 36 37     public static class GroupEditorQuery { 38         private static final String[] PROJECTION = new String[] { 39             Data.CONTACT_ID,           // 0 40             Data.RAW_CONTACT_ID,      // 1 41             Data.DISPLAY_NAME_PRIMARY, // 2 42             Data.PHOTO_URI,           // 3 43             Data.LOOKUP_KEY,          // 4 44         }; 45 46         public static final int CONTACT_ID           = 0; 47         public static final int RAW_CONTACT_ID      = 1; 48         public static final int CONTACT_DISPLAY_NAME_PRIMARY = 2; 49         public static final int CONTACT_PHOTO_URI   = 3; 50         public static final int CONTACT_LOOKUP_KEY  = 4; 51     } 52 53     public static class GroupDetailQuery { 54         private static final String[] PROJECTION = new String[] { 55             Data.CONTACT_ID,           // 0 56             Data.PHOTO_URI,           // 1 57             Data.LOOKUP_KEY,          // 2 58             Data.DISPLAY_NAME_PRIMARY, // 3 59             Data.CONTACT_PRESENCE,    // 4 60             Data.CONTACT_STATUS,      // 5 61         }; 62 63         public static final int CONTACT_ID           = 0; 64         public static final int CONTACT_PHOTO_URI   = 1; 65         public static final int CONTACT_LOOKUP_KEY  = 2; 66         public static final int CONTACT_DISPLAY_NAME_PRIMARY = 3; 67         public static final int CONTACT_PRESENCE_STATUS = 4; 68         public static final int CONTACT_STATUS      = 5; 69     } 70 71     private final long mGroupId;</pre> <p><a href="https://android.googlesource.com/platform/packages/apps/Contacts/+/nougat-mr1-release/src/com/android/contacts/GroupMemberLoader.java">https://android.googlesource.com/platform/packages/apps/Contacts/+/nougat-mr1-release/src/com/android/contacts/GroupMemberLoader.java</a></p>

**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC
	<pre>24  * Group loader for the group list that includes details such as the number of contacts per group 25  * and number of groups per account. This list is sorted by account type, account name, where the 26  * group names are in alphabetical order. Note that the list excludes default, favorite, and deleted 27  * groups. 28  */ 29  public final class GroupListLoader extends CursorLoader { 30 31      private final static String[] COLUMNS = new String[] { 32          Groups.ACCOUNT_NAME, 33          Groups.ACCOUNT_TYPE, 34          Groups.DATA_SET, 35          Groups._ID, 36          Groups.TITLE, 37          Groups.SUMMARY_COUNT, 38      }; 39 40      public final static int ACCOUNT_NAME = 0; 41      public final static int ACCOUNT_TYPE = 1; 42      public final static int DATA_SET = 2; 43      public final static int GROUP_ID = 3; 44      public final static int TITLE = 4; 45      public final static int MEMBER_COUNT = 5; 46 47      private static final Uri GROUP_LIST_URI = Groups.CONTENT_SUMMARY_URI; 48 49      public GroupListLoader(Context context) { 50          super(context, GROUP_LIST_URI, COLUMNS, Groups.ACCOUNT_TYPE + " NOT NULL AND " 51              + Groups.ACCOUNT_NAME + " NOT NULL AND " + Groups.AUTO_ADD + "=0 AND " + 52              Groups.FAVORITES + "=0 AND " + Groups.DELETED + "=0", null, 53              Groups.ACCOUNT_TYPE + ", " + Groups.ACCOUNT_NAME + ", " + Groups.DATA_SET + ", " + 54              Groups.TITLE + " COLLATE LOCALIZED ASC"); 55      } 56  }</pre> <p><a href="https://android.googlesource.com/platform/packages/apps/Contacts+/nougat-mr1-release/src/com/android/contacts/GroupListLoader.java">https://android.googlesource.com/platform/packages/apps/Contacts+/nougat-mr1-release/src/com/android/contacts/GroupListLoader.java</a></p>

**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC
	<pre> 24  * Group meta-data loader. Loads all groups or just a single group from the 25  * database (if given a {@link Uri}). 26  */ 27  public final class GroupMetaLoader extends CursorLoader { 28 29      private final static String[] COLUMNS = new String[] { 30          Groups.ACCOUNT_NAME, 31          Groups.ACCOUNT_TYPE, 32          Groups.DATA_SET, 33          Groups._ID, 34          Groups.TITLE, 35          Groups.AUTO_ADD, 36          Groups.FAVORITES, 37          Groups.GROUP_IS_READ_ONLY, 38          Groups.DELETED, 39      }; 40 41      public final static int ACCOUNT_NAME = 0; 42      public final static int ACCOUNT_TYPE = 1; 43      public final static int DATA_SET = 2; 44      public final static int GROUP_ID = 3; 45      public final static int TITLE = 4; 46      public final static int AUTO_ADD = 5; 47      public final static int FAVORITES = 6; 48      public final static int IS_READ_ONLY = 7; 49      public final static int DELETED = 8; 50 51      public GroupMetaLoader(Context context, Uri groupUri) { 52          super(context, ensureIsGroupUri(groupUri), COLUMNS, Groups.ACCOUNT_TYPE + " NOT NULL AND " 53              + Groups.ACCOUNT_NAME + " NOT NULL", null, null); 54      } 55 56      /** 57       * Ensures that this is a valid group URI. If invalid, then an exception is 58       * thrown. Otherwise, the original URI is returned. 59       */ 60      private static Uri ensureIsGroupUri(final Uri groupUri) { 61          // TODO: Fix ContactsProvider2 getType method to resolve the group Uris 62          if (groupUri == null) { 63              throw new IllegalArgumentException("Uri must not be null"); 64          } 65          if (!groupUri.toString().startsWith(Groups.CONTENT_URI.toString())) { 66              throw new IllegalArgumentException("Invalid group Uri: " + groupUri); 67          } 68          return groupUri; 69      } 70  } </pre> <p><a href="https://android.googlesource.com/platform/packages/apps/Contacts/+nougat-mr1-release/src/com/android/contacts/GroupMetaLoader.java">https://android.googlesource.com/platform/packages/apps/Contacts/+nougat-mr1-release/src/com/android/contacts/GroupMetaLoader.java</a></p>

**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC
	<pre>19  * Meta-data for a contact group. We load all groups associated with the contact's 20  * constituent accounts. 21  */ 22  public final class GroupMetaData { 23      private String mAccountName; 24      private String mAccountType; 25      private String mDataSet; 26      private long mGroupId; 27      private String mTitle; 28      private boolean mDefaultGroup; 29      private boolean mFavorites; 30 31      public GroupMetaData(String accountName, String accountType, String dataSet, long groupId, 32                          String title, boolean defaultGroup, boolean favorites) { 33          this.mAccountName = accountName; 34          this.mAccountType = accountType; 35          this.mDataSet = dataSet; 36          this.mGroupId = groupId; 37          this.mTitle = title; 38          this.mDefaultGroup = defaultGroup; 39          this.mFavorites = favorites; 40      } 41 42      public String getAccountName() { 43          return mAccountName; 44      } 45 46      public String getAccountType() { 47          return mAccountType; 48      } 49 50      public String getDataSet() { 51          return mDataSet; 52      } 53 54      public long getGroupId() { 55          return mGroupId; 56      } 57 58      public String getTitle() { 59          return mTitle; 60      } 61 62      public boolean isDefaultGroup() { 63          return mDefaultGroup; 64      } 65 66      public boolean isFavorites() { 67          return mFavorites; 68      } </pre> <p><a href="https://android.googlesource.com/platform/packages/apps/ContactsCommon/+nougat-mr1-release/src/com/android/contacts/common/GroupMetaData.java">https://android.googlesource.com/platform/packages/apps/ContactsCommon/+nougat-mr1-release/src/com/android/contacts/common/GroupMetaData.java</a></p>



**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC
	<pre>44 * Class that sends chat message via SMS. 45 * 46 * The interface emulates a blocking sending similar to making an HTTP request. 47 * It calls the SmsManager to send a (potentially multipart) message and waits 48 * on the sent status on each part. The waiting has a timeout so it won't wait 49 * forever. Once the sent status of all parts received, the call returns. 50 * A successful sending requires success status for all parts. Otherwise, we 51 * pick the highest level of failure as the error for the whole message, which 52 * is used to determine if we need to retry the sending. 53 */ 54 public class SmsSender { 55     private static final String TAG = LogUtil.BUGLE_TAG; 56 57     public static final String EXTRA_PART_ID = "part_id"; 58 59     /* 60     * A map for pending sms messages. The key is the random request UUID. 61     */ 62     private static ConcurrentHashMap&lt;Uri, SendResult&gt; sPendingMessageMap = 63         new ConcurrentHashMap&lt;Uri, SendResult&gt;(); 64 65     private static final Random RANDOM = new Random(); 66 67     // Whether we should send multipart SMS as separate messages 68     private static Boolean sSendMultipartSmsAsSeparateMessages = null; 69</pre> <p><a href="https://android.googlesource.com/platform/packages/apps/Messaging/+nougat-mr1-release/src/com/android/messaging/sms/SmsSender.java">https://android.googlesource.com/platform/packages/apps/Messaging/+nougat-mr1-release/src/com/android/messaging/sms/SmsSender.java</a></p>

**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC
	<pre> 253 // Actually sending the message using SmsManager 254 private static void sendInternal(final Context context, final int subId, String dest, 255     final ArrayList&lt;String&gt; messages, final String serviceCenter, 256     final boolean requireDeliveryReport, final Uri messageUri) throws SmsException { 257     Assert.notNull(context); 258     final SmsManager smsManager = PhoneUtils.get(subId).getSmsManager(); 259     final int messageCount = messages.size(); 260     final ArrayList&lt;PendingIntent&gt; deliveryIntents = new ArrayList&lt;PendingIntent&gt;(messageCount); 261     final ArrayList&lt;PendingIntent&gt; sentIntents = new ArrayList&lt;PendingIntent&gt;(messageCount); 262     for (int i = 0; i &lt; messageCount; i++) { 263         // Make pending intents different for each message part 264         final int partId = (messageCount &lt;= 1 ? 0 : i + 1); 265         if (requireDeliveryReport &amp;&amp; (i == (messageCount - 1))) { 266             // TODO we only care about the delivery status of the last part 267             // Shall we have better tracking of delivery status of all parts? 268             deliveryIntents.add(PendingIntent.getBroadcast( 269                 context, 270                 partId, 271                 getSendStatusIntent(context, SendStatusReceiver.MESSAGE_DELIVERED_ACTION, 272                     messageUri, partId, subId), 273                 0/*flag*/)); 274         } else { 275             deliveryIntents.add(null); 276         } 277         sentIntents.add(PendingIntent.getBroadcast( 278             context, 279             partId, 280             getSendStatusIntent(context, SendStatusReceiver.MESSAGE_SENT_ACTION, 281                 messageUri, partId, subId), 282             0/*flag*/)); 283     } 284     if (sSendMultipartSmsAsSeparateMessages == null) { 285         sSendMultipartSmsAsSeparateMessages = MmsConfig.get(subId) 286             .getSendMultipartSmsAsSeparateMessages(); 287     } </pre> <p><a href="https://android.googlesource.com/platform/packages/apps/Messaging/+nougat-mr1-release/src/com/android/messaging/sms/SmsSender.java">https://android.googlesource.com/platform/packages/apps/Messaging/+nougat-mr1-release/src/com/android/messaging/sms/SmsSender.java</a></p>

**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC
	<pre> 288     try { 289         if (sSendMultipartSmsAsSeparateMessages) { 290             // If multipart sms is not supported, send them as separate messages 291             for (int i = 0; i &lt; messageCount; i++) { 292                 smsManager.sendTextMessage(dest, 293                     serviceCenter, 294                     messages.get(i), 295                     sentIntents.get(i), 296                     deliveryIntents.get(i)); 297             } 298         } else { 299             smsManager.sendMultipartTextMessage( 300                 dest, serviceCenter, messages, sentIntents, deliveryIntents); 301         } 302     } catch (final Exception e) { 303         throw new SmsException("SmsSender: caught exception in sending " + e); 304     } 305 } </pre> <p><a href="https://android.googlesource.com/platform/packages/apps/Messaging/+nougat-mr1-release/src/com/android/messaging/sms/SmsSender.java">https://android.googlesource.com/platform/packages/apps/Messaging/+nougat-mr1-release/src/com/android/messaging/sms/SmsSender.java</a></p> <pre> 56  * Class that receives incoming SMS messages through android.provider.Telephony.SMS_RECEIVED 57  * 58  * This class serves two purposes: 59  * - Process phone verification SMS messages 60  * - Handle SMS messages when the user has enabled us to be the default SMS app (Pre-KLP) 61  */ 62  public final class SmsReceiver extends BroadcastReceiver { 63      private static final String TAG = LogUtil.BUGLE_TAG; 64 65      private static ArrayList&lt;Pattern&gt; sIgnoreSmsPatterns; 66 </pre> <p><a href="https://android.googlesource.com/platform/packages/apps/Messaging/+nougat-mr1-release/src/com/android/messaging/receiver/SmsReceiver.java">https://android.googlesource.com/platform/packages/apps/Messaging/+nougat-mr1-release/src/com/android/messaging/receiver/SmsReceiver.java</a></p>

**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC
	<pre>200 public static void deliverSmsMessages(final Context context, final int subId, 201     final int errorCode, final android.telephony.SmsMessage[] messages) { 202     final ContentValues messageValues = 203         MmsUtils.parseReceivedSmsMessage(context, messages, errorCode); 204 205     LogUtil.v(TAG, "SmsReceiver.deliverSmsMessages"); 206 207     final long nowInMillis = System.currentTimeMillis(); 208     final long receivedTimestampMs = MmsUtils.getMessageDate(messages[0], nowInMillis); 209 210     messageValues.put(Sms.Inbox.DATE, receivedTimestampMs); 211     // Default to unread and unseen for us but ReceiveSmsMessageAction will override 212     // seen for the telephony db. 213     messageValues.put(Sms.Inbox.READ, 0); 214     messageValues.put(Sms.Inbox.SEEN, 0); 215     if (OsUtil.isAtLeastL_MR1()) { 216         messageValues.put(Sms.SUBSCRIPTION_ID, subId); 217     } 218 219     if (messages[0].getMessageClass() == android.telephony.SmsMessage.MessageClass.CLASS_0    220         DebugUtils.debugClassZeroSmsEnabled()) { 221         Factory.get().getUIIntents().launchClassZeroActivity(context, messageValues); 222     } else { 223         final ReceiveSmsMessageAction action = new ReceiveSmsMessageAction(messageValues); 224         action.start(); 225     } 226 }</pre> <p><a href="https://android.googlesource.com/platform/packages/apps/Messaging/+nougat-mr1-release/src/com/android/messaging/receiver/SmsReceiver.java">https://android.googlesource.com/platform/packages/apps/Messaging/+nougat-mr1-release/src/com/android/messaging/receiver/SmsReceiver.java</a></p>

**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC
	<pre>228     @Override 229     public void onReceive(final Context context, final Intent intent) { 230         LogUtil.v(TAG, "SmsReceiver.onReceive " + intent); 231         // On KLP+ we only take delivery of SMS messages in SmsDeliverReceiver. 232         if (PhoneUtils.getDefault().isSmsEnabled()) { 233             final String action = intent.getAction(); 234             if (OsUtil.isSecondaryUser() &amp;&amp; 235                 (Telephony.Sms.Intents.SMS_RECEIVED_ACTION.equals(action)    236                  // TODO: update this with the actual constant from Telephony 237                  "android.provider.Telephony.MMS_DOWNLOADED".equals(action))) { 238                 postNewMessageSecondaryUserNotification(); 239             } else if (!OsUtil.isAtLeastKLP()) { 240                 deliverSmsIntent(context, intent); 241             } 242         } 243     }</pre> <p><a href="https://android.googlesource.com/platform/packages/apps/Messaging/+/nougat-mr1-release/src/com/android/messaging/receiver/SmsReceiver.java">https://android.googlesource.com/platform/packages/apps/Messaging/+/nougat-mr1-release/src/com/android/messaging/receiver/SmsReceiver.java</a></p>

**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC
	<pre>52  * Class that sends chat message via MMS. 53  * 54  * The interface emulates a blocking send similar to making an HTTP request. 55  */ 56  public class MmsSender { 57      private static final String TAG = LogUtil.BUGLE_TAG; 58 59      /** 60       * Send an MMS message. 61       * 62       * @param context Context 63       * @param messageUri The unique URI of the message for identifying it during sending 64       * @param sendReq The SendReq PDU of the message 65       * @throws MmsFailureException 66       */ 67      public static void sendMms(final Context context, final int subId, final Uri messageUri, 68                               final SendReq sendReq, final Bundle sentIntentExtras) throws MmsFailureException { 69          sendMms(context, 70                subId, 71                messageUri, 72                null /* locationUri */, 73                sendReq, 74                true /* responseImportant */, 75                sentIntentExtras); 76      }</pre> <p><a href="https://android.googlesource.com/platform/packages/apps/Messaging/+/nougat-mr1-release/src/com/android/messaging/sms/MmsSender.java">https://android.googlesource.com/platform/packages/apps/Messaging/+/nougat-mr1-release/src/com/android/messaging/sms/MmsSender.java</a></p>

**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC
	<pre>240     * Download an MMS message. 241     * 242     * @param context Context 243     * @param contentLocation The url of the MMS message 244     * @throws MmsFailureException 245     * @throws InvalidHeaderValueException 246     */ 247     public static void downloadMms(final Context context, final int subId, 248         final String contentLocation, Bundle extras) throws MmsFailureException, 249         InvalidHeaderValueException { 250         final Uri requestUri = Uri.parse(contentLocation); 251         final Uri contentUri = MmsFileProvider.buildRawMmsUri(); 252 253         final Intent downloadedIntent = new Intent(SendStatusReceiver.MMS_DOWNLOADED_ACTION, 254             requestUri, 255             context, 256             SendStatusReceiver.class); 257         downloadedIntent.putExtra(SendMessageAction.EXTRA_CONTENT_URI, contentUri); 258         if (extras != null) { 259             downloadedIntent.putExtras(extras); 260         } 261         final PendingIntent downloadedPendingIntent = PendingIntent.getBroadcast( 262             context, 263             0 /*request code*/, 264             downloadedIntent, 265             PendingIntent.FLAG_UPDATE_CURRENT); 266 267         MmsManager.downloadMultimediaMessage(subId, context, contentLocation, contentUri, 268             downloadedPendingIntent); 269     }</pre> <p><a href="https://android.googlesource.com/platform/packages/apps/Messaging/+nougat-mr1-release/src/com/android/messaging/sms/MmsSender.java">https://android.googlesource.com/platform/packages/apps/Messaging/+nougat-mr1-release/src/com/android/messaging/sms/MmsSender.java</a></p>

**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC
	<pre> 97      * Execute an MMS HTTP request, either a POST (sending) or a GET (downloading) 98      * 99      * @param urlString The request URL, for sending it is usually the MMSC, and for downloading 100     *           it is the message URL 101     * @param pdu For POST (sending) only, the PDU to send 102     * @param method HTTP method, POST for sending and GET for downloading 103     * @param isProxySet Is there a proxy for the MMSC 104     * @param proxyHost The proxy host 105     * @param proxyPort The proxy port 106     * @param mmsConfig The MMS config to use 107     * @param userAgent The user agent header value 108     * @param uaProfUrl The UA Prof URL header value 109     * @return The HTTP response body 110     * @throws MmsHttpException For any failures 111     */ 112     public byte[] execute(String urlString, byte[] pdu, String method, boolean isProxySet, 113                          String proxyHost, int proxyPort, Bundle mmsConfig, String userAgent, String uaProfUrl) 114         throws MmsHttpException { 115         Log.d(MmsService.TAG, "HTTP: " + method + " " + Utils.redactUrlForNonVerbose(urlString) 116             + (isProxySet ? (" proxy=" + proxyHost + ":" + proxyPort) : "")) 117             + ", PDU size=" + (pdu != null ? pdu.length : 0)); 118         checkMethod(method); 119         HttpURLConnection connection = null; 120         try { 121             Proxy proxy = Proxy.NO_PROXY; 122             if (isProxySet) { 123                 proxy = new Proxy(Proxy.Type.HTTP, new InetSocketAddress(proxyHost, proxyPort)); 124             } 125             final URL url = new URL(urlString); 126             // Now get the connection 127             connection = (HttpURLConnection) url.openConnection(proxy); 128             connection.setDoInput(true); 129             connection.setConnectTimeout( 130                 mmsConfig.getInt(CarrierConfigValuesLoader.CONFIG_HTTP_SOCKET_TIMEOUT, 131                     CarrierConfigValuesLoader.CONFIG_HTTP_SOCKET_TIMEOUT_DEFAULT)); </pre> <p data-bbox="506 1219 1596 1287"> <a href="https://android.googlesource.com/platform/packages/apps/Messaging/+nougat-mr1-release/src/android/support/v7/mms/MmsHttpClient.java">https://android.googlesource.com/platform/packages/apps/Messaging/+nougat-mr1-release/src/android/support/v7/mms/MmsHttpClient.java</a> </p>



**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC
	<pre> 132 // ----- COMMON HEADERS ----- 133 // Header: Accept 134 connection.setRequestProperty(HEADER_ACCEPT, HEADER_VALUE_ACCEPT); 135 // Header: Accept-Language 136 connection.setRequestProperty( 137     HEADER_ACCEPT_LANGUAGE, getCurrentAcceptLanguage(Locale.getDefault())); 138 // Header: User-Agent 139 Log.i(MmsService.TAG, "HTTP: User-Agent=" + userAgent); 140 connection.setRequestProperty(HEADER_USER_AGENT, userAgent); 141 // Header: x-wap-profile 142 final String uaProfUrlTagName = mmsConfig.getString( 143     CarrierConfigValuesLoader.CONFIG_UA_PROF_TAG_NAME, 144     CarrierConfigValuesLoader.CONFIG_UA_PROF_TAG_NAME_DEFAULT); 145 if (uaProfUrl != null) { 146     Log.i(MmsService.TAG, "HTTP: UaProfUrl=" + uaProfUrl); 147     connection.setRequestProperty(uaProfUrlTagName, uaProfUrl); 148 } 149 // Add extra headers specified by mms_config.xml's httpparams 150 addExtraHeaders(connection, mmsConfig); 151 // Different stuff for GET and POST 152 if (METHOD_POST.equals(method)) { 153     if (pdu == null    pdu.length &lt; 1) { 154         Log.e(MmsService.TAG, "HTTP: empty pdu"); 155         throw new MmsHttpException(0/*statusCode*/, "Sending empty PDU"); 156     } 157     connection.setDoOutput(true); 158     connection.setRequestMethod(METHOD_POST); 159     if (mmsConfig.getBoolean( 160         CarrierConfigValuesLoader.CONFIG_SUPPORT_HTTP_CHARSET_HEADER, 161         CarrierConfigValuesLoader.CONFIG_SUPPORT_HTTP_CHARSET_HEADER_DEFAULT)) { 162         connection.setRequestProperty(HEADER_CONTENT_TYPE, 163             HEADER_VALUE_CONTENT_TYPE_WITH_CHARSET); 164     } else { 165         connection.setRequestProperty(HEADER_CONTENT_TYPE, 166             HEADER_VALUE_CONTENT_TYPE_WITHOUT_CHARSET); </pre> <p><a href="https://android.googlesource.com/platform/packages/apps/Messaging/+nougat-mr1-release/src/android/support/v7/mms/MmsHttpClient.java">https://android.googlesource.com/platform/packages/apps/Messaging/+nougat-mr1-release/src/android/support/v7/mms/MmsHttpClient.java</a></p>

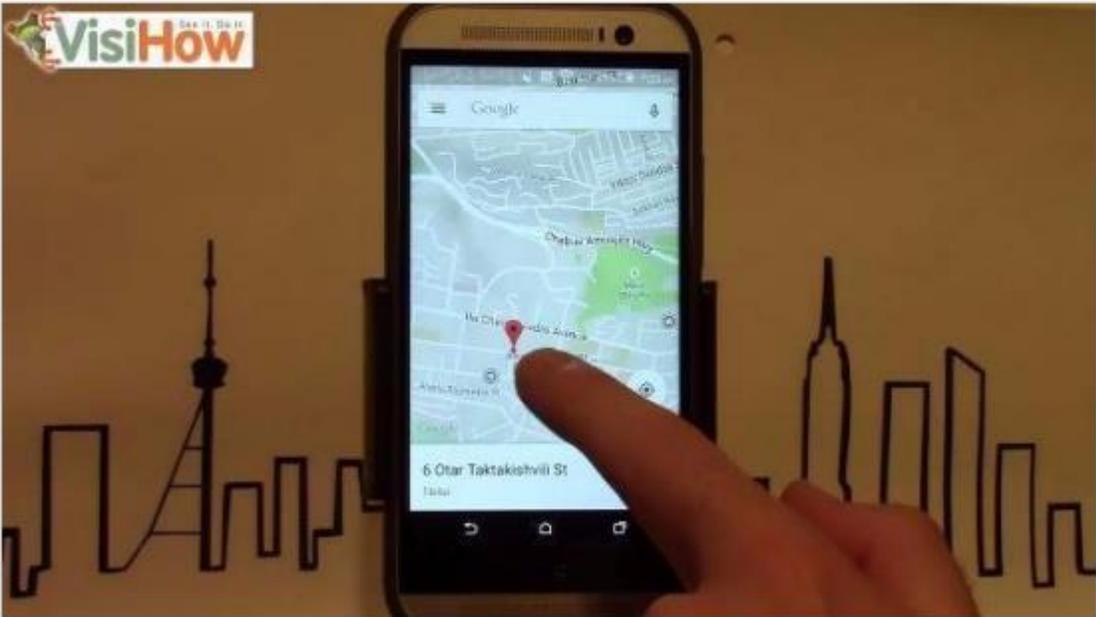
**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC
	<pre> 167     } 168     if (Log.isLoggable(MmsService.TAG, Log.VERBOSE)) { 169         logHttpHeaders(connection.getRequestProperties()); 170     } 171     connection.setFixedLengthStreamingMode(pdu.length); 172     // Sending request body 173     final OutputStream out = 174         new BufferedOutputStream(connection.getOutputStream()); 175     out.write(pdu); 176     out.flush(); 177     out.close(); 178 } else if (METHOD_GET.equals(method)) { 179     if (Log.isLoggable(MmsService.TAG, Log.VERBOSE)) { 180         logHttpHeaders(connection.getRequestProperties()); 181     } 182     connection.setRequestMethod(METHOD_GET); 183 } 184 // Get response 185 final int responseCode = connection.getResponseCode(); 186 final String responseMessage = connection.getResponseMessage(); 187 Log.d(MmsService.TAG, "HTTP: " + responseCode + " " + responseMessage); 188 if (Log.isLoggable(MmsService.TAG, Log.VERBOSE)) { 189     logHttpHeaders(connection.getHeaderFields()); 190 } 191 if (responseCode / 100 != 2) { 192     throw new MmsHttpException(responseCode, responseMessage); 193 } 194 final InputStream in = new BufferedInputStream(connection.getInputStream()); 195 final ByteArrayOutputStream byteOut = new ByteArrayOutputStream(); 196 final byte[] buf = new byte[4096]; 197 int count = 0; 198 while ((count = in.read(buf)) &gt; 0) { 199     byteOut.write(buf, 0, count); 200 } 201 in.close(); 202 final byte[] responseBody = byteOut.toByteArray(); 203 Log.d(MmsService.TAG, "HTTP: response size=" 204     + (responseBody != null ? responseBody.length : 0)); 205 return responseBody; </pre> <p><a href="https://android.goesource.com/platform/packages/apps/Messaging/+nougat-mr1-release/src/android/support/v7/mms/MmsHttpClient.java">https://android.goesource.com/platform/packages/apps/Messaging/+nougat-mr1-release/src/android/support/v7/mms/MmsHttpClient.java</a></p>

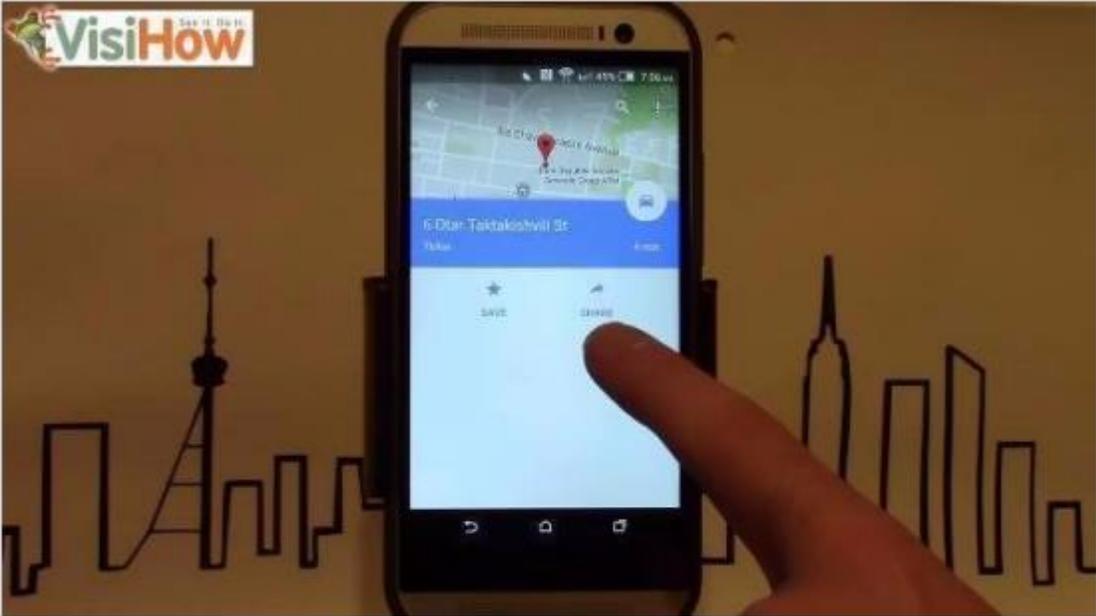
**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC
<p>[41C] receiving respective IP-based responses to the SMS messages, wherein the IP-based responses to the SMS messages include location information of the respective second devices;</p>	<p>HTC infringes directly and/or indirectly by performing, inducing others to perform, and/or contributing to the performance of receiving respective IP-based responses to the SMS messages, wherein the IP-based responses to the SMS messages include location information of the respective second devices. See claims 1[C] and 28[C], which are incorporated herein by reference in their entirety.</p> <p>For example, the HTC accused devices running Maps are configured to receive IP-based communications from the respective second devices that include location information of the second devices.</p> <p><b>Share Locations from Google Maps on HTC One M8</b></p>

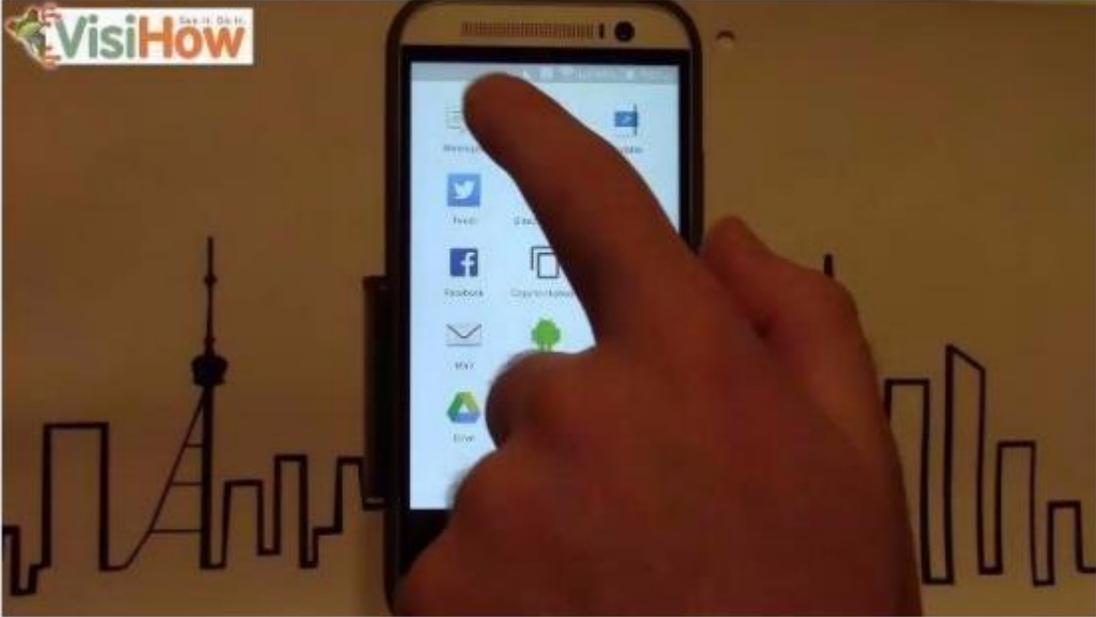
**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC
	<p data-bbox="531 228 829 256"><b>Place a pin on the map.</b></p> <p data-bbox="531 269 1640 380">We can do this anywhere in the entire world, anywhere, on any map that we find in the Google Maps application. We are going to choose a location, press down on the screen, and hold down until we can see that a red pin has been dropped.</p> 

**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC
	<p data-bbox="527 240 1207 305"><b>We have two options:</b> "SAVE" and "SHARE". Press down on the word "SHARE".</p>  <p data-bbox="527 1057 1633 1208"><b>A menu appears with all of the different ways that we can share this location.</b> We can share this location via messages, Bluetooth, Twitter, Facebook, mail, Gmail and Viber. If you have other applications, such as WhatsApp, or any other chat or email applications, you will be able to use them as well and will see them here.</p>

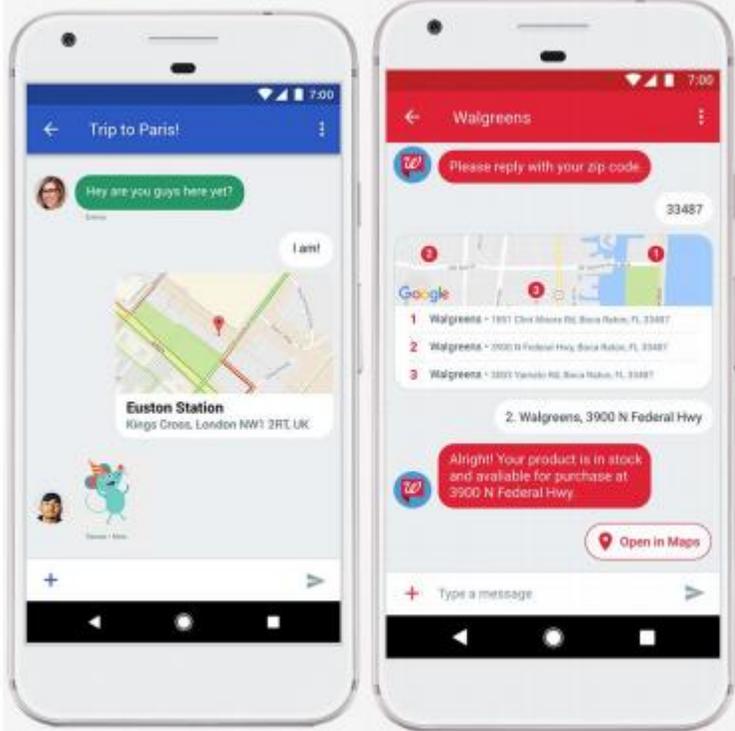
**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC
	<p data-bbox="527 237 856 261"><b>Let's choose "Messages".</b></p> <p data-bbox="527 277 1633 345">This option is located on the top left-hand corner of the screen. All options will have the same basic processes. They will ask us to send the location. Press down on the "Messages" icon.</p> 

**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC
	<p data-bbox="520 228 1260 256"><b>Press the box next to the contact who will be the recipient.</b></p> <p data-bbox="520 266 1549 293">A checkmark will appear in the box. We can select multiple contacts as well if we desire.</p>  <p data-bbox="520 1045 911 1073"><b>Press on the word "DONE (1)".</b></p> <p data-bbox="520 1083 1629 1154">This icon is located in the bottom right-hand corner of the screen. The bracketed number indicates the number of contacts we selected.</p> <p data-bbox="520 1164 1409 1192"><a href="http://visihow.com/Share_Locations_from_Google_Maps_on_HTC_One_M8">http://visihow.com/Share_Locations_from_Google_Maps_on_HTC_One_M8</a></p>

# Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products

US9408055B2	HTC
	<p data-bbox="533 240 884 280"><b>Send your location</b></p> <ol data-bbox="533 305 947 505" style="list-style-type: none"><li>1. Open the Android Messages app .</li><li>2. Open or start a conversation.</li><li>3. Tap Attach .</li><li>4. Tap Location on .</li><li>5. To send your location, tap Send .</li></ol> <p data-bbox="520 524 1539 557"><a href="https://support.google.com/pixelphone/answer/6159880?hl=en&amp;ref_topic=6211804">https://support.google.com/pixelphone/answer/6159880?hl=en&amp;ref_topic=6211804</a></p> <div data-bbox="520 594 1255 1325"></div> <p data-bbox="520 1328 1518 1360"><a href="https://blog.google/topics/rcs/delivering-rcs-messaging-android-users-worldwide/">https://blog.google/topics/rcs/delivering-rcs-messaging-android-users-worldwide/</a></p>



**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC
	<div data-bbox="533 342 915 378" data-label="Section-Header"> <h3>Share a location or place</h3> </div> <div data-bbox="558 423 743 444" data-label="Section-Header"> <h4>Share your location</h4> </div> <div data-bbox="590 477 1104 597" data-label="List-Group"> <ol style="list-style-type: none"> <li>1. On your Android phone or tablet, open the Hangouts app</li> <li>2. Open a conversation.</li> <li>3. Tap Location</li> <li>4. Tap <b>Select this location</b> &gt; <b>Select</b>.</li> </ol> </div> <div data-bbox="558 678 688 699" data-label="Section-Header"> <h4>Share a place</h4> </div> <div data-bbox="590 732 1104 886" data-label="List-Group"> <ol style="list-style-type: none"> <li>1. On your Android phone or tablet, open the Hangouts app</li> <li>2. Open a conversation.</li> <li>3. Tap Location &gt; Search</li> <li>4. Type in a location or address.</li> <li>5. Tap <b>Select</b>.</li> </ol> </div> <div data-bbox="520 922 1577 1019" data-label="Text"> <p><a href="https://support.google.com/hangouts/answer/3115410?visit_id=1-636271867303650973-2491837168&amp;rd=1&amp;co=GENIE.Platform%3DAndroid&amp;oco=1">https://support.google.com/hangouts/answer/3115410?visit_id=1-636271867303650973-2491837168&amp;rd=1&amp;co=GENIE.Platform%3DAndroid&amp;oco=1</a>  <a href="https://play.google.com/store/apps/details?id=com.google.android.talk&amp;hl=en">https://play.google.com/store/apps/details?id=com.google.android.talk&amp;hl=en</a></p> </div> <div data-bbox="512 1057 1902 1308" data-label="Text"> <p><b>Regarding Google Maps,</b> Google Latitude, Google Plus, Google Hangouts, Google Messages, Google Allo, Google Duo, Google Chrome, and Android Messenger, the Accused Products require a user to join the corresponding network by signing-in to the device with an identifier (e.g., Google Account). When the signed-in user enables Google Maps Share Location and Location Services on the first device, the user shares its location and the first device's location is sent to a server. When the user enables sharing to one or more contacts (of respective devices) and the one or more contacts enable sharing their location to the user of the first device, the user of the first device receives the locations of the one or more contacts.</p> </div> <div data-bbox="512 1352 1892 1417" data-label="Text"> <p>The first device's participation in the group is based on receiving the message from the second device, i.e. a message indicating that the second device is sharing its location.</p> </div> <div data-bbox="1360 228 1749 919" data-label="Image"> </div>

**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC
	<p>By participating in the Maps location sharing functionality, the device sends location information to a server (e.g., a network server provided by an ISP such as AT&amp;T and/or a server running Google's services). The device also receives location information from the server indicating the location of other devices that are sharing location information via Maps.</p> <p><b><u>Further regarding Google Maps</u></b>, Google Latitude, Google Plus, Google Hangouts, Google Messages, Google Allo, Google Duo, Google Chrome, and Android Messenger, the Accused Products alternatively require a user to send a request containing the identifier (e.g., Google Account). When the signed-in user enables Google Maps Share Location and Location Services on the first device, the user shares its location and the first device's location is sent to a server. When the user sends a message to another contact through Google Maps, Google Messages, and/or another means from within the Google Maps application, the message including location information are sent to a server before transmission to the intended contact. When one or more contacts enable sharing their location to the user of the first device, or alternatively send a message containing location information, or alternatively accept a request to share their location with the first user, the user of the first device receives the locations of the one or more contacts.</p> <p><b><u>Exemplary Support for Google Maps:</u></b></p>

## Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products

US9408055B2	HTC
	<p data-bbox="548 240 968 261">COMPUTER <b>ANDROID</b> IPHONE &amp; IPAD</p> <hr data-bbox="533 289 1566 293"/> <h3 data-bbox="533 347 1024 383">If they have a Google Account</h3> <ol data-bbox="533 406 1419 698" style="list-style-type: none"><li data-bbox="533 406 1220 427">1. If you haven't already, add their Gmail address to your <a href="#">Google Contacts</a>.</li><li data-bbox="533 443 1419 464">2. On your Android phone or tablet, open the Google Maps app  and sign in. <a href="#">Learn how to sign in.</a></li><li data-bbox="533 480 1031 501">3. Tap Menu  &gt; <b>Location sharing</b> &gt; Add People .</li><li data-bbox="533 518 1003 539">4. Choose how long you want to share your location.</li><li data-bbox="533 555 1140 613">5. Tap <b>Select People</b>.<ul data-bbox="569 589 1140 610" style="list-style-type: none"><li data-bbox="569 589 1140 610">• If you're asked about your contacts, give Google Maps access.</li></ul></li><li data-bbox="533 639 884 660">6. Choose who you want to share with.</li><li data-bbox="533 677 663 698">7. Tap <b>Share</b>.</li></ol> <h3 data-bbox="533 756 1110 792">If they don't have a Google Account</h3> <ol data-bbox="533 815 1560 940" style="list-style-type: none"><li data-bbox="533 815 1419 836">1. On your Android phone or tablet, open the Google Maps app  and sign in. <a href="#">Learn how to sign in.</a></li><li data-bbox="533 852 1031 873">2. Tap Menu  &gt; <b>Location sharing</b> &gt; Add People .</li><li data-bbox="533 889 1560 940">3. Tap More  &gt; <b>Copy to clipboard</b>. People with this link can see your location for as long as you choose, up to 72 hours.</li></ol> <h3 data-bbox="533 989 869 1024">Share using another app</h3> <p data-bbox="533 1044 1205 1065">You can also share through messaging apps. Tap More  &gt; select an app.</p> <h3 data-bbox="533 1123 743 1159">Stop sharing</h3> <ol data-bbox="533 1182 1205 1281" style="list-style-type: none"><li data-bbox="533 1182 842 1203">1. Open the Google Maps app .</li><li data-bbox="533 1219 869 1240">2. Tap Menu  &gt; <b>Location sharing</b>.</li><li data-bbox="533 1256 1205 1278">3. Next to the person with whom you want to stop sharing, tap Remove .</li></ol> <p data-bbox="512 1299 1701 1325"><a href="https://support.google.com/maps/answer/7326816?co=GENIE.Platform%3DAndroid&amp;oco=1">https://support.google.com/maps/answer/7326816?co=GENIE.Platform%3DAndroid&amp;oco=1</a></p>

## Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products

US9408055B2	HTC
	<h3>Share your E.T.A</h3> <p>After you start your drive, you can share your destination, estimated arrival time, and where you are on the route.</p> <ol style="list-style-type: none"><li>1. Open the Google Maps app .</li><li>2. Set a driving destination. <a href="#">Learn how to navigate to a place.</a></li><li>3. After you start navigation, tap More  &gt; <b>Share trip progress.</b></li><li>4. Choose a person from the list.</li><li>5. Tap <b>Share.</b></li><li>6. Location Sharing will stop when you reach your destination or stop navigating.</li></ol> <ul style="list-style-type: none"><li>• To stop sharing before you arrive, tap More  &gt; <b>Stop sharing.</b></li></ul> <h3>See where someone is</h3> <p>If someone shares their location with you, you can see them on the map.</p> <ol style="list-style-type: none"><li>1. Open the Google Maps app .</li><li>2. Tap Menu  &gt; <b>Location sharing.</b></li><li>3. Choose someone.</li></ol> <ul style="list-style-type: none"><li>• To see an updated location, tap on a friend's icon &gt; More  &gt; <b>Refresh.</b></li></ul> <h3>Stop seeing someone's location</h3> <ol style="list-style-type: none"><li>1. Open the Google Maps app .</li><li>2. On the map, tap their icon.</li><li>3. At the bottom, tap More .</li><li>4. To temporarily hide someone, tap <b>Hide from map.</b> You can stop hiding them at any time.</li></ol> <p><b>Note:</b> You can stop someone's location from ever appearing on your map. <a href="#">Learn how to block another person's account.</a></p> <p><a href="https://support.google.com/maps/answer/7326816?co=GENIE.Platform%3DAndroid&amp;oco=1">https://support.google.com/maps/answer/7326816?co=GENIE.Platform%3DAndroid&amp;oco=1</a></p>

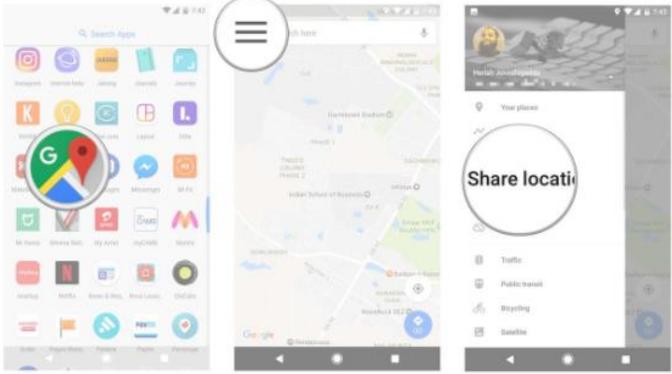
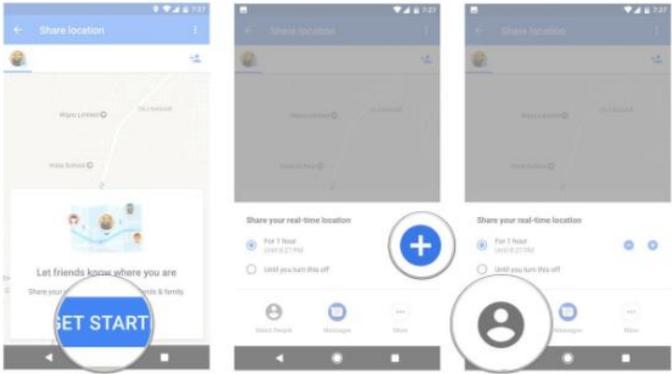
## Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products

US9408055B2	HTC
	<h3>Create a list of places</h3> <p>In Google Maps, you can create a list of places, like your favorite places or places you want to visit.</p> <p>COMPUTER <b>ANDROID</b> IPHONE &amp; IPAD</p> <hr/> <h3>Make a new list</h3> <ol style="list-style-type: none"><li>1. On your Android phone or tablet, open the Google Maps app .</li><li>2. Tap Menu  &gt; <b>Your places</b> &gt; <b>Saved</b>.</li><li>3. In the bottom right, tap Add .</li><li>4. Enter a name and description.</li><li>5. Tap <b>Save</b>.</li></ol> <h3>Save a place to a list</h3> <ol style="list-style-type: none"><li>1. Open the Google Maps app .</li><li>2. Search for a place or tap it on the map.</li><li>3. At the bottom, tap the place's name or address.</li><li>4. Tap <b>Save</b>.</li><li>5. Choose a list. To create a new list, tap <b>New list</b> .</li></ol> <h3>See your lists</h3> <ol style="list-style-type: none"><li>1. Open the Google Maps app .</li><li>2. Tap Menu  &gt; <b>Your places</b> &gt; <b>Saved</b>.</li></ol> <p><a href="https://support.google.com/maps/answer/7280933?hl=en&amp;ref_topic=7301134&amp;co=GENIE.Platform%3DAndroid&amp;oco=1">https://support.google.com/maps/answer/7280933?hl=en&amp;ref_topic=7301134&amp;co=GENIE.Platform%3DAndroid&amp;oco=1</a></p>

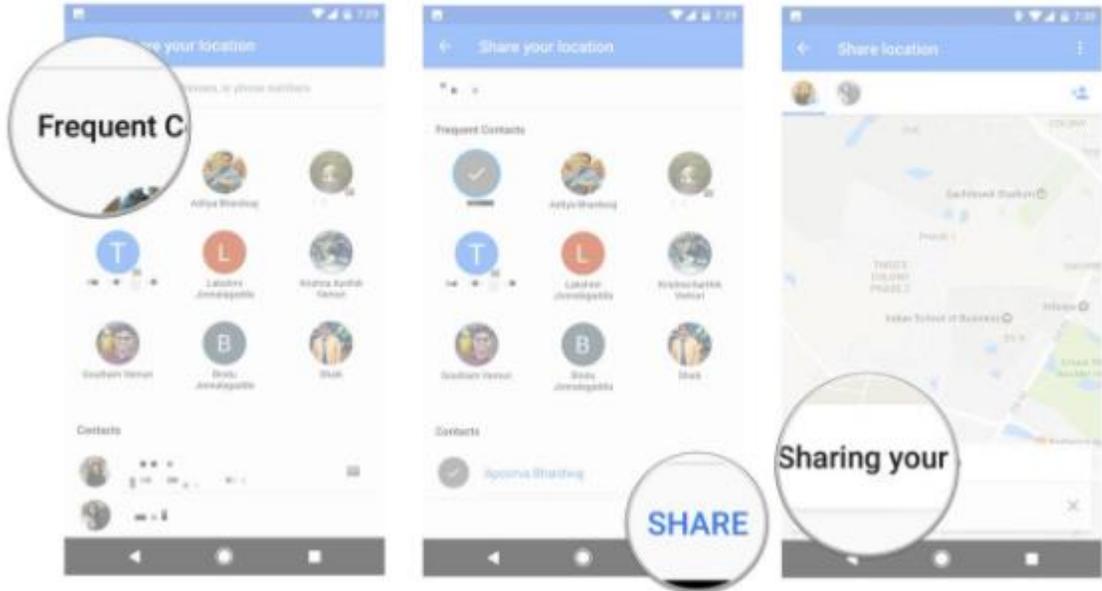
## Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products

US9408055B2	HTC
	<h3 data-bbox="541 245 877 282">Hide or share lists</h3> <p data-bbox="541 313 907 337"><b>Note:</b> You can't share starred places.</p> <ol data-bbox="554 367 1251 475" style="list-style-type: none"><li data-bbox="554 367 890 391">1. Open the Google Maps app .</li><li data-bbox="554 407 968 431">2. Tap Menu  &gt; <b>Your places</b> &gt; <b>Saved</b>.</li><li data-bbox="554 448 1251 475">3. Next to the list you want to share, tap More  &gt; choose an option:</li></ol> <ul data-bbox="583 492 1682 634" style="list-style-type: none"><li data-bbox="583 492 1440 516">• <b>Hide/Show on your map:</b> Display or hide your saved places when looking at the map.</li><li data-bbox="583 532 1058 557">• <b>Share list:</b> Allow others to see your saved list.</li><li data-bbox="583 573 1682 634">• <b>Sharing options:</b> You can make your list public, private, or shared. To let anyone with the link see your list, tap <b>Shared</b>. To let anyone find and follow your list, tap <b>Public</b>.</li></ul> <h3 data-bbox="541 699 764 737">Follow a list</h3> <p data-bbox="541 768 1728 824">If you follow a list made by someone else, their saved places will show up in Your Places. The places will also appear as suggested locations in Google Maps.</p> <h3 data-bbox="541 878 915 915">Follow a list using a link</h3> <ol data-bbox="554 938 1352 1047" style="list-style-type: none"><li data-bbox="554 938 957 963">1. Tap on the link you received to open it.</li><li data-bbox="554 979 1272 1003">2. Tap <b>Follow</b>. This list will now be added to the group of lists you follow.</li><li data-bbox="554 1019 1352 1047">3. <b>Optional:</b> To unfollow a list someone shared with you, tap the list &gt; <b>Following</b>.</li></ol> <h3 data-bbox="541 1101 924 1138">See lists made by others</h3> <p data-bbox="541 1161 1335 1185">If a user has any Google Maps lists that were made public, you can follow them.</p> <ol data-bbox="554 1216 1136 1325" style="list-style-type: none"><li data-bbox="554 1216 1136 1240">1. Tap on the name of a user whose list you want to follow.</li><li data-bbox="554 1256 680 1281">2. Tap <b>Lists</b>.</li><li data-bbox="554 1297 1136 1325">3. Tap on the list you want to follow &gt; More  &gt; <b>Follow</b>.</li></ol> <p data-bbox="512 1341 1898 1401"><a href="https://support.google.com/maps/answer/7280933?hl=en&amp;ref_topic=7301134&amp;co=GENIE.Platform%3DAndroid&amp;oco=1">https://support.google.com/maps/answer/7280933?hl=en&amp;ref_topic=7301134&amp;co=GENIE.Platform%3DAndroid&amp;oco=1</a></p>

**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

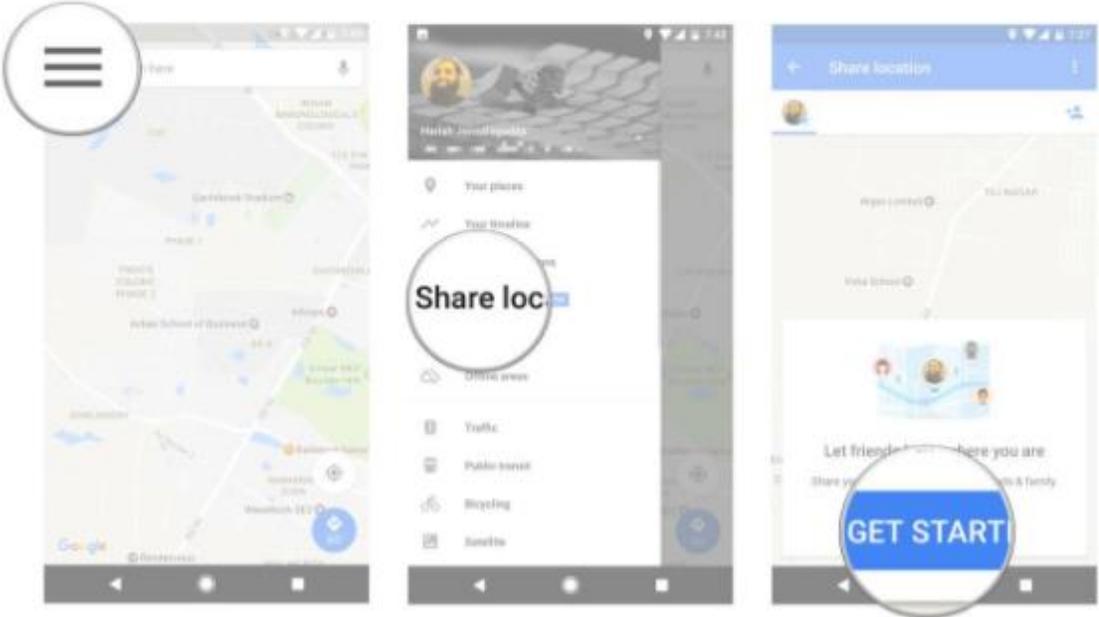
US9408055B2	HTC
	<p data-bbox="520 240 1150 272"><b>How to share your location in Google Maps</b></p> <ol data-bbox="520 300 1134 389" style="list-style-type: none"> <li>1. Open Google Maps from the app drawer or the home screen.</li> <li>2. Tap the hamburger menu (the three horizontal lines) on the top left corner of the screen.</li> <li>3. Select Share location.</li> </ol>  <ol data-bbox="520 824 1165 933" style="list-style-type: none"> <li>4. Tap Get Started.</li> <li>5. Use the + icon to select a time period or select the <b>Until you turn this off</b> setting to share your location indefinitely.</li> <li>6. Tap Select People.</li> </ol>  <p data-bbox="520 1339 1354 1372"><a href="https://www.androidcentral.com/how-share-location-google-maps">https://www.androidcentral.com/how-share-location-google-maps</a></p>

**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

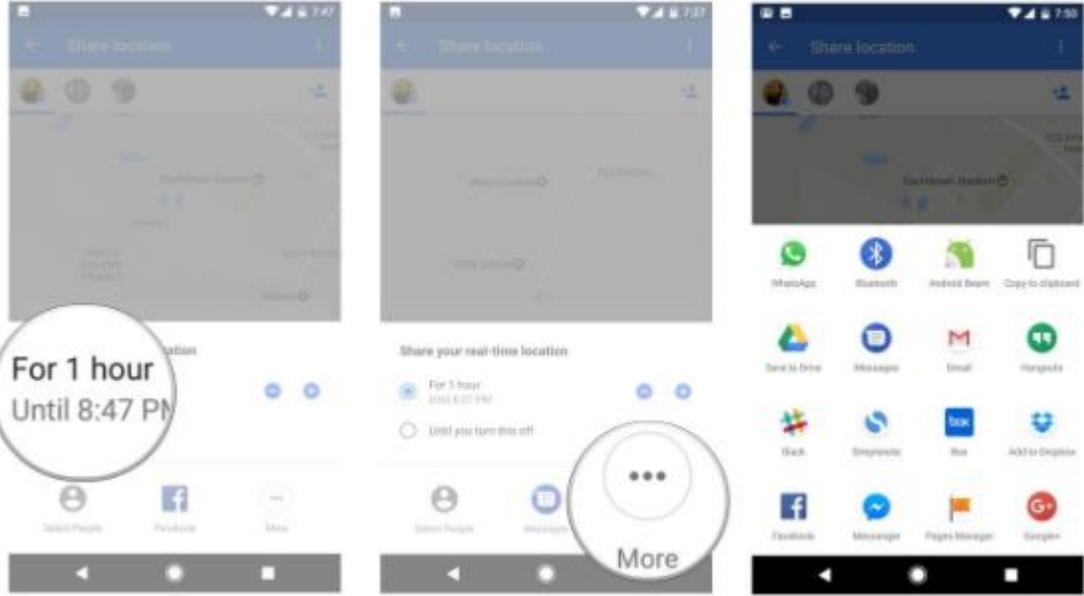
US9408055B2	HTC
	<p data-bbox="527 250 1577 423">7. You'll see a list of your frequent contacts at the top, along with a full list of contacts. Pick the contacts by tapping their name. 8. Once you've selected the contacts you want to share your location to, tap <b>Share</b>. 9. You'll see a message saying that the selected contact can view your location.</p>  <p data-bbox="506 1101 1356 1133"><a href="https://www.androidcentral.com/how-share-location-google-maps">https://www.androidcentral.com/how-share-location-google-maps</a></p>



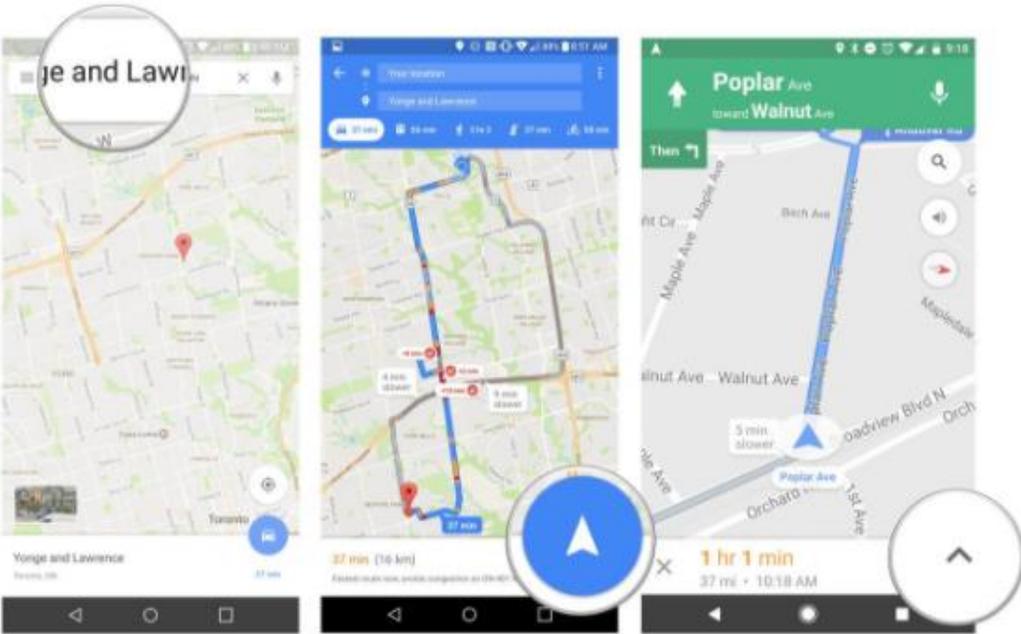
## Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products

US9408055B2	HTC
	<h3 data-bbox="520 245 1255 289">How to create a shareable link</h3> <p data-bbox="520 331 1461 358">You can also create a link and use it to share your location easily. Here's how to do it:</p> <ol data-bbox="520 410 1234 548" style="list-style-type: none"><li data-bbox="520 410 1234 438">1. Tap the hamburger menu on the top left corner of the screen.</li><li data-bbox="520 467 800 495">2. Select Share location.</li><li data-bbox="520 524 737 552">3. Tap Get Started.</li></ol>  <p data-bbox="506 1230 1356 1263"><a href="https://www.androidcentral.com/how-share-location-google-maps">https://www.androidcentral.com/how-share-location-google-maps</a></p>

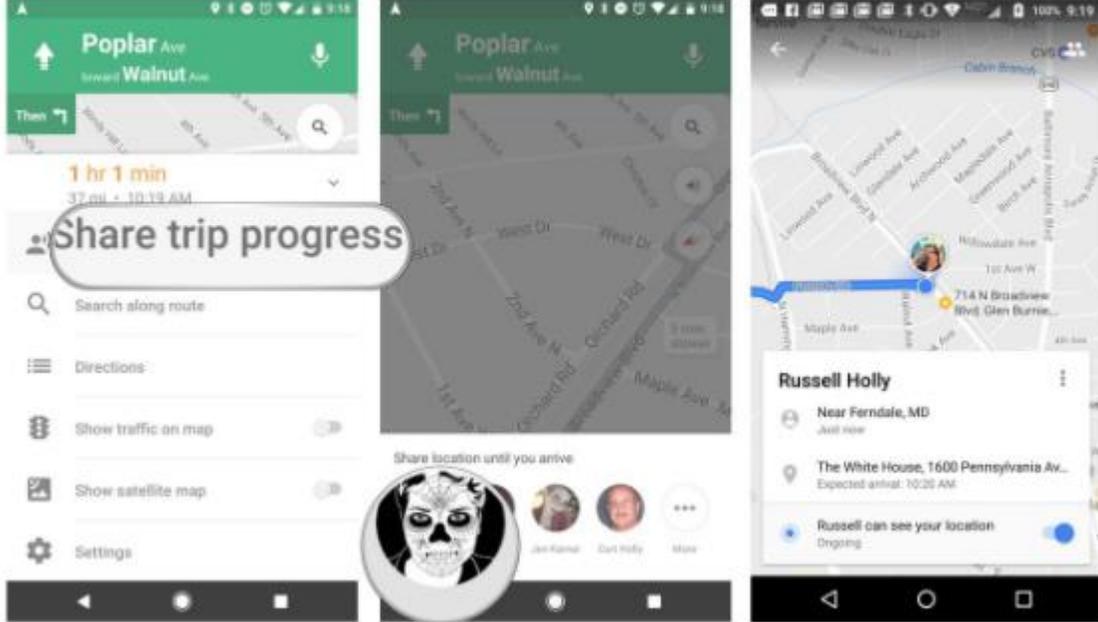
**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC
	<p data-bbox="520 245 1213 272">4. Select the <b>amount of time</b> you want to share your location.</p> <p data-bbox="520 302 680 329">5. Tap <b>More</b>.</p> <p data-bbox="520 358 1633 423">6. Select <b>your app of choice</b> to create and send a <b>unique URL</b> that broadcasts your current location. You can email it, send the link via Messenger, or even tweet it to the intended recipient.</p>  <p data-bbox="506 1084 1360 1117"><a href="https://www.androidcentral.com/how-share-location-google-maps">https://www.androidcentral.com/how-share-location-google-maps</a></p>

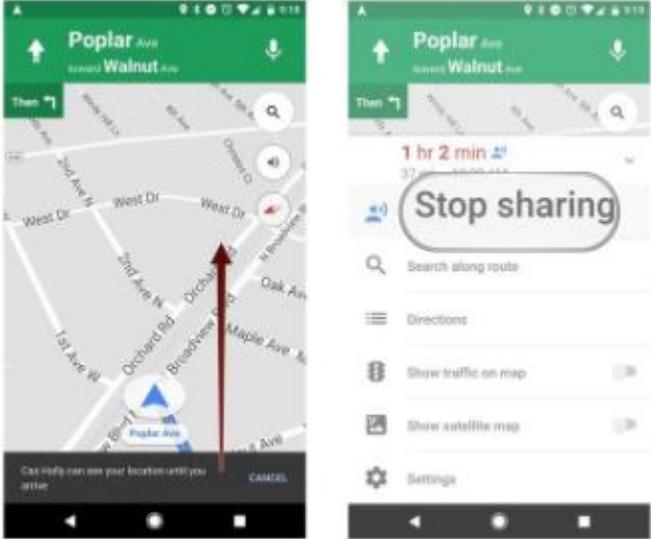
## Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products

US9408055B2	HTC
	<h3 data-bbox="527 240 1423 337">How to share your navigation directions while you walk, drive or transit</h3> <p data-bbox="527 375 1556 464">One of the best ways to share your location with someone is during a drive, walk or transit. If you're meeting a friend or family member somewhere, or navigating towards their home, Google Maps lets you share your location with them for the duration of the trip. It's magic!</p> <ol data-bbox="527 513 1394 643" style="list-style-type: none"><li>1. In the <b>search bar</b> enter your destination.</li><li>2. Pick your navigation type (drive, transit, walk) and press the blue <b>navigate</b> button.</li><li>3. Tap the arrow next to the <b>time-to-destination</b> number at the bottom of the screen.</li></ol>  <p data-bbox="512 1328 1356 1357"><a href="https://www.androidcentral.com/how-share-location-google-maps">https://www.androidcentral.com/how-share-location-google-maps</a></p>

**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC
	<p data-bbox="527 240 835 267">4. Tap Share trip progress.</p> <p data-bbox="527 297 1150 324">5. Choose one or more contacts to share trip progress.</p>  <p data-bbox="535 1031 1339 1058">You can also stop sharing your location with someone before a trip ends.</p> <p data-bbox="514 1068 1354 1096"><a href="https://www.androidcentral.com/how-share-location-google-maps">https://www.androidcentral.com/how-share-location-google-maps</a></p>

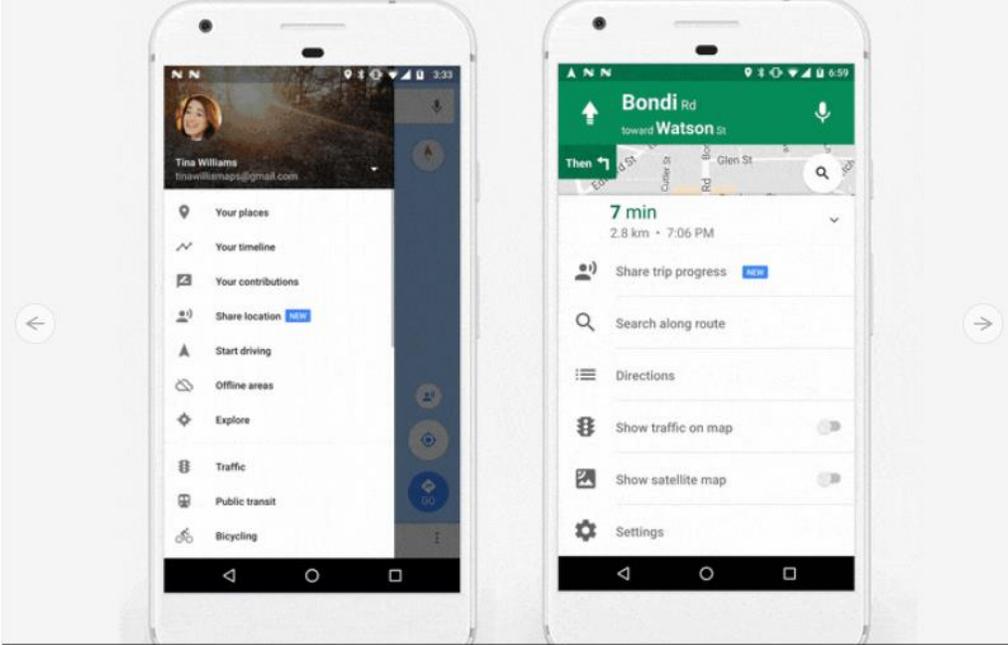
**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC
	<ol style="list-style-type: none"><li data-bbox="533 245 1465 272">1. Tap the arrow next to the <b>time-to-destination</b> number at the bottom of the screen.</li><li data-bbox="533 302 768 329">2. Tap <b>Stop sharing</b>.</li></ol> <div data-bbox="747 383 1398 922"></div> <p data-bbox="541 976 636 1003">That's it!</p> <p data-bbox="541 1045 1608 1073">Are you excited that location sharing is back in Google Maps? How often do you use the feature?</p> <p data-bbox="512 1084 1356 1112"><a href="https://www.androidcentral.com/how-share-location-google-maps">https://www.androidcentral.com/how-share-location-google-maps</a></p> <p data-bbox="512 1122 1419 1149">As shown below, a group may also be defined within Google Contacts.</p>

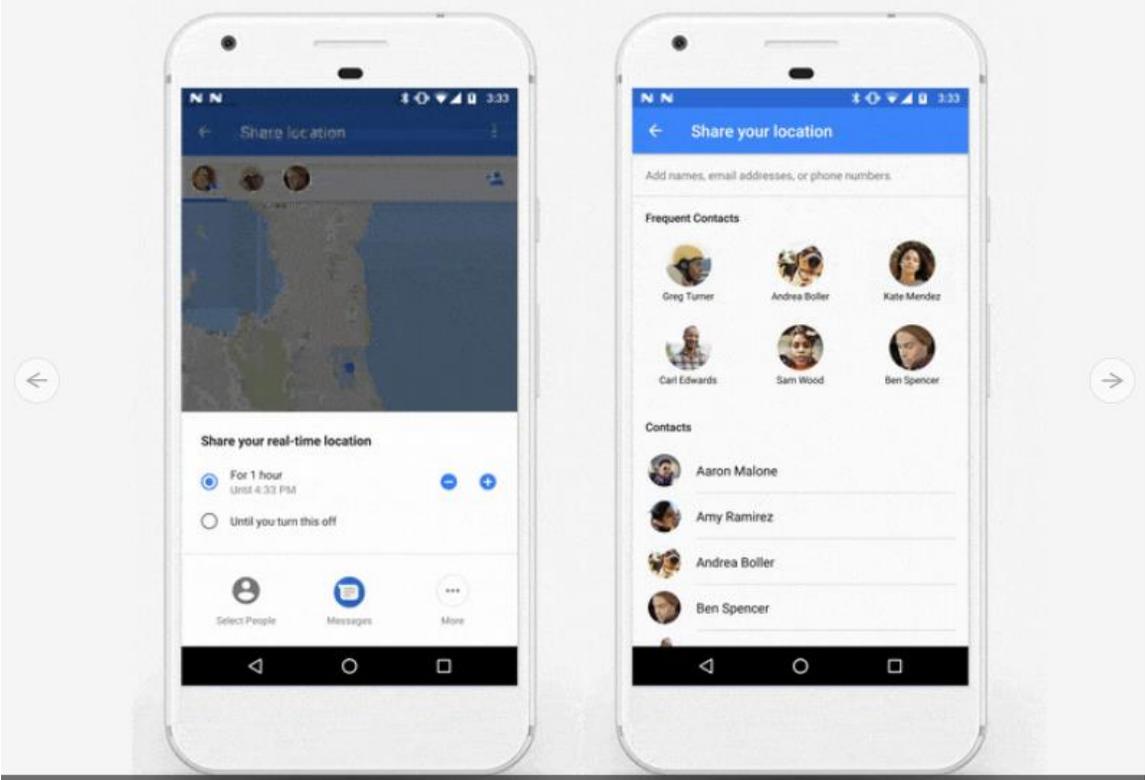
## Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products

US9408055B2	HTC
	<p data-bbox="541 237 894 277"><b>See your contacts</b></p> <ol data-bbox="552 305 968 378" style="list-style-type: none"><li>1. Open your device's Contacts app .</li><li>2. Tap Menu .</li></ol> <ul data-bbox="552 410 1734 605" style="list-style-type: none"><li>• <b>See contacts by label:</b> Choose a label from the list.</li><li>• <b>See contacts for another account:</b> Tap Down arrow  &gt; pick an account.</li><li>• <b>See the contacts for all your accounts:</b> Choose <b>All contacts</b>. <b>Tip:</b> If you have multiple contacts with the same information, the information will be grouped into one contact.</li><li>• <b>See your Google Account contacts on the web:</b> Go to <a href="#">Google Contacts</a> .</li></ul> <p data-bbox="510 638 1535 670"><a href="https://support.google.com/android/answer/6118731?hl=en&amp;ref_topic=6118711">https://support.google.com/android/answer/6118731?hl=en&amp;ref_topic=6118711</a></p> <p data-bbox="541 678 894 719"><b>Label your contacts</b></p> <p data-bbox="541 743 982 768">You can group contacts together using labels.</p> <ol data-bbox="552 800 930 898" style="list-style-type: none"><li>1. Open your device's Contacts app .</li><li>2. Tap Menu  &gt; <b>Create label</b>.</li><li>3. Enter a label name and tap <b>Ok</b>.</li></ol> <ul data-bbox="552 930 1713 995" style="list-style-type: none"><li>• <b>Add one contact to a label:</b> Tap Add contact  &gt; choose a contact.</li><li>• <b>Add multiple contacts to a label:</b> Tap Add contact  &gt; touch and hold a contact &gt; tap the other contacts &gt; tap <b>Add</b>.</li></ul> <p data-bbox="510 1011 1535 1044"><a href="https://support.google.com/android/answer/6118731?hl=en&amp;ref_topic=6118711">https://support.google.com/android/answer/6118731?hl=en&amp;ref_topic=6118711</a></p> <p data-bbox="552 1092 940 1133"><b>Share your contacts</b></p> <ol data-bbox="562 1166 1045 1320" style="list-style-type: none"><li>1. Open your device's Contacts app .</li><li>2. Tap a contact in the list.</li><li>3. Tap More  &gt; <b>Share</b>.</li><li>4. Choose how you want to share the contact.</li></ol> <p data-bbox="510 1344 1535 1377"><a href="https://support.google.com/android/answer/6118731?hl=en&amp;ref_topic=6118711">https://support.google.com/android/answer/6118731?hl=en&amp;ref_topic=6118711</a></p>

**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

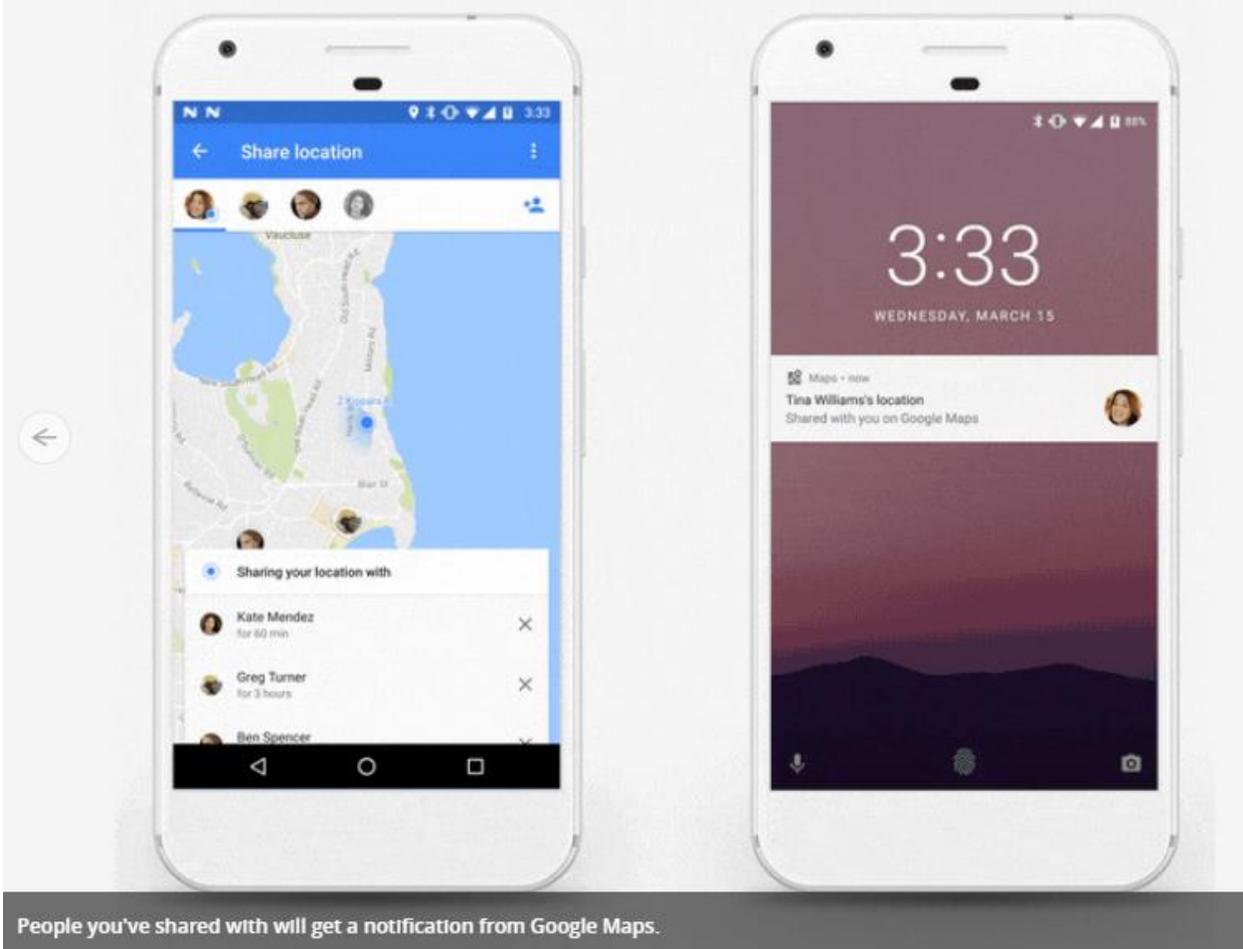
US9408055B2	HTC
	 <p data-bbox="514 893 1522 950">Ways to start location sharing: (left) from the app navigation drawer, and (right) from the menu, you can drag up while navigating somewhere.</p> <p data-bbox="514 958 1522 990"><a href="https://arstechnica.com/gadgets/2017/03/location-sharing-finally-returns-to-google-maps/">https://arstechnica.com/gadgets/2017/03/location-sharing-finally-returns-to-google-maps/</a></p>

**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

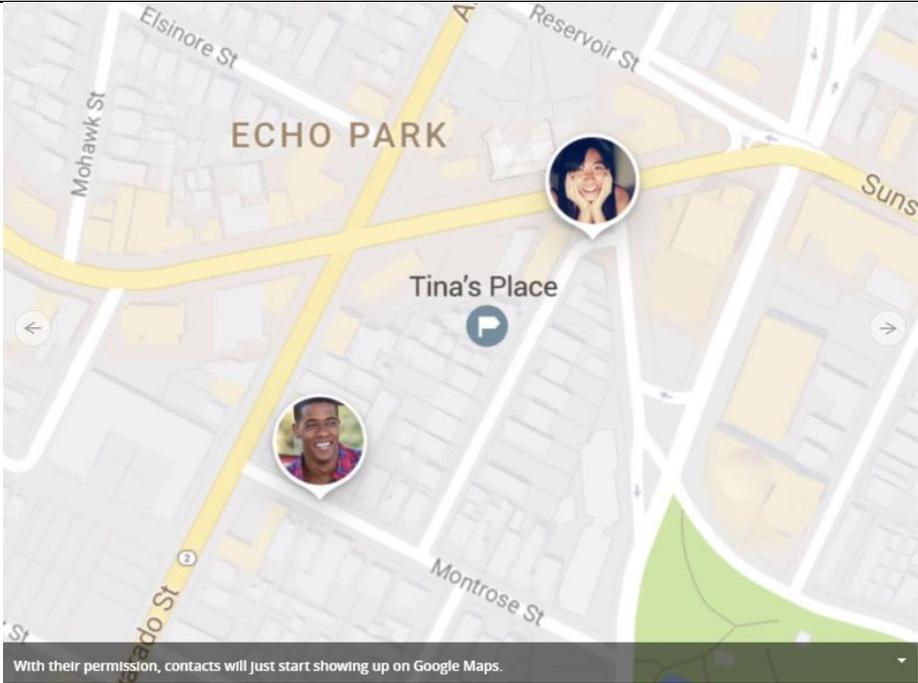
US9408055B2	HTC
	 <p data-bbox="514 1023 1659 1063">You can pick how long you want to share your location for, and with whom you want to share it.</p> <p data-bbox="514 1063 1659 1096"><a href="https://arstechnica.com/gadgets/2017/03/location-sharing-finally-returns-to-google-maps/">https://arstechnica.com/gadgets/2017/03/location-sharing-finally-returns-to-google-maps/</a></p>



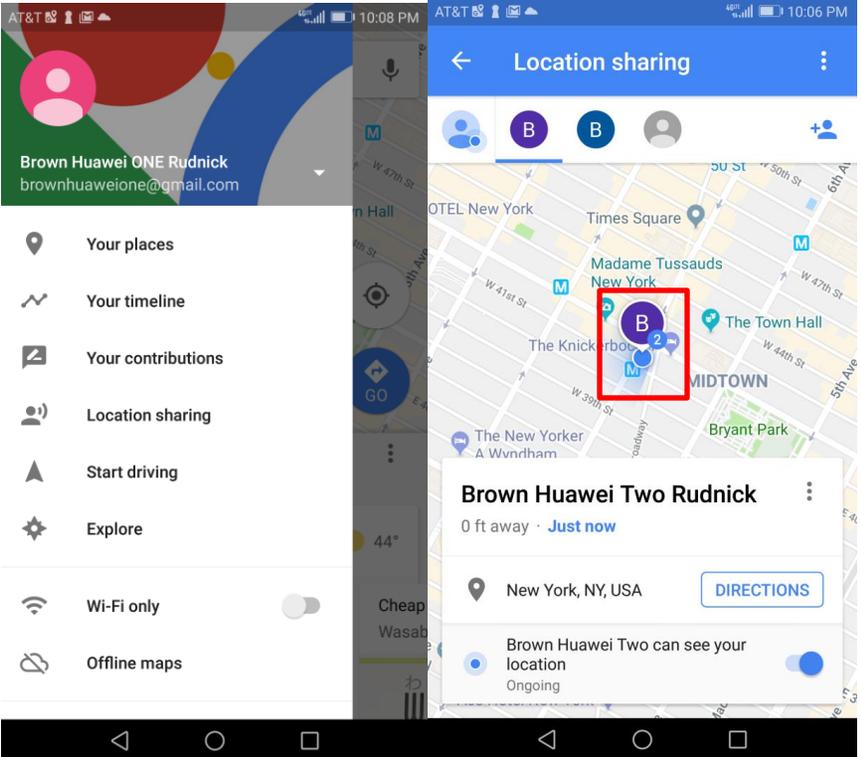
**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC
	 <p data-bbox="512 1143 1176 1170">People you've shared with will get a notification from Google Maps.</p> <p data-bbox="512 1187 1656 1219"><a href="https://arstechnica.com/gadgets/2017/03/location-sharing-finally-returns-to-google-maps/">https://arstechnica.com/gadgets/2017/03/location-sharing-finally-returns-to-google-maps/</a></p>

**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC
	 <p data-bbox="512 911 1430 950"><a href="https://arstechnica.com/gadgets/2017/03/location-sharing-finally-returns-to-google-maps/">https://arstechnica.com/gadgets/2017/03/location-sharing-finally-returns-to-google-maps/</a></p> <p data-bbox="512 950 1430 982"><b><u>Exemplary Google Maps Screenshots</u></b></p>

**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC
	 <p>The screenshot shows an Android phone interface. On the left, a menu is open with options: 'Your places', 'Your timeline', 'Your contributions', 'Location sharing', 'Start driving', 'Explore', 'Wi-Fi only', and 'Offline maps'. The 'Location sharing' option is selected. On the right, a map of New York City is displayed with a location pin labeled 'B' in a red box. Below the map, a card for 'Brown Huawei Two Rudnick' shows '0 ft away · Just now', 'New York, NY, USA', and a 'DIRECTIONS' button. At the bottom of the card, it says 'Brown Huawei Two can see your location Ongoing' with a toggle switch.</p> <p>Location information is shared via IP-based communication resulting in map that displays location information</p> <p><b><u>Exemplary Source Code:</u></b> The above functionality is performed at least in part by the following publicly available source code and/or source code that invokes or is invoked by the following source code (or a substantially similar copy compiled and loaded onto the Accused Products by HTC). AGIS reserves the right to amend these contentions to include additional source code as discovery progresses and as additional source code is made available.</p>

**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC
	<h2 data-bbox="527 240 1016 293">Contacts Provider</h2> <p data-bbox="527 329 1472 591">The Contacts Provider is a powerful and flexible Android component that manages the device's central repository of data about people. The Contacts Provider is the source of data you see in the device's contacts application, and you can also access its data in your own application and transfer data between the device and online services. The provider accommodates a wide range of data sources and tries to manage as much data as possible for each person, with the result that its organization is complex. Because of this, the provider's API includes an extensive set of contract classes and interfaces that facilitate both data retrieval and modification.</p> <p data-bbox="527 626 835 646">This guide describes the following:</p> <ul data-bbox="527 675 1373 850" style="list-style-type: none"><li data-bbox="527 675 806 695">• The basic provider structure.</li><li data-bbox="527 724 894 743">• How to retrieve data from the provider.</li><li data-bbox="527 773 863 792">• How to modify data in the provider.</li><li data-bbox="527 821 1373 841">• How to write a sync adapter for synchronizing data from your server to the Contacts Provider.</li></ul> <p data-bbox="512 863 1486 889"><a href="https://developer.android.com/guide/topics/providers/contacts-provider.html">https://developer.android.com/guide/topics/providers/contacts-provider.html</a></p>

**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC
	<p><b>Overview</b></p> <p>ContactsContract defines an extensible database of contact-related information. Contact information is stored in a three-tier data model:</p> <ul style="list-style-type: none"> <li>• A row in the <code>ContactsContract.Data</code> table can store any kind of personal data, such as a phone number or email addresses. The set of data kinds that can be stored in this table is open-ended. There is a predefined set of common kinds, but any application can add its own data kinds.</li> <li>• A row in the <code>ContactsContract.RawContacts</code> table represents a set of data describing a person and associated with a single account (for example, one of the user's Gmail accounts).</li> <li>• A row in the <code>ContactsContract.Contacts</code> table represents an aggregate of one or more RawContacts presumably describing the same person. When data in or associated with the RawContacts table is changed, the affected aggregate contacts are updated as necessary.</li> </ul> <p>Other tables include:</p> <ul style="list-style-type: none"> <li>• <code>ContactsContract.Groups</code>, which contains information about raw contact groups such as Gmail contact groups. The current API does not support the notion of groups spanning multiple accounts.</li> <li>• <code>ContactsContract.StatusUpdates</code>, which contains social status updates including IM availability.</li> <li>• <code>ContactsContract.AggregationExceptions</code>, which is used for manual aggregation and disaggregation of raw contacts</li> <li>• <code>ContactsContract.Settings</code>, which contains visibility and sync settings for accounts and groups.</li> <li>• <code>ContactsContract.SyncState</code>, which contains free-form data maintained on behalf of sync adapters</li> <li>• <code>ContactsContract.PhoneLookup</code>, which is used for quick caller-ID lookup</li> </ul> <p><a href="https://developer.android.com/reference/android/provider/ContactsContract.html">https://developer.android.com/reference/android/provider/ContactsContract.html</a></p> <p><b>Data</b></p> <p>As noted previously, the data for a raw contact is stored in a <code>ContactsContract.Data</code> row that is linked to the raw contact's <code>_ID</code> value. This allows a single raw contact to have multiple instances of the same type of data such as email addresses or phone numbers. For example, if "Thomas Higginson" for <code>emilyd@gmail.com</code> (the raw contact row for Thomas Higginson associated with the Google account <code>emilyd@gmail.com</code>) has a home email address of <code>thigg@gmail.com</code> and a work email address of <code>thomas.higginson@gmail.com</code>, the Contacts Provider stores the two email address rows and links them both to the raw contact.</p> <p>Notice that different types of data are stored in this single table. Display name, phone number, email, postal address, photo, and website detail rows are all found in the <code>ContactsContract.Data</code> table. To help manage this, the <code>ContactsContract.Data</code> table has some columns with descriptive names, and others with generic names. The contents of a descriptive-name column have the same meaning regardless of the type of data in the row, while the contents of a generic-name column have different meanings depending on the type of data.</p> <p><a href="https://developer.android.com/guide/topics/providers/contacts-provider.html">https://developer.android.com/guide/topics/providers/contacts-provider.html</a></p>

**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC													
	<table border="1"> <thead> <tr> <th>Task</th> <th>Action</th> <th>Data</th> <th>MIME type</th> <th>Notes</th> </tr> </thead> <tbody> <tr> <td>Pick a contact from a list</td> <td><code>ACTION_PICK</code></td> <td>                     One of:                     <ul style="list-style-type: none"> <li><code>Contacts.CONTENT_URI</code>, which displays a list of contacts.</li> <li><code>Phone.CONTENT_URI</code>, which displays a list of phone numbers for a raw contact.</li> <li><code>StructuredPostal.CONTENT_URI</code>, which displays a list of postal addresses for a raw contact.</li> <li><code>Email.CONTENT_URI</code>, which displays a list of email addresses for a raw contact.</li> </ul> </td> <td>Not used</td> <td>                     Displays a list of raw contacts or a list of data from a raw contact, depending on the content URI type you supply. Call <code>startActivityForResult()</code>, which returns the content URI of the selected row. The form of the URI is the table's content URI with the row's <code>LOOKUP_ID</code> appended to it. The device's contacts app delegates read and write permissions to this content URI for the life of your activity. See the <a href="#">Content Provider Basics</a> guide for more details.                 </td> </tr> </tbody> </table>	Task	Action	Data	MIME type	Notes	Pick a contact from a list	<code>ACTION_PICK</code>	One of: <ul style="list-style-type: none"> <li><code>Contacts.CONTENT_URI</code>, which displays a list of contacts.</li> <li><code>Phone.CONTENT_URI</code>, which displays a list of phone numbers for a raw contact.</li> <li><code>StructuredPostal.CONTENT_URI</code>, which displays a list of postal addresses for a raw contact.</li> <li><code>Email.CONTENT_URI</code>, which displays a list of email addresses for a raw contact.</li> </ul>	Not used	Displays a list of raw contacts or a list of data from a raw contact, depending on the content URI type you supply. Call <code>startActivityForResult()</code> , which returns the content URI of the selected row. The form of the URI is the table's content URI with the row's <code>LOOKUP_ID</code> appended to it. The device's contacts app delegates read and write permissions to this content URI for the life of your activity. See the <a href="#">Content Provider Basics</a> guide for more details.	<p><a href="https://developer.android.com/guide/topics/providers/contacts-provider.html">https://developer.android.com/guide/topics/providers/contacts-provider.html</a></p>		
Task	Action	Data	MIME type	Notes										
Pick a contact from a list	<code>ACTION_PICK</code>	One of: <ul style="list-style-type: none"> <li><code>Contacts.CONTENT_URI</code>, which displays a list of contacts.</li> <li><code>Phone.CONTENT_URI</code>, which displays a list of phone numbers for a raw contact.</li> <li><code>StructuredPostal.CONTENT_URI</code>, which displays a list of postal addresses for a raw contact.</li> <li><code>Email.CONTENT_URI</code>, which displays a list of email addresses for a raw contact.</li> </ul>	Not used	Displays a list of raw contacts or a list of data from a raw contact, depending on the content URI type you supply. Call <code>startActivityForResult()</code> , which returns the content URI of the selected row. The form of the URI is the table's content URI with the row's <code>LOOKUP_ID</code> appended to it. The device's contacts app delegates read and write permissions to this content URI for the life of your activity. See the <a href="#">Content Provider Basics</a> guide for more details.										

**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC
	<pre>25  * Parsed form of the intent sent to the Contacts application. 26  */ 27  public class ContactsRequest { 28 29      /** Default mode: browse contacts */ 30      public static final int ACTION_DEFAULT = 10; 31 32      /** Show all contacts */ 33      public static final int ACTION_ALL_CONTACTS = 15; 34 35      /** Show all contacts with phone numbers */ 36      public static final int ACTION_CONTACTS_WITH_PHONES = 17; 37 38      /** Show contents of a specific group */ 39      public static final int ACTION_GROUP = 20; 40 41      /** Show all starred contacts */ 42      public static final int ACTION_STARRED = 30; 43 44      /** Show frequently contacted contacts */ 45      public static final int ACTION_FREQUENT = 40; 46 47      /** Show starred and the frequent */ 48      public static final int ACTION_STREQUENT = 50; 49 50      /** Show all contacts and pick them when clicking */ 51      public static final int ACTION_PICK_CONTACT = 60; 52 53      /** Show all contacts as well as the option to create a new one */ 54      public static final int ACTION_PICK_OR_CREATE_CONTACT = 70; 55 56      /** Show all contacts and pick them for edit when clicking, and allow creating a new contact */ 57      public static final int ACTION_INSERT_OR_EDIT_CONTACT = 80;</pre> <p><a href="https://android.googlesource.com/platform/packages/apps/Contacts/+nougat-mr1-release/src/com/android/contacts/list/ContactsRequest.java">https://android.googlesource.com/platform/packages/apps/Contacts/+nougat-mr1-release/src/com/android/contacts/list/ContactsRequest.java</a></p>

**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC
	<pre>59  /** Show all phone numbers and pick them when clicking */ 60  public static final int ACTION_PICK_PHONE = 90; 61 62  /** Show all postal addresses and pick them when clicking */ 63  public static final int ACTION_PICK_POSTAL = 100; 64 65  /** Show all postal addresses and pick them when clicking */ 66  public static final int ACTION_PICK_EMAIL = 105; 67 68  /** Show all contacts and create a shortcut for the picked contact */ 69  public static final int ACTION_CREATE_SHORTCUT_CONTACT = 110; 70 71  /** Show all phone numbers and create a call shortcut for the picked number */ 72  public static final int ACTION_CREATE_SHORTCUT_CALL = 120; 73 74  /** Show all phone numbers and create an SMS shortcut for the picked number */ 75  public static final int ACTION_CREATE_SHORTCUT_SMS = 130; 76 77  /** Show all contacts and activate the specified one */ 78  public static final int ACTION_VIEW_CONTACT = 140; 79 80  /** Show contacts recommended for joining with a specified target contact */ 81  public static final int ACTION_PICK_JOIN = 150;</pre> <p><a href="https://android.googlesource.com/platform/packages/apps/Contacts+/nougat-mr1-release/src/com/android/contacts/list/ContactsRequest.java">https://android.googlesource.com/platform/packages/apps/Contacts+/nougat-mr1-release/src/com/android/contacts/list/ContactsRequest.java</a></p>



**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC
	<pre> 104  * Displays a list to browse contacts. 105  */ 106  public class PeopleActivity extends ContactsActivity implements 107      View.OnCreateContextMenuListener, 108      View.OnClickListener, 109      ActionBarAdapter.Listener, 110      DialogManager.DialogShowingViewActivity, 111      ContactListFilterController.ContactListFilterListener, 112      ProviderStatusListener, 113      MultiContactDeleteListener, 114      JoinContactsListener { https://android.googlesource.com/platform/packages/apps/Contacts+/nougat-mr1- release/src/com/android/contacts/activities/PeopleActivity.java 145  * Showing a list of Contacts. Also used for showing search results in search mode. 146  */ 147  private MultiSelectContactsListFragment mAllFragment; 148  private ContactTileListFragment mFavoritesFragment; https://android.googlesource.com/platform/packages/apps/Contacts+/nougat-mr1- release/src/com/android/contacts/activities/PeopleActivity.java </pre>

**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC
	<pre>458     private void configureFragments(boolean fromRequest) { 459         if (fromRequest) { 460             ContactListFilter filter = null; 461             int actionCode = mRequest.getActionCode(); 462             boolean searchMode = mRequest.isSearchMode(); 463             final int tabToOpen; 464             switch (actionCode) { 465                 case ContactsRequest.ACTION_ALL_CONTACTS: 466                     filter = ContactListFilter.createFilterWithType( 467                         ContactListFilter.FILTER_TYPE_ALL_ACCOUNTS); 468                     tabToOpen = TabState.ALL; 469                     break; 470                 case ContactsRequest.ACTION_CONTACTS_WITH_PHONES: 471                     filter = ContactListFilter.createFilterWithType( 472                         ContactListFilter.FILTER_TYPE_WITH_PHONE_NUMBERS_ONLY); 473                     tabToOpen = TabState.ALL; 474                     break; 475 476                 case ContactsRequest.ACTION_FREQUENT: 477                 case ContactsRequest.ACTION_STREQUENT: 478                 case ContactsRequest.ACTION_STARRED: 479                     tabToOpen = TabState.FAVORITES; 480                     break; 481                 case ContactsRequest.ACTION_VIEW_CONTACT: 482                     tabToOpen = TabState.ALL; 483                     break; 484                 default: 485                     tabToOpen = -1; 486                     break; 487             } </pre> <p data-bbox="506 1321 1566 1390"><a href="https://android.googlesource.com/platform/packages/apps/Contacts+/nougat-mr1-release/src/com/android/contacts/activities/PeopleActivity.java">https://android.googlesource.com/platform/packages/apps/Contacts+/nougat-mr1-release/src/com/android/contacts/activities/PeopleActivity.java</a></p>

**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC
	<pre>488         if (tabToOpen != -1) { 489             mActionBarAdapter.setCurrentTab(tabToOpen); 490         } 491 492         if (filter != null) { 493             mContactListFilterController.setContactListFilter(filter, false); 494             searchMode = false; 495         } 496 497         if (mRequest.getContactUri() != null) { 498             searchMode = false; 499         } 500 501         mActionBarAdapter.setSearchMode(searchMode); 502         configureContactListFragmentForRequest(); 503     } 504 505     configureContactListFragment(); 506 507     invalidateOptionsMenuIfNeeded(); 508 }</pre> <p><a href="https://android.googlesource.com/platform/packages/apps/Contacts+/nougat-mr1-release/src/com/android/contacts/activities/PeopleActivity.java">https://android.googlesource.com/platform/packages/apps/Contacts+/nougat-mr1-release/src/com/android/contacts/activities/PeopleActivity.java</a></p>

**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC
	<pre>35 public class ProfileAndContactsLoader extends CursorLoader { 36 37     private boolean mLoadProfile; 38 39     private String[] mProjection; 40 41     private Uri mExtraUri; 42     private String[] mExtraProjection; 43     private String mExtraSelection; 44     private String[] mExtraSelectionArgs; 45     private boolean mMergeExtraContactsAfterPrimary; 46 47     public ProfileAndContactsLoader(Context context) { 48         super(context); 49     }</pre> <p><a href="https://android.googlesource.com/platform/packages/apps/ContactsCommon/+/nougat-mr1-release/src/com/android/contacts/common/list/ProfileAndContactsLoader.java">https://android.googlesource.com/platform/packages/apps/ContactsCommon/+/nougat-mr1-release/src/com/android/contacts/common/list/ProfileAndContactsLoader.java</a></p>

**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC
	<pre>35 public final class GroupMemberLoader extends CursorLoader { 36 37     public static class GroupEditorQuery { 38         private static final String[] PROJECTION = new String[] { 39             Data.CONTACT_ID,           // 0 40             Data.RAW_CONTACT_ID,      // 1 41             Data.DISPLAY_NAME_PRIMARY, // 2 42             Data.PHOTO_URI,           // 3 43             Data.LOOKUP_KEY,          // 4 44         }; 45 46         public static final int CONTACT_ID           = 0; 47         public static final int RAW_CONTACT_ID      = 1; 48         public static final int CONTACT_DISPLAY_NAME_PRIMARY = 2; 49         public static final int CONTACT_PHOTO_URI   = 3; 50         public static final int CONTACT_LOOKUP_KEY  = 4; 51     } 52 53     public static class GroupDetailQuery { 54         private static final String[] PROJECTION = new String[] { 55             Data.CONTACT_ID,           // 0 56             Data.PHOTO_URI,           // 1 57             Data.LOOKUP_KEY,          // 2 58             Data.DISPLAY_NAME_PRIMARY, // 3 59             Data.CONTACT_PRESENCE,    // 4 60             Data.CONTACT_STATUS,      // 5 61         }; 62 63         public static final int CONTACT_ID           = 0; 64         public static final int CONTACT_PHOTO_URI   = 1; 65         public static final int CONTACT_LOOKUP_KEY  = 2; 66         public static final int CONTACT_DISPLAY_NAME_PRIMARY = 3; 67         public static final int CONTACT_PRESENCE_STATUS = 4; 68         public static final int CONTACT_STATUS      = 5; 69     } 70 71     private final long mGroupId;</pre> <p><a href="https://android.googlesource.com/platform/packages/apps/Contacts/+nougat-mr1-release/src/com/android/contacts/GroupMemberLoader.java">https://android.googlesource.com/platform/packages/apps/Contacts/+nougat-mr1-release/src/com/android/contacts/GroupMemberLoader.java</a></p>

**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC
	<pre>24 * Group loader for the group list that includes details such as the number of contacts per group 25 * and number of groups per account. This list is sorted by account type, account name, where the 26 * group names are in alphabetical order. Note that the list excludes default, favorite, and deleted 27 * groups. 28 */ 29 public final class GroupListLoader extends CursorLoader { 30 31     private final static String[] COLUMNS = new String[] { 32         Groups.ACCOUNT_NAME, 33         Groups.ACCOUNT_TYPE, 34         Groups.DATA_SET, 35         Groups._ID, 36         Groups.TITLE, 37         Groups.SUMMARY_COUNT, 38     }; 39 40     public final static int ACCOUNT_NAME = 0; 41     public final static int ACCOUNT_TYPE = 1; 42     public final static int DATA_SET = 2; 43     public final static int GROUP_ID = 3; 44     public final static int TITLE = 4; 45     public final static int MEMBER_COUNT = 5; 46 47     private static final Uri GROUP_LIST_URI = Groups.CONTENT_SUMMARY_URI; 48 49     public GroupListLoader(Context context) { 50         super(context, GROUP_LIST_URI, COLUMNS, Groups.ACCOUNT_TYPE + " NOT NULL AND " 51             + Groups.ACCOUNT_NAME + " NOT NULL AND " + Groups.AUTO_ADD + "=0 AND " + 52             Groups.FAVORITES + "=0 AND " + Groups.DELETED + "=0", null, 53             Groups.ACCOUNT_TYPE + ", " + Groups.ACCOUNT_NAME + ", " + Groups.DATA_SET + ", " + 54             Groups.TITLE + " COLLATE LOCALIZED ASC"); 55     } 56 }</pre> <p><a href="https://android.googlesource.com/platform/packages/apps/Contacts+/nougat-mr1-release/src/com/android/contacts/GroupListLoader.java">https://android.googlesource.com/platform/packages/apps/Contacts+/nougat-mr1-release/src/com/android/contacts/GroupListLoader.java</a></p>

**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC
	<pre> 24  * Group meta-data loader. Loads all groups or just a single group from the 25  * database (if given a {@link Uri}). 26  */ 27  public final class GroupMetaLoader extends CursorLoader { 28 29      private final static String[] COLUMNS = new String[] { 30          Groups.ACCOUNT_NAME, 31          Groups.ACCOUNT_TYPE, 32          Groups.DATA_SET, 33          Groups._ID, 34          Groups.TITLE, 35          Groups.AUTO_ADD, 36          Groups.FAVORITES, 37          Groups.GROUP_IS_READ_ONLY, 38          Groups.DELETED, 39      }; 40 41      public final static int ACCOUNT_NAME = 0; 42      public final static int ACCOUNT_TYPE = 1; 43      public final static int DATA_SET = 2; 44      public final static int GROUP_ID = 3; 45      public final static int TITLE = 4; 46      public final static int AUTO_ADD = 5; 47      public final static int FAVORITES = 6; 48      public final static int IS_READ_ONLY = 7; 49      public final static int DELETED = 8; 50 51      public GroupMetaLoader(Context context, Uri groupUri) { 52          super(context, ensureIsGroupUri(groupUri), COLUMNS, Groups.ACCOUNT_TYPE + " NOT NULL AND " 53              + Groups.ACCOUNT_NAME + " NOT NULL", null, null); 54      } 55 56      /** 57       * Ensures that this is a valid group URI. If invalid, then an exception is 58       * thrown. Otherwise, the original URI is returned. 59       */ 60      private static Uri ensureIsGroupUri(final Uri groupUri) { 61          // TODO: Fix ContactsProvider2 getType method to resolve the group Uris 62          if (groupUri == null) { 63              throw new IllegalArgumentException("Uri must not be null"); 64          } 65          if (!groupUri.toString().startsWith(Groups.CONTENT_URI.toString())) { 66              throw new IllegalArgumentException("Invalid group Uri: " + groupUri); 67          } 68          return groupUri; 69      } 70  } </pre> <p><a href="https://android.googlesource.com/platform/packages/apps/Contacts/+nougat-mr1-release/src/com/android/contacts/GroupMetaLoader.java">https://android.googlesource.com/platform/packages/apps/Contacts/+nougat-mr1-release/src/com/android/contacts/GroupMetaLoader.java</a></p>

**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC
	<pre>19  * Meta-data for a contact group. We load all groups associated with the contact's 20  * constituent accounts. 21  */ 22  public final class GroupMetaData { 23      private String mAccountName; 24      private String mAccountType; 25      private String mDataSet; 26      private long mGroupId; 27      private String mTitle; 28      private boolean mDefaultGroup; 29      private boolean mFavorites; 30 31      public GroupMetaData(String accountName, String accountType, String dataSet, long groupId, 32                          String title, boolean defaultGroup, boolean favorites) { 33          this.mAccountName = accountName; 34          this.mAccountType = accountType; 35          this.mDataSet = dataSet; 36          this.mGroupId = groupId; 37          this.mTitle = title; 38          this.mDefaultGroup = defaultGroup; 39          this.mFavorites = favorites; 40      } 41 42      public String getAccountName() { 43          return mAccountName; 44      } 45 46      public String getAccountType() { 47          return mAccountType; 48      } 49 50      public String getDataSet() { 51          return mDataSet; 52      } 53 54      public long getGroupId() { 55          return mGroupId; 56      } 57 58      public String getTitle() { 59          return mTitle; 60      } 61 62      public boolean isDefaultGroup() { 63          return mDefaultGroup; 64      } 65 66      public boolean isFavorites() { 67          return mFavorites; 68      } 69  }</pre> <p><a href="https://android.googlesource.com/platform/packages/apps/ContactsCommon/+nougat-mr1-release/src/com/android/contacts/common/GroupMetaData.java">https://android.googlesource.com/platform/packages/apps/ContactsCommon/+nougat-mr1-release/src/com/android/contacts/common/GroupMetaData.java</a></p>



**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC
	<pre>44 * Class that sends chat message via SMS. 45 * 46 * The interface emulates a blocking sending similar to making an HTTP request. 47 * It calls the SmsManager to send a (potentially multipart) message and waits 48 * on the sent status on each part. The waiting has a timeout so it won't wait 49 * forever. Once the sent status of all parts received, the call returns. 50 * A successful sending requires success status for all parts. Otherwise, we 51 * pick the highest level of failure as the error for the whole message, which 52 * is used to determine if we need to retry the sending. 53 */ 54 public class SmsSender { 55     private static final String TAG = LogUtil.BUGLE_TAG; 56 57     public static final String EXTRA_PART_ID = "part_id"; 58 59     /* 60      * A map for pending sms messages. The key is the random request UUID. 61      */ 62     private static ConcurrentHashMap&lt;Uri, SendResult&gt; sPendingMessageMap = 63         new ConcurrentHashMap&lt;Uri, SendResult&gt;(); 64 65     private static final Random RANDOM = new Random(); 66 67     // Whether we should send multipart SMS as separate messages 68     private static Boolean sSendMultipartSmsAsSeparateMessages = null; 69</pre> <p><a href="https://android.googlesource.com/platform/packages/apps/Messaging/+nougat-mr1-release/src/com/android/messaging/sms/SmsSender.java">https://android.googlesource.com/platform/packages/apps/Messaging/+nougat-mr1-release/src/com/android/messaging/sms/SmsSender.java</a></p>

**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC
	<pre>253 // Actually sending the message using SmsManager 254 private static void sendInternal(final Context context, final int subId, String dest, 255     final ArrayList&lt;String&gt; messages, final String serviceCenter, 256     final boolean requireDeliveryReport, final Uri messageUri) throws SmsException { 257     Assert.notNull(context); 258     final SmsManager smsManager = PhoneUtils.get(subId).getSmsManager(); 259     final int messageCount = messages.size(); 260     final ArrayList&lt;PendingIntent&gt; deliveryIntents = new ArrayList&lt;PendingIntent&gt;(messageCount); 261     final ArrayList&lt;PendingIntent&gt; sentIntents = new ArrayList&lt;PendingIntent&gt;(messageCount); 262     for (int i = 0; i &lt; messageCount; i++) { 263         // Make pending intents different for each message part 264         final int partId = (messageCount &lt;= 1 ? 0 : i + 1); 265         if (requireDeliveryReport &amp;&amp; (i == (messageCount - 1))) { 266             // TODO we only care about the delivery status of the last part 267             // Shall we have better tracking of delivery status of all parts? 268             deliveryIntents.add(PendingIntent.getBroadcast( 269                 context, 270                 partId, 271                 getSendStatusIntent(context, SendStatusReceiver.MESSAGE_DELIVERED_ACTION, 272                     messageUri, partId, subId), 273                 0/*flag*/)); 274         } else { 275             deliveryIntents.add(null); 276         } 277         sentIntents.add(PendingIntent.getBroadcast( 278             context, 279             partId, 280             getSendStatusIntent(context, SendStatusReceiver.MESSAGE_SENT_ACTION, 281                 messageUri, partId, subId), 282             0/*flag*/)); 283     } 284     if (sSendMultipartSmsAsSeparateMessages == null) { 285         sSendMultipartSmsAsSeparateMessages = MmsConfig.get(subId) 286             .getSendMultipartSmsAsSeparateMessages(); 287     }</pre> <p><a href="https://android.googlesource.com/platform/packages/apps/Messaging/+nougat-mr1-release/src/com/android/messaging/sms/SmsSender.java">https://android.googlesource.com/platform/packages/apps/Messaging/+nougat-mr1-release/src/com/android/messaging/sms/SmsSender.java</a></p>

**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC
	<pre> 288     try { 289         if (sSendMultipartSmsAsSeparateMessages) { 290             // If multipart sms is not supported, send them as separate messages 291             for (int i = 0; i &lt; messageCount; i++) { 292                 smsManager.sendTextMessage(dest, 293                     serviceCenter, 294                     messages.get(i), 295                     sentIntents.get(i), 296                     deliveryIntents.get(i)); 297             } 298         } else { 299             smsManager.sendMultipartTextMessage( 300                 dest, serviceCenter, messages, sentIntents, deliveryIntents); 301         } 302     } catch (final Exception e) { 303         throw new SmsException("SmsSender: caught exception in sending " + e); 304     } 305 } </pre> <p><a href="https://android.googlesource.com/platform/packages/apps/Messaging/+nougat-mr1-release/src/com/android/messaging/sms/SmsSender.java">https://android.googlesource.com/platform/packages/apps/Messaging/+nougat-mr1-release/src/com/android/messaging/sms/SmsSender.java</a></p> <pre> 56  * Class that receives incoming SMS messages through android.provider.Telephony.SMS_RECEIVED 57  * 58  * This class serves two purposes: 59  * - Process phone verification SMS messages 60  * - Handle SMS messages when the user has enabled us to be the default SMS app (Pre-KLP) 61  */ 62  public final class SmsReceiver extends BroadcastReceiver { 63      private static final String TAG = LogUtil.BUGLE_TAG; 64 65      private static ArrayList&lt;Pattern&gt; sIgnoreSmsPatterns; 66 </pre> <p><a href="https://android.googlesource.com/platform/packages/apps/Messaging/+nougat-mr1-release/src/com/android/messaging/receiver/SmsReceiver.java">https://android.googlesource.com/platform/packages/apps/Messaging/+nougat-mr1-release/src/com/android/messaging/receiver/SmsReceiver.java</a></p>

**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC
	<pre>200     public static void deliverSmsMessages(final Context context, final int subId, 201         final int errorCode, final android.telephony.SmsMessage[] messages) { 202         final ContentValues messageValues = 203             MmsUtils.parseReceivedSmsMessage(context, messages, errorCode); 204 205         LogUtil.v(TAG, "SmsReceiver.deliverSmsMessages"); 206 207         final long nowInMillis = System.currentTimeMillis(); 208         final long receivedTimestampMs = MmsUtils.getMessageDate(messages[0], nowInMillis); 209 210         messageValues.put(Sms.Inbox.DATE, receivedTimestampMs); 211         // Default to unread and unseen for us but ReceiveSmsMessageAction will override 212         // seen for the telephony db. 213         messageValues.put(Sms.Inbox.READ, 0); 214         messageValues.put(Sms.Inbox.SEEN, 0); 215         if (OsUtil.isAtLeastL_MR1()) { 216             messageValues.put(Sms.SUBSCRIPTION_ID, subId); 217         } 218 219         if (messages[0].getMessageClass() == android.telephony.SmsMessage.MessageClass.CLASS_0    220             DebugUtils.debugClassZeroSmsEnabled()) { 221             Factory.get().getUIIntents().launchClassZeroActivity(context, messageValues); 222         } else { 223             final ReceiveSmsMessageAction action = new ReceiveSmsMessageAction(messageValues); 224             action.start(); 225         } 226     }</pre> <p><a href="https://android.googlesource.com/platform/packages/apps/Messaging/+nougat-mr1-release/src/com/android/messaging/receiver/SmsReceiver.java">https://android.googlesource.com/platform/packages/apps/Messaging/+nougat-mr1-release/src/com/android/messaging/receiver/SmsReceiver.java</a></p>

**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC
	<pre>228     @Override 229     public void onReceive(final Context context, final Intent intent) { 230         LogUtil.v(TAG, "SmsReceiver.onReceive " + intent); 231         // On KLP+ we only take delivery of SMS messages in SmsDeliverReceiver. 232         if (PhoneUtils.getDefault().isSmsEnabled()) { 233             final String action = intent.getAction(); 234             if (OsUtil.isSecondaryUser() &amp;&amp; 235                 (Telephony.Sms.Intents.SMS_RECEIVED_ACTION.equals(action)    236                  // TODO: update this with the actual constant from Telephony 237                  "android.provider.Telephony.MMS_DOWNLOADED".equals(action))) { 238                 postNewMessageSecondaryUserNotification(); 239             } else if (!OsUtil.isAtLeastKLP()) { 240                 deliverSmsIntent(context, intent); 241             } 242         } 243     }</pre> <p><a href="https://android.googlesource.com/platform/packages/apps/Messaging/+/nougat-mr1-release/src/com/android/messaging/receiver/SmsReceiver.java">https://android.googlesource.com/platform/packages/apps/Messaging/+/nougat-mr1-release/src/com/android/messaging/receiver/SmsReceiver.java</a></p>

**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC
	<pre>52  * Class that sends chat message via MMS. 53  * 54  * The interface emulates a blocking send similar to making an HTTP request. 55  */ 56  public class MmsSender { 57      private static final String TAG = LogUtil.BUGLE_TAG; 58 59      /** 60       * Send an MMS message. 61       * 62       * @param context Context 63       * @param messageUri The unique URI of the message for identifying it during sending 64       * @param sendReq The SendReq PDU of the message 65       * @throws MmsFailureException 66       */ 67      public static void sendMms(final Context context, final int subId, final Uri messageUri, 68                               final SendReq sendReq, final Bundle sentIntentExtras) throws MmsFailureException { 69          sendMms(context, 70                subId, 71                messageUri, 72                null /* locationUri */, 73                sendReq, 74                true /* responseImportant */, 75                sentIntentExtras); 76      }</pre> <p><a href="https://android.googlesource.com/platform/packages/apps/Messaging/+nougat-mr1-release/src/com/android/messaging/sms/MmsSender.java">https://android.googlesource.com/platform/packages/apps/Messaging/+nougat-mr1-release/src/com/android/messaging/sms/MmsSender.java</a></p>

**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC
	<pre>240     * Download an MMS message. 241     * 242     * @param context Context 243     * @param contentLocation The url of the MMS message 244     * @throws MmsFailureException 245     * @throws InvalidHeaderValueException 246     */ 247     public static void downloadMms(final Context context, final int subId, 248         final String contentLocation, Bundle extras) throws MmsFailureException, 249         InvalidHeaderValueException { 250         final Uri requestUri = Uri.parse(contentLocation); 251         final Uri contentUri = MmsFileProvider.buildRawMmsUri(); 252 253         final Intent downloadedIntent = new Intent(SendStatusReceiver.MMS_DOWNLOADED_ACTION, 254             requestUri, 255             context, 256             SendStatusReceiver.class); 257         downloadedIntent.putExtra(SendMessageAction.EXTRA_CONTENT_URI, contentUri); 258         if (extras != null) { 259             downloadedIntent.putExtras(extras); 260         } 261         final PendingIntent downloadedPendingIntent = PendingIntent.getBroadcast( 262             context, 263             0 /*request code*/, 264             downloadedIntent, 265             PendingIntent.FLAG_UPDATE_CURRENT); 266 267         MmsManager.downloadMultimediaMessage(subId, context, contentLocation, contentUri, 268             downloadedPendingIntent); 269     }</pre> <p><a href="https://android.googlesource.com/platform/packages/apps/Messaging/+nougat-mr1-release/src/com/android/messaging/sms/MmsSender.java">https://android.googlesource.com/platform/packages/apps/Messaging/+nougat-mr1-release/src/com/android/messaging/sms/MmsSender.java</a></p>

**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC
	<pre> 97      * Execute an MMS HTTP request, either a POST (sending) or a GET (downloading) 98      * 99      * @param urlString The request URL, for sending it is usually the MMSC, and for downloading 100     *           it is the message URL 101     * @param pdu For POST (sending) only, the PDU to send 102     * @param method HTTP method, POST for sending and GET for downloading 103     * @param isProxySet Is there a proxy for the MMSC 104     * @param proxyHost The proxy host 105     * @param proxyPort The proxy port 106     * @param mmsConfig The MMS config to use 107     * @param userAgent The user agent header value 108     * @param uaProfUrl The UA Prof URL header value 109     * @return The HTTP response body 110     * @throws MmsHttpException For any failures 111     */ 112     public byte[] execute(String urlString, byte[] pdu, String method, boolean isProxySet, 113         String proxyHost, int proxyPort, Bundle mmsConfig, String userAgent, String uaProfUrl) 114         throws MmsHttpException { 115         Log.d(MmsService.TAG, "HTTP: " + method + " " + Utils.redactUrlForNonVerbose(urlString) 116             + (isProxySet ? (" proxy=" + proxyHost + ":" + proxyPort) : "")) 117             + ", PDU size=" + (pdu != null ? pdu.length : 0)); 118         checkMethod(method); 119         HttpURLConnection connection = null; 120         try { 121             Proxy proxy = Proxy.NO_PROXY; 122             if (isProxySet) { 123                 proxy = new Proxy(Proxy.Type.HTTP, new InetSocketAddress(proxyHost, proxyPort)); 124             } 125             final URL url = new URL(urlString); 126             // Now get the connection 127             connection = (HttpURLConnection) url.openConnection(proxy); 128             connection.setDoInput(true); 129             connection.setConnectTimeout( 130                 mmsConfig.getInt(CarrierConfigValuesLoader.CONFIG_HTTP_SOCKET_TIMEOUT, 131                     CarrierConfigValuesLoader.CONFIG_HTTP_SOCKET_TIMEOUT_DEFAULT)); </pre> <p data-bbox="506 1219 1596 1289"><a href="https://android.googlesource.com/platform/packages/apps/Messaging/+nougat-mr1-release/src/android/support/v7/mms/MmsHttpClient.java">https://android.googlesource.com/platform/packages/apps/Messaging/+nougat-mr1-release/src/android/support/v7/mms/MmsHttpClient.java</a></p>



**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC
	<pre> 132 // ----- COMMON HEADERS ----- 133 // Header: Accept 134 connection.setRequestProperty(HEADER_ACCEPT, HEADER_VALUE_ACCEPT); 135 // Header: Accept-Language 136 connection.setRequestProperty( 137     HEADER_ACCEPT_LANGUAGE, getCurrentAcceptLanguage(Locale.getDefault())); 138 // Header: User-Agent 139 Log.i(MmsService.TAG, "HTTP: User-Agent=" + userAgent); 140 connection.setRequestProperty(HEADER_USER_AGENT, userAgent); 141 // Header: x-wap-profile 142 final String uaProfUrlTagName = mmsConfig.getString( 143     CarrierConfigValuesLoader.CONFIG_UA_PROF_TAG_NAME, 144     CarrierConfigValuesLoader.CONFIG_UA_PROF_TAG_NAME_DEFAULT); 145 if (uaProfUrl != null) { 146     Log.i(MmsService.TAG, "HTTP: UaProfUrl=" + uaProfUrl); 147     connection.setRequestProperty(uaProfUrlTagName, uaProfUrl); 148 } 149 // Add extra headers specified by mms_config.xml's httpparams 150 addExtraHeaders(connection, mmsConfig); 151 // Different stuff for GET and POST 152 if (METHOD_POST.equals(method)) { 153     if (pdu == null    pdu.length &lt; 1) { 154         Log.e(MmsService.TAG, "HTTP: empty pdu"); 155         throw new MmsHttpException(0/*statusCode*/, "Sending empty PDU"); 156     } 157     connection.setDoOutput(true); 158     connection.setRequestMethod(METHOD_POST); 159     if (mmsConfig.getBoolean( 160         CarrierConfigValuesLoader.CONFIG_SUPPORT_HTTP_CHARSET_HEADER, 161         CarrierConfigValuesLoader.CONFIG_SUPPORT_HTTP_CHARSET_HEADER_DEFAULT)) { 162         connection.setRequestProperty(HEADER_CONTENT_TYPE, 163             HEADER_VALUE_CONTENT_TYPE_WITH_CHARSET); 164     } else { 165         connection.setRequestProperty(HEADER_CONTENT_TYPE, 166             HEADER_VALUE_CONTENT_TYPE_WITHOUT_CHARSET); </pre> <p data-bbox="506 1256 1591 1321"><a href="https://android.googlesource.com/platform/packages/apps/Messaging/+nougat-mr1-release/src/android/support/v7/mms/MmsHttpClient.java">https://android.googlesource.com/platform/packages/apps/Messaging/+nougat-mr1-release/src/android/support/v7/mms/MmsHttpClient.java</a></p>

**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC
	<pre> 167         } 168         if (Log.isLoggable(MmsService.TAG, Log.VERBOSE)) { 169             logHttpHeaders(connection.getRequestProperties()); 170         } 171         connection.setFixedLengthStreamingMode(pdu.length); 172         // Sending request body 173         final OutputStream out = 174             new BufferedOutputStream(connection.getOutputStream()); 175         out.write(pdu); 176         out.flush(); 177         out.close(); 178     } else if (METHOD_GET.equals(method)) { 179         if (Log.isLoggable(MmsService.TAG, Log.VERBOSE)) { 180             logHttpHeaders(connection.getRequestProperties()); 181         } 182         connection.setRequestMethod(METHOD_GET); 183     } 184     // Get response 185     final int responseCode = connection.getResponseCode(); 186     final String responseMessage = connection.getResponseMessage(); 187     Log.d(MmsService.TAG, "HTTP: " + responseCode + " " + responseMessage); 188     if (Log.isLoggable(MmsService.TAG, Log.VERBOSE)) { 189         logHttpHeaders(connection.getHeaderFields()); 190     } 191     if (responseCode / 100 != 2) { 192         throw new MmsHttpException(responseCode, responseMessage); 193     } 194     final InputStream in = new BufferedInputStream(connection.getInputStream()); 195     final ByteArrayOutputStream byteOut = new ByteArrayOutputStream(); 196     final byte[] buf = new byte[4096]; 197     int count = 0; 198     while ((count = in.read(buf)) &gt; 0) { 199         byteOut.write(buf, 0, count); 200     } 201     in.close(); 202     final byte[] responseBody = byteOut.toByteArray(); 203     Log.d(MmsService.TAG, "HTTP: response size=" 204         + (responseBody != null ? responseBody.length : 0)); 205     return responseBody; </pre> <p><a href="https://android.googlesource.com/platform/packages/apps/Messaging/+nougat-mr1-release/src/android/support/v7/mms/MmsHttpClient.java">https://android.googlesource.com/platform/packages/apps/Messaging/+nougat-mr1-release/src/android/support/v7/mms/MmsHttpClient.java</a></p>

**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC
	<pre>38  * Request to send an MMS 39  */ 40  class SendRequest extends MmsRequest { 41      // Max send response PDU size in bytes (exceeding this may cause problem with 42      // system intent delivery). 43      private static final int MAX_SEND_RESPONSE_SIZE = 1000 * 1024; 44 45      private byte[] mPduData; 46 47      SendRequest(final String locationUri, final Uri pduUri, final PendingIntent sentIntent) { 48          super(locationUri, pduUri, sentIntent); 49      } 50 51      @Override 52      protected boolean loadRequest(final Context context, final Bundle mmsConfig) { 53          mPduData = readPduFromContentUri( 54              context, 55              mPduUri, 56              mmsConfig.getInt( 57                  CarrierConfigValuesLoader.CONFIG_MAX_MESSAGE_SIZE, 58                  CarrierConfigValuesLoader.CONFIG_MAX_MESSAGE_SIZE_DEFAULT)); 59          return (mPduData != null); 60      } 61 62      @Override 63      protected boolean transferResponse(final Context context, final Intent fillIn, 64          final byte[] response) { 65          // SendConf pdus are always small and can be included in the intent 66          if (response != null &amp;&amp; fillIn != null) { 67              if (response.length &gt; MAX_SEND_RESPONSE_SIZE) { 68                  // If the response PDU is too large, it won't be able to fit in 69                  // the PendingIntent to be transferred via system IPC. 70                  return false; 71              } 72              fillIn.putExtra(SmsManager.EXTRA_MMS_DATA, response); 73          } 74          return true; 75      } }</pre>

**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC
	<p><a href="https://android.googlesource.com/platform/packages/apps/Messaging/+/nougat-mr1-release/src/android/support/v7/mms/SendRequest.java">https://android.googlesource.com/platform/packages/apps/Messaging/+/nougat-mr1-release/src/android/support/v7/mms/SendRequest.java</a></p> <pre data-bbox="533 354 1738 399">public static LocationRequest create ()</pre> <p data-bbox="525 423 1029 451">Create a location request with default parameters.</p> <p data-bbox="525 482 1638 542">Default parameters are for a block accuracy, slowly updated location. It can then be adjusted as required by the applications before passing to the <code>FusedLocationProviderApi</code>.</p> <p data-bbox="548 565 630 589"><b>Returns</b></p> <ul data-bbox="554 610 810 634" style="list-style-type: none"> <li>• a new location request</li> </ul> <p><a href="https://developers.google.com/android/reference/com/google/android/gms/location/LocationRequest">https://developers.google.com/android/reference/com/google/android/gms/location/LocationRequest</a></p>

**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC
	<p data-bbox="533 248 1745 285">public static final int <b>PRIORITY_BALANCED_POWER_ACCURACY</b></p> <p data-bbox="533 315 1178 342">Used with <code>setPriority(int)</code> to request "block" level accuracy.</p> <p data-bbox="533 371 1644 431">Block level accuracy is considered to be about 100 meter accuracy. Using a coarse accuracy such as this often consumes less power.</p> <p data-bbox="533 456 730 483">Constant Value: 102</p> <p data-bbox="533 537 1745 574">public static final int <b>PRIORITY_HIGH_ACCURACY</b></p> <p data-bbox="533 604 1335 631">Used with <code>setPriority(int)</code> to request the most accurate locations available.</p> <p data-bbox="533 660 961 688">This will return the finest location available.</p> <p data-bbox="533 717 730 745">Constant Value: 100</p> <p data-bbox="533 799 1745 836">public static final int <b>PRIORITY_LOW_POWER</b></p> <p data-bbox="533 865 1157 893">Used with <code>setPriority(int)</code> to request "city" level accuracy.</p> <p data-bbox="533 922 1738 982">City level accuracy is considered to be about 10km accuracy. Using a coarse accuracy such as this often consumes less power.</p> <p data-bbox="533 1006 730 1034">Constant Value: 104</p> <p data-bbox="533 1049 1797 1076"><a href="https://developers.google.com/android/reference/com/google/android/gms/location/LocationRequest">https://developers.google.com/android/reference/com/google/android/gms/location/LocationRequest</a></p>

**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC
	<pre data-bbox="533 250 1749 285">public Task&lt;Location&gt; getLastLocation ()</pre> <p data-bbox="527 315 1104 341">Returns the best most recent location currently available.</p> <p data-bbox="527 371 1696 431">If a location is not available, which should happen very rarely, null will be returned. The best accuracy available while respecting the location permissions will be returned.</p> <p data-bbox="527 462 1736 522">This method provides a simplified way to get location. It is particularly well suited for applications that do not require an accurate location and that do not want to maintain extra logic for location updates.</p> <pre data-bbox="533 578 1749 613">public Task&lt;LocationAvailability&gt; getLocationAvailability ()</pre> <p data-bbox="527 644 1692 704">Returns the availability of location data. When <code>isLocationAvailable()</code> returns true, then the location returned by <code>getLastLocation()</code> will be reasonably up to date within the hints specified by the active <code>LocationRequest</code>s.</p> <p data-bbox="527 735 1472 761">If the client isn't connected to Google Play services and the request times out, null is returned.</p> <p data-bbox="527 792 1673 852">Note it's always possible for <code>getLastLocation()</code> to return null even when this method returns true (e.g. location settings were disabled between calls).</p> <p data-bbox="512 867 1900 927"><a href="https://developers.google.com/android/reference/com/google/android/gms/location/FusedLocationProviderClient">https://developers.google.com/android/reference/com/google/android/gms/location/FusedLocationProviderClient</a></p>

**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC						
	<p data-bbox="527 245 1749 326"> <code>public Task&lt;Void&gt; requestLocationUpdates (LocationRequest request, LocationCallback callback, Looper looper)</code> </p> <p data-bbox="527 354 1272 378">Requests location updates with a callback on the specified Looper thread.</p> <p data-bbox="527 410 1686 472">This method is suited for the foreground use cases. For background use cases, the <code>PendingIntent</code> version of the method is recommended, see <code>requestLocationUpdates(LocationRequest, PendingIntent)</code>.</p> <p data-bbox="527 505 1371 529">Any previous LocationRequests registered on this LocationListener will be replaced.</p> <p data-bbox="527 561 1686 656">This call will keep the Google Play services connection active, so make sure to call <code>removeLocationUpdates(LocationCallback)</code> when you no longer need it, otherwise you lose the benefits of the automatic connection management.</p> <p data-bbox="527 688 1745 712">Callbacks for <code>LocationCallback</code> will be made on the specified thread, which must already be a prepared looper thread.</p> <p data-bbox="527 737 667 761"><b>Parameters</b></p> <table border="1" data-bbox="527 792 1749 1005"> <tr> <td data-bbox="527 800 625 857"><b>request</b></td> <td data-bbox="632 800 1749 857">The location request for the updates.</td> </tr> <tr> <td data-bbox="527 865 625 922"><b>callback</b></td> <td data-bbox="632 865 1749 922">The callback for the location updates.</td> </tr> <tr> <td data-bbox="527 930 625 1005"><b>looper</b></td> <td data-bbox="632 930 1749 1005">The Looper object whose message queue will be used to implement the callback mechanism, or null to make callbacks on the calling thread.</td> </tr> </table> <p data-bbox="527 1024 1902 1084"> <a href="https://developers.google.com/android/reference/com/google/android/gms/location/FusedLocationProviderClient">https://developers.google.com/android/reference/com/google/android/gms/location/FusedLocationProviderClient</a> </p>	<b>request</b>	The location request for the updates.	<b>callback</b>	The callback for the location updates.	<b>looper</b>	The Looper object whose message queue will be used to implement the callback mechanism, or null to make callbacks on the calling thread.
<b>request</b>	The location request for the updates.						
<b>callback</b>	The callback for the location updates.						
<b>looper</b>	The Looper object whose message queue will be used to implement the callback mechanism, or null to make callbacks on the calling thread.						

**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC				
	<pre data-bbox="533 240 1740 321">public Task&lt;Void&gt; requestLocationUpdates (LocationRequest request, PendingIntent callbackIntent)</pre> <p data-bbox="525 354 1268 378">Requests location updates with a callback on the specified PendingIntent.</p> <p data-bbox="525 410 1732 540">This method is suited for the background use cases, more specifically for receiving location updates, even when the app has been killed by the system. In order to do so, use a <code>PendingIntent</code> for a started service. For foreground use cases, the <code>LocationCallback</code> version of the method is recommended, see <code>requestLocationUpdates(LocationRequest, LocationCallback, Looper)</code>.</p> <p data-bbox="525 573 1724 630">Any previously registered requests that have the same PendingIntent (as defined by <code>equals(Object)</code>) will be replaced by this request.</p> <p data-bbox="525 662 1728 760">Both <code>LocationResult</code> and <code>LocationAvailability</code> are sent to the given PendingIntent. You can extract data from an Intent using <code>hasResult(Intent)</code>, <code>extractResult(Intent)</code>, <code>hasLocationAvailability(Intent)</code>, and <code>extractLocationAvailability(Intent)</code>.</p> <p data-bbox="543 784 667 808"><b>Parameters</b></p> <table border="1" data-bbox="525 833 1740 971"> <tbody> <tr> <td data-bbox="525 833 835 906"><code>request</code></td> <td data-bbox="835 833 1740 906">The location request for the updates.</td> </tr> <tr> <td data-bbox="525 906 835 971"><code>callbackIntent</code></td> <td data-bbox="835 906 1740 971">A pending intent to be sent for each location update.</td> </tr> </tbody> </table> <p data-bbox="543 995 630 1019"><b>Returns</b></p> <ul data-bbox="554 1044 1360 1068" style="list-style-type: none"> <li>• a Task for the call, check <code>isSuccessful()</code> to determine if it was successful.</li> </ul> <p data-bbox="512 1084 1898 1141"><a href="https://developers.google.com/android/reference/com/google/android/gms/location/FusedLocationProviderClient">https://developers.google.com/android/reference/com/google/android/gms/location/FusedLocationProviderClient</a></p>	<code>request</code>	The location request for the updates.	<code>callbackIntent</code>	A pending intent to be sent for each location update.
<code>request</code>	The location request for the updates.				
<code>callbackIntent</code>	A pending intent to be sent for each location update.				



**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC						
	<p data-bbox="533 245 1738 277"><code>public void onLocationAvailability (LocationAvailability locationAvailability)</code></p> <p data-bbox="533 310 1171 334">Called when there is a change in the availability of location data.</p> <p data-bbox="533 367 1738 561">When <code>isLocationAvailable()</code> returns <code>false</code> you can assume that location will not be returned in <code>onLocationResult(LocationResult)</code> until something changes in the device's settings or environment. Even when <code>isLocationAvailable()</code> returns <code>true</code> the <code>onLocationResult(LocationResult)</code> may not always be called regularly, however the device location is known and both the most recently delivered location and <code>getLastLocation(GoogleApiClient)</code> will be reasonably up to date given the hints specified by the active <code>LocationRequest</code> s.</p> <p data-bbox="533 586 667 610"><b>Parameters</b></p> <table border="1" data-bbox="533 643 1738 708"> <tr> <td data-bbox="533 651 961 699"><code>locationAvailability</code></td> <td data-bbox="968 651 1738 699">The current status of location availability.</td> </tr> </table> <p data-bbox="533 756 1738 789"><code>public void onLocationResult (LocationResult result)</code></p> <p data-bbox="533 821 1052 846">Called when device location information is available.</p> <p data-bbox="533 878 1661 943">The most recent location returned by <code>getLastLocation()</code> is not guaranteed to be immediately fresh, but will be reasonably up to date given the hints specified by the active <code>LocationRequest</code> s.</p> <p data-bbox="533 967 667 992"><b>Parameters</b></p> <table border="1" data-bbox="533 1024 1738 1089"> <tr> <td data-bbox="533 1032 768 1081"><code>result</code></td> <td data-bbox="774 1032 1738 1089">The latest location result available.</td> </tr> </table> <p data-bbox="512 1105 1808 1130"><a href="https://developers.google.com/android/reference/com/google/android/gms/location/LocationCallback">https://developers.google.com/android/reference/com/google/android/gms/location/LocationCallback</a></p> <p data-bbox="533 1146 1738 1179"><code>public abstract void onLocationChanged (Location location)</code></p> <p data-bbox="533 1211 915 1235">Called when the location has changed.</p> <p data-bbox="533 1260 667 1284"><b>Parameters</b></p> <table border="1" data-bbox="533 1317 1738 1382"> <tr> <td data-bbox="533 1325 926 1373"><code>location</code></td> <td data-bbox="932 1325 1738 1382">The updated location.</td> </tr> </table>	<code>locationAvailability</code>	The current status of location availability.	<code>result</code>	The latest location result available.	<code>location</code>	The updated location.
<code>locationAvailability</code>	The current status of location availability.						
<code>result</code>	The latest location result available.						
<code>location</code>	The updated location.						

**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC		
	<p><a href="https://developers.google.com/android/reference/com/google/android/gms/location/LocationListener">https://developers.google.com/android/reference/com/google/android/gms/location/LocationListener</a></p> <p>Public Constructors</p> <hr/> <p>public <b>MapView</b> (<a href="#">Context</a> context)</p> <p>public <b>MapView</b> (<a href="#">Context</a> context, <a href="#">AttributeSet</a> attrs)</p> <p>public <b>MapView</b> (<a href="#">Context</a> context, <a href="#">AttributeSet</a> attrs, int defStyle)</p> <p>public <b>MapView</b> (<a href="#">Context</a> context, <a href="#">GoogleMapOptions</a> options)</p> <p><a href="https://developers.google.com/android/reference/com/google/android/gms/maps/MapView">https://developers.google.com/android/reference/com/google/android/gms/maps/MapView</a></p> <p>public void <b>getMapAsync</b> (<a href="#">OnMapReadyCallback</a> callback)</p> <p>Returns a non-null instance of the <a href="#">GoogleMap</a> , ready to be used.</p> <p>Note that:</p> <ul style="list-style-type: none"><li>• This method must be called from the main thread.</li><li>• The callback will be executed in the main thread.</li><li>• In the case where Google Play services is not installed on the user's device, the callback will not be triggered until the user installs it.</li><li>• The <a href="#">GoogleMap</a> object provided by the callback is non-null.</li></ul> <p><b>Parameters</b></p> <table border="1"><tr><td data-bbox="527 1203 688 1268">callback</td><td data-bbox="695 1203 1738 1268">The callback object that will be triggered when the map is ready to be used.</td></tr></table> <p>public final void <b>onCreate</b> (<a href="#">Bundle</a> savedInstanceState)</p> <p>You must call this method from the parent Activity/Fragment's corresponding method.</p>	callback	The callback object that will be triggered when the map is ready to be used.
callback	The callback object that will be triggered when the map is ready to be used.		

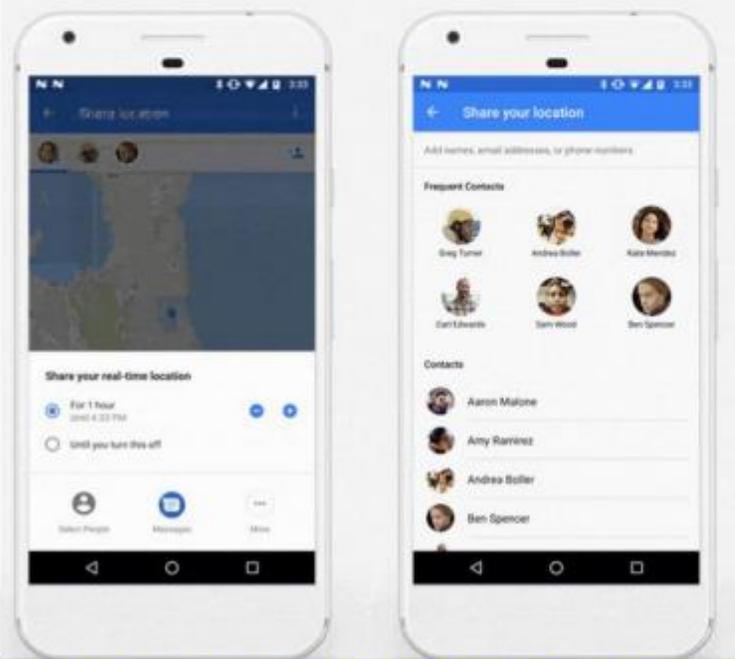
**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC
<p>[41D] transmitting IP-based messages including a location of the first device to the respective second devices;</p>	<p><a href="https://developers.google.com/android/reference/com/google/android/gms/maps/MapView">https://developers.google.com/android/reference/com/google/android/gms/maps/MapView</a></p> <p>HTC infringes directly and/or indirectly by performing, inducing others to perform, and/or contributing to the performance of transmitting IP-based messages including a location of the first device to the respective second devices. See claim 1[D] and 28[D], which are incorporated herein by reference in their entirety.</p> <p>For example, users send their location to a server and receive the location of other devices with whom the location is being shared. To send a location to the network, a user enables location service which enables the device to determine and send its location. If location service is already enabled, the device sends its location to the server as needed by the application (e.g. Google Maps). If location service is not enabled, the application will ask the user to enable location service in order to continue with full functionality, which includes using the device's location. Google Maps applications receive the location of other devices when those devices have location service enabled while using the same respective application. Android Device Manager and Google Maps use the received locations to display those locations on the map, indicating the locations of other devices.</p> <p>See, e.g., location sharing including corresponding code described above with regard to limitation [1C].</p> <p>Using Google Maps, a user enables location services to send its location the network, but the user can also choose to share its location, as shown below. Again, each device that participates is able to see the location of the other device using Google Maps' share your location feature. For example:</p>

**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC
	<p data-bbox="527 233 1304 277"><b>HTC One V™ – Google Location Service &amp; GPS</b></p> <p data-bbox="527 321 1717 428">Google Maps lets you track your current location, view real-time traffic situations, and receive detailed directions to your destination. It also provides a search tool where you can locate a place of interest or an address on a vector or aerial map, or view locations in street level.</p> <p data-bbox="527 472 898 500"><b>Turning on Location Services</b></p> <div data-bbox="527 516 1717 954"> </div> <ol data-bbox="527 980 1717 1154" style="list-style-type: none"> <li>1. From the Home Screen, slide the <b>Notifications</b> panel open.</li> <li>2. In the top right corner, tap <b>Settings</b>.</li> <li>3. Tap <b>Location</b>.</li> <li>4. Make your selection by tapping <b>Google's location service</b>, <b>Use GPS satellites</b>, or both.</li> </ol> <p data-bbox="1461 1084 1717 1154"><b>Note:</b> You will need to accept the location consent terms and conditions.</p>

## Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products

US9408055B2	HTC
	<ol style="list-style-type: none"><li>1. If you haven't already, add their Gmail address to your <a href="#">Google Contacts</a>.</li><li>2. On your Android phone or tablet, open the Google Maps app and sign in. <a href="#">Learn how to sign in.</a></li><li>3. Tap the Menu ≡ &gt; <b>Share location</b> &gt; Add People +.</li><li>4. Choose how long you want to share your location.</li><li>5. Tap <b>Select People</b>.<ul style="list-style-type: none"><li>• If you're asked about your contacts, give Google Maps access.</li></ul></li><li>6. Choose who you want to share with.</li><li>7. Tap <b>Share</b>.</li></ol> <p><a href="https://support.google.com/plus/answer/3302509?co=GENIE.Platform%3DAndroid&amp;hl=en">https://support.google.com/plus/answer/3302509?co=GENIE.Platform%3DAndroid&amp;hl=en</a></p>  <p><a href="https://arstechnica.com/gadgets/2017/03/location-sharing-finally-returns-to-google-maps/">https://arstechnica.com/gadgets/2017/03/location-sharing-finally-returns-to-google-maps/</a></p>

**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC
	 <p><a href="https://arstechnica.com/gadgets/2017/03/location-sharing-finally-returns-to-google-maps/">https://arstechnica.com/gadgets/2017/03/location-sharing-finally-returns-to-google-maps/</a></p> <p>Below are exemplary methods used by Google applications to obtain, send, and receive locations.</p>

**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC
	<p data-bbox="527 233 982 261">The Google Play services Location API</p> <p data-bbox="527 298 1356 347">The Google Play services <a href="#">Location API</a> is the preferred method for adding location awareness to your Android application. It includes functionality that lets you:</p> <ul data-bbox="552 371 1184 505" style="list-style-type: none"> <li>• Determine the device location.</li> <li>• Listen for location changes.</li> <li>• Determine the mode of transportation, if the device is moving.</li> <li>• Create and monitor predefined geographical regions, known as geofences.</li> </ul> <p data-bbox="527 529 1360 662">The location APIs make it easy for you to build power efficient, location-aware applications. Like the Google Maps Android API, the Location API is distributed as part of the Google Play services SDK. For more information on the Location API, please refer to the Android training class <a href="#">Making Your App Location Aware</a> or the <a href="#">Location API Reference</a>. Code examples are included as part of the Google Play services SDK.</p> <p data-bbox="527 667 1331 695"><a href="https://developers.google.com/maps/documentation/android-api/location">https://developers.google.com/maps/documentation/android-api/location</a></p> <div data-bbox="527 776 1339 1179"> <p>The diagram illustrates the architecture of the Google API Client. On the left, a box labeled 'Device' contains 'Your app' (represented by an Android robot icon) and the 'Google Play services library'. Inside the library is the 'Google API Client'. On the right, a dashed box labeled 'Google Play services' contains three service boxes: 'Games service', 'Drive service', and 'Other services'. Bidirectional arrows connect 'Your app' to the 'Google API Client', and the 'Google API Client' to each of the three service boxes.</p> </div> <p data-bbox="527 1192 1331 1268">Figure 1: An illustration showing how the Google API Client provides an interface for connecting and making calls to any of the available Google Play services such as Google Play Games and Google Drive.</p> <p data-bbox="527 1273 1247 1300"><a href="https://developers.google.com/android/guides/api-client#Starting">https://developers.google.com/android/guides/api-client#Starting</a></p>

**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC
	<h3 data-bbox="527 233 919 261">Get the Last Known Location</h3> <p data-bbox="527 289 1545 386">Once you have connected to Google Play services and the location services API, you can get the last known location of a user's device. When your app is connected to these you can use the fused location provider's <code>getLastLocation()</code> method to retrieve the device location. The precision of the location returned by this call is determined by the permission setting you put in your app manifest, as described in the <a href="#">Specify App Permissions</a> section of this document.</p> <p data-bbox="527 407 1545 480">To request the last known location, call the <code>getLastLocation()</code> method, passing it your instance of the <code>GoogleApiClient</code> object. Do this in the <code>onConnected()</code> callback provided by Google API Client, which is called when the client is ready. The following code snippet illustrates the request and a simple handling of the response:</p> <pre data-bbox="527 493 1545 781">public class MainActivity extends ActionBarActivity implements     ConnectionCallbacks, OnConnectionFailedListener {     ...     @Override     public void onConnected(Bundle connectionHint) {         mLstLocation = LocationServices.FusedLocationApi.getLastLocation(             mGoogleApiClient);         if (mLstLocation != null) {             mLatitudeText.setText(String.valueOf(mLstLocation.getLatitude()));             MLongitudeText.setText(String.valueOf(mLstLocation.getLongitude()));         }     } }</pre> <p data-bbox="527 797 1545 842">The <code>getLastLocation()</code> method returns a <code>Location</code> object from which you can retrieve the latitude and longitude coordinates of a geographic location. The location object returned may be null in rare cases when the location is not available.</p> <p data-bbox="527 849 1289 878"><a href="https://developer.android.com/training/location/retrieve-current.html">https://developer.android.com/training/location/retrieve-current.html</a></p>



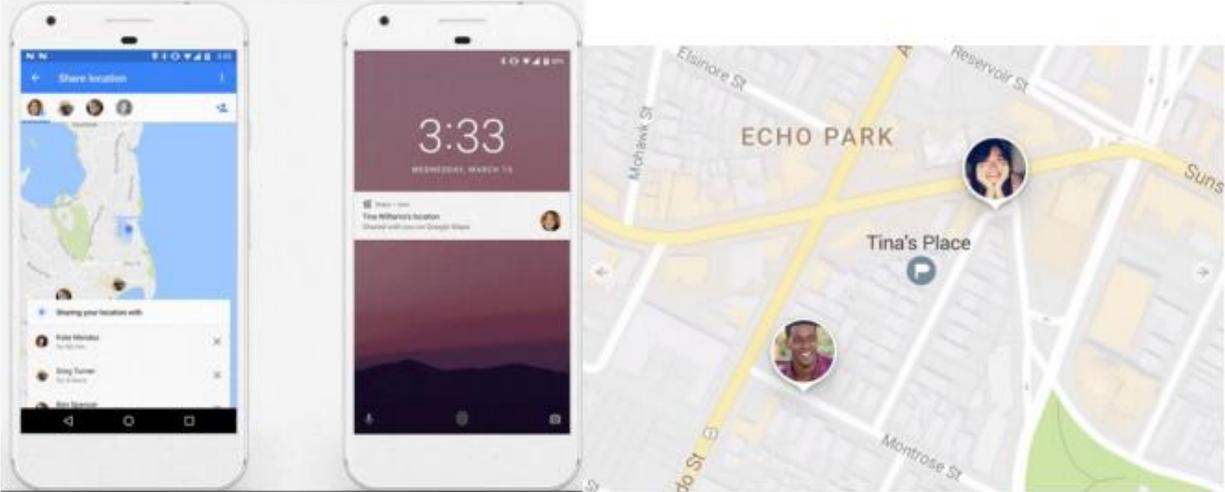
**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC
	<p data-bbox="537 240 1062 272">Determining the user's current location</p> <hr/> <p data-bbox="537 313 1331 334">The Geolocation API offers a simple, "one-shot" method to obtain the user's location:</p> <p data-bbox="537 345 1404 396"><code>getCurrentPosition()</code> . A call to this method asynchronously reports on the user's current location.</p> <pre data-bbox="537 435 1415 672"> window.onload = function() {   var startPos;   var geoSuccess = function(position) {     startPos = position;     document.getElementById('startLat').innerHTML = startPos.coords.latitude;     document.getElementById('startLon').innerHTML = startPos.coords.longitude;   };   navigator.geolocation.getCurrentPosition(geoSuccess); }; </pre> <p data-bbox="537 711 1488 797">If this is the first time that an application on this domain has requested permissions, the browser typically checks for user consent. Depending on the browser, there may also be preferences to always allow—or disallow—permission lookups, in which case the confirmation process is bypassed.</p> <p data-bbox="537 829 1470 915">Depending on the location device your browser is using, the position object might actually contain a lot more than just latitude and longitude; for example, it might include an altitude or a direction. You can't tell what extra information that location system uses until it actually returns the data.</p> <p data-bbox="525 954 1419 976"><a href="https://developers.google.com/web/fundamentals/native-hardware/user-location/">https://developers.google.com/web/fundamentals/native-hardware/user-location/</a></p>
[41E] presenting, via an interactive display of the first device, an interactive map and a plurality of user-selectable symbols corresponding to the plurality of second devices, wherein the symbols are positioned on the map at respective positions	<p data-bbox="514 997 1871 1170">HTC infringes directly and/or indirectly by performing, inducing others to perform, and/or contributing to the performance of presenting, via an interactive display of the first device, an interactive map and a plurality of user-selectable symbols corresponding to the plurality of second devices, wherein the symbols are positioned on the map at respective positions corresponding to the respective locations of the second devices. See claims 1[E] and 28[E], which are incorporated herein by reference in their entirety.</p> <p data-bbox="514 1214 1881 1281">For example, the Accused Products use Android Device Manager, and Google Maps to display an interface with a map and symbols representing devices.</p> <p data-bbox="514 1325 1860 1424">Using Android Device Manager, the user is presented with a map that appears to be based on or imported from Google Maps. The map is interactive because the user can pan, zoom, and make selections, among other interactivities. Depending on the number of devices linked to the Google Account, Android Device</p>

**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC
<p>corresponding to the respective locations of the second devices;</p>	<p>Manager places symbols on the map and in a list, where those symbols represent the linked devices and their positions. The symbols are selectable by touching the display. The map is georeferenced, at least because its contents, including the symbols, are placed and spaced according to geographical coordinates.</p> <p><b><u>Regarding Google Maps</u></b>, Google Latitude, Google Plus, Google Hangouts, Google Messages, Google Allo, Google Duo, Google Chrome, and Android Messenger, the Accused Products display, to the user on the display of the first device, a map with one or more symbols corresponding to one or more second users (or second devices corresponding to the second users). The map is interactive because the user may control the display of the map, e.g., pan, zoom, and/or effect change to the map in an otherwise interactive manner. The map is georeferenced for at least the reason that one or more symbols are associated with spatial locations, i.e., coordinates. The symbols are user-selectable because a user may touch the display to select the user or device associated with the symbol. For example, the user may input a touch selection directly on or near the portion of the display corresponding to the symbol's coordinates on the map to effect a selection of the user or device.</p> <p><b><u>Exemplary Support for Google Maps:</u></b></p> <p>Using Google Maps and its location sharing feature, the user is presented with a map that is interactive because the user can pan, zoom, and make selections, among other interactivities. Depending on how many other devices or Google Accounts are sharing their locations, Google Maps places symbols on the map and in a list, where those symbols represent the linked devices and their positions. The symbols are selectable by touching the display. The map is georeferenced, at least because its contents, including the symbols, are placed and spaced according to geographical coordinates.</p>

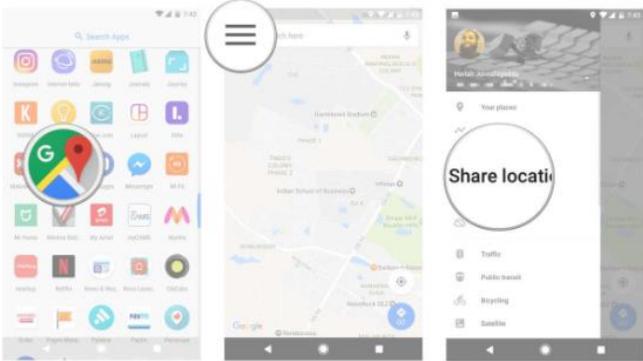
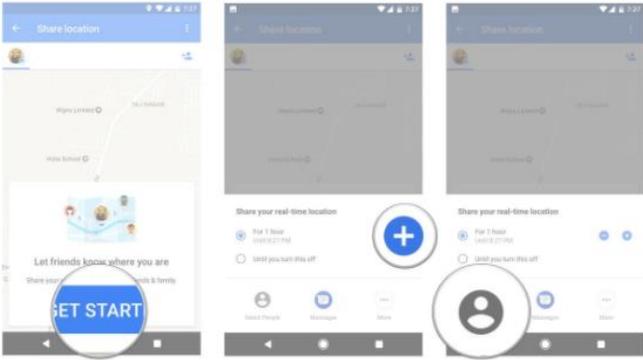
**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC
	 <p><a href="https://arstechnica.com/gadgets/2017/03/location-sharing-finally-returns-to-google-maps/">https://arstechnica.com/gadgets/2017/03/location-sharing-finally-returns-to-google-maps/</a></p> <p><b>See where someone is</b></p> <p>If someone shares their location with you, you can see them on the map.</p> <ol style="list-style-type: none"><li>1. Open the Google Maps app</li><li>2. Tap the Menu ≡ &gt; <b>Share location</b>.</li><li>3. Choose someone.</li></ol> <p>• To see an updated location, tap on a friend's icon &gt; More ⋮ &gt; <b>Refresh</b>.</p> <p><b>Stop seeing someone's location</b></p> <ol style="list-style-type: none"><li>1. Open the Google Maps app</li><li>2. On the map, tap their icon.</li><li>3. At the bottom, tap More ⋮</li><li>4. To temporarily hide someone, tap <b>Hide from map</b>. You can stop hiding them at any time.</li></ol> <p><b>Note:</b> You can stop someone's location from ever appearing on your map. <a href="#">Learn how to block another person's account.</a></p> <p><a href="https://support.google.com/maps/answer/7326816?hl=en&amp;ref_topic=3092425&amp;co=GENIE.Platform%3DAndroid&amp;oco=1">https://support.google.com/maps/answer/7326816?hl=en&amp;ref_topic=3092425&amp;co=GENIE.Platform%3DAndroid&amp;oco=1</a></p>

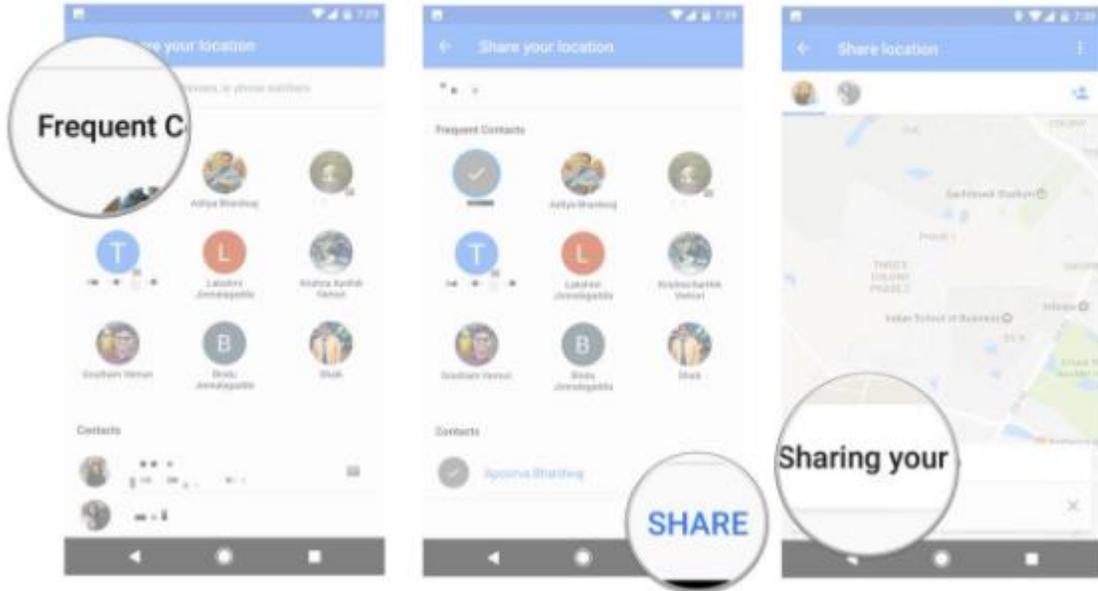
## Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products

US9408055B2	HTC
	<h3>Share your E.T.A</h3> <p>After you start your drive, you can share your destination, estimated arrival time, and where you are on the route.</p> <ol style="list-style-type: none"><li>1. Open the Google Maps app .</li><li>2. Set a driving destination. <a href="#">Learn how to navigate to a place.</a></li><li>3. After you start navigation, tap More ^ &gt; <b>Share trip progress.</b></li><li>4. Choose a person from the list.</li><li>5. Tap <b>Share.</b></li><li>6. Location Sharing will stop when you reach your destination or stop navigating.</li></ol> <ul style="list-style-type: none"><li>• To stop sharing before you arrive, tap More ^ &gt; <b>Stop sharing.</b></li></ul> <h3>See where someone is</h3> <p>If someone shares their location with you, you can see them on the map.</p> <ol style="list-style-type: none"><li>1. Open the Google Maps app .</li><li>2. Tap Menu ≡ &gt; <b>Location sharing.</b></li><li>3. Choose someone.</li></ol> <ul style="list-style-type: none"><li>• To see an updated location, tap on a friend's icon &gt; More ⋮ &gt; <b>Refresh.</b></li></ul> <h3>Stop seeing someone's location</h3> <ol style="list-style-type: none"><li>1. Open the Google Maps app .</li><li>2. On the map, tap their icon.</li><li>3. At the bottom, tap More ^ .</li><li>4. To temporarily hide someone, tap <b>Hide from map.</b> You can stop hiding them at any time.</li></ol> <p><b>Note:</b> You can stop someone's location from ever appearing on your map. Learn how to <a href="#">block another person's account.</a></p> <p><a href="https://support.google.com/maps/answer/7326816?co=GENIE.Platform%3DAndroid&amp;oco=1">https://support.google.com/maps/answer/7326816?co=GENIE.Platform%3DAndroid&amp;oco=1</a></p>

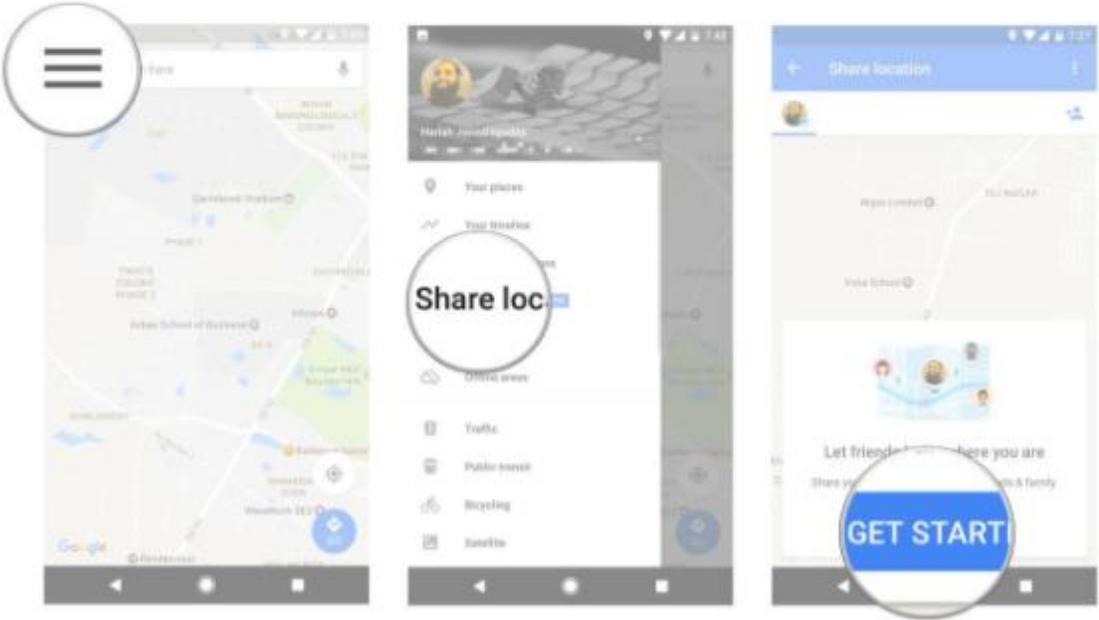
**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC
	<p data-bbox="520 237 1125 269"><b>How to share your location in Google Maps</b></p> <ol data-bbox="520 297 1108 383" style="list-style-type: none"> <li>1. Open Google Maps from the app drawer or the home screen.</li> <li>2. Tap the <b>hamburger menu</b> (the three horizontal lines) on the top left corner of the screen.</li> <li>3. Select <b>Share location</b>.</li> </ol>  <ol data-bbox="520 805 1140 907" style="list-style-type: none"> <li>4. Tap <b>Get Started</b>.</li> <li>5. Use the <b>+</b> icon to select a time period or select the <b>Until you turn this off</b> setting to share your location indefinitely.</li> <li>6. Tap <b>Select People</b>.</li> </ol>  <p data-bbox="512 1295 1356 1326"><a href="https://www.androidcentral.com/how-share-location-google-maps">https://www.androidcentral.com/how-share-location-google-maps</a></p>

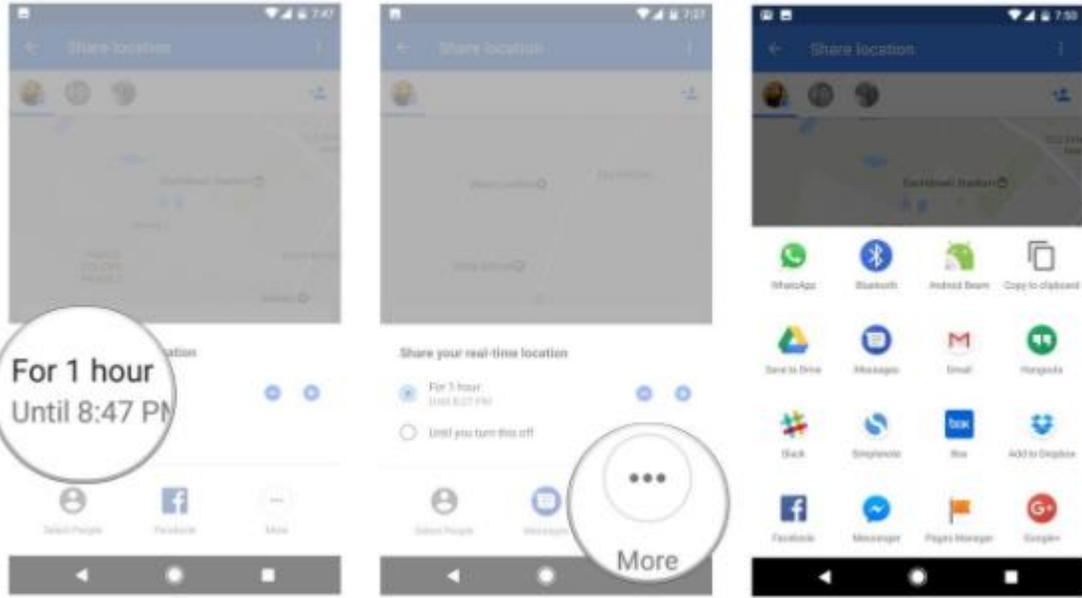
**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC
	<p data-bbox="527 250 1577 310">7. You'll see a list of your frequent contacts at the top, along with a full list of contacts. Pick the contacts by tapping their name.</p> <p data-bbox="527 337 1457 367">8. Once you've selected the contacts you want to share your location to, tap <b>Share</b>.</p> <p data-bbox="527 394 1419 423">9. You'll see a message saying that the selected contact can view your location.</p>  <p data-bbox="506 1101 1356 1130"><a href="https://www.androidcentral.com/how-share-location-google-maps">https://www.androidcentral.com/how-share-location-google-maps</a></p>

# Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products

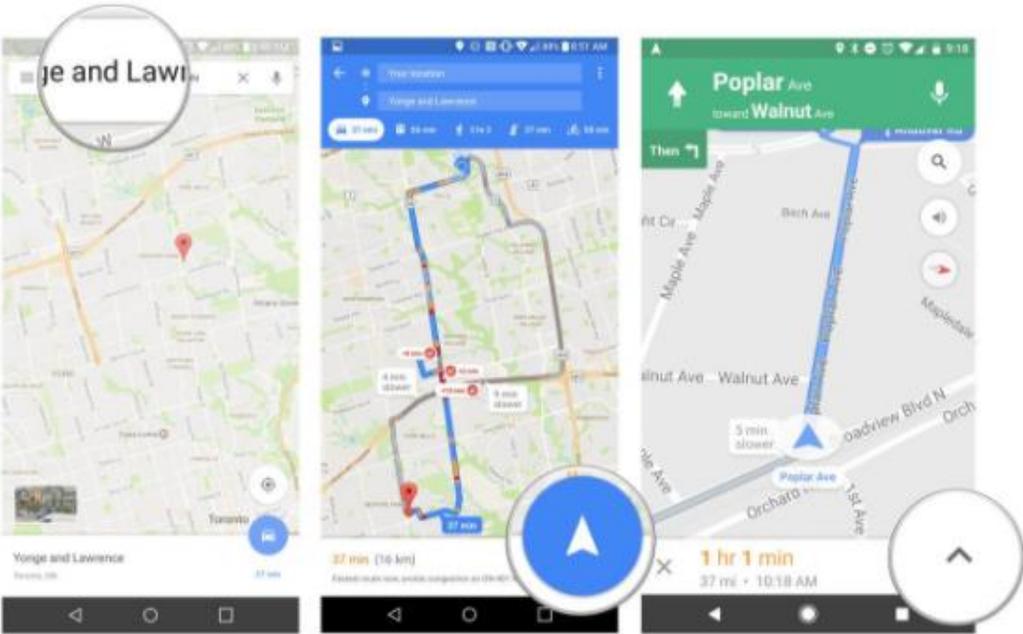
US9408055B2	HTC
	<h2 data-bbox="520 245 1255 289">How to create a shareable link</h2> <p data-bbox="520 331 1461 358">You can also create a link and use it to share your location easily. Here's how to do it:</p> <ol data-bbox="520 410 1234 548" style="list-style-type: none"><li data-bbox="520 410 1234 438">1. Tap the hamburger menu on the top left corner of the screen.</li><li data-bbox="520 467 800 495">2. Select Share location.</li><li data-bbox="520 524 737 552">3. Tap Get Started.</li></ol>  <p data-bbox="510 1230 1356 1263"><a href="https://www.androidcentral.com/how-share-location-google-maps">https://www.androidcentral.com/how-share-location-google-maps</a></p>

**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

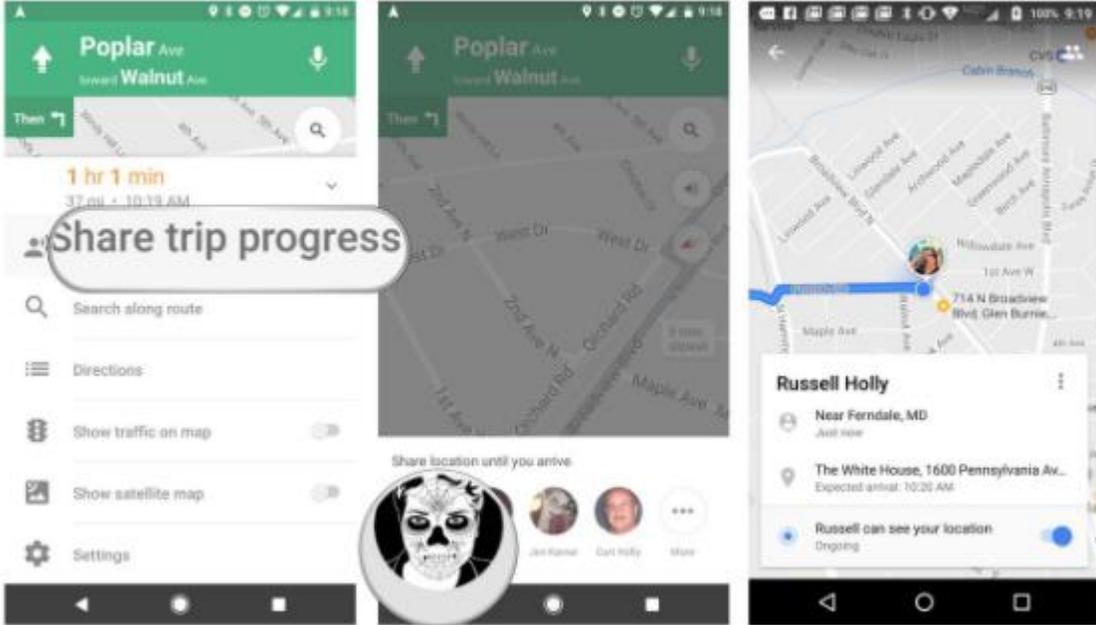
US9408055B2	HTC
	<p data-bbox="520 245 1213 272">4. Select the <b>amount of time</b> you want to share your location.</p> <p data-bbox="520 302 680 329">5. Tap <b>More</b>.</p> <p data-bbox="520 358 1633 418">6. Select <b>your app of choice</b> to create and send a <b>unique URL</b> that broadcasts your current location. You can email it, send the link via Messenger, or even tweet it to the intended recipient.</p>  <p data-bbox="506 1081 1356 1117"><a href="https://www.androidcentral.com/how-share-location-google-maps">https://www.androidcentral.com/how-share-location-google-maps</a></p>



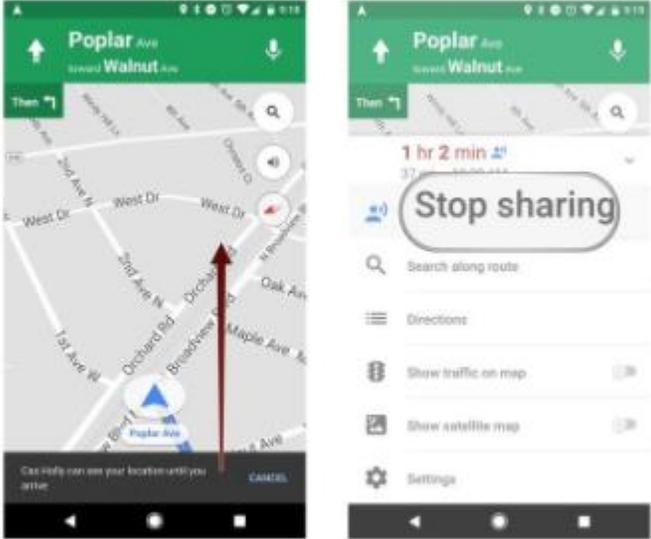
## Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products

US9408055B2	HTC
	<h3 data-bbox="527 240 1428 337">How to share your navigation directions while you walk, drive or transit</h3> <p data-bbox="527 375 1556 464">One of the best ways to share your location with someone is during a drive, walk or transit. If you're meeting a friend or family member somewhere, or navigating towards their home, Google Maps lets you share your location with them for the duration of the trip. It's magic!</p> <ol data-bbox="527 513 1396 643" style="list-style-type: none"><li>1. In the <b>search bar</b> enter your destination.</li><li>2. Pick your navigation type (drive, transit, walk) and press the blue <b>navigate</b> button.</li><li>3. Tap the arrow next to the <b>time-to-destination</b> number at the bottom of the screen.</li></ol>  <p data-bbox="512 1328 1356 1356"><a href="https://www.androidcentral.com/how-share-location-google-maps">https://www.androidcentral.com/how-share-location-google-maps</a></p>

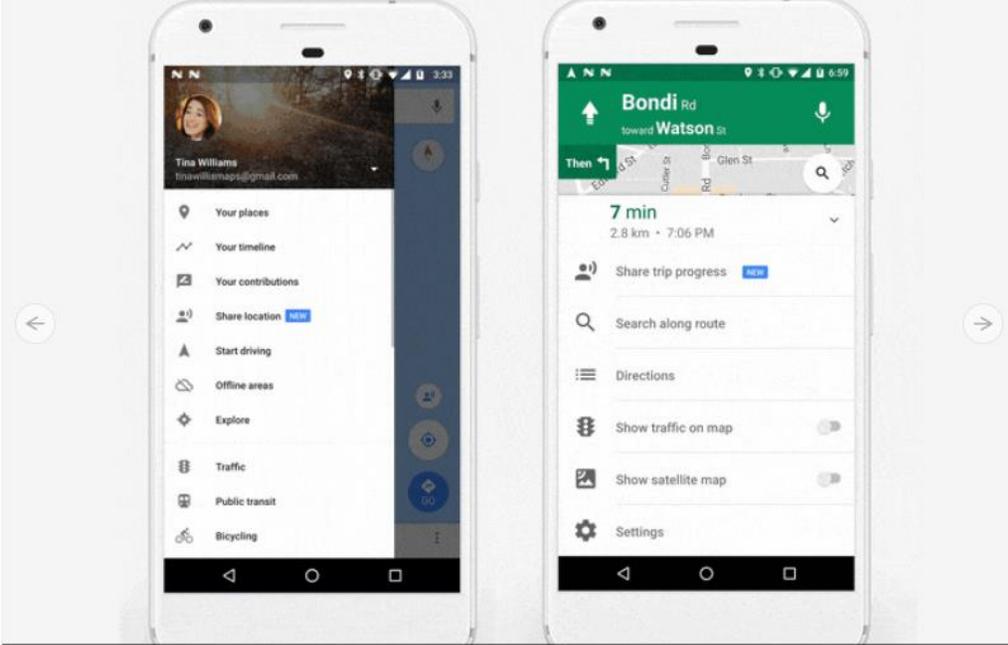
**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC
	<p data-bbox="527 240 835 267">4. Tap Share trip progress.</p> <p data-bbox="527 297 1150 324">5. Choose one or more contacts to share trip progress.</p>  <p data-bbox="527 1027 1360 1097">You can also stop sharing your location with someone before a trip ends. <a href="https://www.androidcentral.com/how-share-location-google-maps">https://www.androidcentral.com/how-share-location-google-maps</a></p>

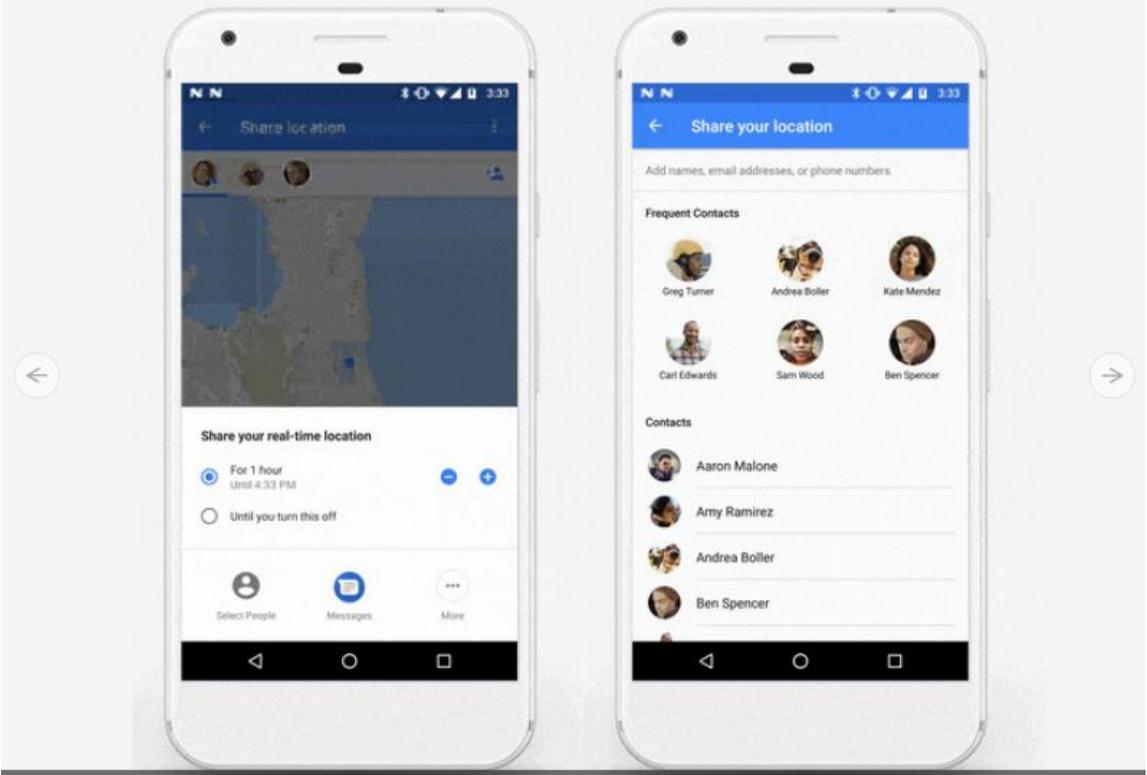
**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC
	<ol style="list-style-type: none"><li data-bbox="533 245 1465 272">1. Tap the arrow next to the <b>time-to-destination</b> number at the bottom of the screen.</li><li data-bbox="533 302 768 329">2. Tap Stop sharing.</li></ol> <div data-bbox="747 383 1398 922"></div> <p data-bbox="541 976 638 1003">That's it!</p> <p data-bbox="541 1045 1608 1073">Are you excited that location sharing is back in Google Maps? How often do you use the feature?</p> <p data-bbox="512 1084 1356 1112"><a href="https://www.androidcentral.com/how-share-location-google-maps">https://www.androidcentral.com/how-share-location-google-maps</a></p>

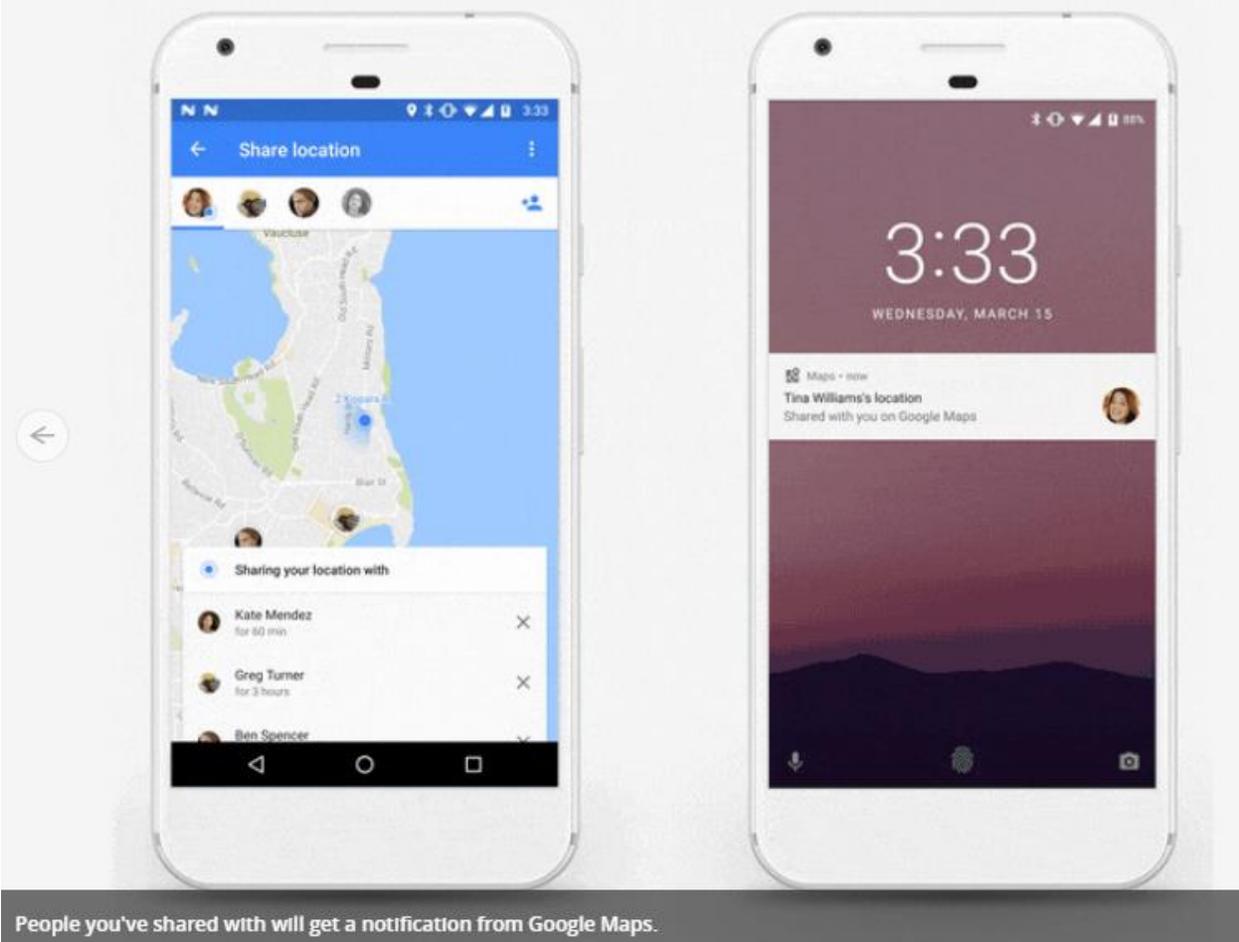
**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC
	 <p data-bbox="514 893 1522 950">Ways to start location sharing: (left) from the app navigation drawer, and (right) from the menu, you can drag up while navigating somewhere.</p> <p data-bbox="514 958 1522 990"><a href="https://arstechnica.com/gadgets/2017/03/location-sharing-finally-returns-to-google-maps/">https://arstechnica.com/gadgets/2017/03/location-sharing-finally-returns-to-google-maps/</a></p>

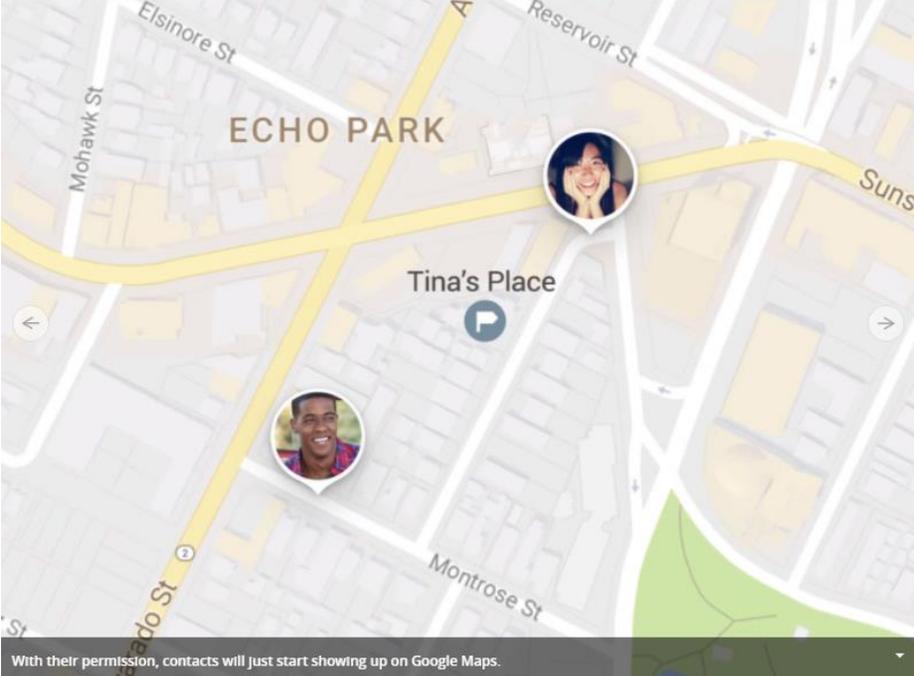
**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC
	 <p data-bbox="514 1023 1661 1063">You can pick how long you want to share your location for, and with whom you want to share it.</p> <p data-bbox="514 1068 1661 1096"><a href="https://arstechnica.com/gadgets/2017/03/location-sharing-finally-returns-to-google-maps/">https://arstechnica.com/gadgets/2017/03/location-sharing-finally-returns-to-google-maps/</a></p>

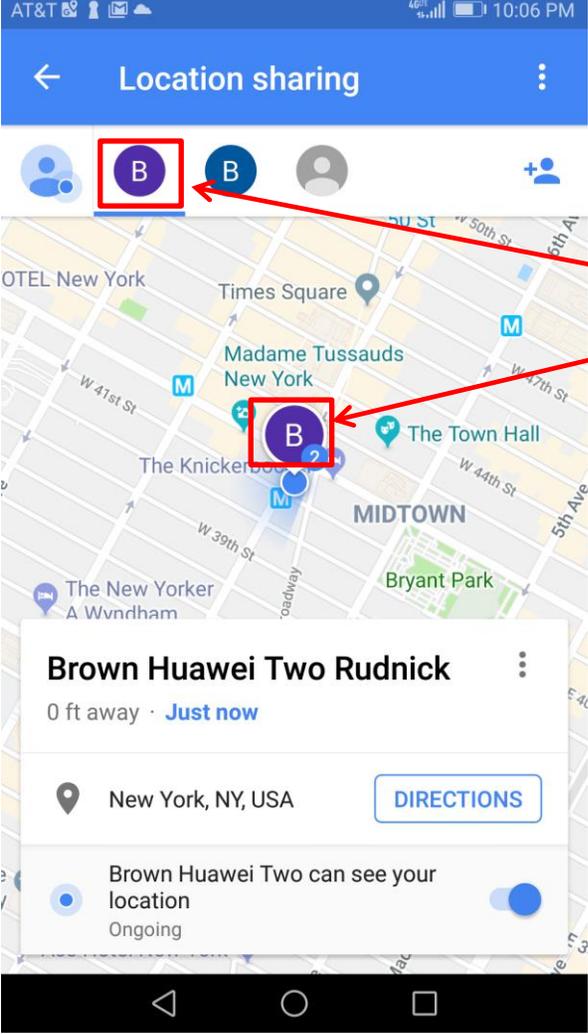
**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC
	 <p data-bbox="514 1144 1176 1177">People you've shared with will get a notification from Google Maps.</p> <p data-bbox="514 1185 1659 1226"><a href="https://arstechnica.com/gadgets/2017/03/location-sharing-finally-returns-to-google-maps/">https://arstechnica.com/gadgets/2017/03/location-sharing-finally-returns-to-google-maps/</a></p>

**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC
	 <p data-bbox="514 885 1018 901">With their permission, contacts will just start showing up on Google Maps.</p> <p data-bbox="514 917 1659 950"><a href="https://arstechnica.com/gadgets/2017/03/location-sharing-finally-returns-to-google-maps/">https://arstechnica.com/gadgets/2017/03/location-sharing-finally-returns-to-google-maps/</a></p> <p data-bbox="514 982 1018 1015"><b><u>Exemplary Google Maps Screenshots</u></b></p>

**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC
	 <p data-bbox="1323 470 1648 641">Exemplary User Selectable Symbols</p> <p data-bbox="514 1347 850 1380"><b><u>Exemplary Source Code:</u></b></p> <p data-bbox="514 1388 1879 1421">The above functionality is performed at least in part by the following publicly available source code and/or</p>



**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC
	<p>source code that invokes or is invoked by the following source code (or a substantially similar copy compiled and loaded onto the Accused Products by HTC). AGIS reserves the right to supplement these contentions with additional source code as discovery progresses and as additional source code is made available.</p> <pre data-bbox="533 428 1738 472">public static LocationRequest create ()</pre> <p>Create a location request with default parameters.</p> <p>Default parameters are for a block accuracy, slowly updated location. It can then be adjusted as required by the applications before passing to the <code>FusedLocationProviderApi</code>.</p> <p><b>Returns</b></p> <ul data-bbox="554 683 814 711" style="list-style-type: none"><li>• a new location request</li></ul> <p><a data-bbox="512 724 1793 753" href="https://developers.google.com/android/reference/com/google/android/gms/location/LocationRequest">https://developers.google.com/android/reference/com/google/android/gms/location/LocationRequest</a></p>

**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC
	<p><code>public static final int PRIORITY_BALANCED_POWER_ACCURACY</code></p> <p>Used with <code>setPriority(int)</code> to request "block" level accuracy.</p> <p>Block level accuracy is considered to be about 100 meter accuracy. Using a coarse accuracy such as this often consumes less power.</p> <p>Constant Value: 102</p> <p><code>public static final int PRIORITY_HIGH_ACCURACY</code></p> <p>Used with <code>setPriority(int)</code> to request the most accurate locations available.</p> <p>This will return the finest location available.</p> <p>Constant Value: 100</p> <p><code>public static final int PRIORITY_LOW_POWER</code></p> <p>Used with <code>setPriority(int)</code> to request "city" level accuracy.</p> <p>City level accuracy is considered to be about 10km accuracy. Using a coarse accuracy such as this often consumes less power.</p> <p>Constant Value: 104</p> <p><a href="https://developers.google.com/android/reference/com/google/android/gms/location/LocationRequest">https://developers.google.com/android/reference/com/google/android/gms/location/LocationRequest</a></p>

**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC
	<p data-bbox="533 248 1749 285"><code>public Task&lt;Location&gt; getLastLocation ()</code></p> <p data-bbox="533 313 1104 337">Returns the best most recent location currently available.</p> <p data-bbox="533 370 1696 430">If a location is not available, which should happen very rarely, null will be returned. The best accuracy available while respecting the location permissions will be returned.</p> <p data-bbox="533 462 1738 522">This method provides a simplified way to get location. It is particularly well suited for applications that do not require an accurate location and that do not want to maintain extra logic for location updates.</p> <p data-bbox="533 576 1749 613"><code>public Task&lt;LocationAvailability&gt; getLocationAvailability ()</code></p> <p data-bbox="533 646 1692 706">Returns the availability of location data. When <code>isLocationAvailable()</code> returns true, then the location returned by <code>getLastLocation()</code> will be reasonably up to date within the hints specified by the active <code>LocationRequest</code>s.</p> <p data-bbox="533 738 1472 763">If the client isn't connected to Google Play services and the request times out, null is returned.</p> <p data-bbox="533 795 1675 855">Note it's always possible for <code>getLastLocation()</code> to return null even when this method returns true (e.g. location settings were disabled between calls).</p> <p data-bbox="512 863 1898 930"><a href="https://developers.google.com/android/reference/com/google/android/gms/location/FusedLocationProviderClient">https://developers.google.com/android/reference/com/google/android/gms/location/FusedLocationProviderClient</a></p>

**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC						
	<p><code>public Task&lt;Void&gt; requestLocationUpdates (LocationRequest request, LocationCallback callback, Looper looper)</code></p> <p>Requests location updates with a callback on the specified Looper thread.</p> <p>This method is suited for the foreground use cases. For background use cases, the <code>PendingIntent</code> version of the method is recommended, see <code>requestLocationUpdates(LocationRequest, PendingIntent)</code>.</p> <p>Any previous LocationRequests registered on this LocationListener will be replaced.</p> <p>This call will keep the Google Play services connection active, so make sure to call <code>removeLocationUpdates(LocationCallback)</code> when you no longer need it, otherwise you lose the benefits of the automatic connection management.</p> <p>Callbacks for <code>LocationCallback</code> will be made on the specified thread, which must already be a prepared looper thread.</p> <p><b>Parameters</b></p> <table border="1"> <tr> <td data-bbox="520 792 625 857"><b>request</b></td> <td data-bbox="632 792 1745 857">The location request for the updates.</td> </tr> <tr> <td data-bbox="520 862 625 927"><b>callback</b></td> <td data-bbox="632 862 1745 927">The callback for the location updates.</td> </tr> <tr> <td data-bbox="520 932 625 997"><b>looper</b></td> <td data-bbox="632 932 1745 997">The Looper object whose message queue will be used to implement the callback mechanism, or null to make callbacks on the calling thread.</td> </tr> </table> <p><a href="https://developers.google.com/android/reference/com/google/android/gms/location/FusedLocationProviderClient">https://developers.google.com/android/reference/com/google/android/gms/location/FusedLocationProviderClient</a></p>	<b>request</b>	The location request for the updates.	<b>callback</b>	The callback for the location updates.	<b>looper</b>	The Looper object whose message queue will be used to implement the callback mechanism, or null to make callbacks on the calling thread.
<b>request</b>	The location request for the updates.						
<b>callback</b>	The callback for the location updates.						
<b>looper</b>	The Looper object whose message queue will be used to implement the callback mechanism, or null to make callbacks on the calling thread.						

**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC				
	<pre data-bbox="533 240 1740 321">public Task&lt;Void&gt; requestLocationUpdates (LocationRequest request, PendingIntent callbackIntent)</pre> <p data-bbox="525 354 1268 378">Requests location updates with a callback on the specified PendingIntent.</p> <p data-bbox="525 410 1732 537">This method is suited for the background use cases, more specifically for receiving location updates, even when the app has been killed by the system. In order to do so, use a <code>PendingIntent</code> for a started service. For foreground use cases, the <code>LocationCallback</code> version of the method is recommended, see <code>requestLocationUpdates(LocationRequest, LocationCallback, Looper)</code>.</p> <p data-bbox="525 570 1724 630">Any previously registered requests that have the same PendingIntent (as defined by <code>equals(Object)</code>) will be replaced by this request.</p> <p data-bbox="525 662 1728 755">Both <code>LocationResult</code> and <code>LocationAvailability</code> are sent to the given PendingIntent. You can extract data from an Intent using <code>hasResult(Intent)</code>, <code>extractResult(Intent)</code>, <code>hasLocationAvailability(Intent)</code>, and <code>extractLocationAvailability(Intent)</code>.</p> <p data-bbox="543 781 667 805"><b>Parameters</b></p> <table border="1" data-bbox="525 833 1740 971"> <tbody> <tr> <td data-bbox="525 833 835 902"><code>request</code></td> <td data-bbox="835 833 1740 902">The location request for the updates.</td> </tr> <tr> <td data-bbox="525 902 835 971"><code>callbackIntent</code></td> <td data-bbox="835 902 1740 971">A pending intent to be sent for each location update.</td> </tr> </tbody> </table> <p data-bbox="543 997 630 1021"><b>Returns</b></p> <ul data-bbox="552 1044 1358 1068" style="list-style-type: none"> <li>• a Task for the call, check <code>isSuccessful()</code> to determine if it was successful.</li> </ul> <p data-bbox="512 1079 1898 1144"><a href="https://developers.google.com/android/reference/com/google/android/gms/location/FusedLocationProviderClient">https://developers.google.com/android/reference/com/google/android/gms/location/FusedLocationProviderClient</a></p>	<code>request</code>	The location request for the updates.	<code>callbackIntent</code>	A pending intent to be sent for each location update.
<code>request</code>	The location request for the updates.				
<code>callbackIntent</code>	A pending intent to be sent for each location update.				

**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC						
	<p data-bbox="533 245 1738 277"><code>public void onLocationAvailability (LocationAvailability locationAvailability)</code></p> <p data-bbox="533 310 1171 334">Called when there is a change in the availability of location data.</p> <p data-bbox="533 367 1738 561">When <code>isLocationAvailable()</code> returns <code>false</code> you can assume that location will not be returned in <code>onLocationResult(LocationResult)</code> until something changes in the device's settings or environment. Even when <code>isLocationAvailable()</code> returns <code>true</code> the <code>onLocationResult(LocationResult)</code> may not always be called regularly, however the device location is known and both the most recently delivered location and <code>getLastLocation(GoogleApiClient)</code> will be reasonably up to date given the hints specified by the active <code>LocationRequest</code>s.</p> <p data-bbox="533 586 667 610"><b>Parameters</b></p> <table border="1" data-bbox="533 643 1738 708"> <tr> <td data-bbox="533 651 961 699"><code>locationAvailability</code></td> <td data-bbox="961 651 1738 699">The current status of location availability.</td> </tr> </table> <p data-bbox="533 756 1738 789"><code>public void onLocationResult (LocationResult result)</code></p> <p data-bbox="533 821 1052 846">Called when device location information is available.</p> <p data-bbox="533 878 1661 943">The most recent location returned by <code>getLastLocation()</code> is not guaranteed to be immediately fresh, but will be reasonably up to date given the hints specified by the active <code>LocationRequest</code>s.</p> <p data-bbox="533 967 667 992"><b>Parameters</b></p> <table border="1" data-bbox="533 1024 1738 1089"> <tr> <td data-bbox="533 1032 768 1081"><code>result</code></td> <td data-bbox="768 1032 1738 1089">The latest location result available.</td> </tr> </table> <p data-bbox="533 1105 1801 1130"><a href="https://developers.google.com/android/reference/com/google/android/gms/location/LocationCallback">https://developers.google.com/android/reference/com/google/android/gms/location/LocationCallback</a></p> <p data-bbox="533 1146 1738 1179"><code>public abstract void onLocationChanged (Location location)</code></p> <p data-bbox="533 1211 915 1235">Called when the location has changed.</p> <p data-bbox="533 1260 667 1284"><b>Parameters</b></p> <table border="1" data-bbox="533 1317 1738 1382"> <tr> <td data-bbox="533 1325 926 1373"><code>location</code></td> <td data-bbox="926 1325 1738 1382">The updated location.</td> </tr> </table>	<code>locationAvailability</code>	The current status of location availability.	<code>result</code>	The latest location result available.	<code>location</code>	The updated location.
<code>locationAvailability</code>	The current status of location availability.						
<code>result</code>	The latest location result available.						
<code>location</code>	The updated location.						

**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC
	<p data-bbox="508 235 1797 264"><a href="https://developers.google.com/android/reference/com/google/android/gms/location/LocationListener">https://developers.google.com/android/reference/com/google/android/gms/location/LocationListener</a></p> <p data-bbox="508 316 814 345">Public Constructors</p> <hr data-bbox="508 358 1740 362"/> <p data-bbox="508 410 1740 440">public <b>MapView</b> (<a href="#">Context</a> context)</p> <p data-bbox="508 500 1740 529">public <b>MapView</b> (<a href="#">Context</a> context, <a href="#">AttributeSet</a> attrs)</p> <p data-bbox="508 589 1740 618">public <b>MapView</b> (<a href="#">Context</a> context, <a href="#">AttributeSet</a> attrs, int defStyle)</p> <p data-bbox="508 678 1740 708">public <b>MapView</b> (<a href="#">Context</a> context, <a href="#">GoogleMapOptions</a> options)</p> <p data-bbox="508 748 1671 777"><a href="https://developers.google.com/android/reference/com/google/android/gms/maps/MapView">https://developers.google.com/android/reference/com/google/android/gms/maps/MapView</a></p>

**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

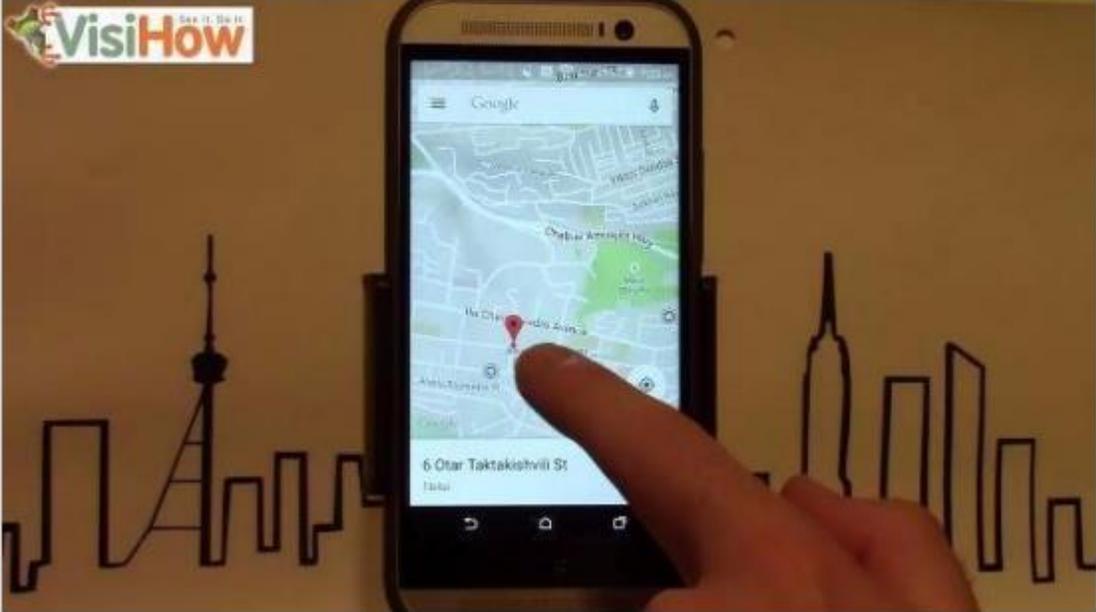
US9408055B2	HTC		
	<p><code>public void <b>getMapAsync</b> (<a href="#">OnMapReadyCallback</a> callback)</code></p> <p>Returns a non-null instance of the <a href="#">GoogleMap</a>, ready to be used.</p> <p>Note that:</p> <ul style="list-style-type: none"> <li>• This method must be called from the main thread.</li> <li>• The callback will be executed in the main thread.</li> <li>• In the case where Google Play services is not installed on the user's device, the callback will not be triggered until the user installs it.</li> <li>• The <a href="#">GoogleMap</a> object provided by the callback is non-null.</li> </ul> <p><b>Parameters</b></p> <table border="1" data-bbox="527 688 1738 753"> <tr> <td><code>callback</code></td> <td>The callback object that will be triggered when the map is ready to be used.</td> </tr> </table> <p><code>public final void <b>onCreate</b> (<a href="#">Bundle</a> savedInstanceState)</code></p> <p>You must call this method from the parent Activity/Fragment's corresponding method.</p> <p><a href="https://developers.google.com/android/reference/com/google/android/gms/maps/MapView">https://developers.google.com/android/reference/com/google/android/gms/maps/MapView</a></p>	<code>callback</code>	The callback object that will be triggered when the map is ready to be used.
<code>callback</code>	The callback object that will be triggered when the map is ready to be used.		
<p>[41F] identifying user interaction with the interactive display selecting one or more of the user-selectable symbols corresponding to one or more of the second devices and user interaction with the display specifying an</p>	<p>HTC infringes directly and/or indirectly by performing, inducing others to perform, and/or contributing to the performance of identifying user interaction with the interactive display selecting one or more of the user-selectable symbols corresponding to one or more of the second devices and user interaction with the display specifying an action and, based thereon, sending data to the one or more second devices. See claims 1[F] and 28[F], which are incorporated herein by reference in their entirety.</p> <p><b>Regarding Google Maps</b>, Google Latitude, Google Plus, Google Hangouts, Google Messages, and Android Messenger, the Accused Products are configured to allow a user of the first device to interact with the display, to select a user, contact, or device, and to select an action to be performed, such as: sending a message, initiating a call, initiating a data conference, sharing a location, stop sharing a location, block a user from location sharing, sending a location, requesting a location, or sending other data. When an action is</p>		



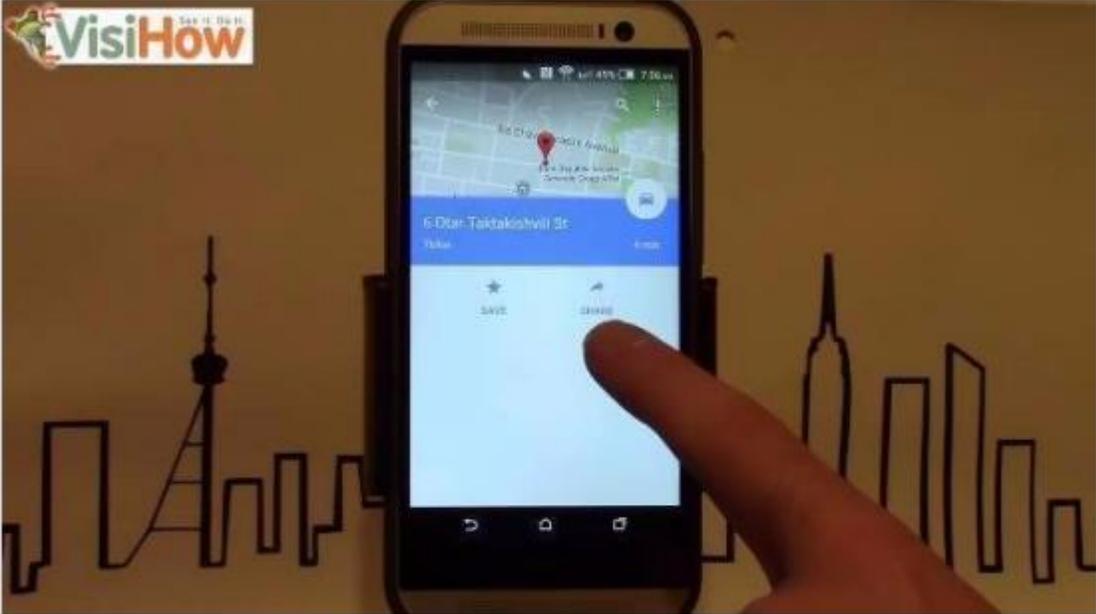
**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC
<p>action and, based thereon, sending data to the one or more second devices;</p>	<p>specified, data is sent from the first device to the second device via a server.</p> <p><b><u>Exemplary Support for Google Maps:</u></b></p> <p>Using Google Maps, a user may choose a symbol and send data to that device. For example, a user who is already sharing her location with another user can stop sharing by making a selection resulting in the second device no longer displaying the first device’s location. Additionally, a user can share an ETA message with another user or send another user a link in a message to share her location. Additionally, a user who is sharing a location until she arrives can make a selection to stop her location from showing on the second device.</p> <p><b>Share Locations from Google Maps on HTC One M8</b></p>

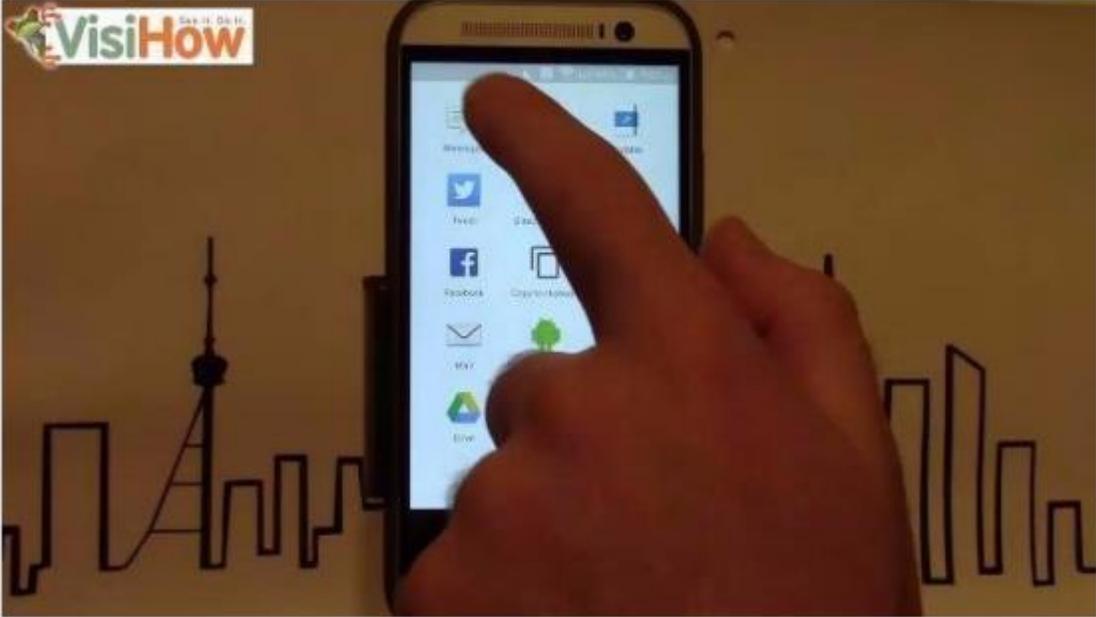
**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC
	<p data-bbox="531 233 827 256"><b>Place a pin on the map.</b></p> <p data-bbox="531 272 1640 380">We can do this anywhere in the entire world, anywhere, on any map that we find in the Google Maps application. We are going to choose a location, press down on the screen, and hold down until we can see that a red pin has been dropped.</p> 

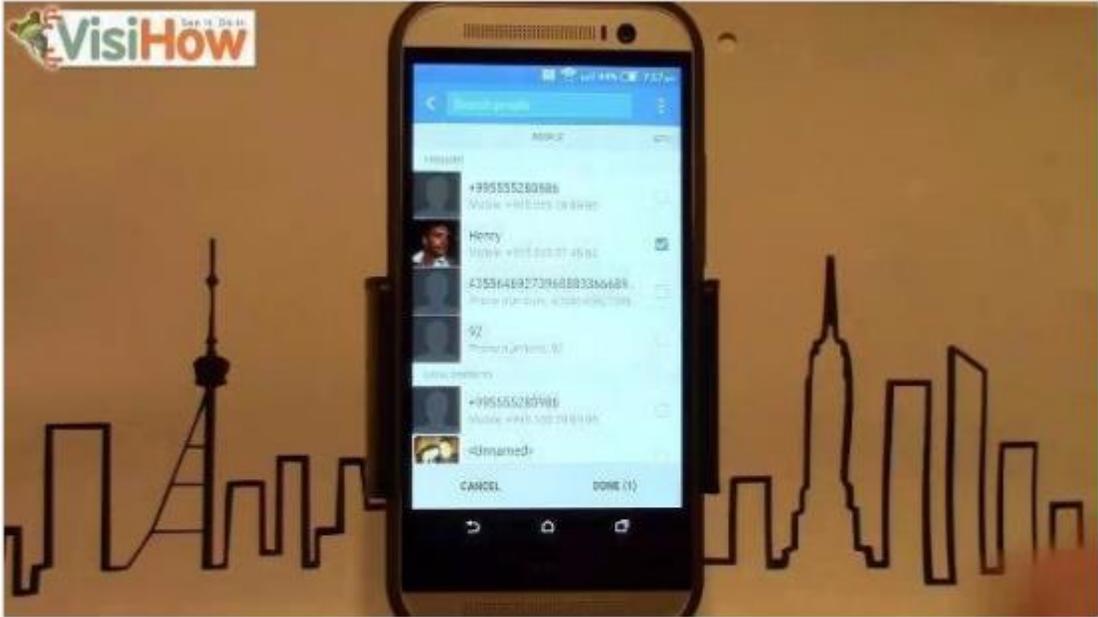
**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC
	<p data-bbox="527 240 1205 305"><b>We have two options:</b> "SAVE" and "SHARE". Press down on the word "SHARE".</p>  <p data-bbox="527 1057 1633 1208"><b>A menu appears with all of the different ways that we can share this location.</b> We can share this location via messages, Bluetooth, Twitter, Facebook, mail, Gmail and Viber. If you have other applications, such as WhatsApp, or any other chat or email applications, you will be able to use them as well and will see them here.</p>

**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC
	<p data-bbox="527 237 856 261"><b>Let's choose "Messages".</b></p> <p data-bbox="527 277 1629 342">This option is located on the top left-hand corner of the screen. All options will have the same basic processes. They will ask us to send the location. Press down on the "Messages" icon.</p> 

**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC
	<p data-bbox="520 228 1260 256"><b>Press the box next to the contact who will be the recipient.</b></p> <p data-bbox="520 264 1549 292">A checkmark will appear in the box. We can select multiple contacts as well if we desire.</p>  <p data-bbox="520 1044 911 1071"><b>Press on the word "DONE (1)".</b></p> <p data-bbox="520 1081 1629 1153">This icon is located in the bottom right-hand corner of the screen. The bracketed number indicates the number of contacts we selected.</p> <p data-bbox="520 1162 1407 1190"><a href="http://visihow.com/Share_Locations_from_Google_Maps_on_HTC_One_M8">http://visihow.com/Share_Locations_from_Google_Maps_on_HTC_One_M8</a></p>

## Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products

US9408055B2	HTC
	<p data-bbox="548 240 968 261">COMPUTER <b>ANDROID</b> IPHONE &amp; IPAD</p> <hr data-bbox="533 289 1566 293"/> <h3 data-bbox="533 347 1024 383">If they have a Google Account</h3> <ol data-bbox="533 406 1419 698" style="list-style-type: none"><li data-bbox="533 406 1220 427">1. If you haven't already, add their Gmail address to your <a href="#">Google Contacts</a>.</li><li data-bbox="533 443 1419 464">2. On your Android phone or tablet, open the Google Maps app  and sign in. <a href="#">Learn how to sign in.</a></li><li data-bbox="533 480 1035 501">3. Tap Menu ≡ &gt; <b>Location sharing</b> &gt; Add People +.</li><li data-bbox="533 518 1005 539">4. Choose how long you want to share your location.</li><li data-bbox="533 555 1140 613">5. Tap <b>Select People</b>.<ul data-bbox="569 589 1140 610" style="list-style-type: none"><li data-bbox="569 589 1140 610">• If you're asked about your contacts, give Google Maps access.</li></ul></li><li data-bbox="533 639 884 660">6. Choose who you want to share with.</li><li data-bbox="533 677 663 698">7. Tap <b>Share</b>.</li></ol> <h3 data-bbox="533 756 1110 792">If they don't have a Google Account</h3> <ol data-bbox="533 815 1560 940" style="list-style-type: none"><li data-bbox="533 815 1419 836">1. On your Android phone or tablet, open the Google Maps app  and sign in. <a href="#">Learn how to sign in.</a></li><li data-bbox="533 852 1035 873">2. Tap Menu ≡ &gt; <b>Location sharing</b> &gt; Add People +.</li><li data-bbox="533 889 1560 940">3. Tap More ... &gt; <b>Copy to clipboard</b>. People with this link can see your location for as long as you choose, up to 72 hours.</li></ol> <h3 data-bbox="533 989 869 1024">Share using another app</h3> <p data-bbox="533 1044 1205 1065">You can also share through messaging apps. Tap More ... &gt; select an app.</p> <h3 data-bbox="533 1123 743 1159">Stop sharing</h3> <ol data-bbox="533 1182 1205 1281" style="list-style-type: none"><li data-bbox="533 1182 842 1203">1. Open the Google Maps app .</li><li data-bbox="533 1219 869 1240">2. Tap Menu ≡ &gt; <b>Location sharing</b>.</li><li data-bbox="533 1256 1205 1278">3. Next to the person with whom you want to stop sharing, tap Remove ✕.</li></ol> <p data-bbox="512 1299 1701 1326"><a href="https://support.google.com/maps/answer/7326816?co=GENIE.Platform%3DAndroid&amp;oco=1">https://support.google.com/maps/answer/7326816?co=GENIE.Platform%3DAndroid&amp;oco=1</a></p>

## Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products

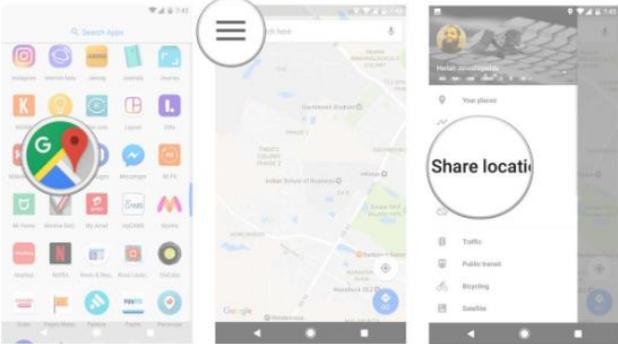
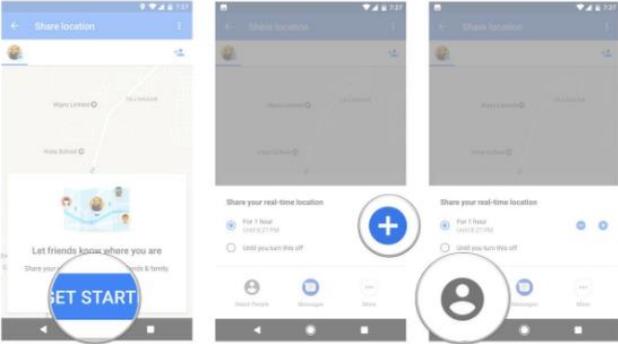
US9408055B2	HTC
	<h3>Share your E.T.A</h3> <p>After you start your drive, you can share your destination, estimated arrival time, and where you are on the route.</p> <ol style="list-style-type: none"><li>1. Open the Google Maps app .</li><li>2. Set a driving destination. <a href="#">Learn how to navigate to a place.</a></li><li>3. After you start navigation, tap More  &gt; <b>Share trip progress.</b></li><li>4. Choose a person from the list.</li><li>5. Tap <b>Share.</b></li><li>6. Location Sharing will stop when you reach your destination or stop navigating.</li></ol> <ul style="list-style-type: none"><li>• To stop sharing before you arrive, tap More  &gt; <b>Stop sharing.</b></li></ul> <h3>See where someone is</h3> <p>If someone shares their location with you, you can see them on the map.</p> <ol style="list-style-type: none"><li>1. Open the Google Maps app .</li><li>2. Tap Menu  &gt; <b>Location sharing.</b></li><li>3. Choose someone.</li></ol> <ul style="list-style-type: none"><li>• To see an updated location, tap on a friend's icon &gt; More  &gt; <b>Refresh.</b></li></ul> <h3>Stop seeing someone's location</h3> <ol style="list-style-type: none"><li>1. Open the Google Maps app .</li><li>2. On the map, tap their icon.</li><li>3. At the bottom, tap More .</li><li>4. To temporarily hide someone, tap <b>Hide from map.</b> You can stop hiding them at any time.</li></ol> <p><b>Note:</b> You can stop someone's location from ever appearing on your map. <a href="#">Learn how to block another person's account.</a></p> <p><a href="https://support.google.com/maps/answer/7326816?co=GENIE.Platform%3DAndroid&amp;oco=1">https://support.google.com/maps/answer/7326816?co=GENIE.Platform%3DAndroid&amp;oco=1</a></p>

## Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products

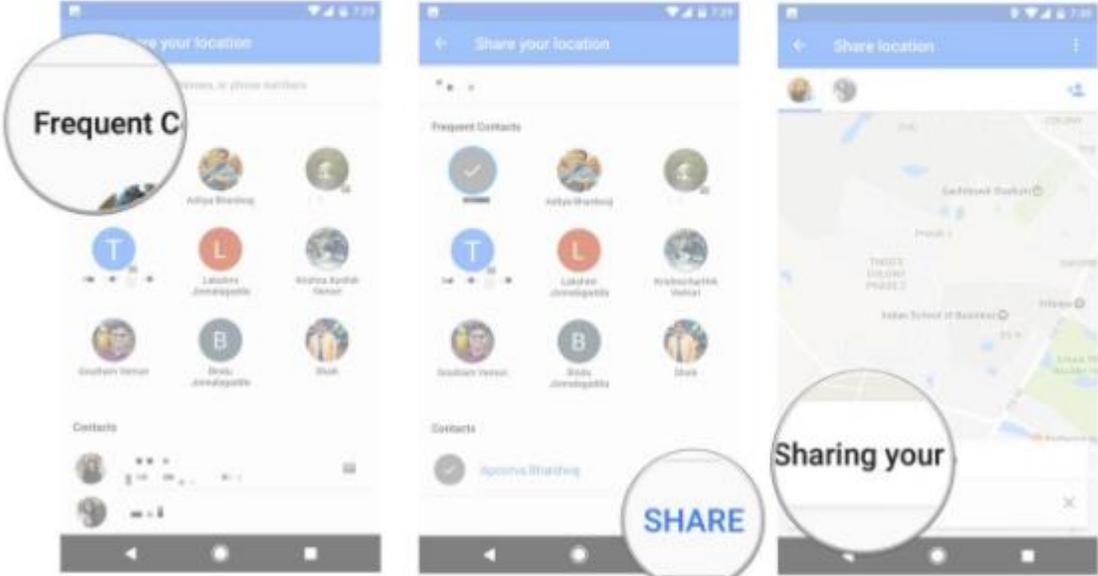
US9408055B2	HTC
	<h3 data-bbox="541 282 877 321">Hide or share lists</h3> <p data-bbox="541 347 909 370"><b>Note:</b> You can't share starred places.</p> <ol data-bbox="554 402 1682 667" style="list-style-type: none"><li data-bbox="554 402 890 425">1. Open the Google Maps app .</li><li data-bbox="554 444 968 467">2. Tap Menu  &gt; <b>Your places</b> &gt; <b>Saved</b>.</li><li data-bbox="554 487 1682 667">3. Next to the list you want to share, tap More  &gt; choose an option:<ul data-bbox="583 526 1682 667" style="list-style-type: none"><li data-bbox="583 526 1444 548">• <b>Hide/Show on your map:</b> Display or hide your saved places when looking at the map.</li><li data-bbox="583 568 1058 591">• <b>Share list:</b> Allow others to see your saved list.</li><li data-bbox="583 610 1682 667">• <b>Sharing options:</b> You can make your list public, private, or shared. To let anyone with the link see your list, tap <b>Shared</b>. To let anyone find and follow your list, tap <b>Public</b>.</li></ul></li></ol> <h3 data-bbox="541 737 764 776">Follow a list</h3> <p data-bbox="541 802 1724 862">If you follow a list made by someone else, their saved places will show up in Your Places. The places will also appear as suggested locations in Google Maps.</p> <h3 data-bbox="541 915 915 954">Follow a list using a link</h3> <ol data-bbox="554 980 1356 1084" style="list-style-type: none"><li data-bbox="554 980 957 1003">1. Tap on the link you received to open it.</li><li data-bbox="554 1023 1272 1045">2. Tap <b>Follow</b>. This list will now be added to the group of lists you follow.</li><li data-bbox="554 1065 1356 1088">3. <b>Optional:</b> To unfollow a list someone shared with you, tap the list &gt; <b>Following</b>.</li></ol> <h3 data-bbox="541 1138 924 1177">See lists made by others</h3> <p data-bbox="541 1203 1335 1226">If a user has any Google Maps lists that were made public, you can follow them.</p> <ol data-bbox="554 1252 1136 1356" style="list-style-type: none"><li data-bbox="554 1252 1136 1274">1. Tap on the name of a user whose list you want to follow.</li><li data-bbox="554 1294 680 1317">2. Tap <b>Lists</b>.</li><li data-bbox="554 1336 1136 1359">3. Tap on the list you want to follow &gt; More  &gt; <b>Follow</b>.</li></ol> <p data-bbox="512 1373 1898 1396"><a href="https://support.google.com/maps/answer/7280933?hl=en&amp;ref_topic=7301134&amp;co=GENIE.Platform%3DAn">https://support.google.com/maps/answer/7280933?hl=en&amp;ref_topic=7301134&amp;co=GENIE.Platform%3DAn</a></p>



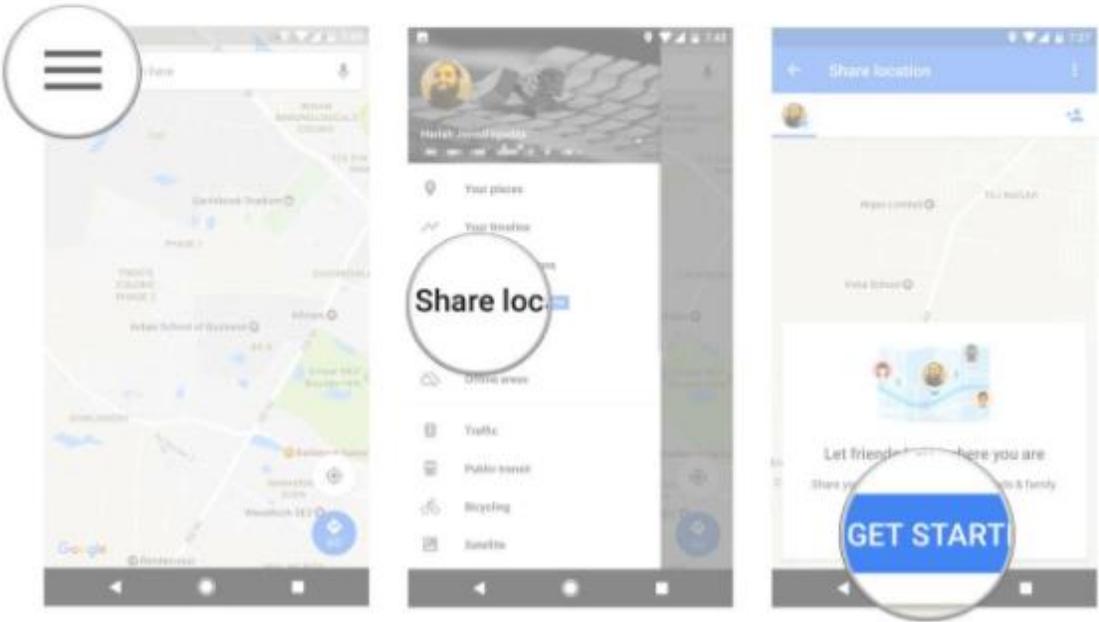
# Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products

US9408055B2	HTC
	<p data-bbox="514 233 682 261">droid&amp;oco=1</p> <h3 data-bbox="514 310 1102 342">How to share your location in Google Maps</h3> <ol data-bbox="514 367 1087 451" style="list-style-type: none"><li>1. Open Google Maps from the app drawer or the home screen.</li><li>2. Tap the <b>hamburger menu</b> (the three horizontal lines) on the top left corner of the screen.</li><li>3. Select <b>Share location</b>.</li></ol>  <ol data-bbox="514 854 1117 954" style="list-style-type: none"><li>4. Tap <b>Get Started</b>.</li><li>5. Use the <b>+</b> icon to select a time period or select the <b>Until you turn this off</b> setting to share your location indefinitely.</li><li>6. Tap <b>Select People</b>.</li></ol>  <p data-bbox="514 1333 1354 1365"><a href="https://www.androidcentral.com/how-share-location-google-maps">https://www.androidcentral.com/how-share-location-google-maps</a></p>

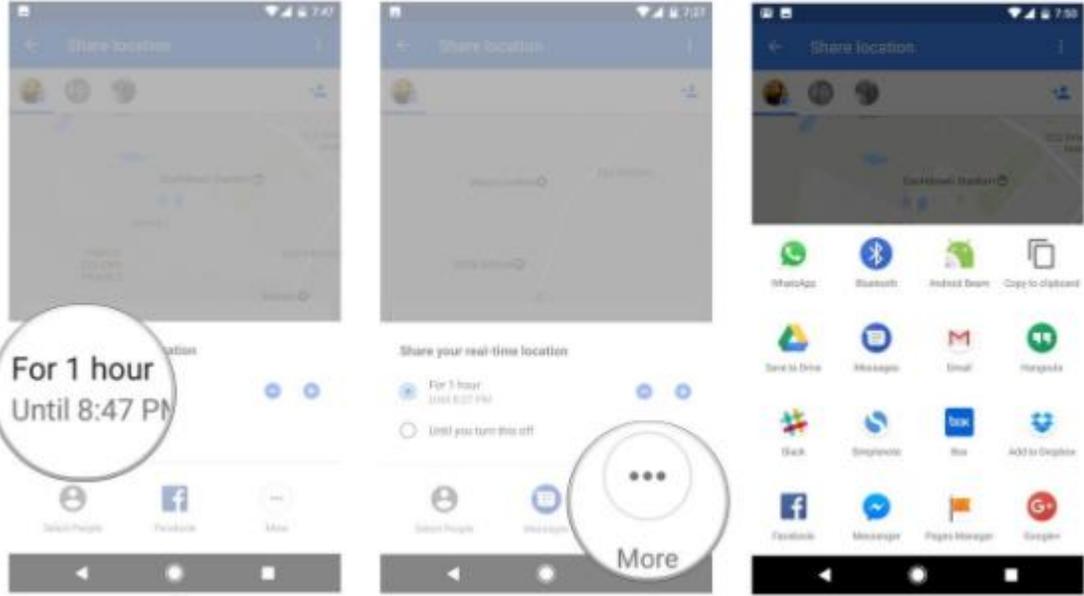
**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC
	<p data-bbox="527 253 1577 427"><b>7.</b> You'll see a list of your frequent contacts at the top, along with a full list of contacts. <b>Pick the contacts</b> by tapping their name.</p> <p data-bbox="527 342 1457 367"><b>8.</b> Once you've selected the contacts you want to share your location to, tap <b>Share</b>.</p> <p data-bbox="527 399 1419 423"><b>9.</b> You'll see a message saying that the selected contact can view your location.</p>  <p data-bbox="510 1101 1356 1133"><a href="https://www.androidcentral.com/how-share-location-google-maps">https://www.androidcentral.com/how-share-location-google-maps</a></p>

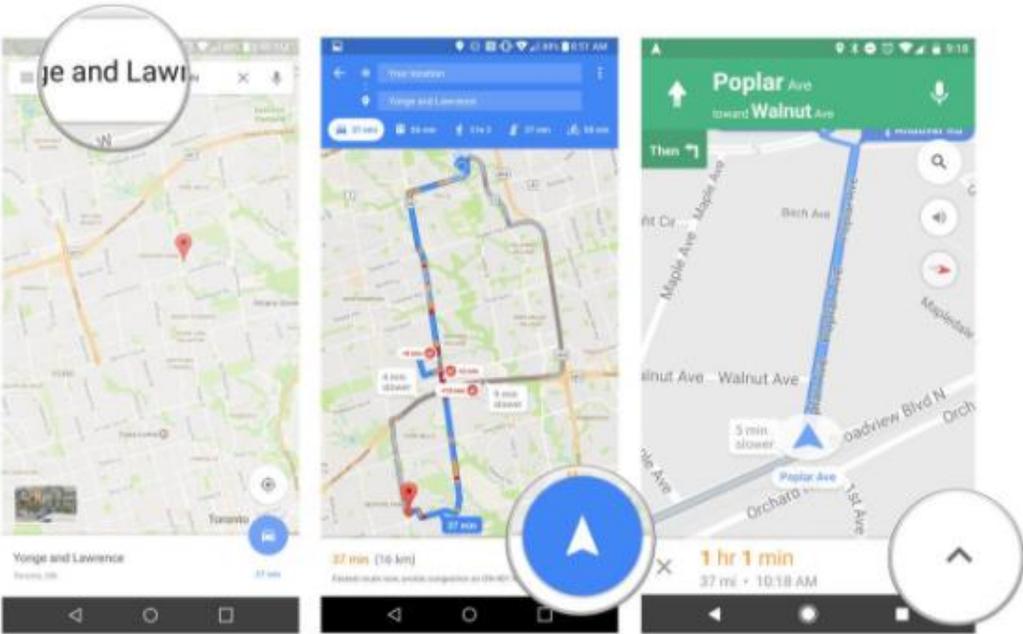
## Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products

US9408055B2	HTC
	<h3 data-bbox="520 240 1255 289">How to create a shareable link</h3> <p data-bbox="520 329 1461 358">You can also create a link and use it to share your location easily. Here's how to do it:</p> <ol data-bbox="520 407 1234 548" style="list-style-type: none"><li data-bbox="520 407 1234 436">1. Tap the hamburger menu on the top left corner of the screen.</li><li data-bbox="520 464 800 493">2. Select Share location.</li><li data-bbox="520 521 737 550">3. Tap Get Started.</li></ol>  <p data-bbox="506 1230 1356 1263"><a href="https://www.androidcentral.com/how-share-location-google-maps">https://www.androidcentral.com/how-share-location-google-maps</a></p>

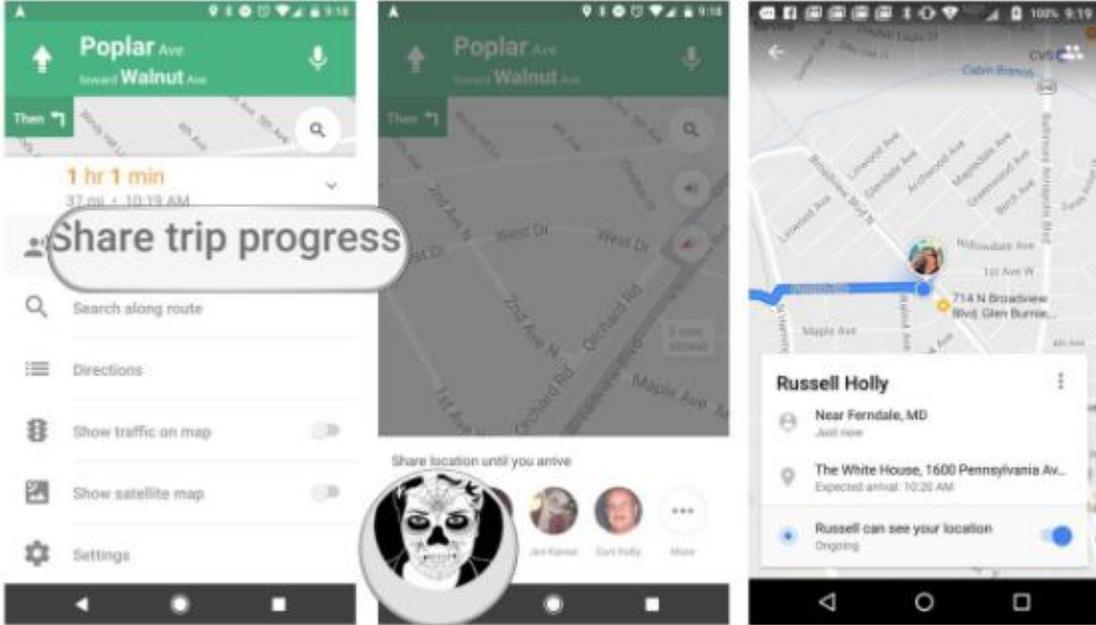
## Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products

US9408055B2	HTC
	<p data-bbox="520 245 1213 272">4. Select the <b>amount of time</b> you want to share your location.</p> <p data-bbox="520 302 680 329">5. Tap <b>More</b>.</p> <p data-bbox="520 358 1633 418">6. Select <b>your app of choice</b> to create and send a <b>unique URL</b> that broadcasts your current location. You can email it, send the link via Messenger, or even tweet it to the intended recipient.</p>  <p data-bbox="506 1084 1360 1117"><a href="https://www.androidcentral.com/how-share-location-google-maps">https://www.androidcentral.com/how-share-location-google-maps</a></p>

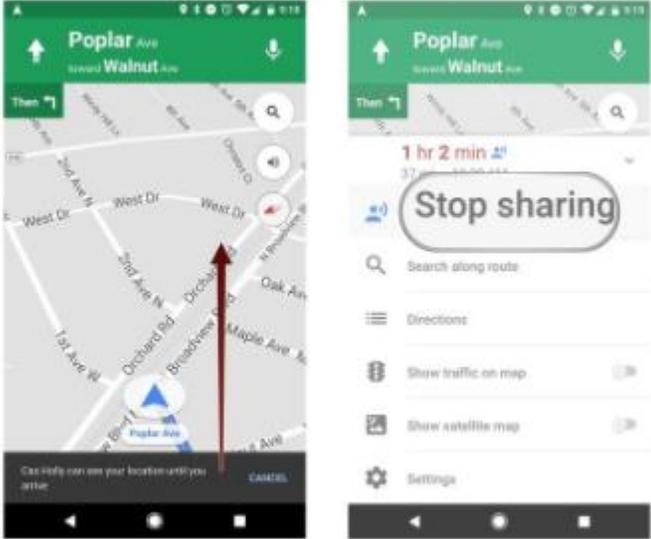
## Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products

US9408055B2	HTC
	<h3 data-bbox="527 240 1428 337">How to share your navigation directions while you walk, drive or transit</h3> <p data-bbox="527 375 1556 464">One of the best ways to share your location with someone is during a drive, walk or transit. If you're meeting a friend or family member somewhere, or navigating towards their home, Google Maps lets you share your location with them for the duration of the trip. It's magic!</p> <ol data-bbox="527 513 1396 643" style="list-style-type: none"><li>1. In the <b>search bar</b> enter your destination.</li><li>2. Pick your navigation type (drive, transit, walk) and press the blue <b>navigate</b> button.</li><li>3. Tap the arrow next to the <b>time-to-destination</b> number at the bottom of the screen.</li></ol>  <p data-bbox="512 1328 1356 1360"><a href="https://www.androidcentral.com/how-share-location-google-maps">https://www.androidcentral.com/how-share-location-google-maps</a></p>

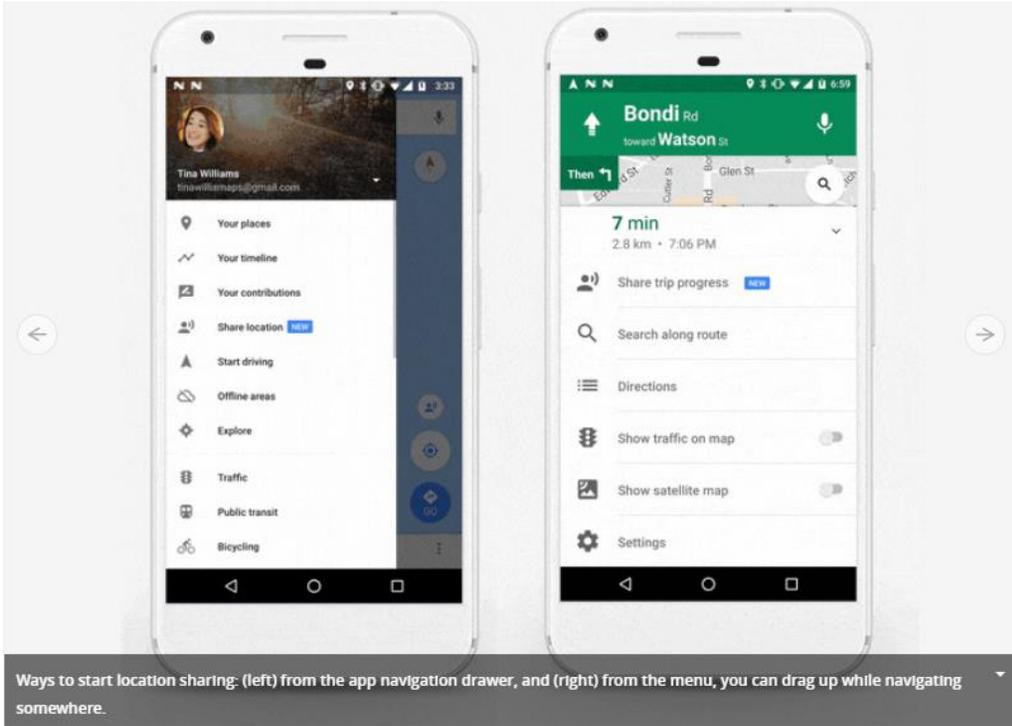
**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC
	<p data-bbox="527 277 835 305">4. Tap Share trip progress.</p> <p data-bbox="527 334 1150 362">5. Choose one or more contacts to share trip progress.</p>  <p data-bbox="527 1065 1360 1133">You can also stop sharing your location with someone before a trip ends. <a href="https://www.androidcentral.com/how-share-location-google-maps">https://www.androidcentral.com/how-share-location-google-maps</a></p>

**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

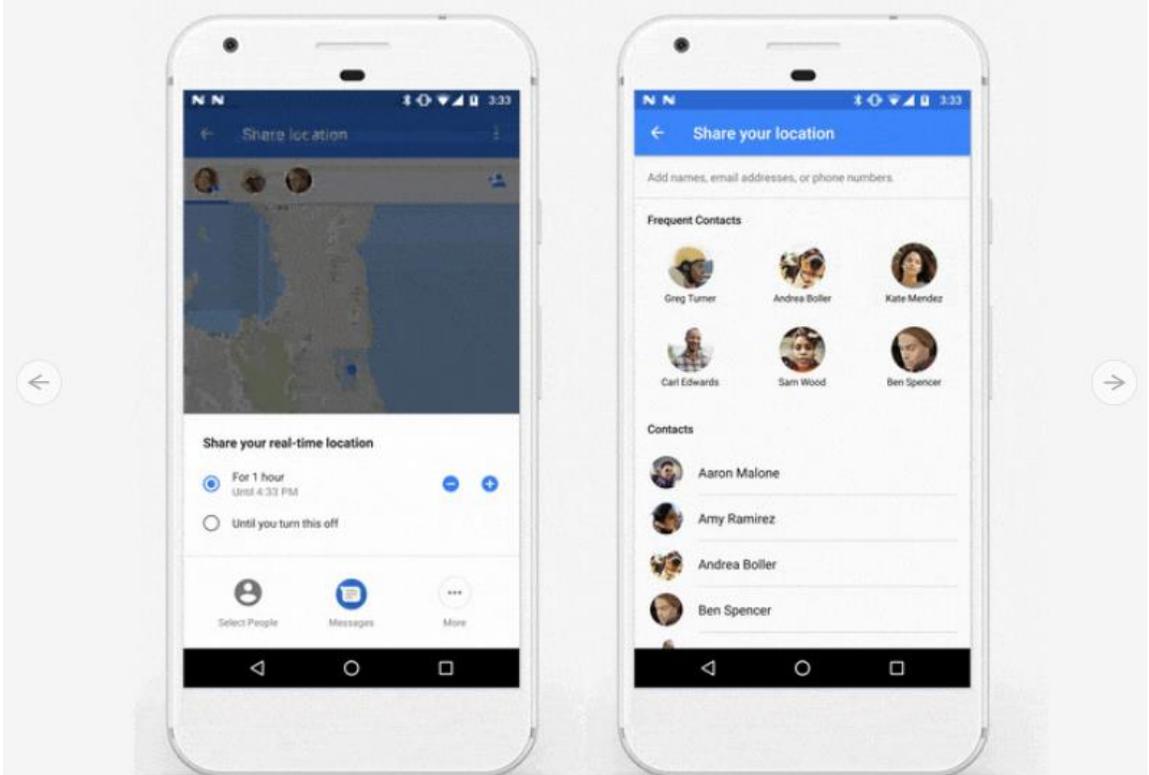
US9408055B2	HTC
	<ol style="list-style-type: none"><li data-bbox="533 245 1465 272">1. Tap the arrow next to the <b>time-to-destination</b> number at the bottom of the screen.</li><li data-bbox="533 302 768 329">2. Tap Stop sharing.</li></ol> <div data-bbox="747 383 1398 922"></div> <p data-bbox="541 976 638 1003">That's it!</p> <p data-bbox="541 1045 1608 1073">Are you excited that location sharing is back in Google Maps? How often do you use the feature?</p> <p data-bbox="512 1084 1356 1112"><a href="https://www.androidcentral.com/how-share-location-google-maps">https://www.androidcentral.com/how-share-location-google-maps</a></p> <p data-bbox="512 1195 1419 1222">As shown below, a group may also be defined within Google Contacts.</p>

# Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products

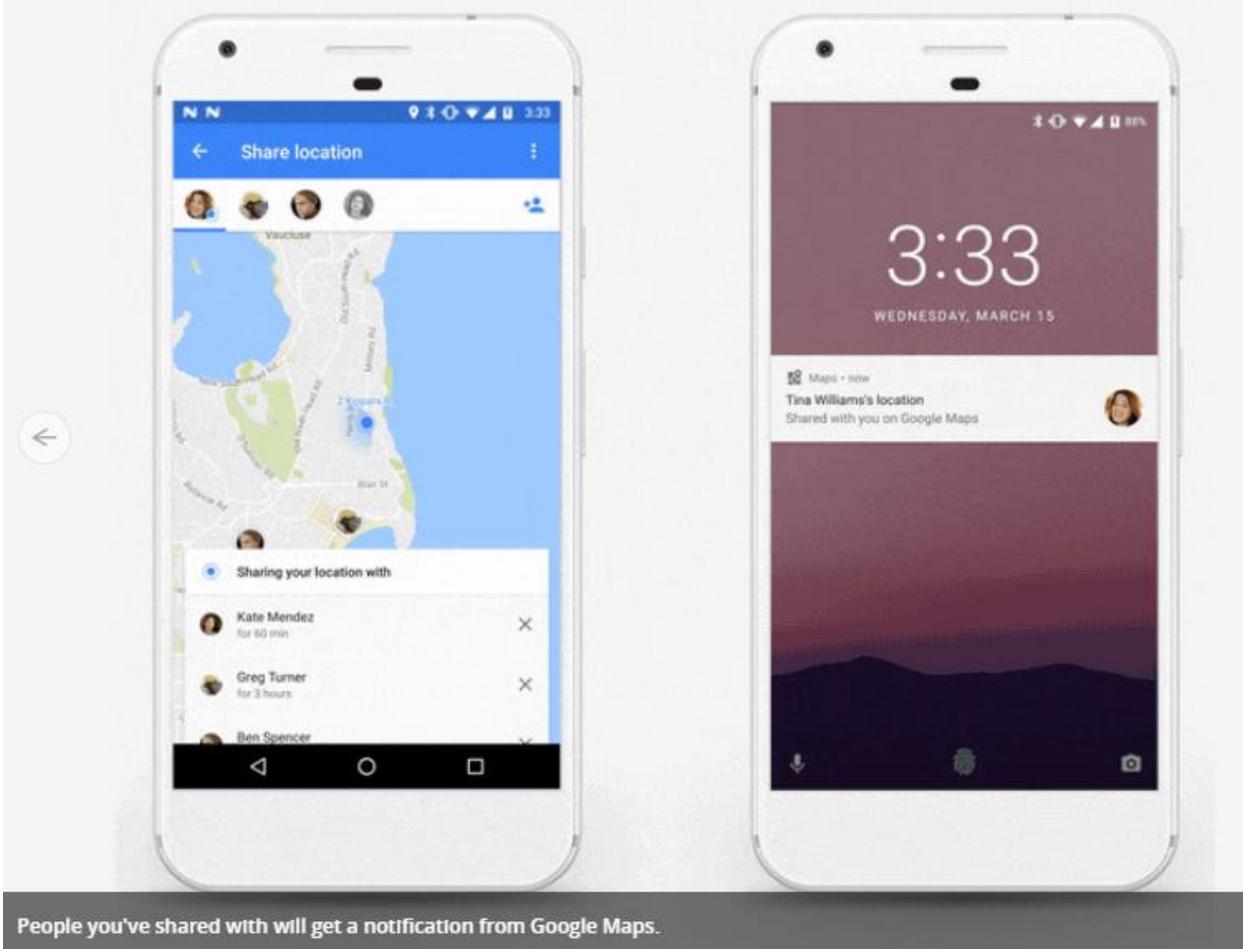
US9408055B2	HTC
	<p data-bbox="548 245 945 289"><b>Share your contacts</b></p> <ol data-bbox="562 316 1045 479" style="list-style-type: none"><li>1. Open your device's Contacts app .</li><li>2. Tap a contact in the list.</li><li>3. Tap More  &gt; <b>Share</b>.</li><li>4. Choose how you want to share the contact.</li></ol> <p data-bbox="512 495 1535 527"><a href="https://support.google.com/android/answer/6118731?hl=en&amp;ref_topic=6118711">https://support.google.com/android/answer/6118731?hl=en&amp;ref_topic=6118711</a></p>  <p data-bbox="520 1230 1516 1284">Ways to start location sharing: (left) from the app navigation drawer, and (right) from the menu, you can drag up while navigating somewhere.</p> <p data-bbox="512 1295 1656 1323"><a href="https://arstechnica.com/gadgets/2017/03/location-sharing-finally-returns-to-google-maps/">https://arstechnica.com/gadgets/2017/03/location-sharing-finally-returns-to-google-maps/</a></p>



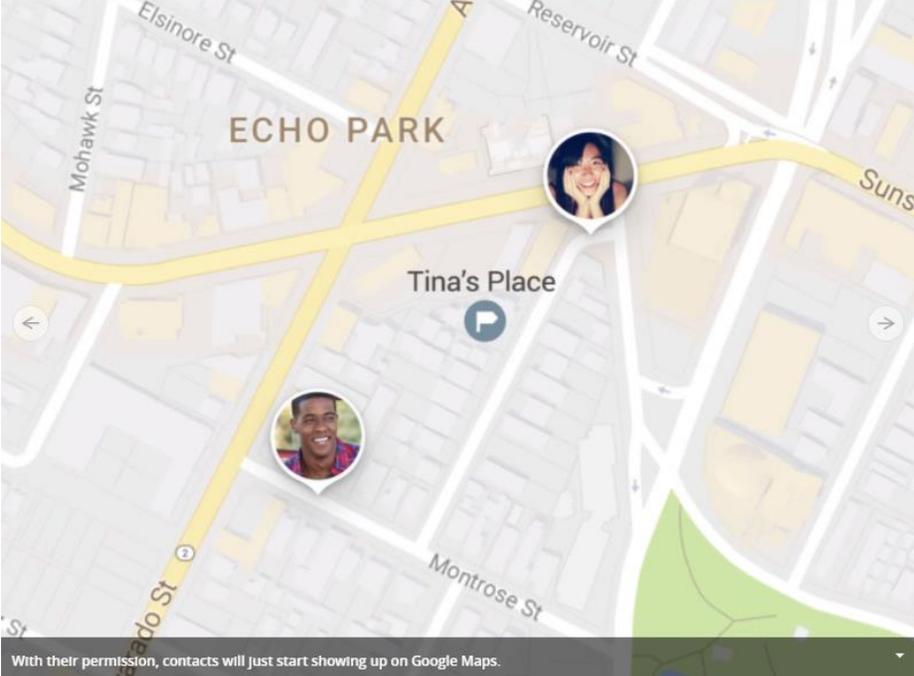
**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC
	 <p data-bbox="512 1008 1661 1057">You can pick how long you want to share your location for, and with whom you want to share it.</p> <p data-bbox="512 1057 1661 1097"><a href="https://arstechnica.com/gadgets/2017/03/location-sharing-finally-returns-to-google-maps/">https://arstechnica.com/gadgets/2017/03/location-sharing-finally-returns-to-google-maps/</a></p>

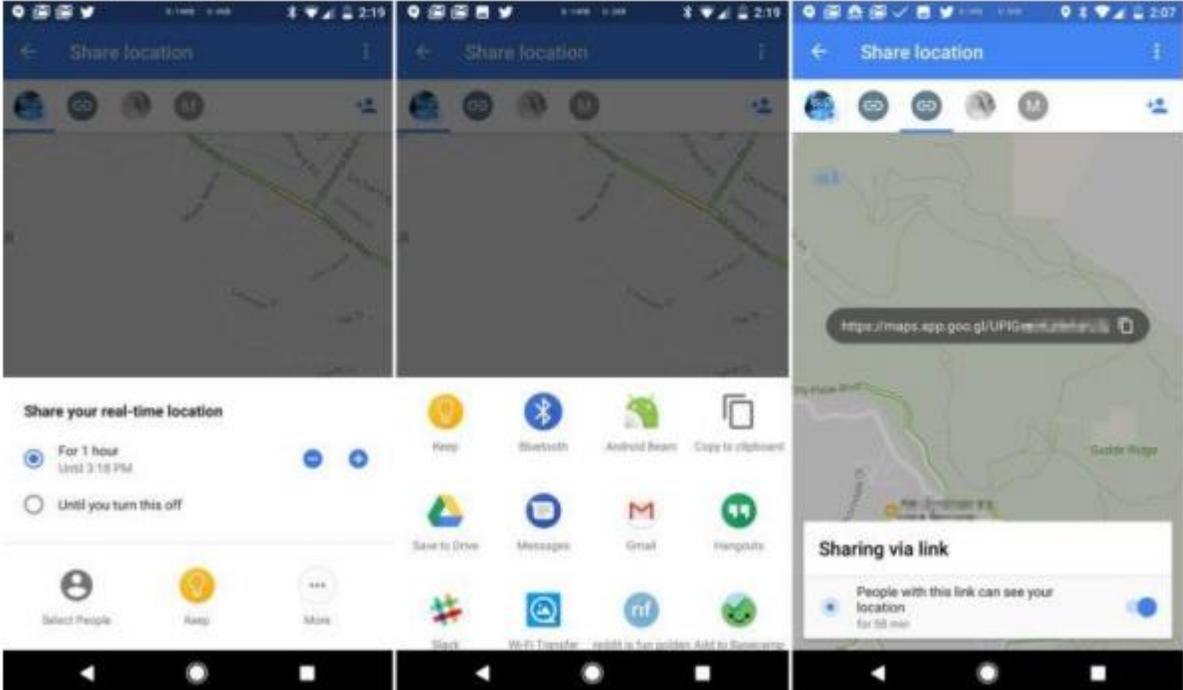
**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC
	 <p data-bbox="512 1143 1176 1170">People you've shared with will get a notification from Google Maps.</p> <p data-bbox="512 1187 1656 1219"><a href="https://arstechnica.com/gadgets/2017/03/location-sharing-finally-returns-to-google-maps/">https://arstechnica.com/gadgets/2017/03/location-sharing-finally-returns-to-google-maps/</a></p>

**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC
	 <p data-bbox="514 885 1018 901">With their permission, contacts will just start showing up on Google Maps.</p> <p data-bbox="514 917 1648 950"><a href="https://arstechnica.com/gadgets/2017/03/location-sharing-finally-returns-to-google-maps/">https://arstechnica.com/gadgets/2017/03/location-sharing-finally-returns-to-google-maps/</a></p> <h3 data-bbox="514 1023 777 1063">Stop sharing</h3> <ol data-bbox="514 1096 1354 1218" style="list-style-type: none"><li>1. Open the Google Maps app 📍.</li><li>2. Tap the Menu ☰ &gt; Share location.</li><li>3. Next to the person with whom you want to stop sharing, tap Remove ✕.</li></ol>

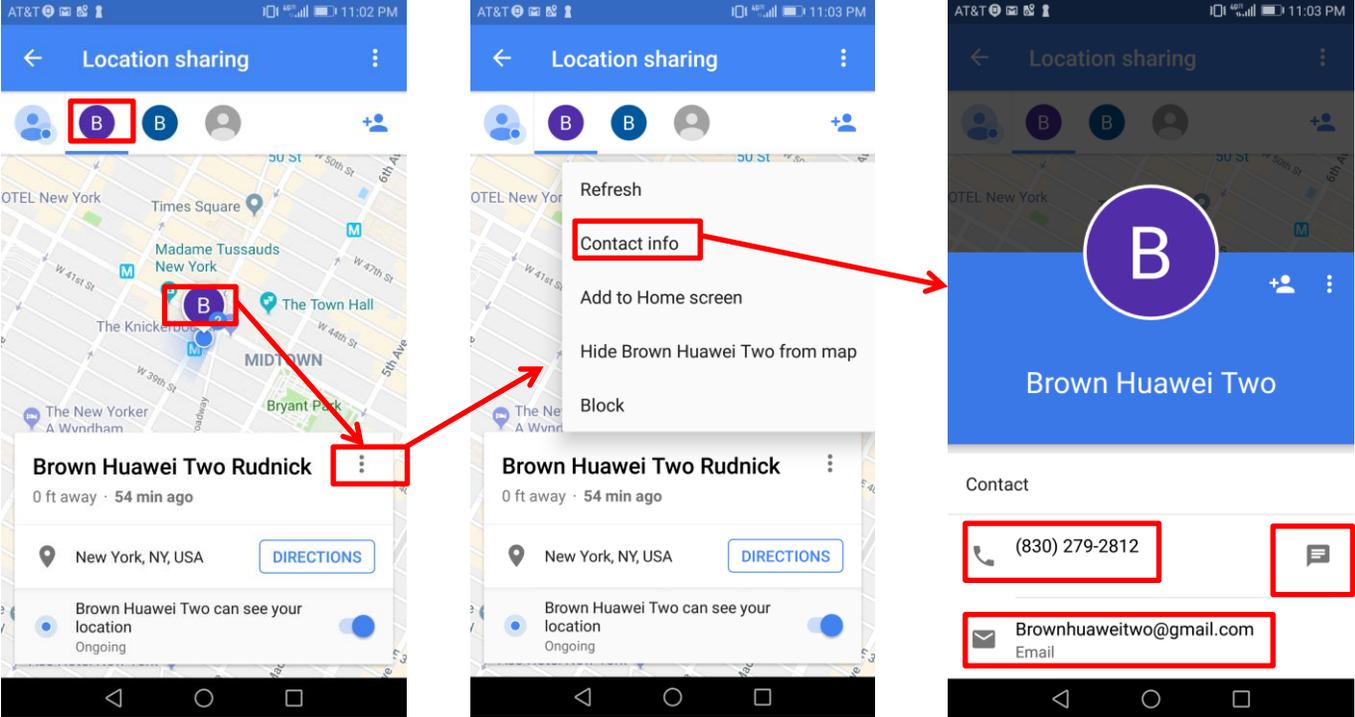
**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC
	<p data-bbox="541 233 856 272"><b>Share your E.T.A</b></p> <p data-bbox="541 302 1703 326">After you start your drive, you can share your destination, estimated arrival time, and where you are on the route.</p> <ol data-bbox="541 358 1388 602" style="list-style-type: none"><li>1. Open the Google Maps app .</li><li>2. Set a driving destination. <a href="#">Learn how to navigate to a place.</a></li><li>3. After you start navigation, tap <b>More</b>  <b>&gt; Share trip progress.</b></li><li>4. Choose a person from the list.</li><li>5. Tap <b>Share.</b></li><li>6. Location Sharing will stop when you reach your destination or stop navigating.</li></ol> <p data-bbox="541 634 1230 659">• To stop sharing before you arrive, tap <b>More</b>  <b>&gt; Stop sharing.</b></p> 

**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC
	 <p><b><u>Exemplary Maps Screenshots:</u></b></p>

**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC
	 <p><b>Exemplary Source Code:</b></p> <p>The above functionality is performed at least in part by the following publicly available source code and/or source code that invokes or is invoked by the following source code (or a substantially similar copy compiled and loaded onto the Accused Products by HTC). AGIS reserves the right to supplement these contentions with additional source code as discovery progresses and as additional source code is made available. AGIS reserves the right to supplement these contentions with additional source code as discovery progresses and as additional source code is made available.</p>

**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC
	<pre>44 * Class that sends chat message via SMS. 45 * 46 * The interface emulates a blocking sending similar to making an HTTP request. 47 * It calls the SmsManager to send a (potentially multipart) message and waits 48 * on the sent status on each part. The waiting has a timeout so it won't wait 49 * forever. Once the sent status of all parts received, the call returns. 50 * A successful sending requires success status for all parts. Otherwise, we 51 * pick the highest level of failure as the error for the whole message, which 52 * is used to determine if we need to retry the sending. 53 */ 54 public class SmsSender { 55     private static final String TAG = LogUtil.BUGLE_TAG; 56 57     public static final String EXTRA_PART_ID = "part_id"; 58 59     /* 60     * A map for pending sms messages. The key is the random request UUID. 61     */ 62     private static ConcurrentHashMap&lt;Uri, SendResult&gt; sPendingMessageMap = 63         new ConcurrentHashMap&lt;Uri, SendResult&gt;(); 64 65     private static final Random RANDOM = new Random(); 66 67     // Whether we should send multipart SMS as separate messages 68     private static Boolean sSendMultipartSmsAsSeparateMessages = null; 69</pre> <p><a href="https://android.googlesource.com/platform/packages/apps/Messaging/+nougat-mr1-release/src/com/android/messaging/sms/SmsSender.java">https://android.googlesource.com/platform/packages/apps/Messaging/+nougat-mr1-release/src/com/android/messaging/sms/SmsSender.java</a></p>

**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC
	<pre> 253 // Actually sending the message using SmsManager 254 private static void sendInternal(final Context context, final int subId, String dest, 255     final ArrayList&lt;String&gt; messages, final String serviceCenter, 256     final boolean requireDeliveryReport, final Uri messageUri) throws SmsException { 257     Assert.notNull(context); 258     final SmsManager smsManager = PhoneUtils.get(subId).getSmsManager(); 259     final int messageCount = messages.size(); 260     final ArrayList&lt;PendingIntent&gt; deliveryIntents = new ArrayList&lt;PendingIntent&gt;(messageCount); 261     final ArrayList&lt;PendingIntent&gt; sentIntents = new ArrayList&lt;PendingIntent&gt;(messageCount); 262     for (int i = 0; i &lt; messageCount; i++) { 263         // Make pending intents different for each message part 264         final int partId = (messageCount &lt;= 1 ? 0 : i + 1); 265         if (requireDeliveryReport &amp;&amp; (i == (messageCount - 1))) { 266             // TODO we only care about the delivery status of the last part 267             // Shall we have better tracking of delivery status of all parts? 268             deliveryIntents.add(PendingIntent.getBroadcast( 269                 context, 270                 partId, 271                 getSendStatusIntent(context, SendStatusReceiver.MESSAGE_DELIVERED_ACTION, 272                     messageUri, partId, subId), 273                 0/*flag*/)); 274         } else { 275             deliveryIntents.add(null); 276         } 277         sentIntents.add(PendingIntent.getBroadcast( 278             context, 279             partId, 280             getSendStatusIntent(context, SendStatusReceiver.MESSAGE_SENT_ACTION, 281                 messageUri, partId, subId), 282             0/*flag*/)); 283     } 284     if (sSendMultipartSmsAsSeparateMessages == null) { 285         sSendMultipartSmsAsSeparateMessages = MmsConfig.get(subId) 286             .getSendMultipartSmsAsSeparateMessages(); 287     } </pre> <p><a href="https://android.googlesource.com/platform/packages/apps/Messaging/+nougat-mr1-release/src/com/android/messaging/sms/SmsSender.java">https://android.googlesource.com/platform/packages/apps/Messaging/+nougat-mr1-release/src/com/android/messaging/sms/SmsSender.java</a></p>



**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC
	<pre>288     try { 289         if (sSendMultipartSmsAsSeparateMessages) { 290             // If multipart sms is not supported, send them as separate messages 291             for (int i = 0; i &lt; messageCount; i++) { 292                 smsManager.sendTextMessage(dest, 293                     serviceCenter, 294                     messages.get(i), 295                     sentIntents.get(i), 296                     deliveryIntents.get(i)); 297             } 298         } else { 299             smsManager.sendMultipartTextMessage( 300                 dest, serviceCenter, messages, sentIntents, deliveryIntents); 301         } 302     } catch (final Exception e) { 303         throw new SmsException("SmsSender: caught exception in sending " + e); 304     } 305 }</pre> <p><a href="https://android.googlesource.com/platform/packages/apps/Messaging/+nougat-mr1-release/src/com/android/messaging/sms/SmsSender.java">https://android.googlesource.com/platform/packages/apps/Messaging/+nougat-mr1-release/src/com/android/messaging/sms/SmsSender.java</a></p> <pre>56  * Class that receives incoming SMS messages through android.provider.Telephony.SMS_RECEIVED 57  * 58  * This class serves two purposes: 59  * - Process phone verification SMS messages 60  * - Handle SMS messages when the user has enabled us to be the default SMS app (Pre-KLP) 61  */ 62  public final class SmsReceiver extends BroadcastReceiver { 63      private static final String TAG = LogUtil.BUGLE_TAG; 64 65      private static ArrayList&lt;Pattern&gt; sIgnoreSmsPatterns; 66  }</pre> <p><a href="https://android.googlesource.com/platform/packages/apps/Messaging/+nougat-mr1-release/src/com/android/messaging/receiver/SmsReceiver.java">https://android.googlesource.com/platform/packages/apps/Messaging/+nougat-mr1-release/src/com/android/messaging/receiver/SmsReceiver.java</a></p>

**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC
	<pre>200 public static void deliverSmsMessages(final Context context, final int subId, 201     final int errorCode, final android.telephony.SmsMessage[] messages) { 202     final ContentValues messageValues = 203         MmsUtils.parseReceivedSmsMessage(context, messages, errorCode); 204 205     LogUtil.v(TAG, "SmsReceiver.deliverSmsMessages"); 206 207     final long nowInMillis = System.currentTimeMillis(); 208     final long receivedTimestampMs = MmsUtils.getMessageDate(messages[0], nowInMillis); 209 210     messageValues.put(Sms.Inbox.DATE, receivedTimestampMs); 211     // Default to unread and unseen for us but ReceiveSmsMessageAction will override 212     // seen for the telephony db. 213     messageValues.put(Sms.Inbox.READ, 0); 214     messageValues.put(Sms.Inbox.SEEN, 0); 215     if (OsUtil.isAtLeastL_MR1()) { 216         messageValues.put(Sms.SUBSCRIPTION_ID, subId); 217     } 218 219     if (messages[0].getMessageClass() == android.telephony.SmsMessage.MessageClass.CLASS_0    220         DebugUtils.debugClassZeroSmsEnabled()) { 221         Factory.get().getUIIntents().launchClassZeroActivity(context, messageValues); 222     } else { 223         final ReceiveSmsMessageAction action = new ReceiveSmsMessageAction(messageValues); 224         action.start(); 225     } 226 }</pre> <p><a href="https://android.googlesource.com/platform/packages/apps/Messaging/+nougat-mr1-release/src/com/android/messaging/receiver/SmsReceiver.java">https://android.googlesource.com/platform/packages/apps/Messaging/+nougat-mr1-release/src/com/android/messaging/receiver/SmsReceiver.java</a></p>

**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC
	<pre>228     @Override 229     public void onReceive(final Context context, final Intent intent) { 230         LogUtil.v(TAG, "SmsReceiver.onReceive " + intent); 231         // On KLP+ we only take delivery of SMS messages in SmsDeliverReceiver. 232         if (PhoneUtils.getDefault().isSmsEnabled()) { 233             final String action = intent.getAction(); 234             if (OsUtil.isSecondaryUser() &amp;&amp; 235                 (Telephony.Sms.Intents.SMS_RECEIVED_ACTION.equals(action)    236                  // TODO: update this with the actual constant from Telephony 237                  "android.provider.Telephony.MMS_DOWNLOADED".equals(action))) { 238                 postNewMessageSecondaryUserNotification(); 239             } else if (!OsUtil.isAtLeastKLP()) { 240                 deliverSmsIntent(context, intent); 241             } 242         } 243     }</pre> <p><a href="https://android.googlesource.com/platform/packages/apps/Messaging/+/nougat-mr1-release/src/com/android/messaging/receiver/SmsReceiver.java">https://android.googlesource.com/platform/packages/apps/Messaging/+/nougat-mr1-release/src/com/android/messaging/receiver/SmsReceiver.java</a></p>

**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC
	<pre>52  * Class that sends chat message via MMS. 53  * 54  * The interface emulates a blocking send similar to making an HTTP request. 55  */ 56  public class MmsSender { 57      private static final String TAG = LogUtil.BUGLE_TAG; 58 59      /** 60       * Send an MMS message. 61       * 62       * @param context Context 63       * @param messageUri The unique URI of the message for identifying it during sending 64       * @param sendReq The SendReq PDU of the message 65       * @throws MmsFailureException 66       */ 67      public static void sendMms(final Context context, final int subId, final Uri messageUri, 68                               final SendReq sendReq, final Bundle sentIntentExtras) throws MmsFailureException { 69          sendMms(context, 70                subId, 71                messageUri, 72                null /* locationUri */, 73                sendReq, 74                true /* responseImportant */, 75                sentIntentExtras); 76      }</pre> <p><a href="https://android.googlesource.com/platform/packages/apps/Messaging/+/nougat-mr1-release/src/com/android/messaging/sms/MmsSender.java">https://android.googlesource.com/platform/packages/apps/Messaging/+/nougat-mr1-release/src/com/android/messaging/sms/MmsSender.java</a></p>

**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC
	<pre>240     * Download an MMS message. 241     * 242     * @param context Context 243     * @param contentLocation The url of the MMS message 244     * @throws MmsFailureException 245     * @throws InvalidHeaderValueException 246     */ 247     public static void downloadMms(final Context context, final int subId, 248         final String contentLocation, Bundle extras) throws MmsFailureException, 249         InvalidHeaderValueException { 250         final Uri requestUri = Uri.parse(contentLocation); 251         final Uri contentUri = MmsFileProvider.buildRawMmsUri(); 252 253         final Intent downloadedIntent = new Intent(SendStatusReceiver.MMS_DOWNLOADED_ACTION, 254             requestUri, 255             context, 256             SendStatusReceiver.class); 257         downloadedIntent.putExtra(SendMessageAction.EXTRA_CONTENT_URI, contentUri); 258         if (extras != null) { 259             downloadedIntent.putExtras(extras); 260         } 261         final PendingIntent downloadedPendingIntent = PendingIntent.getBroadcast( 262             context, 263             0 /*request code*/, 264             downloadedIntent, 265             PendingIntent.FLAG_UPDATE_CURRENT); 266 267         MmsManager.downloadMultimediaMessage(subId, context, contentLocation, contentUri, 268             downloadedPendingIntent); 269     }</pre> <p><a href="https://android.googlesource.com/platform/packages/apps/Messaging/+nougat-mr1-release/src/com/android/messaging/sms/MmsSender.java">https://android.googlesource.com/platform/packages/apps/Messaging/+nougat-mr1-release/src/com/android/messaging/sms/MmsSender.java</a></p>

**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC
	<pre> 97      * Execute an MMS HTTP request, either a POST (sending) or a GET (downloading) 98      * 99      * @param urlString The request URL, for sending it is usually the MMSC, and for downloading 100     *           it is the message URL 101     * @param pdu For POST (sending) only, the PDU to send 102     * @param method HTTP method, POST for sending and GET for downloading 103     * @param isProxySet Is there a proxy for the MMSC 104     * @param proxyHost The proxy host 105     * @param proxyPort The proxy port 106     * @param mmsConfig The MMS config to use 107     * @param userAgent The user agent header value 108     * @param uaProfUrl The UA Prof URL header value 109     * @return The HTTP response body 110     * @throws MmsHttpException For any failures 111     */ 112     public byte[] execute(String urlString, byte[] pdu, String method, boolean isProxySet, 113                          String proxyHost, int proxyPort, Bundle mmsConfig, String userAgent, String uaProfUrl) 114         throws MmsHttpException { 115         Log.d(MmsService.TAG, "HTTP: " + method + " " + Utils.redactUrlForNonVerbose(urlString) 116             + (isProxySet ? (" proxy=" + proxyHost + ":" + proxyPort) : "")) 117             + ", PDU size=" + (pdu != null ? pdu.length : 0)); 118         checkMethod(method); 119         HttpURLConnection connection = null; 120         try { 121             Proxy proxy = Proxy.NO_PROXY; 122             if (isProxySet) { 123                 proxy = new Proxy(Proxy.Type.HTTP, new InetSocketAddress(proxyHost, proxyPort)); 124             } 125             final URL url = new URL(urlString); 126             // Now get the connection 127             connection = (HttpURLConnection) url.openConnection(proxy); 128             connection.setDoInput(true); 129             connection.setConnectTimeout( 130                 mmsConfig.getInt(CarrierConfigValuesLoader.CONFIG_HTTP_SOCKET_TIMEOUT, 131                     CarrierConfigValuesLoader.CONFIG_HTTP_SOCKET_TIMEOUT_DEFAULT)); </pre> <p><a href="https://android.googlesource.com/platform/packages/apps/Messaging/+nougat-mr1-release/src/android/support/v7/mms/MmsHttpClient.java">https://android.googlesource.com/platform/packages/apps/Messaging/+nougat-mr1-release/src/android/support/v7/mms/MmsHttpClient.java</a></p>

**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC
	<pre> 132 // ----- COMMON HEADERS ----- 133 // Header: Accept 134 connection.setRequestProperty(HEADER_ACCEPT, HEADER_VALUE_ACCEPT); 135 // Header: Accept-Language 136 connection.setRequestProperty( 137     HEADER_ACCEPT_LANGUAGE, getCurrentAcceptLanguage(Locale.getDefault())); 138 // Header: User-Agent 139 Log.i(MmsService.TAG, "HTTP: User-Agent=" + userAgent); 140 connection.setRequestProperty(HEADER_USER_AGENT, userAgent); 141 // Header: x-wap-profile 142 final String uaProfUrlTagName = mmsConfig.getString( 143     CarrierConfigValuesLoader.CONFIG_UA_PROF_TAG_NAME, 144     CarrierConfigValuesLoader.CONFIG_UA_PROF_TAG_NAME_DEFAULT); 145 if (uaProfUrl != null) { 146     Log.i(MmsService.TAG, "HTTP: UaProfUrl=" + uaProfUrl); 147     connection.setRequestProperty(uaProfUrlTagName, uaProfUrl); 148 } 149 // Add extra headers specified by mms_config.xml's httpparams 150 addExtraHeaders(connection, mmsConfig); 151 // Different stuff for GET and POST 152 if (METHOD_POST.equals(method)) { 153     if (pdu == null    pdu.length &lt; 1) { 154         Log.e(MmsService.TAG, "HTTP: empty pdu"); 155         throw new MmsHttpException(0/*statusCode*/, "Sending empty PDU"); 156     } 157     connection.setDoOutput(true); 158     connection.setRequestMethod(METHOD_POST); 159     if (mmsConfig.getBoolean( 160         CarrierConfigValuesLoader.CONFIG_SUPPORT_HTTP_CHARSET_HEADER, 161         CarrierConfigValuesLoader.CONFIG_SUPPORT_HTTP_CHARSET_HEADER_DEFAULT)) { 162         connection.setRequestProperty(HEADER_CONTENT_TYPE, 163             HEADER_VALUE_CONTENT_TYPE_WITH_CHARSET); 164     } else { 165         connection.setRequestProperty(HEADER_CONTENT_TYPE, 166             HEADER_VALUE_CONTENT_TYPE_WITHOUT_CHARSET); </pre> <p><a href="https://android.googlesource.com/platform/packages/apps/Messaging/+/nougat-mr1-release/src/android/support/v7/mms/MmsHttpClient.java">https://android.googlesource.com/platform/packages/apps/Messaging/+/nougat-mr1-release/src/android/support/v7/mms/MmsHttpClient.java</a></p>

**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC
	<pre>167     } 168     if (Log.isLoggable(MmsService.TAG, Log.VERBOSE)) { 169         logHttpHeaders(connection.getRequestProperties()); 170     } 171     connection.setFixedLengthStreamingMode(pdu.length); 172     // Sending request body 173     final OutputStream out = 174         new BufferedOutputStream(connection.getOutputStream()); 175     out.write(pdu); 176     out.flush(); 177     out.close(); 178 } else if (METHOD_GET.equals(method)) { 179     if (Log.isLoggable(MmsService.TAG, Log.VERBOSE)) { 180         logHttpHeaders(connection.getRequestProperties()); 181     } 182     connection.setRequestMethod(METHOD_GET); 183 } 184 // Get response 185 final int responseCode = connection.getResponseCode(); 186 final String responseMessage = connection.getResponseMessage(); 187 Log.d(MmsService.TAG, "HTTP: " + responseCode + " " + responseMessage); 188 if (Log.isLoggable(MmsService.TAG, Log.VERBOSE)) { 189     logHttpHeaders(connection.getHeaderFields()); 190 } 191 if (responseCode / 100 != 2) { 192     throw new MmsHttpException(responseCode, responseMessage); 193 } 194 final InputStream in = new BufferedInputStream(connection.getInputStream()); 195 final ByteArrayOutputStream byteOut = new ByteArrayOutputStream(); 196 final byte[] buf = new byte[4096]; 197 int count = 0; 198 while ((count = in.read(buf)) &gt; 0) { 199     byteOut.write(buf, 0, count); 200 } 201 in.close(); 202 final byte[] responseBody = byteOut.toByteArray(); 203 Log.d(MmsService.TAG, "HTTP: response size=" 204     + (responseBody != null ? responseBody.length : 0)); 205 return responseBody;</pre> <p><a href="https://android.googlesource.com/platform/packages/apps/Messaging/+nougat-mr1-release/src/android/support/v7/mms/MmsHttpClient.java">https://android.googlesource.com/platform/packages/apps/Messaging/+nougat-mr1-release/src/android/support/v7/mms/MmsHttpClient.java</a></p>



**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC
	<pre>38  * Request to send an MMS 39  */ 40  class SendRequest extends MmsRequest { 41      // Max send response PDU size in bytes (exceeding this may cause problem with 42      // system intent delivery). 43      private static final int MAX_SEND_RESPONSE_SIZE = 1000 * 1024; 44 45      private byte[] mPduData; 46 47      SendRequest(final String locationUri, final Uri pduUri, final PendingIntent sentIntent) { 48          super(locationUri, pduUri, sentIntent); 49      } 50 51      @Override 52      protected boolean loadRequest(final Context context, final Bundle mmsConfig) { 53          mPduData = readPduFromContentUri( 54              context, 55              mPduUri, 56              mmsConfig.getInt( 57                  CarrierConfigValuesLoader.CONFIG_MAX_MESSAGE_SIZE, 58                  CarrierConfigValuesLoader.CONFIG_MAX_MESSAGE_SIZE_DEFAULT)); 59          return (mPduData != null); 60      } 61 62      @Override 63      protected boolean transferResponse(final Context context, final Intent fillIn, 64          final byte[] response) { 65          // SendConf pdus are always small and can be included in the intent 66          if (response != null &amp;&amp; fillIn != null) { 67              if (response.length &gt; MAX_SEND_RESPONSE_SIZE) { 68                  // If the response PDU is too large, it won't be able to fit in 69                  // the PendingIntent to be transferred via system IPC. 70                  return false; 71              } 72              fillIn.putExtra(SmsManager.EXTRA_MMS_DATA, response); 73          } 74          return true; 75      } }</pre>

**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC
	<p><a href="https://android.googlesource.com/platform/packages/apps/Messaging/+/nougat-mr1-release/src/android/support/v7/mms/SendRequest.java">https://android.googlesource.com/platform/packages/apps/Messaging/+/nougat-mr1-release/src/android/support/v7/mms/SendRequest.java</a></p> <pre data-bbox="533 354 1738 397">public static LocationRequest create ()</pre> <p>Create a location request with default parameters.</p> <p>Default parameters are for a block accuracy, slowly updated location. It can then be adjusted as required by the applications before passing to the <a href="#">FusedLocationProviderApi</a>.</p> <p><b>Returns</b></p> <ul data-bbox="554 613 814 636" style="list-style-type: none"><li>• a new location request</li></ul> <p><a href="https://developers.google.com/android/reference/com/google/android/gms/location/LocationRequest">https://developers.google.com/android/reference/com/google/android/gms/location/LocationRequest</a></p>

**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC
	<p><code>public static final int PRIORITY_BALANCED_POWER_ACCURACY</code></p> <p>Used with <code>setPriority(int)</code> to request "block" level accuracy.</p> <p>Block level accuracy is considered to be about 100 meter accuracy. Using a coarse accuracy such as this often consumes less power.</p> <p>Constant Value: 102</p> <hr/> <p><code>public static final int PRIORITY_HIGH_ACCURACY</code></p> <p>Used with <code>setPriority(int)</code> to request the most accurate locations available.</p> <p>This will return the finest location available.</p> <p>Constant Value: 100</p> <hr/> <p><code>public static final int PRIORITY_LOW_POWER</code></p> <p>Used with <code>setPriority(int)</code> to request "city" level accuracy.</p> <p>City level accuracy is considered to be about 10km accuracy. Using a coarse accuracy such as this often consumes less power.</p> <p>Constant Value: 104</p> <p><a href="https://developers.google.com/android/reference/com/google/android/gms/location/LocationRequest">https://developers.google.com/android/reference/com/google/android/gms/location/LocationRequest</a></p>

**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC
	<pre data-bbox="533 248 1751 285">public Task&lt;Location&gt; getLastLocation ()</pre> <p data-bbox="525 313 1104 337">Returns the best most recent location currently available.</p> <p data-bbox="525 370 1696 430">If a location is not available, which should happen very rarely, null will be returned. The best accuracy available while respecting the location permissions will be returned.</p> <p data-bbox="525 462 1738 522">This method provides a simplified way to get location. It is particularly well suited for applications that do not require an accurate location and that do not want to maintain extra logic for location updates.</p> <pre data-bbox="533 578 1751 615">public Task&lt;LocationAvailability&gt; getLocationAvailability ()</pre> <p data-bbox="525 643 1692 703">Returns the availability of location data. When <code>isLocationAvailable()</code> returns true, then the location returned by <code>getLastLocation()</code> will be reasonably up to date within the hints specified by the active <code>LocationRequest</code>s.</p> <p data-bbox="525 735 1472 760">If the client isn't connected to Google Play services and the request times out, null is returned.</p> <p data-bbox="525 792 1675 852">Note it's always possible for <code>getLastLocation()</code> to return null even when this method returns true (e.g. location settings were disabled between calls).</p> <p data-bbox="510 862 1902 930"><a href="https://developers.google.com/android/reference/com/google/android/gms/location/FusedLocationProviderClient">https://developers.google.com/android/reference/com/google/android/gms/location/FusedLocationProviderClient</a></p>

**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC						
	<p data-bbox="527 245 1747 326">public <code>Task&lt;Void&gt; requestLocationUpdates</code> (<code>LocationRequest</code> request, <code>LocationCallback</code> callback, <code>Looper</code> looper)</p> <p data-bbox="527 354 1272 378">Requests location updates with a callback on the specified Looper thread.</p> <p data-bbox="527 412 1686 472">This method is suited for the foreground use cases. For background use cases, the <code>PendingIntent</code> version of the method is recommended, see <code>requestLocationUpdates(LocationRequest, PendingIntent)</code>.</p> <p data-bbox="527 506 1371 531">Any previous <code>LocationRequests</code> registered on this <code>LocationListener</code> will be replaced.</p> <p data-bbox="527 565 1686 657">This call will keep the Google Play services connection active, so make sure to call <code>removeLocationUpdates(LocationCallback)</code> when you no longer need it, otherwise you lose the benefits of the automatic connection management.</p> <p data-bbox="527 691 1745 716">Callbacks for <code>LocationCallback</code> will be made on the specified thread, which must already be a prepared looper thread.</p> <p data-bbox="527 740 667 764"><b>Parameters</b></p> <table border="1" data-bbox="527 792 1747 1008"> <tbody> <tr> <td data-bbox="527 800 625 857"><b>request</b></td> <td data-bbox="632 800 1747 857">The location request for the updates.</td> </tr> <tr> <td data-bbox="527 865 625 922"><b>callback</b></td> <td data-bbox="632 865 1747 922">The callback for the location updates.</td> </tr> <tr> <td data-bbox="527 930 625 1003"><b>looper</b></td> <td data-bbox="632 930 1747 1003">The <code>Looper</code> object whose message queue will be used to implement the callback mechanism, or null to make callbacks on the calling thread.</td> </tr> </tbody> </table> <p data-bbox="527 1024 1902 1089"><a href="https://developers.google.com/android/reference/com/google/android/gms/location/FusedLocationProviderClient">https://developers.google.com/android/reference/com/google/android/gms/location/FusedLocationProviderClient</a></p>	<b>request</b>	The location request for the updates.	<b>callback</b>	The callback for the location updates.	<b>looper</b>	The <code>Looper</code> object whose message queue will be used to implement the callback mechanism, or null to make callbacks on the calling thread.
<b>request</b>	The location request for the updates.						
<b>callback</b>	The callback for the location updates.						
<b>looper</b>	The <code>Looper</code> object whose message queue will be used to implement the callback mechanism, or null to make callbacks on the calling thread.						

**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC				
	<p data-bbox="533 245 1740 321"> <code>public Task&lt;Void&gt; requestLocationUpdates (LocationRequest request, PendingIntent callbackIntent)</code> </p> <p data-bbox="533 354 1268 378">Requests location updates with a callback on the specified PendingIntent.</p> <p data-bbox="533 410 1740 540">This method is suited for the background use cases, more specifically for receiving location updates, even when the app has been killed by the system. In order to do so, use a <code>PendingIntent</code> for a started service. For foreground use cases, the <code>LocationCallback</code> version of the method is recommended, see <code>requestLocationUpdates(LocationRequest, LocationCallback, Looper)</code>.</p> <p data-bbox="533 573 1740 630">Any previously registered requests that have the same PendingIntent (as defined by <code>equals(Object)</code>) will be replaced by this request.</p> <p data-bbox="533 662 1740 751">Both <code>LocationResult</code> and <code>LocationAvailability</code> are sent to the given PendingIntent. You can extract data from an Intent using <code>hasResult(Intent)</code>, <code>extractResult(Intent)</code>, <code>hasLocationAvailability(Intent)</code>, and <code>extractLocationAvailability(Intent)</code>.</p> <p data-bbox="533 784 667 808"><b>Parameters</b></p> <table border="1" data-bbox="533 833 1740 971"> <tr> <td data-bbox="533 833 835 906"><code>request</code></td> <td data-bbox="835 833 1740 906">The location request for the updates.</td> </tr> <tr> <td data-bbox="533 906 835 971"><code>callbackIntent</code></td> <td data-bbox="835 906 1740 971">A pending intent to be sent for each location update.</td> </tr> </table> <p data-bbox="533 995 630 1019"><b>Returns</b></p> <ul data-bbox="533 1044 1360 1068" style="list-style-type: none"> <li>• a Task for the call, check <code>isSuccessful()</code> to determine if it was successful.</li> </ul> <p data-bbox="533 1076 1898 1146"> <a href="https://developers.google.com/android/reference/com/google/android/gms/location/FusedLocationProviderClient">https://developers.google.com/android/reference/com/google/android/gms/location/FusedLocationProviderClient</a> </p>	<code>request</code>	The location request for the updates.	<code>callbackIntent</code>	A pending intent to be sent for each location update.
<code>request</code>	The location request for the updates.				
<code>callbackIntent</code>	A pending intent to be sent for each location update.				

**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC						
	<p><code>public void onLocationAvailability (LocationAvailability locationAvailability)</code></p> <p>Called when there is a change in the availability of location data.</p> <p>When <code>isLocationAvailable()</code> returns <code>false</code> you can assume that location will not be returned in <code>onLocationResult(LocationResult)</code> until something changes in the device's settings or environment. Even when <code>isLocationAvailable()</code> returns <code>true</code> the <code>onLocationResult(LocationResult)</code> may not always be called regularly, however the device location is known and both the most recently delivered location and <code>getLastLocation(GoogleApiClient)</code> will be reasonably up to date given the hints specified by the active <code>LocationRequest</code>s.</p> <p><b>Parameters</b></p> <table border="1" data-bbox="527 639 1738 704"> <tr> <td><code>locationAvailability</code></td> <td>The current status of location availability.</td> </tr> </table> <p><code>public void onLocationResult (LocationResult result)</code></p> <p>Called when device location information is available.</p> <p>The most recent location returned by <code>getLastLocation()</code> is not guaranteed to be immediately fresh, but will be reasonably up to date given the hints specified by the active <code>LocationRequest</code>s.</p> <p><b>Parameters</b></p> <table border="1" data-bbox="527 1019 1738 1084"> <tr> <td><code>result</code></td> <td>The latest location result available.</td> </tr> </table> <p><a href="https://developers.google.com/android/reference/com/google/android/gms/location/LocationCallback">https://developers.google.com/android/reference/com/google/android/gms/location/LocationCallback</a></p> <p><code>public abstract void onLocationChanged (Location location)</code></p> <p>Called when the location has changed.</p> <p><b>Parameters</b></p> <table border="1" data-bbox="527 1318 1738 1383"> <tr> <td><code>location</code></td> <td>The updated location.</td> </tr> </table>	<code>locationAvailability</code>	The current status of location availability.	<code>result</code>	The latest location result available.	<code>location</code>	The updated location.
<code>locationAvailability</code>	The current status of location availability.						
<code>result</code>	The latest location result available.						
<code>location</code>	The updated location.						

**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

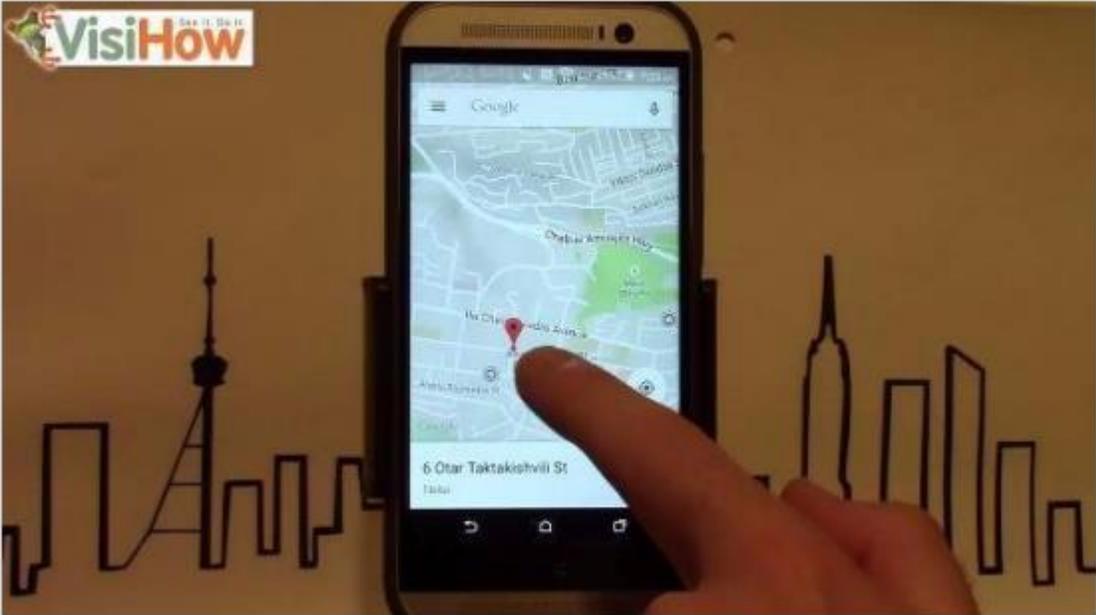
US9408055B2	HTC
	<p data-bbox="508 235 1797 264"><a href="https://developers.google.com/android/reference/com/google/android/gms/location/LocationListener">https://developers.google.com/android/reference/com/google/android/gms/location/LocationListener</a></p> <p data-bbox="508 316 814 345">Public Constructors</p> <hr data-bbox="508 358 1743 362"/> <p data-bbox="508 410 1743 440">public <b>MapView</b> (<a href="#">Context</a> context)</p> <p data-bbox="508 500 1743 529">public <b>MapView</b> (<a href="#">Context</a> context, <a href="#">AttributeSet</a> attrs)</p> <p data-bbox="508 589 1743 618">public <b>MapView</b> (<a href="#">Context</a> context, <a href="#">AttributeSet</a> attrs, int defStyle)</p> <p data-bbox="508 678 1743 708">public <b>MapView</b> (<a href="#">Context</a> context, <a href="#">GoogleMapOptions</a> options)</p> <p data-bbox="508 743 1671 773"><a href="https://developers.google.com/android/reference/com/google/android/gms/maps/MapView">https://developers.google.com/android/reference/com/google/android/gms/maps/MapView</a></p>



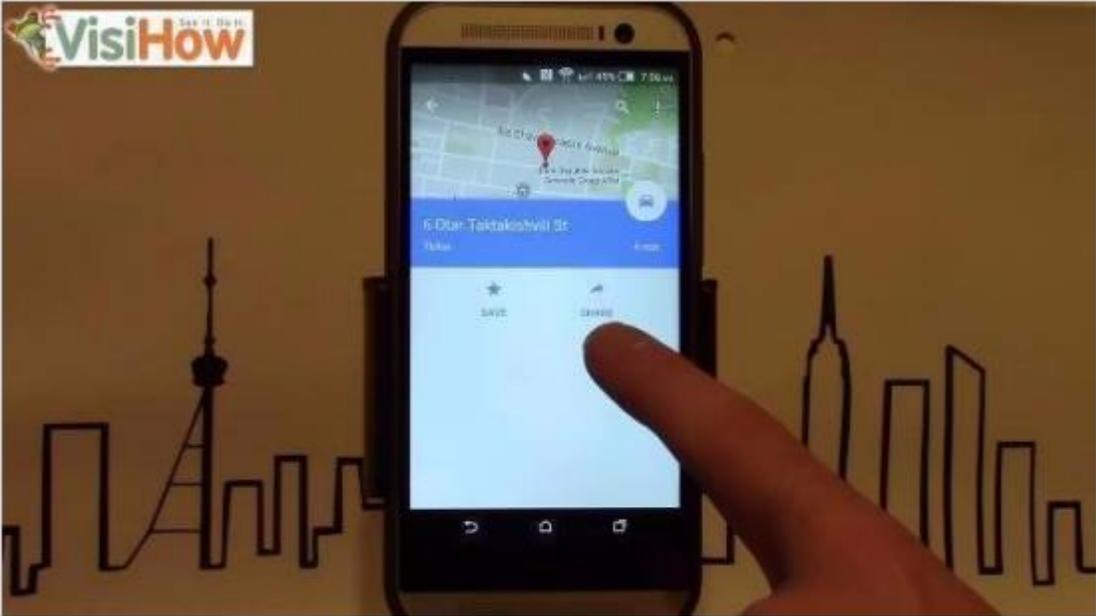
**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC		
	<p><code>public void <b>getMapAsync</b> (<a href="#">OnMapReadyCallback</a> callback)</code></p> <p>Returns a non-null instance of the <a href="#">GoogleMap</a>, ready to be used.</p> <p>Note that:</p> <ul style="list-style-type: none"> <li>• This method must be called from the main thread.</li> <li>• The callback will be executed in the main thread.</li> <li>• In the case where Google Play services is not installed on the user's device, the callback will not be triggered until the user installs it.</li> <li>• The <a href="#">GoogleMap</a> object provided by the callback is non-null.</li> </ul> <p><b>Parameters</b></p> <table border="1" data-bbox="527 688 1738 753"> <tr> <td><code>callback</code></td> <td>The callback object that will be triggered when the map is ready to be used.</td> </tr> </table> <p><code>public final void <b>onCreate</b> (<a href="#">Bundle</a> savedInstanceState)</code></p> <p>You must call this method from the parent Activity/Fragment's corresponding method.</p> <p><a href="https://developers.google.com/android/reference/com/google/android/gms/maps/MapView">https://developers.google.com/android/reference/com/google/android/gms/maps/MapView</a></p>	<code>callback</code>	The callback object that will be triggered when the map is ready to be used.
<code>callback</code>	The callback object that will be triggered when the map is ready to be used.		
<p>[41G] receiving user input via user interaction with the interactive display of the first device, the user input specifying a location and a symbol corresponding to an entity other than the first device and the second devices;</p>	<p>HTC infringes directly and/or indirectly by performing, inducing others to perform, and/or contributing to the performance of receiving user input via user interaction with the interactive display of the first device, the user input specifying a location and a symbol corresponding to an entity other than the first device and the second devices. See claims 1[G] and 28[G], which are incorporated herein by reference in their entirety.</p> <p><b>Share Locations from Google Maps on HTC One M8</b></p>		

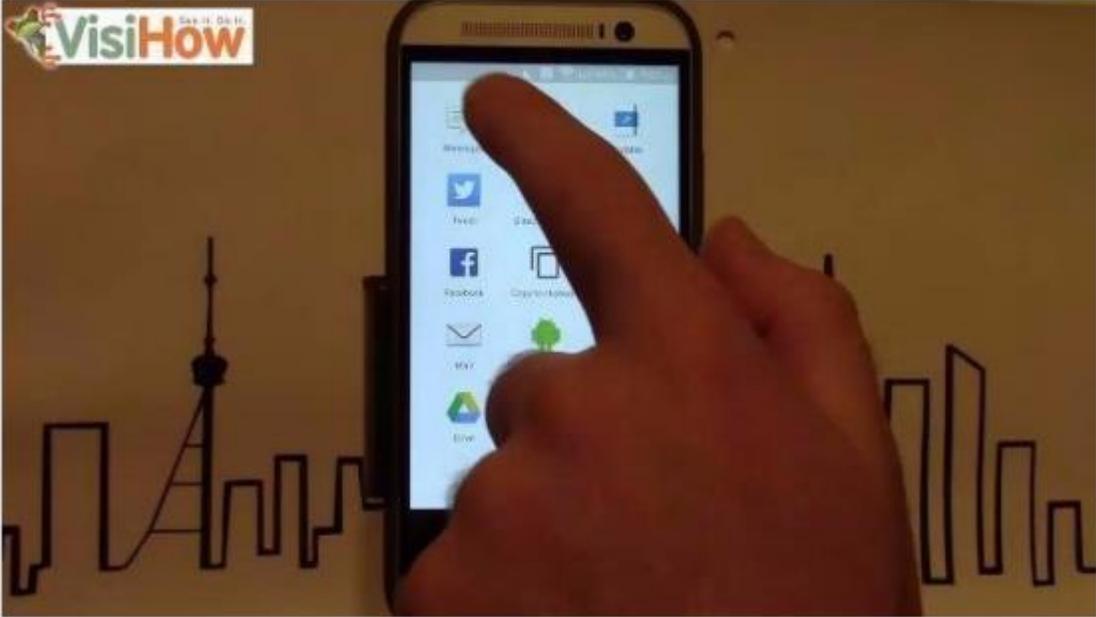
**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC
	<p data-bbox="531 233 827 256"><b>Place a pin on the map.</b></p> <p data-bbox="531 272 1640 380">We can do this anywhere in the entire world, anywhere, on any map that we find in the Google Maps application. We are going to choose a location, press down on the screen, and hold down until we can see that a red pin has been dropped.</p> 

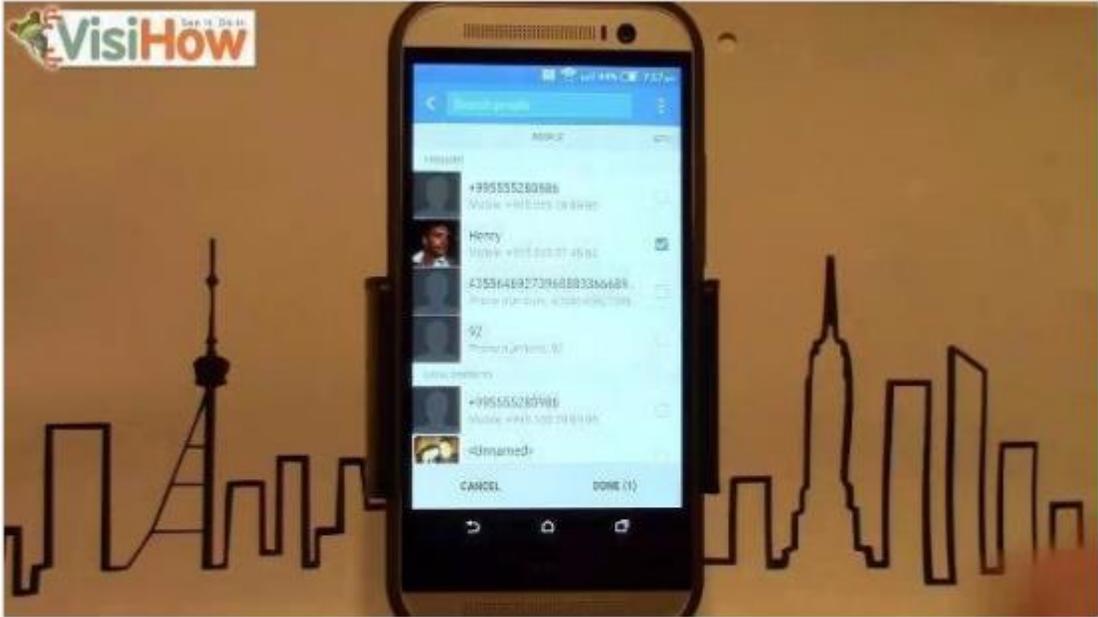
**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC
	<p data-bbox="527 240 1207 305"><b>We have two options:</b> "SAVE" and "SHARE". Press down on the word "SHARE".</p>  <hr/> <p data-bbox="527 1057 1633 1208"><b>A menu appears with all of the different ways that we can share this location.</b> We can share this location via messages, Bluetooth, Twitter, Facebook, mail, Gmail and Viber. If you have other applications, such as WhatsApp, or any other chat or email applications, you will be able to use them as well and will see them here.</p>

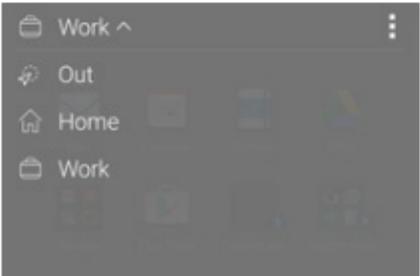
**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC
	<p data-bbox="527 237 856 264"><b>Let's choose "Messages".</b></p> <p data-bbox="527 277 1633 347">This option is located on the top left-hand corner of the screen. All options will have the same basic processes. They will ask us to send the location. Press down on the "Messages" icon.</p> 

**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC
	<p data-bbox="520 228 1260 256"><b>Press the box next to the contact who will be the recipient.</b></p> <p data-bbox="520 264 1549 292">A checkmark will appear in the box. We can select multiple contacts as well if we desire.</p>  <p data-bbox="520 1044 911 1071"><b>Press on the word "DONE (1)".</b></p> <p data-bbox="520 1081 1629 1151">This icon is located in the bottom right-hand corner of the screen. The bracketed number indicates the number of contacts we selected.</p> <p data-bbox="520 1161 1409 1192"><a href="http://visihow.com/Share_Locations_from_Google_Maps_on_HTC_One_M8">http://visihow.com/Share_Locations_from_Google_Maps_on_HTC_One_M8</a></p>

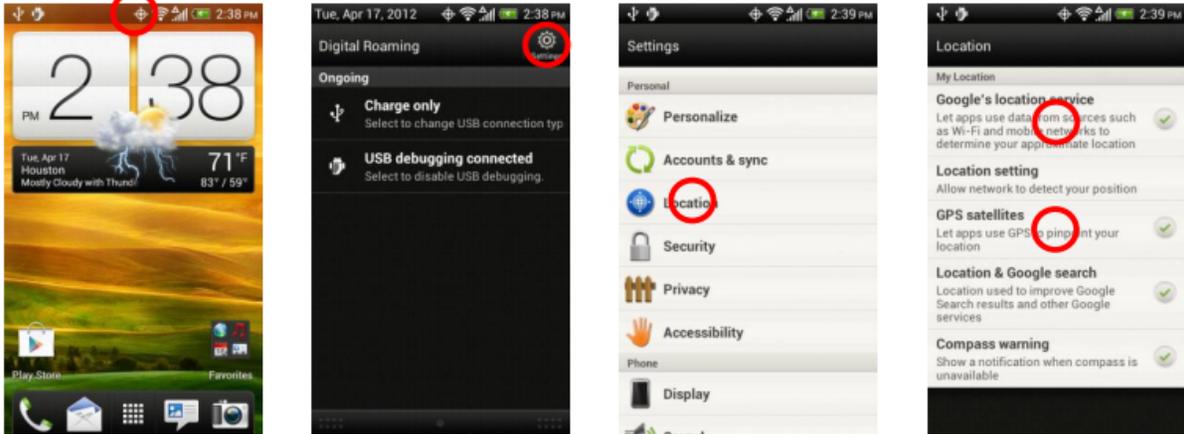
## Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products

US9408055B2	HTC
	<h3 data-bbox="520 264 1115 310">Manually switching locations</h3> <p data-bbox="520 358 1413 467">The HTC Sense Home widget automatically changes locations based on where you are. You can also manually change the location in the HTC Sense Home widget.</p> <p data-bbox="520 508 1402 613">For the HTC Sense Home widget to change locations automatically, you need to make sure that location services is turned on. See <a href="#">Turning location services on or off</a>.</p> <ol data-bbox="583 654 1392 776" style="list-style-type: none"><li>1. On your Home screen, slide right or left until you see the HTC Sense Home widget.</li><li>2. Tap , and then tap the location you want.</li></ol>  <p data-bbox="527 1101 1409 1125">HTC One (M8) - Manually switching locations - Support   HTC United States</p>

**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC
	<h2 data-bbox="548 269 1331 318">Setting your home and work locations</h2> <p data-bbox="548 367 1415 435">In the HTC Sense Home widget, set your home and work locations based on your address, Wi-Fi network, or both.</p> <p data-bbox="548 475 1425 621">You can associate multiple addresses and Wi-Fi networks to each of these locations. Using your set addresses or Wi-Fi networks, the HTC Sense Home widget will be able to determine where you are and display the appropriate apps.</p> <ol data-bbox="621 662 1457 1146" style="list-style-type: none"><li data-bbox="621 662 1423 727">1. On the Home screen, swipe right or left until you see the HTC Sense Home widget.</li><li data-bbox="621 751 972 784">2. Tap  &gt;  &gt; <b>Set locations</b>.</li><li data-bbox="621 816 1062 849">3. Choose the location you want to set.</li><li data-bbox="621 865 1457 1092">4. Tap  and do one of the following:<ul data-bbox="701 930 1457 1092" style="list-style-type: none"><li data-bbox="701 930 1444 995">• Tap <b>Address</b> and then enter your street address or select it on the map.</li><li data-bbox="701 1027 1457 1092">• Tap <b>Wi-Fi network</b> and select one or more Wi-Fi networks you want to associate with the location.</li></ul></li><li data-bbox="621 1117 1457 1146">5. When you've finished setting your home and work locations, press .</li></ol> <p data-bbox="520 1182 1503 1214">HTC One (M8) - Setting your home and work locations - Support   HTC United States</p>

## Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products

US9408055B2	HTC
	<p data-bbox="533 233 1304 272"><b>HTC One V™ – Google Location Service &amp; GPS</b></p> <p data-bbox="533 318 1717 423">Google Maps lets you track your current location, view real-time traffic situations, and receive detailed directions to your destination. It also provides a search tool where you can locate a place of interest or an address on a vector or aerial map, or view locations in street level.</p> <p data-bbox="533 469 905 496"><b>Turning on Location Services</b></p> <div data-bbox="533 509 1717 943"></div> <ol data-bbox="533 976 1717 1146" style="list-style-type: none"><li>1. From the Home Screen, slide the <b>Notifications</b> panel open.</li><li>2. In the top right corner, tap <b>Settings</b>.</li><li>3. Tap <b>Location</b>.</li><li>4. Make your selection by tapping <b>Google's location service</b>, <b>Use GPS satellites</b>, or both. <b>Note:</b> You will need to accept the location consent terms and conditions.</li></ol> <p data-bbox="512 1203 1906 1414"><b>Regarding Google Maps</b>, Google Latitude, Google Plus, Google Hangouts, Google Messages, and Android Messenger, the Accused Products are configured to allow a user of the first device to interact with the display, to select a user, contact, or device, and to select an action to be performed, such as: sending a message, initiating a call, initiating a data conference, sharing a location, stop sharing a location, block a user from location sharing, sending a location, requesting a location, or sending other data. When an action is specified, data is sent from the first device to the second device via a server. In an example, using Google</p>



**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC
	<p>Maps, a user can interact with the display to specify a location that does not correspond to the first or second devices. Alternatively, a user can share an ETA which includes a route, where the route is not the location of any of the first or second devices.</p> <p>Selection with Markers: <a href="https://developers.google.com/maps/documentation/android-api/marker">https://developers.google.com/maps/documentation/android-api/marker</a></p> <p>Queries with GeoTagging database: <a href="https://developers.google.com/android/reference/com/google/android/gms/location/places/GeoDataClient">https://developers.google.com/android/reference/com/google/android/gms/location/places/GeoDataClient</a></p>

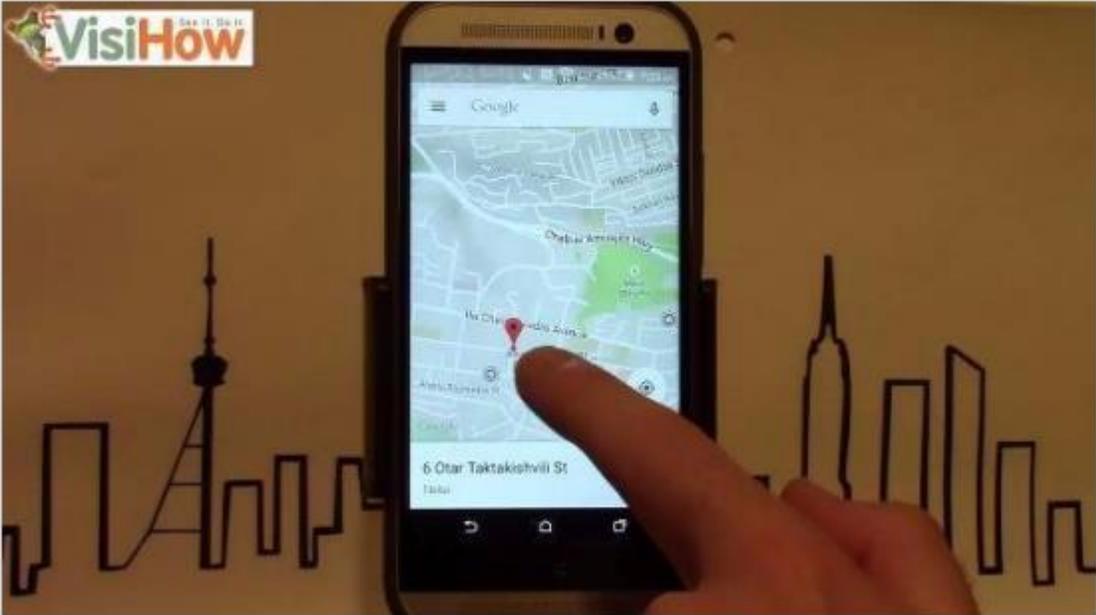
## Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products

US9408055B2	HTC
	<h3 data-bbox="541 245 1255 289">Embed a map or share a location</h3> <p data-bbox="541 313 1516 386">On your computer, you can embed a map, Street View image, driving directions, or search into your website or blog. On your computer, phone, or tablet, you can share a location or map with others over email, Google+, Facebook, Twitter, or text.</p> <p data-bbox="556 464 947 483"><a href="#">ANDROID</a> <a href="#">COMPUTER</a> <a href="#">IPHONE &amp; IPAD</a></p> <hr data-bbox="541 509 1528 513"/> <h3 data-bbox="541 565 848 589">Share a map or location</h3> <ol data-bbox="552 613 1230 769" style="list-style-type: none"><li>1. Open the Google Maps app .</li><li>2. Search for a place. Or find a place on the map then touch and hold to drop a pin.</li><li>3. At the bottom, tap the place's name or address.</li><li>4. Tap Share . If you don't see this, tap More  &gt; <b>Share</b>.</li><li>5. Select an app. It'll send a link that shows the place in Google Maps.</li></ol> <p data-bbox="522 805 1635 829"><a href="https://support.google.com/maps/answer/144361?co=GENIE.Platform%3DAndroid&amp;hl=en">https://support.google.com/maps/answer/144361?co=GENIE.Platform%3DAndroid&amp;hl=en</a></p> <h3 data-bbox="541 898 732 922">Share your E.T.A</h3> <p data-bbox="541 946 1461 966">After you start your drive, you can share your destination, estimated arrival time, and where you are on the route.</p> <ol data-bbox="546 987 1209 1167" style="list-style-type: none"><li>1. Open the Google Maps app .</li><li>2. Set a driving destination. <a href="#">Learn how to navigate to a place.</a></li><li>3. After you start navigation, tap <b>More</b>  &gt; <b>Share trip progress</b>.</li><li>4. Choose a person from the list.</li><li>5. Tap <b>Share</b>.</li><li>6. Location Sharing will stop when you reach your destination or stop navigating.</li></ol> <p data-bbox="541 1192 1083 1211">• To stop sharing before you arrive, tap <b>More</b>  &gt; <b>Stop sharing</b>.</p> <p data-bbox="512 1219 1696 1243"><a href="https://support.google.com/maps/answer/7326816?co=GENIE.Platform%3DAndroid&amp;hl=en">https://support.google.com/maps/answer/7326816?co=GENIE.Platform%3DAndroid&amp;hl=en</a></p> <p data-bbox="512 1292 1518 1317">Markers (adding location information to the link associated with the database):</p> <div data-bbox="512 1360 1913 1395" style="background-color: black; height: 20px;"></div>

**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC
	<pre>static final LatLng PERTH = new LatLng(-31.90, 115.86); Marker perth = mMap.addMarker(new MarkerOptions()     .position(PERTH)     .draggable(true));</pre>
<p>[41H] and based on the user input, adding the user-specified symbol to the interactive display at a position on the interactive map corresponding to the user-specified location, and transmitting the user-specified symbol and location to the second devices for addition of the user-specified symbol to respective interactive displays of the second devices at respective positions on respective interactive maps corresponding to the user-specified location.</p>	<p>HTC infringes directly and/or indirectly by performing, inducing others to perform, and/or contributing to the performance of [] based on the user input, adding the user-specified symbol to the interactive display at a position on the interactive map corresponding to the user-specified location, and transmitting the user-specified symbol and location to the second devices for addition of the user-specified symbol to respective interactive displays of the second devices at respective positions on respective interactive maps corresponding to the user-specified location. See claims 1[H] and 28[H], which are incorporated herein by reference in their entirety.</p> <p>A user can interact with the display to specify a location that does not correspond to the first or second devices. A user can drop a symbol pin on the specified location. A user can then share that location and transmit the location to one or more second devices using Android Messages, Google Hangouts, or another application.</p> <p>Alternatively, a user can share an ETA which includes a route, where the route is not the location of any of the first or second devices. Again, this route can be shared with users over Android Messages, Google Hangouts, or another application.</p> <p><b>Share Locations from Google Maps on HTC One M8</b></p>

**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC
	<p data-bbox="531 233 827 256"><b>Place a pin on the map.</b></p> <p data-bbox="531 272 1640 380">We can do this anywhere in the entire world, anywhere, on any map that we find in the Google Maps application. We are going to choose a location, press down on the screen, and hold down until we can see that a red pin has been dropped.</p>  <p data-bbox="512 1105 743 1128">Placing a Marker:</p> <p data-bbox="512 1143 1430 1166"><a href="https://developers.google.com/maps/documentation/android-api/marker">https://developers.google.com/maps/documentation/android-api/marker</a></p> <p data-bbox="512 1214 1083 1237">based on queries with GeoTagging database:</p> <p data-bbox="512 1252 1860 1274"><a href="https://developers.google.com/android/reference/com/google/android/gms/location/places/GeoDataClient">https://developers.google.com/android/reference/com/google/android/gms/location/places/GeoDataClient</a></p>

**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC
	<p><b>Embed a map or share a location</b></p> <p>On your computer, you can embed a map, Street View image, driving directions, or search into your website or blog. On your computer, phone, or tablet, you can share a location or map with others over email, Google+, Facebook, Twitter, or text.</p> <p><b>ANDROID</b> COMPUTER IPHONE &amp; IPAD</p> <hr/> <p><b>Share a map or location</b></p> <ol style="list-style-type: none"> <li>1. Open the Google Maps app .</li> <li>2. Search for a place. Or find a place on the map then touch and hold to drop a pin.</li> <li>3. At the bottom, tap the place's name or address.</li> <li>4. Tap Share . If you don't see this, tap More  &gt; <b>Share</b>.</li> <li>5. Select an app. It'll send a link that shows the place in Google Maps.</li> </ol> <p><a href="https://support.google.com/maps/answer/144361?co=GENIE.Platform%3DAndroid&amp;hl=en">https://support.google.com/maps/answer/144361?co=GENIE.Platform%3DAndroid&amp;hl=en</a></p> <p><b>Share your E.T.A</b></p> <p>After you start your drive, you can share your destination, estimated arrival time, and where you are on the route.</p> <ol style="list-style-type: none"> <li>1. Open the Google Maps app .</li> <li>2. Set a driving destination. <a href="#">Learn how to navigate to a place.</a></li> <li>3. After you start navigation, tap <b>More</b>  &gt; <b>Share trip progress</b>.</li> <li>4. Choose a person from the list.</li> <li>5. Tap <b>Share</b>.</li> <li>6. Location Sharing will stop when you reach your destination or stop navigating.</li> </ol> <ul style="list-style-type: none"> <li>• To stop sharing before you arrive, tap <b>More</b>  &gt; <b>Stop sharing</b>.</li> </ul>
<p>42[A]. The storage device of claim 41 wherein the operations</p>	<p>HTC infringes directly and/or indirectly by performing, inducing others to perform, and/or contributing to the performance of wherein the operations further comprise: presenting another symbol on the interactive map corresponding to a fixed location and associated with a telephone number. See claims 2[A] and 41,</p>

**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC
<p>further comprise: presenting another symbol on the interactive map corresponding to a fixed location and associated with a telephone number;</p>	<p>which are incorporated by reference in their entirety.</p>
<p>[42B] and receiving user selection of the other symbol and, based thereon, initiating a telephone call to the telephone number associated with the symbol.</p>	<p>HTC infringes directly and/or indirectly by performing, inducing others to perform, and/or contributing to the performance of [] receiving user selection of the other symbol and, based thereon, initiating a telephone call to the telephone number associated with the symbol. See claims 2[B] and 41, which are incorporated by reference in their entirety.</p>
<p>43. The storage device of claim 41 wherein the data comprises a text message, an image, a video, or a command to cause the second devices corresponding to the selected symbols to convert text to speech.</p>	<p>HTC infringes directly and/or indirectly by performing, inducing others to perform, and/or contributing to the performance of wherein the data comprises a text message, an image, a video, or a command to cause the second devices corresponding to the selected symbols to convert text to speech. See claims 3 and 41, which are incorporated by reference in their entirety.</p>
<p>44[A]. The storage device of claim 41 wherein: the SMS messages include an Internet Protocol (IP) address of the first</p>	<p>HTC infringes directly and/or indirectly by performing, inducing others to perform, and/or contributing to the performance of wherein: the SMS messages include an Internet Protocol (IP) address of the first device. See claims 4[A] and 41, which are incorporated by reference in their entirety.</p>

**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC
device;	
[44B] and the IP-based responses include respective IP addresses of the second devices.	HTC infringes directly and/or indirectly by performing, inducing others to perform, and/or contributing to the performance of [] the IP-based responses include respective IP addresses of the second devices. See claims 4[B] and 41, which are incorporated by reference in their entirety.
45. The storage device of claim 41 wherein the operations further comprise: transmitting location information including an updated location of the first device to the second devices based on displacement of the first device by at least a predetermined distance relative to a previous location of the first device, passage of at least a predetermined time interval since transmitting information including a location of the first device, or a combination of the displacement of the first device and the passage of time.	HTC infringes directly and/or indirectly by performing, inducing others to perform, and/or contributing to the performance of wherein the operations further comprise: transmitting location information including an updated location of the first device to the second devices based on displacement of the first device by at least a predetermined distance relative to a previous location of the first device, passage of at least a predetermined time interval since transmitting information including a location of the first device, or a combination of the displacement of the first device and the passage of time. See claims 5 and 41, which are incorporated by reference in their entirety.
46[A]. The storage	HTC infringes directly and/or indirectly by performing, inducing others to perform, and/or contributing to

**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC
device of claim 41 wherein the operations further comprise: receiving second user selection of one or more of the symbols corresponding to one or more of the second devices;	the performance of wherein the operations further comprise: receiving second user selection of one or more of the symbols corresponding to one or more of the second devices. See claims 6[A] and 41, which are incorporated by reference in their entirety.
[46B] and receiving user input assigning the one or more second devices corresponding to the second selected one or more symbols to a sub-net.	HTC infringes directly and/or indirectly by performing, inducing others to perform, and/or contributing to the performance of [] receiving user input assigning the one or more second devices corresponding to the second selected one or more symbols to a sub-net. See claims 6[B] and 41, which are incorporated by reference in their entirety.
47[A]. The storage device of claim 46 wherein the operations further comprise: receiving user selection of the sub-net;	HTC infringes directly and/or indirectly by performing, inducing others to perform, and/or contributing to the performance of wherein the operations further comprise: receiving user selection of the sub-net. See claims 7[A], 41, and 46, which are incorporated herein by reference in their entirety.
[47B] and establishing a conference among the one or more second devices of the sub-net for sharing voice, text, photographs, or video communications.	HTC infringes directly and/or indirectly by performing, inducing others to perform, and/or contributing to the performance of [] establishing a conference among the one or more second devices of the sub-net for sharing voice, text, photographs, or video communications. See claims 7[B], 41, and 46, which are incorporated herein by reference in their entirety.



**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC
48. The storage device of claim 41, wherein the first device is a cellular phone or a personal digital assistant (PDA).	HTC infringes directly and/or indirectly by performing, inducing others to perform, and/or contributing to the performance of wherein the first device is a cellular phone or a personal digital assistant (PDA). See claims 8 and 41, which are incorporated by reference in their entirety.
49. The storage device of claim 41, wherein the operations further comprise: identifying second user interaction with the interactive display selecting at least one of the user-selectable symbols corresponding to at least one of the second devices and user interaction with the display specifying an action and, based thereon, initiating a phone call or phone conference with the at least one second device.	HTC infringes directly and/or indirectly by performing, inducing others to perform, and/or contributing to the performance of wherein the operations further comprise: identifying second user interaction with the interactive display selecting at least one of the user-selectable symbols corresponding to at least one of the second devices and user interaction with the display specifying an action and, based thereon, initiating a phone call or phone conference with the at least one second device. See claims 9 and 41, which are incorporated by reference in their entirety.
50. The storage device of claim 43, wherein the video comprises a video clip or a video transmission.	HTC infringes directly and/or indirectly by performing, inducing others to perform, and/or contributing to the performance of wherein the video comprises a video clip or a video transmission. See claims 10, 41, and 43, which are incorporated herein by reference in their entirety.
51[A]. The storage	HTC infringes directly and/or indirectly by performing, inducing others to perform, and/or contributing to

**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC
<p>device of claim 46, wherein the operations further comprise: receiving user selection of the sub-net;</p>	<p>the performance of wherein the operations further comprise: receiving user selection of the sub-net. See claims 11[A], 41, and 46, which are incorporated herein by reference in their entirety.</p>
<p>[51B] and causing the one or more second devices of the sub-net to place a call, make a verbal announcement, convert text to speech, vibrate, or increase sound levels.</p>	<p>HTC infringes directly and/or indirectly by performing, inducing others to perform, and/or contributing to the performance of [] causing the one or more second devices of the sub-net to place a call, make a verbal announcement, convert text to speech, vibrate, or increase sound levels. See claims 11[B], 41, and 46, which are incorporated herein by reference in their entirety.</p>
<p>52. The storage device of claim 41, wherein the data sent to the one or more second devices causes at least one of the second devices to play an audio message announcing an emergency.</p>	<p>HTC infringes directly and/or indirectly by performing, inducing others to perform, and/or contributing to the performance of wherein the data sent to the one or more second devices causes at least one of the second devices to play an audio message announcing an emergency. See claims 39 and 41, which are incorporated by reference in their entirety.</p>
<p>53. The storage device of claim 41, wherein the data sent to the one or more second devices causes at least one of the second devices to place a phone call to the first</p>	<p>HTC infringes directly and/or indirectly by performing, inducing others to perform, and/or contributing to the performance of wherein the data sent to the one or more second devices causes at least one of the second devices to place a phone call to the first device. See claims 40 and 41, which are incorporated by reference in their entirety.</p>

**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC
device.	
<p>54[P]. A method comprising: performing by a first device:</p>	<p>HTC infringes directly and/or indirectly by performing, inducing others to perform, and/or contributing to the performance of each step of this method as set forth below. See claims 1[P], 28[P], and 41[P], which are incorporated herein by reference in their entirety.</p> <p>The Accused Products meet the claim limitations by providing device-location tracking features such as those features described below. For example, the Accused Products meet the claim limitations because they are pre-installed with Android mobile operating systems containing code for providing device-location tracking features as provided in the claims limitations herein. For example, the Accused Products run applications and/or software that run within the Android mobile operating system and that use components of the Android mobile operating system to provide device-location tracking features. Upon information and belief, in addition to the components and features of the Android mobile operating system itself, the following applications and/or software run within the Android mobile operating system and use components of the Android mobile operating system to provide device-location tracking features: Google Latitude, Google Plus, Google Hangouts (including Allo and Duo), Google Maps, Google Chrome, Google Messages, and Android Messenger.</p> <p><b><u>Google Maps Share Location</u></b></p> <p>Share Location is currently included as a standard feature on the Accused Devices operating as a feature of Google Maps. Google Maps is a pre-installed software application in Android OS. The Accused Devices have included the Share Location functionalities since 2009 as part of Google Latitude, which was an opt-in feature for Google Maps on Android OS-based mobile devices, such as the Accused Products. Share Location functionalities were briefly shifted from Latitude for Google Maps to Google Plus and Google Hangouts, until reappearing as a standard feature in Google Maps. Upon information and belief, the Share Location method also uses and/or works in conjunction with functionalities associated with Google Maps, Google Messages, Android Messenger, Location Access, and other features, which are pre-installed on the Accused Products. For the purposes of these contentions, AGIS sets forth Google Maps' Share Location feature of the Accused Products as representative of this exemplary software. AGIS reserves the right to supplement these contentions to the extent that defendant requires additional information in accordance with P.R. 3-1 and for any other reason.</p>

**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC
	<p data-bbox="512 235 1904 412"><i>See, e.g.,</i> <a href="https://techcrunch.com/2017/03/22/google-maps-now-lets-you-share-your-location-with-friends-and-family-for-a-specific-period-of-time/">https://techcrunch.com/2017/03/22/google-maps-now-lets-you-share-your-location-with-friends-and-family-for-a-specific-period-of-time/</a>; <a href="https://googleblog.blogspot.com/2009/02/see-where-your-friends-are-with-google.html">https://googleblog.blogspot.com/2009/02/see-where-your-friends-are-with-google.html</a>; <a href="https://googleblog.blogspot.jp/2013/07/a-new-google-maps-app-for-smartphones.html">https://googleblog.blogspot.jp/2013/07/a-new-google-maps-app-for-smartphones.html</a>; <a href="http://googleplusproject.blogspot.com/2013/05/google-for-android-42.html">http://googleplusproject.blogspot.com/2013/05/google-for-android-42.html</a>; <a href="https://googleblog.blogspot.com/2013/10/google-hangouts-and-photos-save-some.html">https://googleblog.blogspot.com/2013/10/google-hangouts-and-photos-save-some.html</a></p> <p data-bbox="550 483 1205 516"><b>Control within reach, even when your device isn't</b></p> <p data-bbox="550 558 1591 776">One of the biggest security risks you're likely to face is simply losing your phone. To help in these times of need, we're launching <a href="#">Find My Device</a> as part of Google Play Protect. With Find My Device you can locate, ring, lock and erase your Android devices—phones, tablets, and even watches. This feature is built in and enabled on all devices; visit <a href="http://android.com/find">android.com/find</a> or check out <a href="#">the app</a>.</p> <p data-bbox="512 818 1444 850"><i>See, e.g.,</i> <a href="https://www.blog.google/products/android/google-play-protect/">https://www.blog.google/products/android/google-play-protect/</a></p> <div data-bbox="512 964 1495 1338"><h3 data-bbox="537 974 1310 1071">Find your device using Android Device Manager</h3><p data-bbox="537 1091 1491 1140">If you've lost a device, you can use Android Device Manager to find its approximate location on a map and when it was last used. When Android Device Manager locates your device, that device will get a notification.</p><p data-bbox="537 1162 1482 1237"><b>Before you can use Android Device Manager to locate your device:</b> Your device's <a href="#">location access</a> need to be turned on and be signed in to your Google Account. Android Device Manager won't work for devices that are turned off or that don't have a mobile data or Wi-Fi connection.</p><p data-bbox="537 1260 1457 1279"><b>Tip:</b> if you've linked your phone to Google, you can locate or ring it by searching for <a href="#">find my phone</a> on <a href="http://google.com">google.com</a>.</p><p data-bbox="520 1308 1192 1338"><b><a href="https://support.google.com/pixelphone/answer/6160491">https://support.google.com/pixelphone/answer/6160491</a></b></p></div>

# Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products

US9408055B2	HTC
	<h2 data-bbox="520 233 911 266">Link your phone to Google</h2> <p data-bbox="520 282 1201 318">You can connect your Android phone to Google, which lets you send information from your computer to your phone. For example, you can send directions you searched for on your computer to Google Maps on your phone.</p> <h3 data-bbox="520 342 737 362">Link your Android phone</h3> <p data-bbox="520 386 737 406"><b>Step 1: Update the Google app</b></p> <ol data-bbox="520 410 884 451" style="list-style-type: none"><li data-bbox="520 410 884 430">1. On your phone, go to the <a href="#">Google app page on the Play Store</a>.</li><li data-bbox="520 435 611 451">2. Tap <b>Update</b>.</li></ol> <p data-bbox="520 472 722 492"><b>Step 2: Turn on Google Now</b></p> <ol data-bbox="520 496 852 581" style="list-style-type: none"><li data-bbox="520 496 768 516">1. On your phone, open the Google app .</li><li data-bbox="520 521 852 540">2. At the top left, tap Menu  &gt; <b>Settings</b> &gt; <b>Now cards</b>.</li><li data-bbox="520 545 659 565">3. Turn on <b>Show cards</b>.</li><li data-bbox="520 570 699 581">4. Turn on <b>Show notifications</b>.</li></ol> <p data-bbox="520 602 768 621"><b>Step 3: Turn on Web &amp; App Activity</b></p> <ol data-bbox="520 626 737 667" style="list-style-type: none"><li data-bbox="520 626 716 646">1. Visit the <a href="#">Account History page</a>.</li><li data-bbox="520 651 737 667">2. Make sure the switch is on (green).</li></ol> <p data-bbox="520 688 737 708"><b>Step 4: Sign in to your browser</b></p> <ol data-bbox="520 712 978 842" style="list-style-type: none"><li data-bbox="520 712 768 732">1. On your phone, open the Google app .</li><li data-bbox="520 737 722 756">2. At the top left, tap the Menu .</li><li data-bbox="520 761 942 781">3. At the top left, you'll see the email address you use for the Google app.</li><li data-bbox="520 786 793 805">4. Visit <a href="http://www.google.com">www.google.com</a>  on your computer.</li><li data-bbox="520 810 978 829">5. If you aren't signed in already, click <b>Sign in</b> in the top right corner of the page.</li><li data-bbox="520 834 890 842">6. Sign in using the Google Account you use for the Google app.</li></ol> <p data-bbox="520 863 800 883"><b>Step 5: Send information to your phone</b></p> <ol data-bbox="520 888 1209 945" style="list-style-type: none"><li data-bbox="520 888 1041 907">1. Do one of the searches below, like <b>note to self</b>, or <b>send directions to my phone</b>.</li><li data-bbox="520 912 1209 945">2. If a box doesn't pop up with the option to send information to your phone, try refreshing the page. If you just turned on Google Now, it may take a few minutes for the box to show up</li></ol> <p data-bbox="520 950 1215 980"><a href="https://support.google.com/websearch/answer/6128427">https://support.google.com/websearch/answer/6128427</a></p>

**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC
	<p data-bbox="520 237 1010 261"><b>What you can do once your phone is linked</b></p> <hr/> <p data-bbox="537 297 667 318"><a href="#">Find my phone</a> </p> <p data-bbox="562 337 1041 358">You can get the current location of your phone if you can't find it.</p> <ol data-bbox="573 375 1388 472" style="list-style-type: none"><li>1. On your computer's browser, search on <a href="http://www.google.com">www.google.com</a>  for <b>find my phone</b>.</li><li>2. If your phone is turned on and connected to the Internet, you'll see your phone's location.</li><li>3. If your phone's location is unavailable, you can still make it ring for 5 minutes on full volume by clicking <b>Ring</b>. You can stop the ringing from your phone when you find it.</li></ol> <p data-bbox="562 488 1388 529"><b>Tip:</b> You can also find your missing phone using the <a href="#">Android Device manager</a>  which lets you find your device or remotely ring, lock, or erase it.</p> <hr/> <p data-bbox="537 589 785 610"><a href="#">Send directions to my phone</a> </p> <p data-bbox="562 630 1388 670">Once you've looked up directions on your computer, you can send them to your phone so you have them on your trip.</p> <ol data-bbox="573 686 1314 789" style="list-style-type: none"><li>1. On your computer's browser, search on <a href="http://www.google.com">www.google.com</a>  for <b>send directions to my phone</b>.</li><li>2. Enter in your destination.</li><li>3. Click <b>Send directions to your phone</b>.</li><li>4. You'll get a notification on your phone. Tap to navigate to your destination using Google Maps.</li></ol> <hr/> <p data-bbox="537 849 751 870"><a href="#">Send a note to my phone</a> </p> <ol data-bbox="573 889 1367 992" style="list-style-type: none"><li>1. On your computer's browser, search on <a href="http://www.google.com">www.google.com</a>  for <b>send a note to my phone</b>.</li><li>2. Type your note in the box.</li><li>3. Click <b>Send note to your phone</b>.</li><li>4. You'll get a notification on your phone with your note that you can either save to one of your apps or copy.</li></ol> <hr/> <p data-bbox="537 1052 646 1073"><a href="#">Set an alarm</a> </p> <ol data-bbox="573 1092 1178 1195" style="list-style-type: none"><li>1. On your computer's browser, search on <a href="http://www.google.com">www.google.com</a>  for <b>set an alarm</b>.</li><li>2. Choose the time you want the alarm to go off.</li><li>3. Click <b>Set an alarm on your phone</b>.</li><li>4. An alarm will now be set on your phone's Clock app.</li></ol> <hr/> <p data-bbox="537 1255 667 1276"><a href="#">Set a reminder</a> </p> <ol data-bbox="573 1295 1335 1365" style="list-style-type: none"><li>1. On your computer's browser, search on <a href="http://www.google.com">www.google.com</a>  for <b>set an reminder</b>.</li><li>2. Type what you want to be reminded about, and either when or where you want the reminder to go off.</li><li>3. Click <b>Remind me on my devices</b>.</li></ol> <hr/> <p data-bbox="512 1373 1220 1398"><a href="https://support.google.com/websearch/answer/6128427">https://support.google.com/websearch/answer/6128427</a></p>

**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC
	<p data-bbox="548 233 1325 277"><b>Share your location using Google Maps</b></p> <p data-bbox="548 297 1465 342">You can't share your location in Google+ anymore. If you used to share your location in Google+ and want to keep sharing it, you'll need to share it again in Google Maps.</p> <p data-bbox="520 358 1633 386"><a href="https://support.google.com/plus/answer/3302509?hl=en&amp;co=GENIE.Platform%3DAndroid&amp;oco=1">https://support.google.com/plus/answer/3302509?hl=en&amp;co=GENIE.Platform%3DAndroid&amp;oco=1</a></p> <p data-bbox="533 402 625 430"><b>Location</b></p> <p data-bbox="533 443 1409 537">Turn on location service, your phone determines your approximate location using Wi-Fi and mobile networks. When you select this option, you're asked whether you consent to allowing Google to use your location when providing these services.</p> <ul data-bbox="562 548 1367 760" style="list-style-type: none"><li>• <b>Mode</b> – Sets the how your current location information is determined.</li><li>• <b>Recent Location Request</b> – Displays applications and services that have recently requested your location information.</li><li>• <b>Camera</b> – Checkmark to tag photos or videos with their locations.</li><li>• <b>Google Location History</b> – Allows you to view and manage your Google location history.</li></ul> <p data-bbox="533 776 716 803"><b>Accounts &amp; sync</b></p> <p data-bbox="533 816 1402 943">Use the Accounts &amp; sync settings menu to add, remove, and manage your Google and other supported accounts. You also use these settings to control how and whether all applications send, receive, and sync data on their own schedules and whether all applications can synchronize user data automatically.</p> <p data-bbox="533 959 1373 1053">Gmail™, Calendar, and other applications may also have their own settings to control how they synchronize data; see the sections on those applications for details. Touch <b>Add account</b> to add new account.</p> <p data-bbox="533 1057 653 1073">-</p>

**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC
	 <p><a href="http://googlemobile.blogspot.com/2009/02/locate-your-friends-in-real-time-with.html">http://googlemobile.blogspot.com/2009/02/locate-your-friends-in-real-time-with.html</a></p> <p>Google's location-sharing feature also appeared in Google+, Google Trust Contacts, and Google Hangouts services until its current integration in Google Maps.</p> <p>HTC makes, uses, sells, and otherwise provides this first device by making, using, selling, and importing Android devices such as HTC mobile devices, HTC tablets, and HTC Smartwatches as well as by providing its servers or using third party servers (e.g., Google servers) for use with Android devices to enable features such as Maps. Below are example HTC Android devices that perform each step of this method as set forth below.</p>



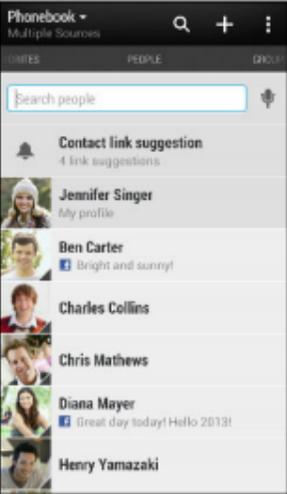
# Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products

US9408055B2	HTC
	<p>Sort by: <a href="#">Popularity</a> <a href="#">Date</a> <a href="#">Price</a></p> <p>HTC LG Samsung Motorola Fly Sony-Ericsson Apple Nokia Mobiado Vertu BenQ-Siemens Sagem Alcatel Philips</p> <p><a href="#">All brands</a></p> <div data-bbox="814 300 1050 479"><p><b>HTC Desire 816</b></p><p>Mobile phone 2014 year Touchscreen: 720 x 1280 5.5 inch. Android 4.4</p></div> <div data-bbox="535 576 777 755"><p><b>HTC One M8</b></p><p>Mobile phone 2014 year Touchscreen: 1080 x 1920 5 inch. Android 4.4</p></div> <div data-bbox="814 576 1050 755"><p><b>HTC Desire 300</b></p><p>Mobile phone 2013 year Touchscreen: 480 x 800 4.3 inch. Android 4.2</p></div> <div data-bbox="535 852 777 1031"><p><b>HTC Desire 601</b></p><p>Mobile phone 2013 year Touchscreen: 540 x 960 4.5 inch. Android 4.4</p></div> <div data-bbox="814 852 1050 1031"><p><b>HTC Desire 700</b></p><p>Mobile phone 2013 year Touchscreen: 540 x 960 5 inch. Android 4.2</p></div> <div data-bbox="535 1128 777 1234"><p><b>HTC Desire 400 Dual Sim</b></p><p>Mobile phone 2013 year Touchscreen: 480 x 800 4.3 inch. Android 4.1</p></div> <div data-bbox="814 1128 1050 1234"><p><b>HTC One Max</b></p><p>Mobile phone 2013 year Touchscreen: 1080 x 1920 5.9 inch. Android 4.3</p></div> <p><a href="https://mob.org/phone/htc/page_3/sort_date_down/">https://mob.org/phone/htc/page_3/sort_date_down/</a></p>

**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC
<p>54[A] obtaining contact information of a plurality of second devices, wherein the contact information comprises respective telephone numbers of the second devices;</p>	<p>HTC infringes directly and/or indirectly by performing, inducing others to perform, and/or contributing to the performance of: obtaining contact information of a plurality of second devices, wherein the contact information comprises respective telephone numbers of the second devices. See claims 1[A], 28[A], and 41[A], which are incorporated herein by reference in their entirety.</p> <p>For example, the Accused products include a contacts app to access contact information for second users using respective second devices.</p>

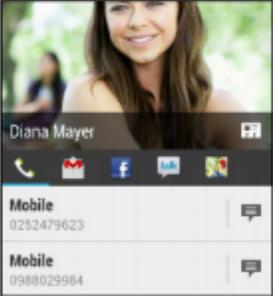
**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC
	<h2 data-bbox="562 233 936 272">Your contacts list</h2> <p data-bbox="562 310 1560 402">The Contacts app lists all contacts you've stored on HTC One and from online accounts you're logged in to. Use the Contacts app to easily manage communications with people that matter to you.</p> <ol data-bbox="590 431 909 456" style="list-style-type: none"><li data-bbox="590 431 909 456">1. Open the Contacts app.</li></ol>  <p data-bbox="590 997 982 1021">2. On your contacts list, you can:</p> <ul data-bbox="653 1049 1556 1289" style="list-style-type: none"><li data-bbox="653 1049 1289 1073">▪ View your profile and edit your contact information.</li><li data-bbox="653 1084 1087 1109">▪ Create, edit, find, or send contacts.</li><li data-bbox="653 1120 909 1144">▪ See status updates.</li><li data-bbox="653 1156 1499 1180">▪ Tap a contact photo to find ways to quickly connect with the contact.</li><li data-bbox="653 1192 1465 1216">▪ See a notification icon when a contact has sent you new messages.</li><li data-bbox="653 1227 1556 1289">▪ Check out who's online in Google Talk™. Online status icons are displayed if you're signed in to Google Talk.</li></ul> <p data-bbox="520 1317 1566 1386"> To sort your contacts by their first or last name, tap  &gt; Settings &gt; Sort contact list.</p>

## Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products

US9408055B2	HTC
	<p data-bbox="531 248 926 277"><b>Filtering your contacts list</b></p> <p data-bbox="531 305 1572 331">When your contacts list gets long, you can choose which contact accounts to show.</p> <ol data-bbox="562 362 1440 488" style="list-style-type: none"><li data-bbox="562 362 957 388">1. On the Contacts tab, tap ▼.</li><li data-bbox="562 410 1440 436">2. Choose the accounts that contain the contacts you want to display.</li><li data-bbox="562 459 716 488">3. Press &lt;.</li></ol> <p data-bbox="531 532 758 561"><b>Finding people</b></p> <p data-bbox="531 589 1514 647">Search for contacts stored on HTC One, your company directory if you have an Exchange ActiveSync account, or social networks you've signed into.</p> <ol data-bbox="562 683 1604 1097" style="list-style-type: none"><li data-bbox="562 683 900 709">1. Open the Contacts app.</li><li data-bbox="562 732 1604 1097">2. On the Contacts tab, you can:<ul data-bbox="627 784 1604 1097" style="list-style-type: none"><li data-bbox="627 784 1604 846">▪ Find people in your contacts list. Tap the Search people box, and then enter the first few letters of the contact name.</li><li data-bbox="627 859 1604 956">▪ Find people on your company directory. Tap the Search people box, enter the first few letters of the contact name, and then tap Search contacts in your Company Directory.</li><li data-bbox="627 969 1604 1097">▪ Search for people you know on your social networks. Tap ☰ &gt; Settings &gt; Find people you know on, and then select the social networks you're signed in to. The Contacts app then uploads your contacts to the selected social networks to help you find friends.</li></ul></li></ol> <p data-bbox="531 1141 1587 1239">) Aside from searching for a contact by name, you can search using a contact's email address or company name. On the Contacts tab, tap ☰ &gt; Settings &gt; Search contacts by, and then choose a search criteria.</p>

## Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products

US9408055B2	HTC
	<p data-bbox="569 240 1283 282"><b>Getting in touch with a contact</b></p> <ol data-bbox="600 329 1625 440" style="list-style-type: none"><li data-bbox="600 329 940 354">1. Open the Contacts app.</li><li data-bbox="600 378 1625 440">2. Tap a contact's photo (not the name), and then choose how you want to get in touch with that contact.</li></ol>  <p data-bbox="527 802 1646 883"> For more ways of getting in touch with your contact, tap an icon below the contact photo.</p>

**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC
	<p>In another example, the Accused products run Android Messages and Google Hangouts which both access contact information for second users using respective second devices.</p> <h2 data-bbox="527 386 1016 440">Contacts Provider</h2> <p data-bbox="527 475 1472 735">The Contacts Provider is a powerful and flexible Android component that manages the device's central repository of data about people. The Contacts Provider is the source of data you see in the device's contacts application, and you can also access its data in your own application and transfer data between the device and online services. The provider accommodates a wide range of data sources and tries to manage as much data as possible for each person, with the result that its organization is complex. Because of this, the provider's API includes an extensive set of contract classes and interfaces that facilitate both data retrieval and modification.</p> <p data-bbox="527 771 835 792">This guide describes the following:</p> <ul data-bbox="527 821 1373 995" style="list-style-type: none"><li>• The basic provider structure.</li><li>• How to retrieve data from the provider.</li><li>• How to modify data in the provider.</li><li>• How to write a sync adapter for synchronizing data from your server to the Contacts Provider.</li></ul> <p data-bbox="512 1011 1486 1037"><a href="https://developer.android.com/guide/topics/providers/contacts-provider.html">https://developer.android.com/guide/topics/providers/contacts-provider.html</a></p>

**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC
	<p><b>Overview</b></p> <p>ContactsContract defines an extensible database of contact-related information. Contact information is stored in a three-tier data model:</p> <ul style="list-style-type: none"> <li>• A row in the <code>ContactsContract.Data</code> table can store any kind of personal data, such as a phone number or email addresses. The set of data kinds that can be stored in this table is open-ended. There is a predefined set of common kinds, but any application can add its own data kinds.</li> <li>• A row in the <code>ContactsContract.RawContacts</code> table represents a set of data describing a person and associated with a single account (for example, one of the user's Gmail accounts).</li> <li>• A row in the <code>ContactsContract.Contacts</code> table represents an aggregate of one or more RawContacts presumably describing the same person. When data in or associated with the RawContacts table is changed, the affected aggregate contacts are updated as necessary.</li> </ul> <p>Other tables include:</p> <ul style="list-style-type: none"> <li>• <code>ContactsContract.Groups</code>, which contains information about raw contact groups such as Gmail contact groups. The current API does not support the notion of groups spanning multiple accounts.</li> <li>• <code>ContactsContract.StatusUpdates</code>, which contains social status updates including IM availability.</li> <li>• <code>ContactsContract.AggregationExceptions</code>, which is used for manual aggregation and disaggregation of raw contacts</li> <li>• <code>ContactsContract.Settings</code>, which contains visibility and sync settings for accounts and groups.</li> <li>• <code>ContactsContract.SyncState</code>, which contains free-form data maintained on behalf of sync adapters</li> <li>• <code>ContactsContract.PhoneLookup</code>, which is used for quick caller-ID lookup</li> </ul> <p><a href="https://developer.android.com/reference/android/provider/ContactsContract.html">https://developer.android.com/reference/android/provider/ContactsContract.html</a></p> <p><b>Data</b></p> <p>As noted previously, the data for a raw contact is stored in a <code>ContactsContract.Data</code> row that is linked to the raw contact's <code>_ID</code> value. This allows a single raw contact to have multiple instances of the same type of data such as email addresses or phone numbers. For example, if "Thomas Higginson" for <code>emilyd@gmail.com</code> (the raw contact row for Thomas Higginson associated with the Google account <code>emilyd@gmail.com</code>) has a home email address of <code>thigg@gmail.com</code> and a work email address of <code>thomas.higginson@gmail.com</code>, the Contacts Provider stores the two email address rows and links them both to the raw contact.</p> <p>Notice that different types of data are stored in this single table. Display name, phone number, email, postal address, photo, and website detail rows are all found in the <code>ContactsContract.Data</code> table. To help manage this, the <code>ContactsContract.Data</code> table has some columns with descriptive names, and others with generic names. The contents of a descriptive-name column have the same meaning regardless of the type of data in the row, while the contents of a generic-name column have different meanings depending on the type of data.</p> <p><a href="https://developer.android.com/guide/topics/providers/contacts-provider.html">https://developer.android.com/guide/topics/providers/contacts-provider.html</a></p>

**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC														
	<table border="1"> <thead> <tr> <th>Task</th> <th>Action</th> <th>Data</th> <th>MIME type</th> <th>Notes</th> </tr> </thead> <tbody> <tr> <td>Pick a contact from a list</td> <td><a href="#">ACTION_PICK</a></td> <td>                     One of:                     <ul style="list-style-type: none"> <li><a href="#">Contacts.CONTENT_URI</a>, which displays a list of contacts.</li> <li><a href="#">Phone.CONTENT_URI</a>, which displays a list of phone numbers for a raw contact.</li> <li><a href="#">StructuredPostal.CONTENT_URI</a>, which displays a list of postal addresses for a raw contact.</li> <li><a href="#">Email.CONTENT_URI</a>, which displays a list of email addresses for a raw contact.</li> </ul> </td> <td>Not used</td> <td>                     Displays a list of raw contacts or a list of data from a raw contact, depending on the content URI type you supply. Call <a href="#">startActivityForResult()</a>, which returns the content URI of the selected row. The form of the URI is the table's content URI with the row's <a href="#">LOOKUP_ID</a> appended to it. The device's contacts app delegates read and write permissions to this content URI for the life of your activity. See the <a href="#">Content Provider Basics</a> guide for more details.                 </td> </tr> </tbody> </table>	Task	Action	Data	MIME type	Notes	Pick a contact from a list	<a href="#">ACTION_PICK</a>	One of: <ul style="list-style-type: none"> <li><a href="#">Contacts.CONTENT_URI</a>, which displays a list of contacts.</li> <li><a href="#">Phone.CONTENT_URI</a>, which displays a list of phone numbers for a raw contact.</li> <li><a href="#">StructuredPostal.CONTENT_URI</a>, which displays a list of postal addresses for a raw contact.</li> <li><a href="#">Email.CONTENT_URI</a>, which displays a list of email addresses for a raw contact.</li> </ul>	Not used	Displays a list of raw contacts or a list of data from a raw contact, depending on the content URI type you supply. Call <a href="#">startActivityForResult()</a> , which returns the content URI of the selected row. The form of the URI is the table's content URI with the row's <a href="#">LOOKUP_ID</a> appended to it. The device's contacts app delegates read and write permissions to this content URI for the life of your activity. See the <a href="#">Content Provider Basics</a> guide for more details.				
Task	Action	Data	MIME type	Notes											
Pick a contact from a list	<a href="#">ACTION_PICK</a>	One of: <ul style="list-style-type: none"> <li><a href="#">Contacts.CONTENT_URI</a>, which displays a list of contacts.</li> <li><a href="#">Phone.CONTENT_URI</a>, which displays a list of phone numbers for a raw contact.</li> <li><a href="#">StructuredPostal.CONTENT_URI</a>, which displays a list of postal addresses for a raw contact.</li> <li><a href="#">Email.CONTENT_URI</a>, which displays a list of email addresses for a raw contact.</li> </ul>	Not used	Displays a list of raw contacts or a list of data from a raw contact, depending on the content URI type you supply. Call <a href="#">startActivityForResult()</a> , which returns the content URI of the selected row. The form of the URI is the table's content URI with the row's <a href="#">LOOKUP_ID</a> appended to it. The device's contacts app delegates read and write permissions to this content URI for the life of your activity. See the <a href="#">Content Provider Basics</a> guide for more details.											
<a href="https://developer.android.com/guide/topics/providers/contacts-provider.html">https://developer.android.com/guide/topics/providers/contacts-provider.html</a>															



**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC
	<pre>25  * Parsed form of the intent sent to the Contacts application. 26  */ 27  public class ContactsRequest { 28 29      /** Default mode: browse contacts */ 30      public static final int ACTION_DEFAULT = 10; 31 32      /** Show all contacts */ 33      public static final int ACTION_ALL_CONTACTS = 15; 34 35      /** Show all contacts with phone numbers */ 36      public static final int ACTION_CONTACTS_WITH_PHONES = 17; 37 38      /** Show contents of a specific group */ 39      public static final int ACTION_GROUP = 20; 40 41      /** Show all starred contacts */ 42      public static final int ACTION_STARRED = 30; 43 44      /** Show frequently contacted contacts */ 45      public static final int ACTION_FREQUENT = 40; 46 47      /** Show starred and the frequent */ 48      public static final int ACTION_STREQUENT = 50; 49 50      /** Show all contacts and pick them when clicking */ 51      public static final int ACTION_PICK_CONTACT = 60; 52 53      /** Show all contacts as well as the option to create a new one */ 54      public static final int ACTION_PICK_OR_CREATE_CONTACT = 70; 55 56      /** Show all contacts and pick them for edit when clicking, and allow creating a new contact */ 57      public static final int ACTION_INSERT_OR_EDIT_CONTACT = 80;</pre> <p><a href="https://android.googlesource.com/platform/packages/apps/Contacts/+nougat-mr1-release/src/com/android/contacts/list/ContactsRequest.java">https://android.googlesource.com/platform/packages/apps/Contacts/+nougat-mr1-release/src/com/android/contacts/list/ContactsRequest.java</a></p>

**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC
	<pre>59  /** Show all phone numbers and pick them when clicking */ 60  public static final int ACTION_PICK_PHONE = 90; 61 62  /** Show all postal addresses and pick them when clicking */ 63  public static final int ACTION_PICK_POSTAL = 100; 64 65  /** Show all postal addresses and pick them when clicking */ 66  public static final int ACTION_PICK_EMAIL = 105; 67 68  /** Show all contacts and create a shortcut for the picked contact */ 69  public static final int ACTION_CREATE_SHORTCUT_CONTACT = 110; 70 71  /** Show all phone numbers and create a call shortcut for the picked number */ 72  public static final int ACTION_CREATE_SHORTCUT_CALL = 120; 73 74  /** Show all phone numbers and create an SMS shortcut for the picked number */ 75  public static final int ACTION_CREATE_SHORTCUT_SMS = 130; 76 77  /** Show all contacts and activate the specified one */ 78  public static final int ACTION_VIEW_CONTACT = 140; 79 80  /** Show contacts recommended for joining with a specified target contact */ 81  public static final int ACTION_PICK_JOIN = 150;</pre> <p><a href="https://android.googlesource.com/platform/packages/apps/Contacts+/nougat-mr1-release/src/com/android/contacts/list/ContactsRequest.java">https://android.googlesource.com/platform/packages/apps/Contacts+/nougat-mr1-release/src/com/android/contacts/list/ContactsRequest.java</a></p>

**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC
	<pre> 104      * Displays a list to browse contacts. 105      */ 106      public class PeopleActivity extends ContactsActivity implements 107          View.OnCreateContextMenuListener, 108          View.OnClickListener, 109          ActionBarAdapter.Listener, 110          DialogManager.DialogShowingViewActivity, 111          ContactListFilterController.ContactListFilterListener, 112          ProviderStatusListener, 113          MultiContactDeleteListener, 114          JoinContactsListener { </pre> <p data-bbox="512 659 1566 727"> <a href="https://android.googlesource.com/platform/packages/apps/Contacts+/nougat-mr1-release/src/com/android/contacts/activities/PeopleActivity.java">https://android.googlesource.com/platform/packages/apps/Contacts+/nougat-mr1-release/src/com/android/contacts/activities/PeopleActivity.java</a> </p> <pre> 145      * Showing a list of Contacts. Also used for showing search results in search mode. 146      */ 147      private MultiSelectContactsListFragment mAllFragment; 148      private ContactTileListFragment mFavoritesFragment; </pre> <p data-bbox="512 870 1566 937"> <a href="https://android.googlesource.com/platform/packages/apps/Contacts+/nougat-mr1-release/src/com/android/contacts/activities/PeopleActivity.java">https://android.googlesource.com/platform/packages/apps/Contacts+/nougat-mr1-release/src/com/android/contacts/activities/PeopleActivity.java</a> </p>

**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC
	<pre>458     private void configureFragments(boolean fromRequest) { 459         if (fromRequest) { 460             ContactListFilter filter = null; 461             int actionCode = mRequest.getActionCode(); 462             boolean searchMode = mRequest.isSearchMode(); 463             final int tabToOpen; 464             switch (actionCode) { 465                 case ContactsRequest.ACTION_ALL_CONTACTS: 466                     filter = ContactListFilter.createFilterWithType( 467                         ContactListFilter.FILTER_TYPE_ALL_ACCOUNTS); 468                     tabToOpen = TabState.ALL; 469                     break; 470                 case ContactsRequest.ACTION_CONTACTS_WITH_PHONES: 471                     filter = ContactListFilter.createFilterWithType( 472                         ContactListFilter.FILTER_TYPE_WITH_PHONE_NUMBERS_ONLY); 473                     tabToOpen = TabState.ALL; 474                     break; 475 476                 case ContactsRequest.ACTION_FREQUENT: 477                 case ContactsRequest.ACTION_STREQUENT: 478                 case ContactsRequest.ACTION_STARRED: 479                     tabToOpen = TabState.FAVORITES; 480                     break; 481                 case ContactsRequest.ACTION_VIEW_CONTACT: 482                     tabToOpen = TabState.ALL; 483                     break; 484                 default: 485                     tabToOpen = -1; 486                     break; 487             } </pre> <p data-bbox="506 1321 1566 1390"><a href="https://android.googlesource.com/platform/packages/apps/Contacts+/nougat-mr1-release/src/com/android/contacts/activities/PeopleActivity.java">https://android.googlesource.com/platform/packages/apps/Contacts+/nougat-mr1-release/src/com/android/contacts/activities/PeopleActivity.java</a></p>

**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC
	<pre>488         if (tabToOpen != -1) { 489             mActionBarAdapter.setCurrentTab(tabToOpen); 490         } 491 492         if (filter != null) { 493             mContactListFilterController.setContactListFilter(filter, false); 494             searchMode = false; 495         } 496 497         if (mRequest.getContactUri() != null) { 498             searchMode = false; 499         } 500 501         mActionBarAdapter.setSearchMode(searchMode); 502         configureContactListFragmentForRequest(); 503     } 504 505     configureContactListFragment(); 506 507     invalidateOptionsMenuIfNeeded(); 508 }</pre> <p data-bbox="506 1019 1570 1084"><a href="https://android.googlesource.com/platform/packages/apps/Contacts+/nougat-mr1-release/src/com/android/contacts/activities/PeopleActivity.java">https://android.googlesource.com/platform/packages/apps/Contacts+/nougat-mr1-release/src/com/android/contacts/activities/PeopleActivity.java</a></p>

**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC
	<pre>35 public class ProfileAndContactsLoader extends CursorLoader { 36 37     private boolean mLoadProfile; 38 39     private String[] mProjection; 40 41     private Uri mExtraUri; 42     private String[] mExtraProjection; 43     private String mExtraSelection; 44     private String[] mExtraSelectionArgs; 45     private boolean mMergeExtraContactsAfterPrimary; 46 47     public ProfileAndContactsLoader(Context context) { 48         super(context); 49     }</pre> <p><a href="https://android.googlesource.com/platform/packages/apps/ContactsCommon/+/nougat-mr1-release/src/com/android/contacts/common/list/ProfileAndContactsLoader.java">https://android.googlesource.com/platform/packages/apps/ContactsCommon/+/nougat-mr1-release/src/com/android/contacts/common/list/ProfileAndContactsLoader.java</a></p>

**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC
	<pre>35 public final class GroupMemberLoader extends CursorLoader { 36 37     public static class GroupEditorQuery { 38         private static final String[] PROJECTION = new String[] { 39             Data.CONTACT_ID,           // 0 40             Data.RAW_CONTACT_ID,      // 1 41             Data.DISPLAY_NAME_PRIMARY, // 2 42             Data.PHOTO_URI,           // 3 43             Data.LOOKUP_KEY,          // 4 44         }; 45 46         public static final int CONTACT_ID           = 0; 47         public static final int RAW_CONTACT_ID      = 1; 48         public static final int CONTACT_DISPLAY_NAME_PRIMARY = 2; 49         public static final int CONTACT_PHOTO_URI   = 3; 50         public static final int CONTACT_LOOKUP_KEY  = 4; 51     } 52 53     public static class GroupDetailQuery { 54         private static final String[] PROJECTION = new String[] { 55             Data.CONTACT_ID,           // 0 56             Data.PHOTO_URI,           // 1 57             Data.LOOKUP_KEY,          // 2 58             Data.DISPLAY_NAME_PRIMARY, // 3 59             Data.CONTACT_PRESENCE,     // 4 60             Data.CONTACT_STATUS,      // 5 61         }; 62 63         public static final int CONTACT_ID           = 0; 64         public static final int CONTACT_PHOTO_URI   = 1; 65         public static final int CONTACT_LOOKUP_KEY   = 2; 66         public static final int CONTACT_DISPLAY_NAME_PRIMARY = 3; 67         public static final int CONTACT_PRESENCE_STATUS = 4; 68         public static final int CONTACT_STATUS      = 5; 69     } 70 71     private final long mGroupId;</pre> <p><a href="https://android.googlesource.com/platform/packages/apps/Contacts/+/nougat-mr1-release/src/com/android/contacts/GroupMemberLoader.java">https://android.googlesource.com/platform/packages/apps/Contacts/+/nougat-mr1-release/src/com/android/contacts/GroupMemberLoader.java</a></p>

**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC
	<pre>24 * Group loader for the group list that includes details such as the number of contacts per group 25 * and number of groups per account. This list is sorted by account type, account name, where the 26 * group names are in alphabetical order. Note that the list excludes default, favorite, and deleted 27 * groups. 28 */ 29 public final class GroupListLoader extends CursorLoader { 30 31     private final static String[] COLUMNS = new String[] { 32         Groups.ACCOUNT_NAME, 33         Groups.ACCOUNT_TYPE, 34         Groups.DATA_SET, 35         Groups._ID, 36         Groups.TITLE, 37         Groups.SUMMARY_COUNT, 38     }; 39 40     public final static int ACCOUNT_NAME = 0; 41     public final static int ACCOUNT_TYPE = 1; 42     public final static int DATA_SET = 2; 43     public final static int GROUP_ID = 3; 44     public final static int TITLE = 4; 45     public final static int MEMBER_COUNT = 5; 46 47     private static final Uri GROUP_LIST_URI = Groups.CONTENT_SUMMARY_URI; 48 49     public GroupListLoader(Context context) { 50         super(context, GROUP_LIST_URI, COLUMNS, Groups.ACCOUNT_TYPE + " NOT NULL AND " 51             + Groups.ACCOUNT_NAME + " NOT NULL AND " + Groups.AUTO_ADD + "=0 AND " + 52             Groups.FAVORITES + "=0 AND " + Groups.DELETED + "=0", null, 53             Groups.ACCOUNT_TYPE + ", " + Groups.ACCOUNT_NAME + ", " + Groups.DATA_SET + ", " + 54             Groups.TITLE + " COLLATE LOCALIZED ASC"); 55     } 56 }</pre> <p><a href="https://android.googlesource.com/platform/packages/apps/Contacts+/nougat-mr1-release/src/com/android/contacts/GroupListLoader.java">https://android.googlesource.com/platform/packages/apps/Contacts+/nougat-mr1-release/src/com/android/contacts/GroupListLoader.java</a></p>



**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC
	<pre>24  * Group meta-data loader. Loads all groups or just a single group from the 25  * database (if given a {@link Uri}). 26  */ 27  public final class GroupMetaLoader extends CursorLoader { 28 29      private final static String[] COLUMNS = new String[] { 30          Groups.ACCOUNT_NAME, 31          Groups.ACCOUNT_TYPE, 32          Groups.DATA_SET, 33          Groups._ID, 34          Groups.TITLE, 35          Groups.AUTO_ADD, 36          Groups.FAVORITES, 37          Groups.GROUP_IS_READ_ONLY, 38          Groups.DELETED, 39      }; 40 41      public final static int ACCOUNT_NAME = 0; 42      public final static int ACCOUNT_TYPE = 1; 43      public final static int DATA_SET = 2; 44      public final static int GROUP_ID = 3; 45      public final static int TITLE = 4; 46      public final static int AUTO_ADD = 5; 47      public final static int FAVORITES = 6; 48      public final static int IS_READ_ONLY = 7; 49      public final static int DELETED = 8; 50 51      public GroupMetaLoader(Context context, Uri groupUri) { 52          super(context, ensureIsGroupUri(groupUri), COLUMNS, Groups.ACCOUNT_TYPE + " NOT NULL AND " 53              + Groups.ACCOUNT_NAME + " NOT NULL", null, null); 54      } 55 56      /** 57       * Ensures that this is a valid group URI. If invalid, then an exception is 58       * thrown. Otherwise, the original URI is returned. 59       */ 60      private static Uri ensureIsGroupUri(final Uri groupUri) { 61          // TODO: Fix ContactsProvider2 getType method to resolve the group Uris 62          if (groupUri == null) { 63              throw new IllegalArgumentException("Uri must not be null"); 64          } 65          if (!groupUri.toString().startsWith(Groups.CONTENT_URI.toString())) { 66              throw new IllegalArgumentException("Invalid group Uri: " + groupUri); 67          } 68          return groupUri; 69      } 70  }</pre> <p><a href="https://android.googlesource.com/platform/packages/apps/Contacts/+nougat-mr1-release/src/com/android/contacts/GroupMetaLoader.java">https://android.googlesource.com/platform/packages/apps/Contacts/+nougat-mr1-release/src/com/android/contacts/GroupMetaLoader.java</a></p>

**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC
	<pre>19  * Meta-data for a contact group. We load all groups associated with the contact's 20  * constituent accounts. 21  */ 22  public final class GroupMetaData { 23      private String mAccountName; 24      private String mAccountType; 25      private String mDataSet; 26      private long mGroupId; 27      private String mTitle; 28      private boolean mDefaultGroup; 29      private boolean mFavorites; 30 31      public GroupMetaData(String accountName, String accountType, String dataSet, long groupId, 32                          String title, boolean defaultGroup, boolean favorites) { 33          this.mAccountName = accountName; 34          this.mAccountType = accountType; 35          this.mDataSet = dataSet; 36          this.mGroupId = groupId; 37          this.mTitle = title; 38          this.mDefaultGroup = defaultGroup; 39          this.mFavorites = favorites; 40      } 41 42      public String getAccountName() { 43          return mAccountName; 44      } 45 46      public String getAccountType() { 47          return mAccountType; 48      } 49 50      public String getDataSet() { 51          return mDataSet; 52      } 53 54      public long getGroupId() { 55          return mGroupId; 56      } 57 58      public String getTitle() { 59          return mTitle; 60      } 61 62      public boolean isDefaultGroup() { 63          return mDefaultGroup; 64      } 65 66      public boolean isFavorites() { 67          return mFavorites; 68      } </pre> <p><a href="https://android.googlesource.com/platform/packages/apps/ContactsCommon/+nougat-mr1-release/src/com/android/contacts/common/GroupMetaData.java">https://android.googlesource.com/platform/packages/apps/ContactsCommon/+nougat-mr1-release/src/com/android/contacts/common/GroupMetaData.java</a></p>

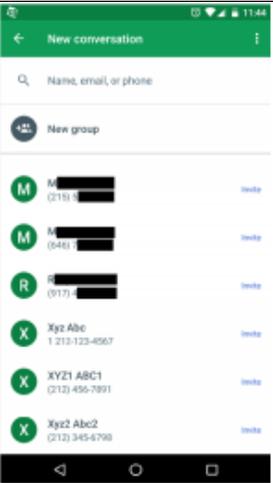
**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC
	<p data-bbox="541 272 1297 370"><b>Send &amp; receive text messages in Android Messages</b></p> <p data-bbox="541 383 1178 402">You can send and receive text messages with friends and contacts on Android Messages.</p> <p data-bbox="531 418 846 451"><b>Start a conversation</b></p> <ol data-bbox="541 475 1528 634" style="list-style-type: none"><li>1. Open the Android Messages app .</li><li>2. Tap Compose .</li><li>3. In "To," enter the names, phone numbers, or email addresses that you'd like to message. You can also pick from your top contacts or your whole contact list.</li><li>4. Tap Next .</li></ol> <p data-bbox="516 651 1472 683"><a href="https://support.google.com/nexus/answer/6080324?hl=en&amp;ref_topic=6080329">https://support.google.com/nexus/answer/6080324?hl=en&amp;ref_topic=6080329</a></p> <p data-bbox="541 776 894 816"><b>See your contacts</b></p> <ol data-bbox="552 846 968 919" style="list-style-type: none"><li>1. Open your device's Contacts app .</li><li>2. Tap Menu .</li></ol> <ul data-bbox="541 946 1738 1146" style="list-style-type: none"><li>• <b>See contacts by label:</b> Choose a label from the list.</li><li>• <b>See contacts for another account:</b> Tap Down arrow  &gt; pick an account.</li><li>• <b>See the contacts for all your accounts:</b> Choose <b>All contacts</b>. <b>Tip:</b> If you have multiple contacts with the same information, the information will be grouped into one contact.</li><li>• <b>See your Google Account contacts on the web:</b> Go to <a href="#">Google Contacts</a> .</li></ul> <p data-bbox="510 1174 1535 1206"><a href="https://support.google.com/android/answer/6118731?hl=en&amp;ref_topic=6118711">https://support.google.com/android/answer/6118731?hl=en&amp;ref_topic=6118711</a></p>

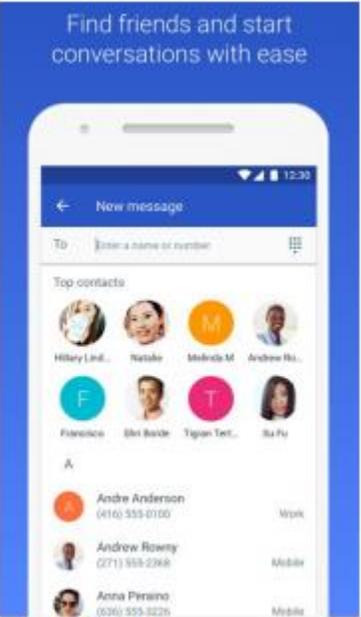
**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC
	<p data-bbox="541 240 894 277"><b>Label your contacts</b></p> <p data-bbox="541 305 982 326">You can group contacts together using labels.</p> <ol data-bbox="541 358 926 461" style="list-style-type: none"><li data-bbox="541 358 926 380">1. Open your device's Contacts app .</li><li data-bbox="541 396 863 417">2. Tap Menu  &gt; <b>Create label</b>.</li><li data-bbox="541 433 863 454">3. Enter a label name and tap <b>Ok</b>.</li></ol> <ul data-bbox="541 493 1713 553" style="list-style-type: none"><li data-bbox="541 493 1234 514">• <b>Add one contact to a label:</b> Tap Add contact  &gt; choose a contact.</li><li data-bbox="541 531 1713 552">• <b>Add multiple contacts to a label:</b> Tap Add contact  &gt; touch and hold a contact &gt; tap the other contacts &gt; tap <b>Add</b>.</li></ul> <p data-bbox="512 570 1535 591"><a href="https://support.google.com/android/answer/6118731?hl=en&amp;ref_topic=6118711">https://support.google.com/android/answer/6118731?hl=en&amp;ref_topic=6118711</a></p> <p data-bbox="541 656 940 693"><b>Share your contacts</b></p> <ol data-bbox="541 725 1045 883" style="list-style-type: none"><li data-bbox="541 725 974 747">1. Open your device's Contacts app .</li><li data-bbox="541 769 842 790">2. Tap a contact in the list.</li><li data-bbox="541 813 831 834">3. Tap More  &gt; <b>Share</b>.</li><li data-bbox="541 857 1045 878">4. Choose how you want to share the contact.</li></ol> <p data-bbox="512 899 1535 920"><a href="https://support.google.com/android/answer/6118731?hl=en&amp;ref_topic=6118711">https://support.google.com/android/answer/6118731?hl=en&amp;ref_topic=6118711</a></p>

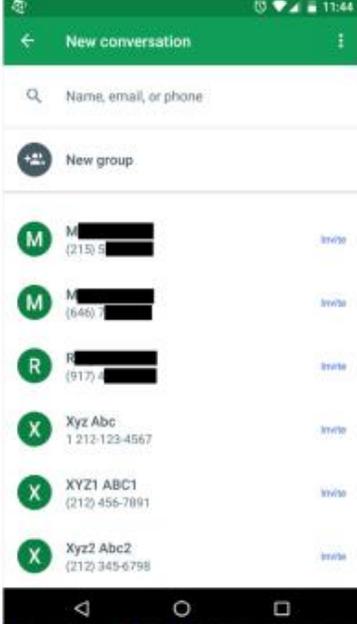
# Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products

US9408055B2	HTC
	<p><b>Start a Hangout</b></p> <p>You can send and receive messages with one person or multiple people.</p> <p>COMPUTER <b>ANDROID</b> IPHONE &amp; IPAD</p> <p><b>Start a conversation</b></p> <ol style="list-style-type: none"><li>1. On your Android phone or tablet, open the Hangouts app .</li><li>2. At the bottom right, tap Add  &gt; New Conversation .</li><li>3. Type and select a person's name.</li><li>4. Enter your message. You can also add emojis, photos, your location, or a sticker.</li><li>5. Tap Send .</li></ol>  <p><a href="https://support.google.com/hangouts/answer/3115553?hl=en&amp;ref_topic=6386410&amp;co=GENIE.Platform%3DAndroid&amp;oco=0">https://support.google.com/hangouts/answer/3115553?hl=en&amp;ref_topic=6386410&amp;co=GENIE.Platform%3DAndroid&amp;oco=0</a></p> <p><b>Contact someone</b></p> <p>You can call, email, or send text messages to your contacts.</p> <ol style="list-style-type: none"><li>1. Open your device's Contacts app .</li><li>2. Tap a contact in the list.</li><li>3. Choose an option:<ul style="list-style-type: none"><li>• Call </li><li>• Email </li><li>• New message </li></ul></li></ol> <p><a href="https://support.google.com/nexus/answer/6118731?hl=en&amp;ref_topic=6118711">https://support.google.com/nexus/answer/6118731?hl=en&amp;ref_topic=6118711</a></p>

**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC
	<p data-bbox="527 688 762 716"><b>Start a conversation</b></p> <ol data-bbox="527 732 1272 862" style="list-style-type: none"><li>1. Open the Android Messages app</li><li>2. Tap Compose</li><li>3. In "To," enter the names, phone numbers, or email addresses that you'd like to message. You can also pick from your top contacts or your whole contact list.</li><li>4. Tap Next</li></ol> <p data-bbox="516 878 1612 943"><a href="https://support.google.com/nexus/answer/6080324?hl=en&amp;ref_topic=6080329">https://support.google.com/nexus/answer/6080324?hl=en&amp;ref_topic=6080329</a> <a href="https://play.google.com/store/apps/details?id=com.google.android.apps.messaging&amp;hl=en">https://play.google.com/store/apps/details?id=com.google.android.apps.messaging&amp;hl=en</a></p> 

# Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products

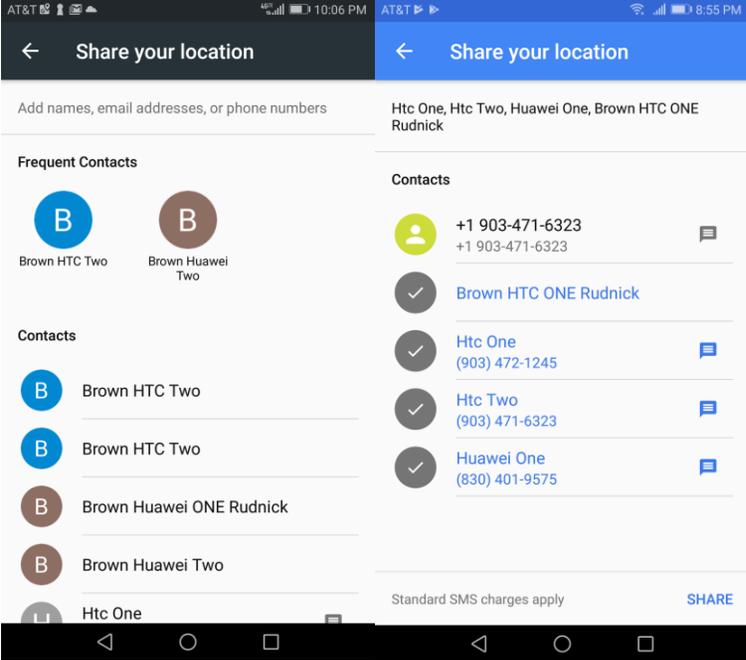
US9408055B2	HTC
	<p data-bbox="533 618 842 651"><b>Start a conversation</b></p> <ol data-bbox="533 672 1220 829" style="list-style-type: none"><li>1. On your Android phone or tablet, open the Hangouts app .</li><li>2. At the bottom right, tap Add  &gt; New Conversation .</li><li>3. Type and select a person's name.</li><li>4. Enter your message. You can also add emojis, photos, your location, or a sticker.</li><li>5. Tap Send .</li></ol> <p data-bbox="520 862 1751 919"><a href="https://support.google.com/hangouts/answer/3115553?hl=en&amp;ref_topic=6386410&amp;co=GENIE.Platform%3DAndroid&amp;oco=0">https://support.google.com/hangouts/answer/3115553?hl=en&amp;ref_topic=6386410&amp;co=GENIE.Platform%3DAndroid&amp;oco=0</a></p> <ol data-bbox="533 980 1352 1138" style="list-style-type: none"><li>1. Open the Hangouts app .</li><li>2. At the bottom, tap Add  &gt; <b>New conversation</b> &gt; <b>New group</b>.</li><li>3. Enter and select the names, phone numbers, or email addresses of people in your group.</li><li>4. Tap Done .</li></ol> <p data-bbox="520 1149 1751 1206"><a href="https://support.google.com/hangouts/answer/3111943?hl=en&amp;ref_topic=6386410&amp;co=GENIE.Platform%3DAndroid&amp;oco=1">https://support.google.com/hangouts/answer/3111943?hl=en&amp;ref_topic=6386410&amp;co=GENIE.Platform%3DAndroid&amp;oco=1</a></p> 

**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

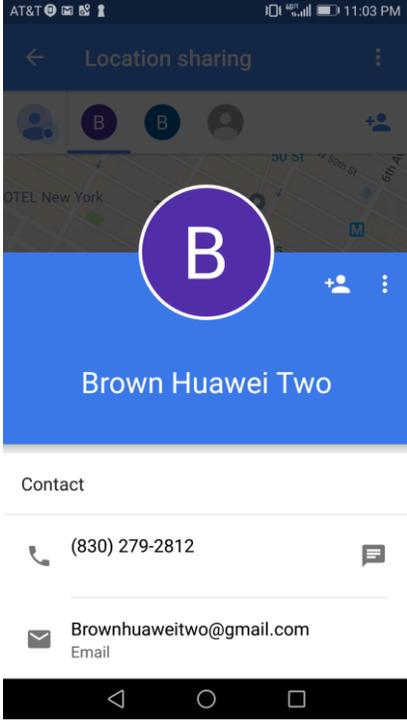
US9408055B2	HTC
	<p data-bbox="541 235 905 272"><b>Contact someone</b></p> <p data-bbox="541 305 1192 332">You can call, email, or send text messages to your contacts.</p> <ol data-bbox="554 365 982 625" style="list-style-type: none"><li data-bbox="554 365 982 397">1. Open your device's Contacts app .</li><li data-bbox="554 414 842 446">2. Tap a contact in the list.</li><li data-bbox="554 462 779 495">3. Choose an option:<ul data-bbox="583 503 808 625" style="list-style-type: none"><li data-bbox="583 503 688 535">• Call </li><li data-bbox="583 552 709 584">• Email </li><li data-bbox="583 600 808 625">• New message </li></ul></li></ol> <p data-bbox="520 641 1472 673"><a href="https://support.google.com/nexus/answer/6118731?hl=en&amp;ref_topic=6118711">https://support.google.com/nexus/answer/6118731?hl=en&amp;ref_topic=6118711</a></p> <p data-bbox="512 711 1031 743"><b><u>Exemplary Google Maps Screenshots:</u></b></p>



**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC
	 <p>For example, the Accused Products include software that obtains contact information including the phone numbers . Furthermore, these phone calls can merge multiple parties into a conference call.</p>

**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC
	 <p>The screenshot shows an Android contact card for 'Brown Huawei Two'. At the top, there is a 'Location sharing' header with a back arrow and a menu icon. Below this is a map showing a location in New York. A large purple circle with a white 'B' is overlaid on the map. Below the map, the contact name 'Brown Huawei Two' is displayed. Underneath the name, there is a 'Contact' section with a phone number '(830) 279-2812' and an email address 'Brownhuaweitwo@gmail.com'. The bottom of the screen shows the standard Android navigation bar with back, home, and recent apps buttons.</p>
<p>[54B] facilitating initiation of Internet Protocol (IP) based communication between the first device and the respective second devices by using the respective telephone numbers to send, to the second devices, respective Short</p>	<p>HTC infringes directly and/or indirectly by performing, inducing others to perform, and/or contributing to the performance of facilitating initiation of Internet Protocol (IP) based communication between the first device and the respective second devices by using the respective telephone numbers to send, to the second devices, respective Short Message Service (SMS) messages including a telephone number of the first device and information usable by the respective second device to send IP-based communication to the first device. See claims 1[B], 28[B], and 41[B], which are incorporated herein by reference in their entirety.</p> <p>For example, the Accused Products utilize SMS-based messages to initiate IP communication between participants of Maps location sharing. For example, both Android Messages and Hangouts, in conjunction with Maps, utilize SMS messages, including group messages from one device to several devices, to send an SMS message, with additional information, to a contact.</p>

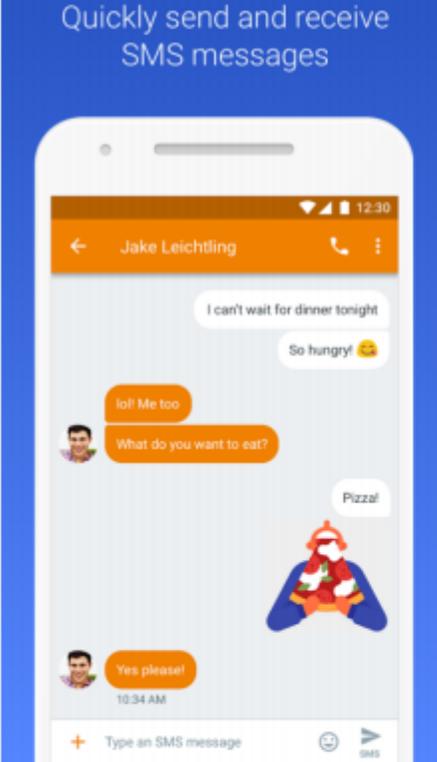
**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC
<p>Message Service (SMS) messages including a telephone number of the first device and information usable by the respective second device to send IP-based communication to the first device;</p>	<p>D2 Technologies Showcases its mCUE IP Communications Interface over WiMAX on HTC's EVO 4G Android Smartphone</p> <p><b>Amsterdam, Netherlands (WiMAX Forum Global Congress) and Santa Barbara, CA – June 14, 2010</b> —D2 Technologies, the market leader in embedded IP communications software platforms, today announced that it is holding private demonstrations of its mCUE® converged communications client for mobile devices and handsets on the HTC EVO™ 4G smartphone on Thursday, June 17 at the WiMAX Forum® Global Congress in Amsterdam. D2's mCUE on the HTC EVO, the first 4G phone to be introduced in the United States, was configured and installed in less than a week – clearly illustrating how OEMs and ODMs can more rapidly develop Android™-based devices by choosing to incorporate the converged presence-based communications user interface (CUI).</p> <p><a href="http://www.d2tech.com/press-releases-year.html?Y=2010">http://www.d2tech.com/press-releases-year.html?Y=2010</a></p> <p>Siemens Enterprise Communications and HTC Simplify Device Choice for Mobile UC   </p> <p><b>Reston, VA and Frankfurt, Germany, Feb 19, 2013</b></p> <p>Enterasys' Mobile IAM and MDM connect™ BYOD solutions now support AirWatch to onboard and manage mobile devices and applications</p> <p>In an ongoing effort to support today's increasingly mobile workforce, Siemens Enterprise Communications and HTC Corporation today announced a strategic global partnership to make it easier for enterprises to embrace mobile unified communications (UC) on HTC enterprise-enabled devices. This partnership makes it even simpler for enterprises to embrace a BYOD strategy for mobile UC, since HTC's popular consumer Android smartphones now fully support Siemens Enterprise Communications' OpenScape Mobile and OpenScape Web Collaboration solutions.</p> <p><b>Key Facts</b></p> <ul style="list-style-type: none"> <li>▪ Siemens Enterprise Communications OpenScape Mobile and OpenScape Web Collaboration solutions will be validated on select HTC devices to increase users' confidence that their chosen device will work seamlessly with their mobile communication tools</li> <li>▪ Siemens Enterprise Communications customers will have a simplified process to secure validated HTCPro devices supporting Siemens Enterprise Communications solutions</li> <li>▪ Siemens Enterprise Communications and HTC will collaborate to simplify deployment of mobile UC through joint marketing and fulfillment efforts</li> <li>▪ This collaboration will take place through HTCPro, a program that provides mobile solutions for companies and their employees and ensures that HTC's entire portfolio is business-ready</li> </ul> <p>Siemens Enterprise Communications OpenScape Mobile and OpenScape Web Collaboration solutions will be validated on select HTC devices to increase users' confidence that their chosen device will work seamlessly with their mobile communication tools.</p>

**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC
	<p data-bbox="512 233 1713 264"><a href="http://www.unify.com/us/news/E853B94A-F94F-4ADA-98DA-8C80BB965953/?isarchive=1">http://www.unify.com/us/news/E853B94A-F94F-4ADA-98DA-8C80BB965953/?isarchive=1</a></p> <p data-bbox="541 334 1514 435">Android Messages makes it easy to communicate with anyone by using SMS, MMS, and more. Stay in touch with friends and family, send group texts, and share your favorite pictures, videos, audio messages.</p> <p data-bbox="533 464 1612 532">• <b>Enhanced features:</b> On supported carriers, you can send messages over Wi-Fi or your data network, see when friends have read your message, and more.</p> <p data-bbox="522 552 1619 583"><a href="https://play.google.com/store/apps/details?id=com.google.android.apps.messaging&amp;hl=en">https://play.google.com/store/apps/details?id=com.google.android.apps.messaging&amp;hl=en</a></p>

# Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products

US9408055B2	HTC
	 <p><a href="https://play.google.com/store/apps/details?id=com.google.android.apps.messaging&amp;hl=en">https://play.google.com/store/apps/details?id=com.google.android.apps.messaging&amp;hl=en</a></p> <h2>Get started with Hangouts</h2> <p>You can use Hangouts to:</p> <ul style="list-style-type: none"><li>• Start a chat conversation or video call.</li><li>• Make phone calls using Wi-Fi or data.</li><li>• Send text messages with your <a href="#">Google Voice</a> or <a href="#">Project Fi</a> phone number.</li></ul> <p>Hangouts sync automatically across devices. If you start a Hangout on your computer, you can continue your chat on another device, like your phone.</p> <p><a href="https://support.google.com/hangouts/answer/2944865?hl=en&amp;ref_topic=6386410">https://support.google.com/hangouts/answer/2944865?hl=en&amp;ref_topic=6386410</a></p>

## Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products

US9408055B2	HTC
	<p data-bbox="541 266 835 305"><b>Start a Hangout</b></p> <p data-bbox="541 323 1052 342">You can send and receive messages with one person or multiple people.</p> <p data-bbox="554 404 888 423">COMPUTER <b>ANDROID</b> IPHONE &amp; IPAD</p> <hr/> <p data-bbox="541 553 806 581"><b>Start a conversation</b></p> <ol data-bbox="548 602 1136 737" style="list-style-type: none"><li>1. On your Android phone or tablet, open the Hangouts app .</li><li>2. At the bottom right, tap Add  &gt; New Conversation .</li><li>3. Type and select a person's name.</li><li>4. Enter your message. You can also add emojis, photos, your location, or a sticker.</li><li>5. Tap Send .</li></ol> <p data-bbox="527 753 1745 813"><a href="https://support.google.com/hangouts/answer/3115553?hl=en&amp;ref_topic=6386410&amp;co=GENIE.Platform%3DAndroid&amp;oco=0">https://support.google.com/hangouts/answer/3115553?hl=en&amp;ref_topic=6386410&amp;co=GENIE.Platform%3DAndroid&amp;oco=0</a></p> <ul data-bbox="562 862 1562 1013" style="list-style-type: none"><li>• Connect your Google Voice account for phone calling, SMS texting, and voicemail integration.</li><li>• Keep in touch with contacts across Android, iOS, and the web, and sync chats across all your devices.</li><li>• Message contacts anytime, even if they're offline.</li></ul> <p data-bbox="527 1045 1398 1073"><a href="https://play.google.com/store/apps/details?id=com.google.android.talk&amp;hl=en">https://play.google.com/store/apps/details?id=com.google.android.talk&amp;hl=en</a></p> <ol data-bbox="548 1084 1493 1344" style="list-style-type: none"><li>1. Open the Hangouts app .</li><li>2. At the bottom right, tap Add .</li><li>3. Choose <b>New SMS</b>.</li><li>4. Type the name or phone number. If you're traveling, use the "+" sign and country code when texting.</li><li>5. Tap the number or contact.</li><li>6. Tap Continue .</li><li>7. Type your message and tap Send .</li></ol> <p data-bbox="527 1360 1255 1388"><a href="https://support.google.com/hangouts/answer/3441321?hl=en">https://support.google.com/hangouts/answer/3441321?hl=en</a></p>

**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC
	<p data-bbox="510 269 911 305"><b><u>Google Maps Share Location</u></b></p> <p data-bbox="510 345 1906 997">Google Latitude, Google Plus, Google Hangouts, Google Messages, and Android Messenger, Google Duo, Google Allo, and Google Chrome, the Accused Products require a user to join the corresponding network by signing-in to the device with an identifier (e.g., Google Account). The sign-in process takes place within the Google Maps software on the Accused Product or by navigating to maps.google.com within the Google Chrome browser on the Accused Product. Alternatively, the sign-in process may partially or completely take place using credentials already provided when the user associates a Google Account with the Accused Product, e.g., during initial setup of the Accused Product. Subject to discovery, one or more additional or substitute identifiers may correspond to the group. The sign-in process involves a user entering its Google Account and additional authentication data on the interface of the Accused Product and sending a message containing the Google Account and additional authentication data over a network to members of a group. The group comprises the multiple identifiers, individuals, profiles, and/or devices associated with the group. Further regarding Google Maps, Google Latitude, Google Plus, Google Hangouts, Google Messages, and Android Messenger, Google Duo, Google Allo, and Google Chrome, the Accused Products alternatively require a user to send a request containing the identifier (e.g., Google Account). Subject to discovery, additional identifiers may be assigned or used to correspond to the group. The request may be an invitation or message that associates a Google Account with one or more Google Accounts for the purposes of sharing locations within the group. The group comprises the multiple identifiers, individuals, profiles, and/or devices associated with the group.</p> <p data-bbox="510 1040 1031 1073"><b><u>Exemplary Support for Google Maps:</u></b></p>

# Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products

US9408055B2	HTC
	<p>COMPUTER <b>ANDROID</b> IPHONE &amp; IPAD</p> <hr/> <h3>If they have a Google Account</h3> <ol style="list-style-type: none"><li>1. If you haven't already, add their Gmail address to your <a href="#">Google Contacts</a>.</li><li>2. On your Android phone or tablet, open the Google Maps app  and sign in. <a href="#">Learn how to sign in.</a></li><li>3. Tap Menu ≡ &gt; <b>Location sharing</b> &gt; Add People .</li><li>4. Choose how long you want to share your location.</li><li>5. Tap <b>Select People</b>.<ul style="list-style-type: none"><li>• If you're asked about your contacts, give Google Maps access.</li></ul></li><li>6. Choose who you want to share with.</li><li>7. Tap <b>Share</b>.</li></ol> <h3>If they don't have a Google Account</h3> <ol style="list-style-type: none"><li>1. On your Android phone or tablet, open the Google Maps app  and sign in. <a href="#">Learn how to sign in.</a></li><li>2. Tap Menu ≡ &gt; <b>Location sharing</b> &gt; Add People .</li><li>3. Tap More ... &gt; <b>Copy to clipboard</b>. People with this link can see your location for as long as you choose, up to 72 hours.</li></ol> <h3>Share using another app</h3> <p>You can also share through messaging apps. Tap More ... &gt; select an app.</p> <h3>Stop sharing</h3> <ol style="list-style-type: none"><li>1. Open the Google Maps app .</li><li>2. Tap Menu ≡ &gt; <b>Location sharing</b>.</li><li>3. Next to the person with whom you want to stop sharing, tap Remove ✕.</li></ol> <p><a href="https://support.google.com/maps/answer/7326816?co=GENIE.Platform%3DAndroid&amp;oco=1">https://support.google.com/maps/answer/7326816?co=GENIE.Platform%3DAndroid&amp;oco=1</a></p>



## Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products

US9408055B2	HTC
	<h3>Share your E.T.A</h3> <p>After you start your drive, you can share your destination, estimated arrival time, and where you are on the route.</p> <ol style="list-style-type: none"><li>1. Open the Google Maps app .</li><li>2. Set a driving destination. <a href="#">Learn how to navigate to a place.</a></li><li>3. After you start navigation, tap More ^ &gt; <b>Share trip progress.</b></li><li>4. Choose a person from the list.</li><li>5. Tap <b>Share.</b></li><li>6. Location Sharing will stop when you reach your destination or stop navigating.</li></ol> <ul style="list-style-type: none"><li>• To stop sharing before you arrive, tap More ^ &gt; <b>Stop sharing.</b></li></ul> <h3>See where someone is</h3> <p>If someone shares their location with you, you can see them on the map.</p> <ol style="list-style-type: none"><li>1. Open the Google Maps app .</li><li>2. Tap Menu ≡ &gt; <b>Location sharing.</b></li><li>3. Choose someone.</li></ol> <ul style="list-style-type: none"><li>• To see an updated location, tap on a friend's icon &gt; More ⋮ &gt; <b>Refresh.</b></li></ul> <h3>Stop seeing someone's location</h3> <ol style="list-style-type: none"><li>1. Open the Google Maps app .</li><li>2. On the map, tap their icon.</li><li>3. At the bottom, tap More ^ .</li><li>4. To temporarily hide someone, tap <b>Hide from map.</b> You can stop hiding them at any time.</li></ol> <p><b>Note:</b> You can stop someone's location from ever appearing on your map. <a href="#">Learn how to block another person's account.</a></p> <p><a href="https://support.google.com/maps/answer/7326816?co=GENIE.Platform%3DAndroid&amp;oco=1">https://support.google.com/maps/answer/7326816?co=GENIE.Platform%3DAndroid&amp;oco=1</a></p>

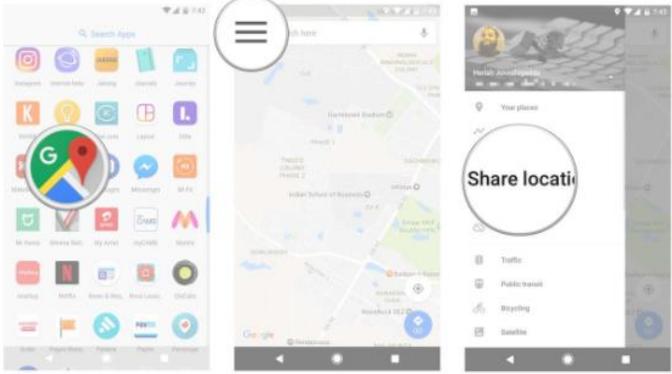
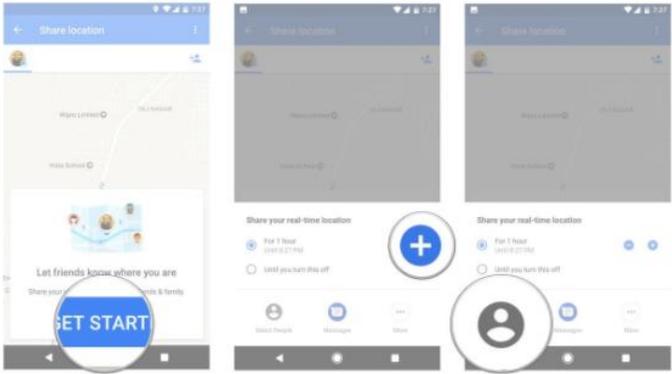
## Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products

US9408055B2	HTC
	<h3>Create a list of places</h3> <p>In Google Maps, you can create a list of places, like your favorite places or places you want to visit.</p> <p>COMPUTER <b>ANDROID</b> IPHONE &amp; IPAD</p> <hr/> <h3>Make a new list</h3> <ol style="list-style-type: none"><li>1. On your Android phone or tablet, open the Google Maps app .</li><li>2. Tap Menu  &gt; <b>Your places</b> &gt; <b>Saved</b>.</li><li>3. In the bottom right, tap Add .</li><li>4. Enter a name and description.</li><li>5. Tap <b>Save</b>.</li></ol> <h3>Save a place to a list</h3> <ol style="list-style-type: none"><li>1. Open the Google Maps app .</li><li>2. Search for a place or tap it on the map.</li><li>3. At the bottom, tap the place's name or address.</li><li>4. Tap <b>Save</b>.</li><li>5. Choose a list. To create a new list, tap <b>New list</b> .</li></ol> <h3>See your lists</h3> <ol style="list-style-type: none"><li>1. Open the Google Maps app .</li><li>2. Tap Menu  &gt; <b>Your places</b> &gt; <b>Saved</b>.</li></ol> <p><a href="https://support.google.com/maps/answer/7280933?hl=en&amp;ref_topic=7301134&amp;co=GENIE.Platform%3DAndroid&amp;oco=1">https://support.google.com/maps/answer/7280933?hl=en&amp;ref_topic=7301134&amp;co=GENIE.Platform%3DAndroid&amp;oco=1</a></p>

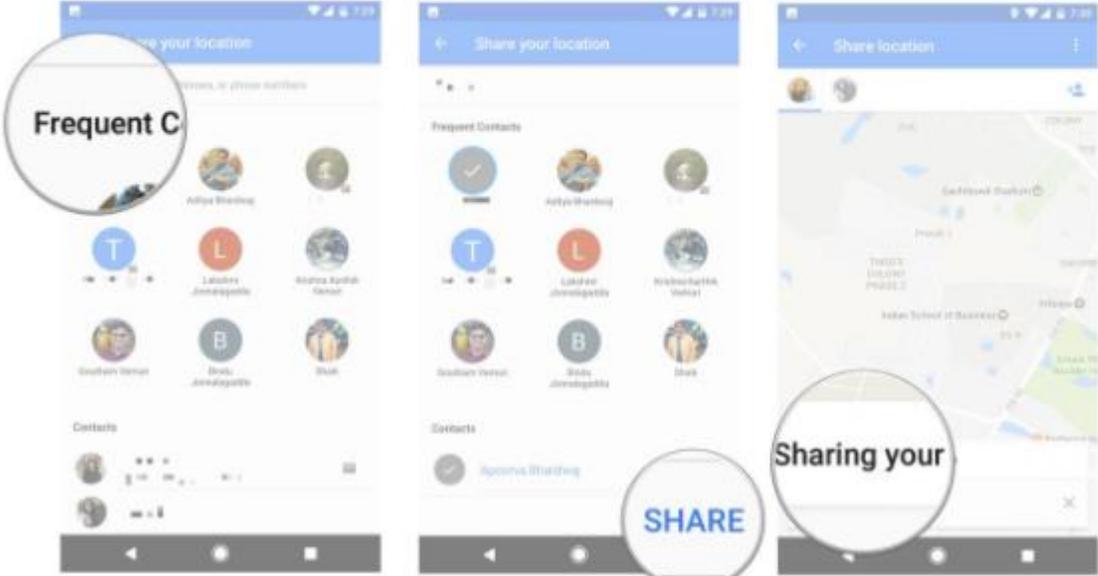
## Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products

US9408055B2	HTC
	<h3 data-bbox="541 245 877 282">Hide or share lists</h3> <p data-bbox="541 313 909 337"><b>Note:</b> You can't share starred places.</p> <ol data-bbox="554 367 1251 475" style="list-style-type: none"><li>1. Open the Google Maps app .</li><li>2. Tap Menu  &gt; <b>Your places</b> &gt; <b>Saved</b>.</li><li>3. Next to the list you want to share, tap More  &gt; choose an option:</li></ol> <ul data-bbox="583 492 1682 634" style="list-style-type: none"><li>• <b>Hide/Show on your map:</b> Display or hide your saved places when looking at the map.</li><li>• <b>Share list:</b> Allow others to see your saved list.</li><li>• <b>Sharing options:</b> You can make your list public, private, or shared. To let anyone with the link see your list, tap <b>Shared</b>. To let anyone find and follow your list, tap <b>Public</b>.</li></ul> <h3 data-bbox="541 699 768 737">Follow a list</h3> <p data-bbox="541 768 1728 824">If you follow a list made by someone else, their saved places will show up in Your Places. The places will also appear as suggested locations in Google Maps.</p> <h3 data-bbox="541 878 915 915">Follow a list using a link</h3> <ol data-bbox="554 938 1356 1047" style="list-style-type: none"><li>1. Tap on the link you received to open it.</li><li>2. Tap <b>Follow</b>. This list will now be added to the group of lists you follow.</li><li>3. <b>Optional:</b> To unfollow a list someone shared with you, tap the list &gt; <b>Following</b>.</li></ol> <h3 data-bbox="541 1101 926 1138">See lists made by others</h3> <p data-bbox="541 1161 1335 1185">If a user has any Google Maps lists that were made public, you can follow them.</p> <ol data-bbox="554 1214 1136 1323" style="list-style-type: none"><li>1. Tap on the name of a user whose list you want to follow.</li><li>2. Tap <b>Lists</b>.</li><li>3. Tap on the list you want to follow &gt; More  &gt; <b>Follow</b>.</li></ol> <p data-bbox="512 1339 1902 1401"><a href="https://support.google.com/maps/answer/7280933?hl=en&amp;ref_topic=7301134&amp;co=GENIE.Platform%3DAndroid&amp;oco=1">https://support.google.com/maps/answer/7280933?hl=en&amp;ref_topic=7301134&amp;co=GENIE.Platform%3DAndroid&amp;oco=1</a></p>

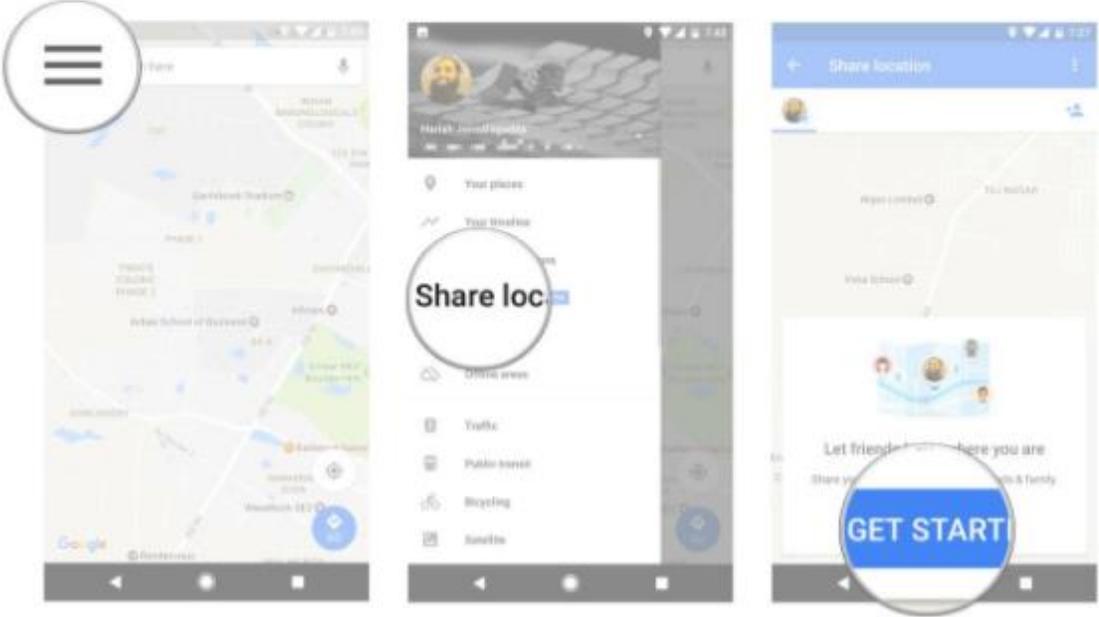
# Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products

US9408055B2	HTC
	<h3 data-bbox="520 240 1150 272">How to share your location in Google Maps</h3> <ol data-bbox="520 300 1134 389" style="list-style-type: none"><li>1. Open Google Maps from the app drawer or the home screen.</li><li>2. Tap the hamburger menu (the three horizontal lines) on the top left corner of the screen.</li><li>3. Select Share location.</li></ol>  <ol data-bbox="520 824 1134 933" style="list-style-type: none"><li>4. Tap Get Started.</li><li>5. Use the + icon to select a time period or select the Until you turn this off setting to share your location indefinitely.</li><li>6. Tap Select People.</li></ol>  <p data-bbox="520 1339 1354 1369"><a href="https://www.androidcentral.com/how-share-location-google-maps">https://www.androidcentral.com/how-share-location-google-maps</a></p>

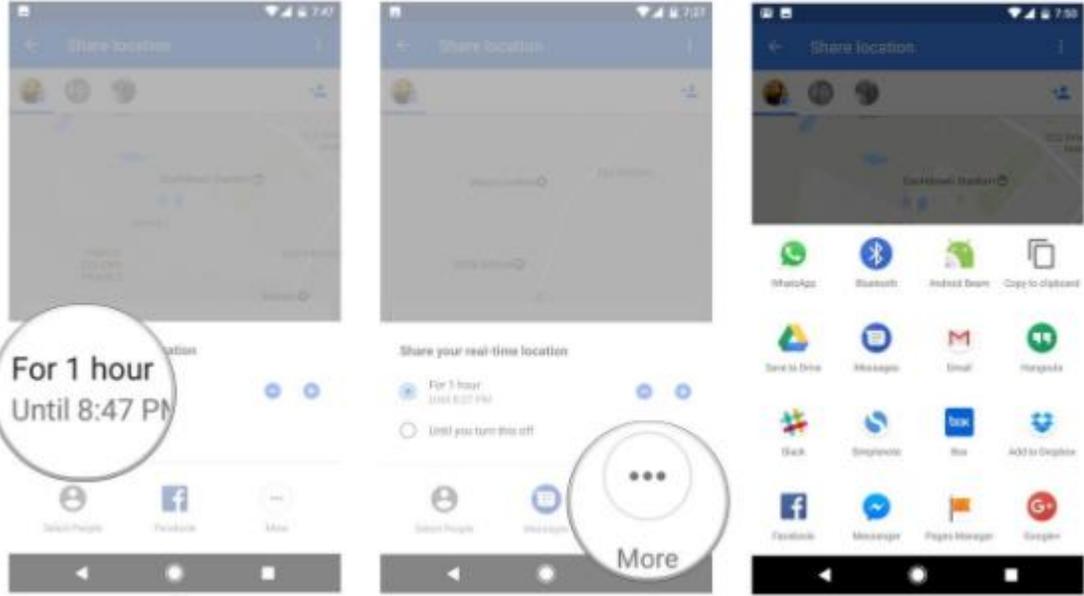
**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC
	<p data-bbox="527 253 1577 427"><b>7.</b> You'll see a list of your frequent contacts at the top, along with a full list of contacts. <b>Pick the contacts</b> by tapping their name.</p> <p data-bbox="527 342 1457 367"><b>8.</b> Once you've selected the contacts you want to share your location to, tap <b>Share</b>.</p> <p data-bbox="527 399 1419 423"><b>9.</b> You'll see a message saying that the selected contact can view your location.</p>  <p data-bbox="510 1065 1356 1097"><a href="https://www.androidcentral.com/how-share-location-google-maps">https://www.androidcentral.com/how-share-location-google-maps</a></p>

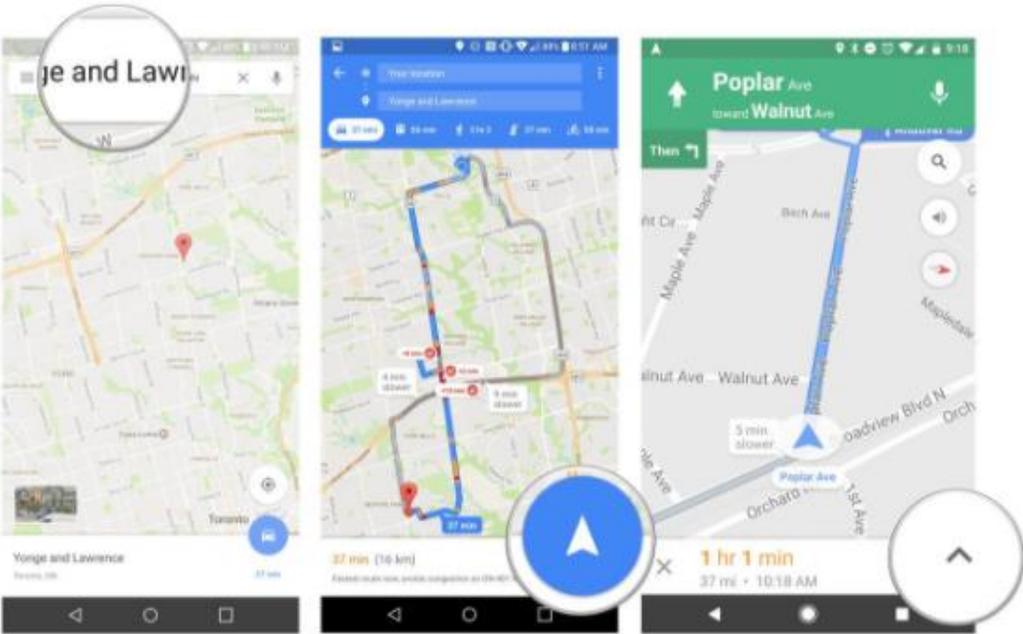
# Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products

US9408055B2	HTC
	<h2 data-bbox="520 245 1255 289">How to create a shareable link</h2> <p data-bbox="520 334 1461 362">You can also create a link and use it to share your location easily. Here's how to do it:</p> <ol data-bbox="520 412 1234 553" style="list-style-type: none"><li data-bbox="520 412 1234 440">1. Tap the hamburger menu on the top left corner of the screen.</li><li data-bbox="520 467 800 495">2. Select Share location.</li><li data-bbox="520 522 737 550">3. Tap Get Started.</li></ol>  <p data-bbox="506 1232 1356 1263"><a href="https://www.androidcentral.com/how-share-location-google-maps">https://www.androidcentral.com/how-share-location-google-maps</a></p>

**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

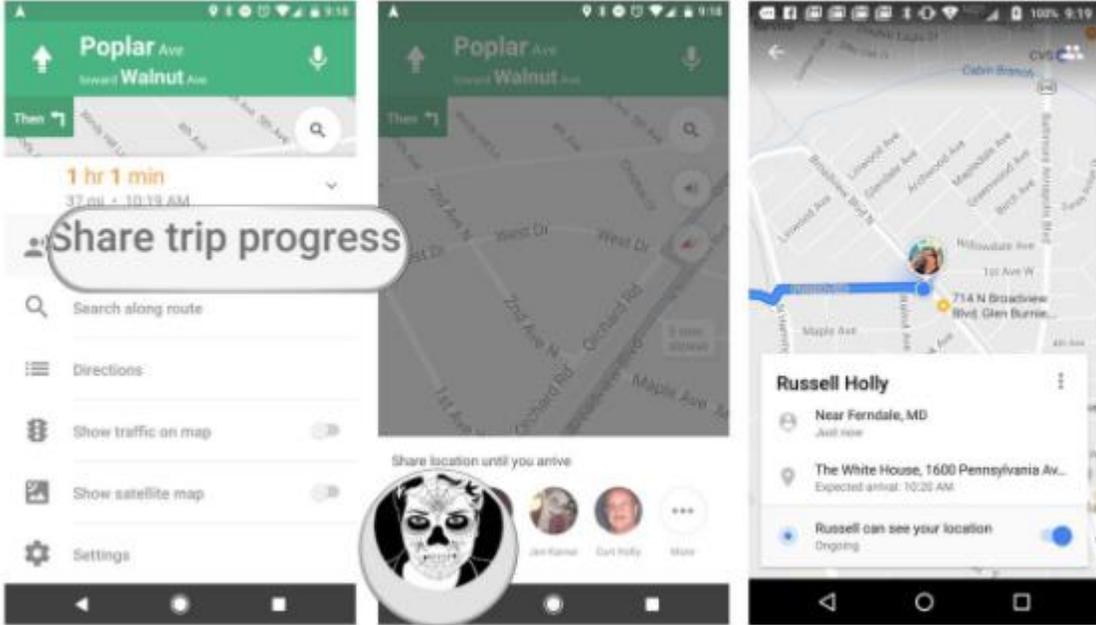
US9408055B2	HTC
	<p data-bbox="520 245 1213 272">4. Select the <b>amount of time</b> you want to share your location.</p> <p data-bbox="520 302 680 329">5. Tap <b>More</b>.</p> <p data-bbox="520 358 1633 418">6. Select <b>your app of choice</b> to create and send a <b>unique URL</b> that broadcasts your current location. You can email it, send the link via Messenger, or even tweet it to the <b>intended recipient</b>.</p>  <p data-bbox="506 1084 1360 1117"><a href="https://www.androidcentral.com/how-share-location-google-maps">https://www.androidcentral.com/how-share-location-google-maps</a></p>

**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

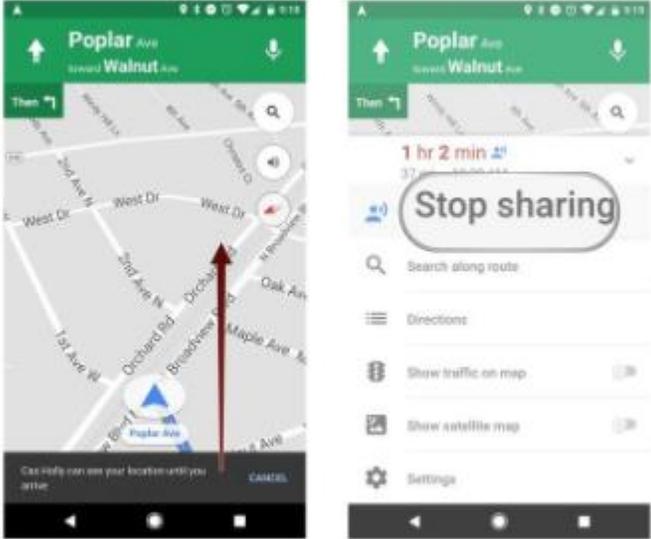
US9408055B2	HTC
	<h2 data-bbox="527 240 1428 337">How to share your navigation directions while you walk, drive or transit</h2> <p data-bbox="527 375 1556 464">One of the best ways to share your location with someone is during a drive, walk or transit. If you're meeting a friend or family member somewhere, or navigating towards their home, Google Maps lets you share your location with them for the duration of the trip. It's magic!</p> <ol data-bbox="527 513 1396 643" style="list-style-type: none"><li data-bbox="527 513 976 537">1. In the <b>search bar</b> enter your destination.</li><li data-bbox="527 561 1396 586">2. Pick your navigation type (drive, transit, walk) and press the blue <b>navigate</b> button.</li><li data-bbox="527 610 1396 634">3. Tap the arrow next to the <b>time-to-destination</b> number at the bottom of the screen.</li></ol>  <p data-bbox="512 1328 1356 1360"><a href="https://www.androidcentral.com/how-share-location-google-maps">https://www.androidcentral.com/how-share-location-google-maps</a></p>



**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC
	<p data-bbox="527 277 835 305">4. Tap Share trip progress.</p> <p data-bbox="527 334 1150 362">5. Choose one or more contacts to share trip progress.</p>  <p data-bbox="527 1065 1339 1092">You can also stop sharing your location with someone before a trip ends.</p> <p data-bbox="512 1101 1356 1128"><a href="https://www.androidcentral.com/how-share-location-google-maps">https://www.androidcentral.com/how-share-location-google-maps</a></p>

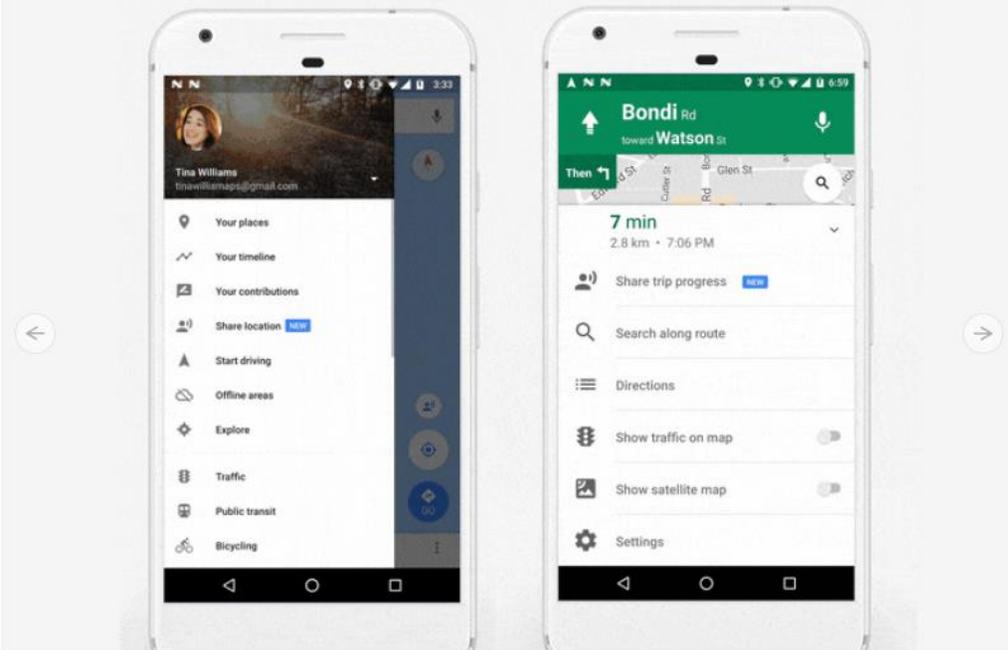
## Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products

US9408055B2	HTC
	<ol style="list-style-type: none"><li data-bbox="533 245 1465 272">1. Tap the arrow next to the <b>time-to-destination</b> number at the bottom of the screen.</li><li data-bbox="533 302 768 329">2. Tap Stop sharing.</li></ol> <div data-bbox="747 383 1398 922"></div> <p data-bbox="541 976 636 1003">That's it!</p> <p data-bbox="541 1045 1608 1073">Are you excited that location sharing is back in Google Maps? How often do you use the feature?</p> <p data-bbox="512 1084 1356 1117"><a href="https://www.androidcentral.com/how-share-location-google-maps">https://www.androidcentral.com/how-share-location-google-maps</a></p> <p data-bbox="512 1195 1419 1227">As shown below, a group may also be defined within Google Contacts.</p>

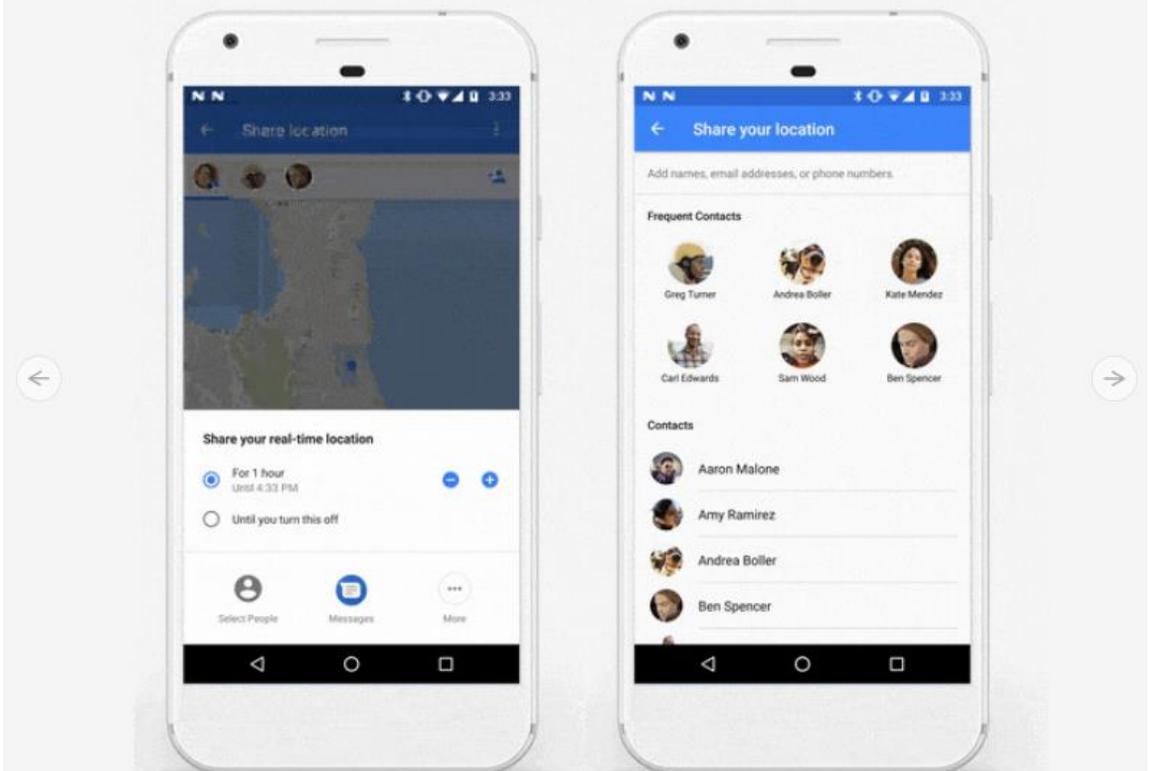
## Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products

US9408055B2	HTC
	<h3 data-bbox="541 237 894 280">See your contacts</h3> <ol data-bbox="552 305 968 378" style="list-style-type: none"><li data-bbox="552 305 968 337">1. Open your device's Contacts app .</li><li data-bbox="552 350 730 378">2. Tap Menu .</li></ol> <ul data-bbox="546 410 1738 605" style="list-style-type: none"><li data-bbox="546 410 1113 438">• <b>See contacts by label:</b> Choose a label from the list.</li><li data-bbox="546 454 1365 482">• <b>See contacts for another account:</b> Tap Down arrow  &gt; pick an account.</li><li data-bbox="546 498 1213 526">• <b>See the contacts for all your accounts:</b> Choose <b>All contacts</b>.</li></ul> <p data-bbox="569 537 1738 565"><b>Tip:</b> If you have multiple contacts with the same information, the information will be grouped into one contact.</p> <ul data-bbox="546 581 1360 605" style="list-style-type: none"><li data-bbox="546 581 1360 605">• <b>See your Google Account contacts on the web:</b> Go to <a href="#">Google Contacts</a> .</li></ul> <p data-bbox="512 638 1535 665"><a href="https://support.google.com/android/answer/6118731?hl=en&amp;ref_topic=6118711">https://support.google.com/android/answer/6118731?hl=en&amp;ref_topic=6118711</a></p> <h3 data-bbox="541 714 894 758">Label your contacts</h3> <p data-bbox="541 779 984 807">You can group contacts together using labels.</p> <ol data-bbox="552 831 930 938" style="list-style-type: none"><li data-bbox="552 831 930 863">1. Open your device's Contacts app .</li><li data-bbox="552 875 863 902">2. Tap Menu  &gt; <b>Create label</b>.</li><li data-bbox="552 914 873 941">3. Enter a label name and tap <b>Ok</b>.</li></ol> <ul data-bbox="546 967 1717 1032" style="list-style-type: none"><li data-bbox="546 967 1234 995">• <b>Add one contact to a label:</b> Tap Add contact  &gt; choose a contact.</li><li data-bbox="546 1008 1717 1032">• <b>Add multiple contacts to a label:</b> Tap Add contact  &gt; touch and hold a contact &gt; tap the other contacts &gt; tap <b>Add</b>.</li></ul> <p data-bbox="512 1044 1535 1071"><a href="https://support.google.com/android/answer/6118731?hl=en&amp;ref_topic=6118711">https://support.google.com/android/answer/6118731?hl=en&amp;ref_topic=6118711</a></p> <h3 data-bbox="552 1130 940 1174">Share your contacts</h3> <ol data-bbox="562 1198 1045 1360" style="list-style-type: none"><li data-bbox="562 1198 978 1230">1. Open your device's Contacts app .</li><li data-bbox="562 1242 842 1269">2. Tap a contact in the list.</li><li data-bbox="562 1286 831 1313">3. Tap More  &gt; <b>Share</b>.</li><li data-bbox="562 1330 1045 1360">4. Choose how you want to share the contact.</li></ol> <p data-bbox="512 1378 1535 1406"><a href="https://support.google.com/android/answer/6118731?hl=en&amp;ref_topic=6118711">https://support.google.com/android/answer/6118731?hl=en&amp;ref_topic=6118711</a></p>

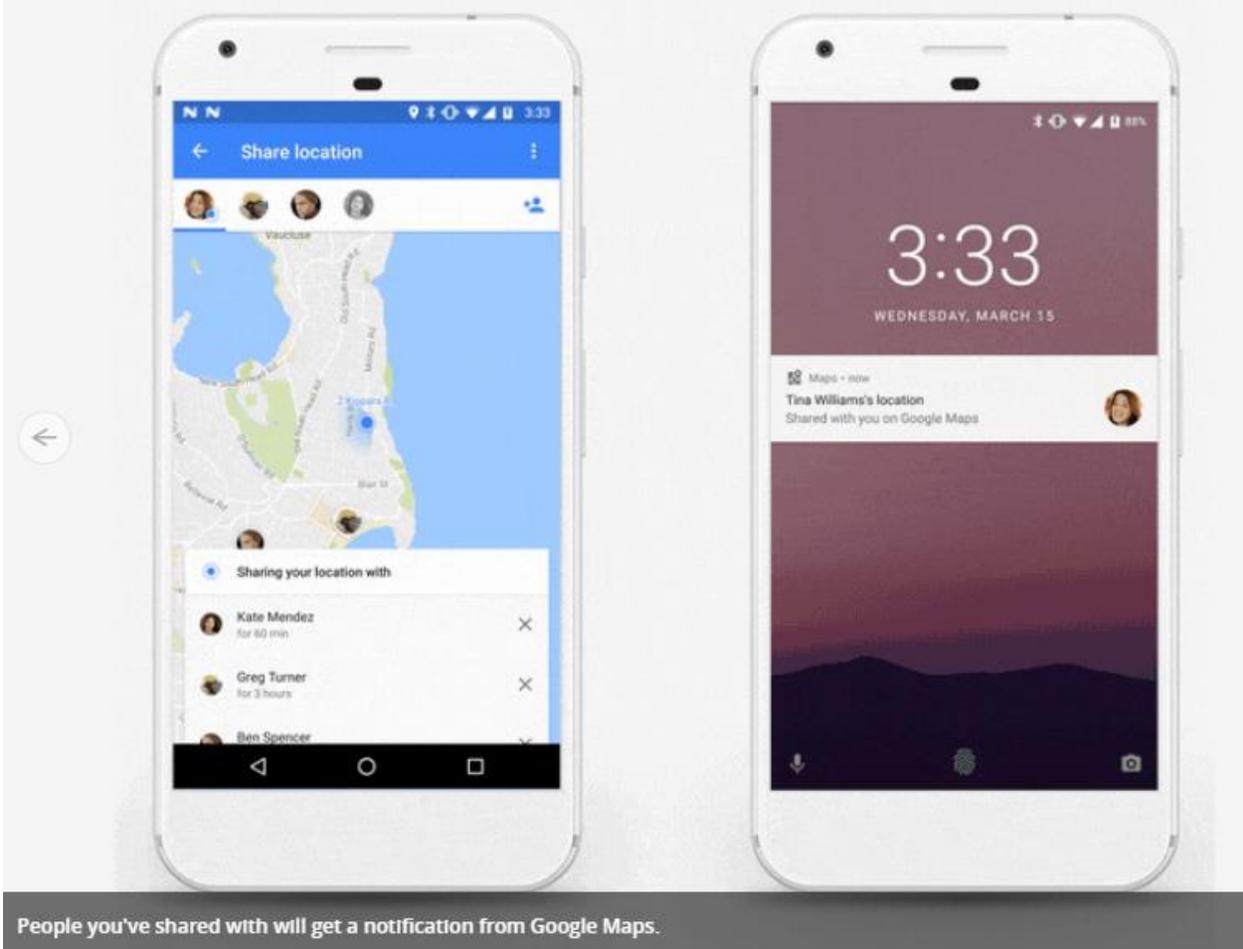
**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC
	 <p>Ways to start location sharing: (left) from the app navigation drawer, and (right) from the menu, you can drag up while navigating somewhere.</p> <p><a href="https://arstechnica.com/gadgets/2017/03/location-sharing-finally-returns-to-google-maps/">https://arstechnica.com/gadgets/2017/03/location-sharing-finally-returns-to-google-maps/</a></p>

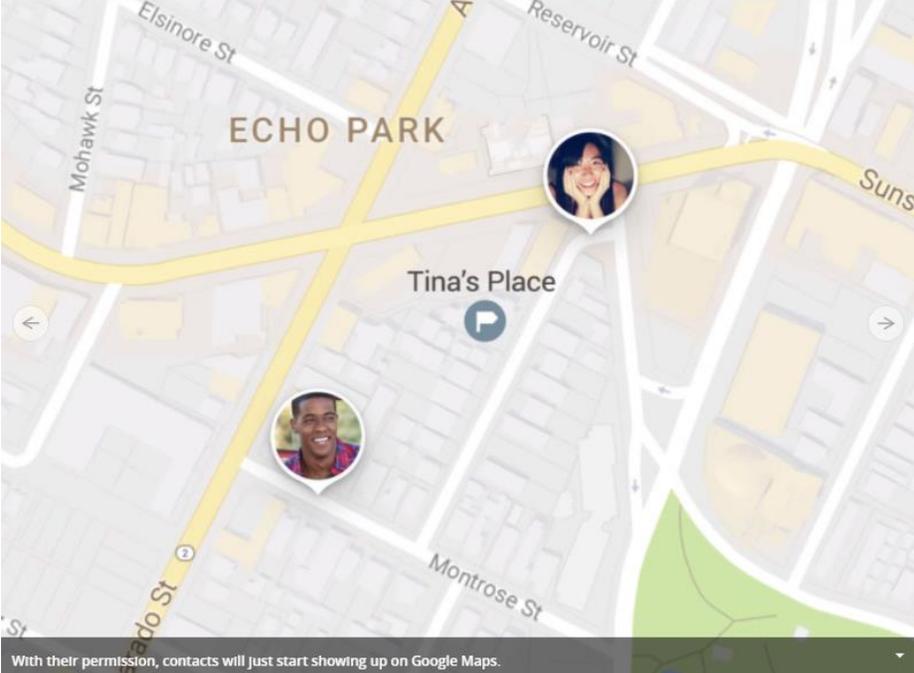
**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC
	 <p data-bbox="512 1008 1661 1057">You can pick how long you want to share your location for, and with whom you want to share it.</p> <p data-bbox="512 1057 1661 1096"><a href="https://arstechnica.com/gadgets/2017/03/location-sharing-finally-returns-to-google-maps/">https://arstechnica.com/gadgets/2017/03/location-sharing-finally-returns-to-google-maps/</a></p>

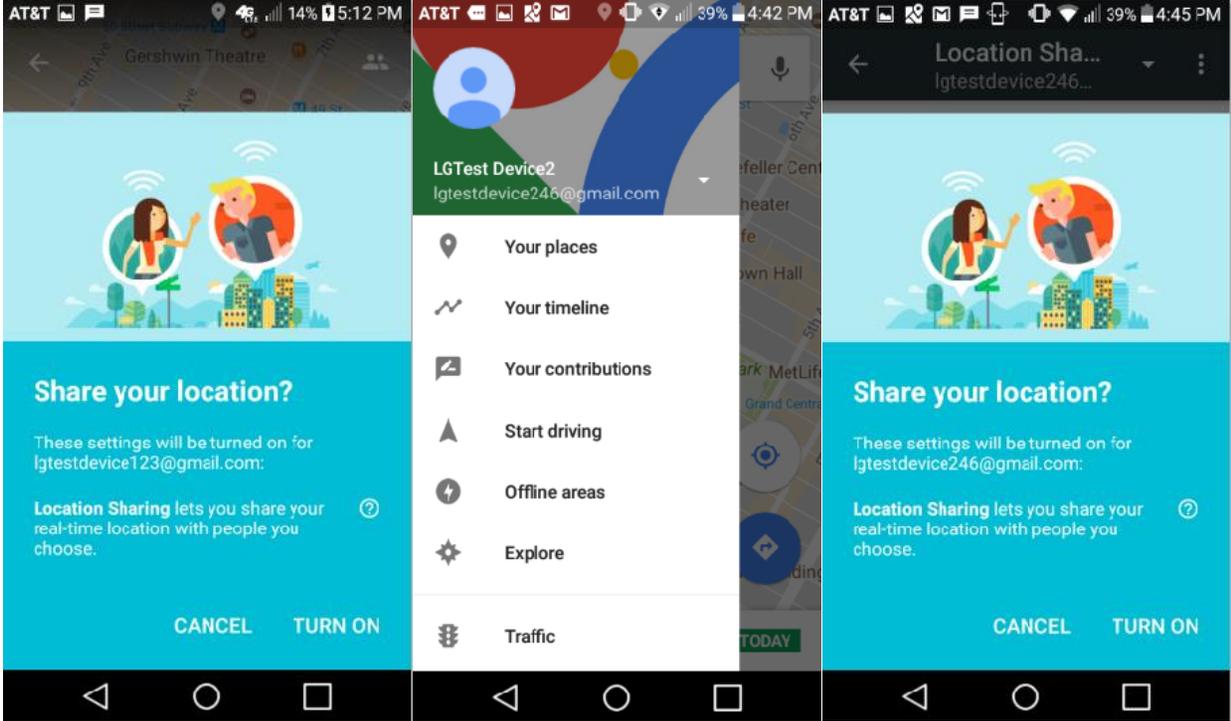
**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC
	 <p data-bbox="512 1143 1176 1170">People you've shared with will get a notification from Google Maps.</p> <p data-bbox="512 1187 1656 1219"><a href="https://arstechnica.com/gadgets/2017/03/location-sharing-finally-returns-to-google-maps/">https://arstechnica.com/gadgets/2017/03/location-sharing-finally-returns-to-google-maps/</a></p>

**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC
	 <p data-bbox="514 914 1654 946"><a href="https://arstechnica.com/gadgets/2017/03/location-sharing-finally-returns-to-google-maps/">https://arstechnica.com/gadgets/2017/03/location-sharing-finally-returns-to-google-maps/</a></p> <p data-bbox="514 987 1020 1019"><b><u>Exemplary Google Maps Screenshots</u></b></p>

**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC
	 <p>Exemplary Source Code:          The above functionality is performed at least in part by the following publicly available source code and/or source code that invokes or is invoked by the following source code (or a substantially similar copy compiled and loaded onto the Accused Products by HTC): AGIS reserves the right to amend these contentions to include additional source code as discovery progresses and as additional source code is made available.</p>



**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC
	<h2 data-bbox="527 240 1016 293">Contacts Provider</h2> <p data-bbox="527 329 1472 591">The Contacts Provider is a powerful and flexible Android component that manages the device's central repository of data about people. The Contacts Provider is the source of data you see in the device's contacts application, and you can also access its data in your own application and transfer data between the device and online services. The provider accommodates a wide range of data sources and tries to manage as much data as possible for each person, with the result that its organization is complex. Because of this, the provider's API includes an extensive set of contract classes and interfaces that facilitate both data retrieval and modification.</p> <p data-bbox="527 626 835 646">This guide describes the following:</p> <ul data-bbox="527 675 1373 850" style="list-style-type: none"><li data-bbox="527 675 806 695">• The basic provider structure.</li><li data-bbox="527 724 894 743">• How to retrieve data from the provider.</li><li data-bbox="527 773 863 792">• How to modify data in the provider.</li><li data-bbox="527 821 1373 841">• How to write a sync adapter for synchronizing data from your server to the Contacts Provider.</li></ul> <p data-bbox="512 863 1486 889"><a href="https://developer.android.com/guide/topics/providers/contacts-provider.html">https://developer.android.com/guide/topics/providers/contacts-provider.html</a></p>

**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC
	<p><b>Overview</b></p> <p>ContactsContract defines an extensible database of contact-related information. Contact information is stored in a three-tier data model:</p> <ul style="list-style-type: none"> <li>• A row in the <code>ContactsContract.Data</code> table can store any kind of personal data, such as a phone number or email addresses. The set of data kinds that can be stored in this table is open-ended. There is a predefined set of common kinds, but any application can add its own data kinds.</li> <li>• A row in the <code>ContactsContract.RawContacts</code> table represents a set of data describing a person and associated with a single account (for example, one of the user's Gmail accounts).</li> <li>• A row in the <code>ContactsContract.Contacts</code> table represents an aggregate of one or more RawContacts presumably describing the same person. When data in or associated with the RawContacts table is changed, the affected aggregate contacts are updated as necessary.</li> </ul> <p>Other tables include:</p> <ul style="list-style-type: none"> <li>• <code>ContactsContract.Groups</code>, which contains information about raw contact groups such as Gmail contact groups. The current API does not support the notion of groups spanning multiple accounts.</li> <li>• <code>ContactsContract.StatusUpdates</code>, which contains social status updates including IM availability.</li> <li>• <code>ContactsContract.AggregationExceptions</code>, which is used for manual aggregation and disaggregation of raw contacts</li> <li>• <code>ContactsContract.Settings</code>, which contains visibility and sync settings for accounts and groups.</li> <li>• <code>ContactsContract.SyncState</code>, which contains free-form data maintained on behalf of sync adapters</li> <li>• <code>ContactsContract.PhoneLookup</code>, which is used for quick caller-ID lookup</li> </ul> <p><a href="https://developer.android.com/reference/android/provider/ContactsContract.html">https://developer.android.com/reference/android/provider/ContactsContract.html</a></p> <p><b>Data</b></p> <p>As noted previously, the data for a raw contact is stored in a <code>ContactsContract.Data</code> row that is linked to the raw contact's <code>_id</code> value. This allows a single raw contact to have multiple instances of the same type of data such as email addresses or phone numbers. For example, if "Thomas Higginson" for <code>emilyd@gmail.com</code> (the raw contact row for Thomas Higginson associated with the Google account <code>emilyd@gmail.com</code>) has a home email address of <code>thigg@gmail.com</code> and a work email address of <code>thomas.higginson@gmail.com</code>, the Contacts Provider stores the two email address rows and links them both to the raw contact.</p> <p>Notice that different types of data are stored in this single table. Display name, phone number, email, postal address, photo, and website detail rows are all found in the <code>ContactsContract.Data</code> table. To help manage this, the <code>ContactsContract.Data</code> table has some columns with descriptive names, and others with generic names. The contents of a descriptive-name column have the same meaning regardless of the type of data in the row, while the contents of a generic-name column have different meanings depending on the type of data.</p> <p><a href="https://developer.android.com/guide/topics/providers/contacts-provider.html">https://developer.android.com/guide/topics/providers/contacts-provider.html</a></p>

**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC														
	<table border="1"> <thead> <tr> <th>Task</th> <th>Action</th> <th>Data</th> <th>MIME type</th> <th>Notes</th> </tr> </thead> <tbody> <tr> <td>Pick a contact from a list</td> <td><a href="#">ACTION_PICK</a></td> <td>                     One of:                     <ul style="list-style-type: none"> <li>• <a href="#">Contacts.CONTENT_URI</a>, which displays a list of contacts.</li> <li>• <a href="#">Phone.CONTENT_URI</a>, which displays a list of phone numbers for a raw contact.</li> <li>• <a href="#">StructuredPostal.CONTENT_URI</a>, which displays a list of postal addresses for a raw contact.</li> <li>• <a href="#">Email.CONTENT_URI</a>, which displays a list of email addresses for a raw contact.</li> </ul> </td> <td>Not used</td> <td>                     Displays a list of raw contacts or a list of data from a raw contact, depending on the content URI type you supply. Call <a href="#">startActivityForResult()</a>, which returns the content URI of the selected row. The form of the URI is the table's content URI with the row's <a href="#">LOOKUP_ID</a> appended to it. The device's contacts app delegates read and write permissions to this content URI for the life of your activity. See the <a href="#">Content Provider Basics</a> guide for more details.                 </td> </tr> </tbody> </table>	Task	Action	Data	MIME type	Notes	Pick a contact from a list	<a href="#">ACTION_PICK</a>	One of: <ul style="list-style-type: none"> <li>• <a href="#">Contacts.CONTENT_URI</a>, which displays a list of contacts.</li> <li>• <a href="#">Phone.CONTENT_URI</a>, which displays a list of phone numbers for a raw contact.</li> <li>• <a href="#">StructuredPostal.CONTENT_URI</a>, which displays a list of postal addresses for a raw contact.</li> <li>• <a href="#">Email.CONTENT_URI</a>, which displays a list of email addresses for a raw contact.</li> </ul>	Not used	Displays a list of raw contacts or a list of data from a raw contact, depending on the content URI type you supply. Call <a href="#">startActivityForResult()</a> , which returns the content URI of the selected row. The form of the URI is the table's content URI with the row's <a href="#">LOOKUP_ID</a> appended to it. The device's contacts app delegates read and write permissions to this content URI for the life of your activity. See the <a href="#">Content Provider Basics</a> guide for more details.				
Task	Action	Data	MIME type	Notes											
Pick a contact from a list	<a href="#">ACTION_PICK</a>	One of: <ul style="list-style-type: none"> <li>• <a href="#">Contacts.CONTENT_URI</a>, which displays a list of contacts.</li> <li>• <a href="#">Phone.CONTENT_URI</a>, which displays a list of phone numbers for a raw contact.</li> <li>• <a href="#">StructuredPostal.CONTENT_URI</a>, which displays a list of postal addresses for a raw contact.</li> <li>• <a href="#">Email.CONTENT_URI</a>, which displays a list of email addresses for a raw contact.</li> </ul>	Not used	Displays a list of raw contacts or a list of data from a raw contact, depending on the content URI type you supply. Call <a href="#">startActivityForResult()</a> , which returns the content URI of the selected row. The form of the URI is the table's content URI with the row's <a href="#">LOOKUP_ID</a> appended to it. The device's contacts app delegates read and write permissions to this content URI for the life of your activity. See the <a href="#">Content Provider Basics</a> guide for more details.											
<a href="https://developer.android.com/guide/topics/providers/contacts-provider.html">https://developer.android.com/guide/topics/providers/contacts-provider.html</a>															

**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC
	<pre>25  * Parsed form of the intent sent to the Contacts application. 26  */ 27  public class ContactsRequest { 28 29      /** Default mode: browse contacts */ 30      public static final int ACTION_DEFAULT = 10; 31 32      /** Show all contacts */ 33      public static final int ACTION_ALL_CONTACTS = 15; 34 35      /** Show all contacts with phone numbers */ 36      public static final int ACTION_CONTACTS_WITH_PHONES = 17; 37 38      /** Show contents of a specific group */ 39      public static final int ACTION_GROUP = 20; 40 41      /** Show all starred contacts */ 42      public static final int ACTION_STARRED = 30; 43 44      /** Show frequently contacted contacts */ 45      public static final int ACTION_FREQUENT = 40; 46 47      /** Show starred and the frequent */ 48      public static final int ACTION_STREQUENT = 50; 49 50      /** Show all contacts and pick them when clicking */ 51      public static final int ACTION_PICK_CONTACT = 60; 52 53      /** Show all contacts as well as the option to create a new one */ 54      public static final int ACTION_PICK_OR_CREATE_CONTACT = 70; 55 56      /** Show all contacts and pick them for edit when clicking, and allow creating a new contact */ 57      public static final int ACTION_INSERT_OR_EDIT_CONTACT = 80;</pre> <p><a href="https://android.googlesource.com/platform/packages/apps/Contacts/+nougat-mr1-release/src/com/android/contacts/list/ContactsRequest.java">https://android.googlesource.com/platform/packages/apps/Contacts/+nougat-mr1-release/src/com/android/contacts/list/ContactsRequest.java</a></p>

**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC
	<pre>59  /** Show all phone numbers and pick them when clicking */ 60  public static final int ACTION_PICK_PHONE = 90; 61 62  /** Show all postal addresses and pick them when clicking */ 63  public static final int ACTION_PICK_POSTAL = 100; 64 65  /** Show all postal addresses and pick them when clicking */ 66  public static final int ACTION_PICK_EMAIL = 105; 67 68  /** Show all contacts and create a shortcut for the picked contact */ 69  public static final int ACTION_CREATE_SHORTCUT_CONTACT = 110; 70 71  /** Show all phone numbers and create a call shortcut for the picked number */ 72  public static final int ACTION_CREATE_SHORTCUT_CALL = 120; 73 74  /** Show all phone numbers and create an SMS shortcut for the picked number */ 75  public static final int ACTION_CREATE_SHORTCUT_SMS = 130; 76 77  /** Show all contacts and activate the specified one */ 78  public static final int ACTION_VIEW_CONTACT = 140; 79 80  /** Show contacts recommended for joining with a specified target contact */ 81  public static final int ACTION_PICK_JOIN = 150;</pre> <p><a href="https://android.googlesource.com/platform/packages/apps/Contacts+/nougat-mr1-release/src/com/android/contacts/list/ContactsRequest.java">https://android.googlesource.com/platform/packages/apps/Contacts+/nougat-mr1-release/src/com/android/contacts/list/ContactsRequest.java</a></p>

**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC
	<pre> 104  * Displays a list to browse contacts. 105  */ 106  public class PeopleActivity extends ContactsActivity implements 107      View.OnCreateContextMenuListener, 108      View.OnClickListener, 109      ActionBarAdapter.Listener, 110      DialogManager.DialogShowingViewActivity, 111      ContactListFilterController.ContactListFilterListener, 112      ProviderStatusListener, 113      MultiContactDeleteListener, 114      JoinContactsListener { </pre> <p data-bbox="506 662 1570 727"><a href="https://android.googlesource.com/platform/packages/apps/Contacts+/nougat-mr1-release/src/com/android/contacts/activities/PeopleActivity.java">https://android.googlesource.com/platform/packages/apps/Contacts+/nougat-mr1-release/src/com/android/contacts/activities/PeopleActivity.java</a></p> <pre> 145  * Showing a list of Contacts. Also used for showing search results in search mode. 146  */ 147  private MultiSelectContactsListFragment mAllFragment; 148  private ContactTileListFragment mFavoritesFragment; </pre> <p data-bbox="506 873 1570 938"><a href="https://android.googlesource.com/platform/packages/apps/Contacts+/nougat-mr1-release/src/com/android/contacts/activities/PeopleActivity.java">https://android.googlesource.com/platform/packages/apps/Contacts+/nougat-mr1-release/src/com/android/contacts/activities/PeopleActivity.java</a></p>

**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC
	<pre>458     private void configureFragments(boolean fromRequest) { 459         if (fromRequest) { 460             ContactListFilter filter = null; 461             int actionCode = mRequest.getActionCode(); 462             boolean searchMode = mRequest.isSearchMode(); 463             final int tabToOpen; 464             switch (actionCode) { 465                 case ContactsRequest.ACTION_ALL_CONTACTS: 466                     filter = ContactListFilter.createFilterWithType( 467                         ContactListFilter.FILTER_TYPE_ALL_ACCOUNTS); 468                     tabToOpen = TabState.ALL; 469                     break; 470                 case ContactsRequest.ACTION_CONTACTS_WITH_PHONES: 471                     filter = ContactListFilter.createFilterWithType( 472                         ContactListFilter.FILTER_TYPE_WITH_PHONE_NUMBERS_ONLY); 473                     tabToOpen = TabState.ALL; 474                     break; 475 476                 case ContactsRequest.ACTION_FREQUENT: 477                 case ContactsRequest.ACTION_STREQUENT: 478                 case ContactsRequest.ACTION_STARRED: 479                     tabToOpen = TabState.FAVORITES; 480                     break; 481                 case ContactsRequest.ACTION_VIEW_CONTACT: 482                     tabToOpen = TabState.ALL; 483                     break; 484                 default: 485                     tabToOpen = -1; 486                     break; 487             } </pre> <p data-bbox="506 1321 1566 1390"><a href="https://android.googlesource.com/platform/packages/apps/Contacts+/nougat-mr1-release/src/com/android/contacts/activities/PeopleActivity.java">https://android.googlesource.com/platform/packages/apps/Contacts+/nougat-mr1-release/src/com/android/contacts/activities/PeopleActivity.java</a></p>

**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC
	<pre>488         if (tabToOpen != -1) { 489             mActionBarAdapter.setCurrentTab(tabToOpen); 490         } 491 492         if (filter != null) { 493             mContactListFilterController.setContactListFilter(filter, false); 494             searchMode = false; 495         } 496 497         if (mRequest.getContactUri() != null) { 498             searchMode = false; 499         } 500 501         mActionBarAdapter.setSearchMode(searchMode); 502         configureContactListFragmentForRequest(); 503     } 504 505     configureContactListFragment(); 506 507     invalidateOptionsMenuIfNeeded(); 508 }</pre> <p data-bbox="506 1019 1570 1084"><a href="https://android.googlesource.com/platform/packages/apps/Contacts+/nougat-mr1-release/src/com/android/contacts/activities/PeopleActivity.java">https://android.googlesource.com/platform/packages/apps/Contacts+/nougat-mr1-release/src/com/android/contacts/activities/PeopleActivity.java</a></p>



**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC
	<pre>35 public class ProfileAndContactsLoader extends CursorLoader { 36 37     private boolean mLoadProfile; 38 39     private String[] mProjection; 40 41     private Uri mExtraUri; 42     private String[] mExtraProjection; 43     private String mExtraSelection; 44     private String[] mExtraSelectionArgs; 45     private boolean mMergeExtraContactsAfterPrimary; 46 47     public ProfileAndContactsLoader(Context context) { 48         super(context); 49     }</pre> <p><a href="https://android.googlesource.com/platform/packages/apps/ContactsCommon/+/nougat-mr1-release/src/com/android/contacts/common/list/ProfileAndContactsLoader.java">https://android.googlesource.com/platform/packages/apps/ContactsCommon/+/nougat-mr1-release/src/com/android/contacts/common/list/ProfileAndContactsLoader.java</a></p>

**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC
	<pre>35 public final class GroupMemberLoader extends CursorLoader { 36 37     public static class GroupEditorQuery { 38         private static final String[] PROJECTION = new String[] { 39             Data.CONTACT_ID,           // 0 40             Data.RAW_CONTACT_ID,      // 1 41             Data.DISPLAY_NAME_PRIMARY, // 2 42             Data.PHOTO_URI,           // 3 43             Data.LOOKUP_KEY,          // 4 44         }; 45 46         public static final int CONTACT_ID           = 0; 47         public static final int RAW_CONTACT_ID      = 1; 48         public static final int CONTACT_DISPLAY_NAME_PRIMARY = 2; 49         public static final int CONTACT_PHOTO_URI   = 3; 50         public static final int CONTACT_LOOKUP_KEY   = 4; 51     } 52 53     public static class GroupDetailQuery { 54         private static final String[] PROJECTION = new String[] { 55             Data.CONTACT_ID,           // 0 56             Data.PHOTO_URI,           // 1 57             Data.LOOKUP_KEY,          // 2 58             Data.DISPLAY_NAME_PRIMARY, // 3 59             Data.CONTACT_PRESENCE,    // 4 60             Data.CONTACT_STATUS,      // 5 61         }; 62 63         public static final int CONTACT_ID           = 0; 64         public static final int CONTACT_PHOTO_URI   = 1; 65         public static final int CONTACT_LOOKUP_KEY   = 2; 66         public static final int CONTACT_DISPLAY_NAME_PRIMARY = 3; 67         public static final int CONTACT_PRESENCE_STATUS = 4; 68         public static final int CONTACT_STATUS      = 5; 69     } 70 71     private final long mGroupId;</pre> <p><a href="https://android.googlesource.com/platform/packages/apps/Contacts/+nougat-mr1-release/src/com/android/contacts/GroupMemberLoader.java">https://android.googlesource.com/platform/packages/apps/Contacts/+nougat-mr1-release/src/com/android/contacts/GroupMemberLoader.java</a></p>

**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC
	<pre>24 * Group loader for the group list that includes details such as the number of contacts per group 25 * and number of groups per account. This list is sorted by account type, account name, where the 26 * group names are in alphabetical order. Note that the list excludes default, favorite, and deleted 27 * groups. 28 */ 29 public final class GroupListLoader extends CursorLoader { 30 31     private final static String[] COLUMNS = new String[] { 32         Groups.ACCOUNT_NAME, 33         Groups.ACCOUNT_TYPE, 34         Groups.DATA_SET, 35         Groups._ID, 36         Groups.TITLE, 37         Groups.SUMMARY_COUNT, 38     }; 39 40     public final static int ACCOUNT_NAME = 0; 41     public final static int ACCOUNT_TYPE = 1; 42     public final static int DATA_SET = 2; 43     public final static int GROUP_ID = 3; 44     public final static int TITLE = 4; 45     public final static int MEMBER_COUNT = 5; 46 47     private static final Uri GROUP_LIST_URI = Groups.CONTENT_SUMMARY_URI; 48 49     public GroupListLoader(Context context) { 50         super(context, GROUP_LIST_URI, COLUMNS, Groups.ACCOUNT_TYPE + " NOT NULL AND " 51             + Groups.ACCOUNT_NAME + " NOT NULL AND " + Groups.AUTO_ADD + "=0 AND " + 52             Groups.FAVORITES + "=0 AND " + Groups.DELETED + "=0", null, 53             Groups.ACCOUNT_TYPE + ", " + Groups.ACCOUNT_NAME + ", " + Groups.DATA_SET + ", " + 54             Groups.TITLE + " COLLATE LOCALIZED ASC"); 55     } 56 }</pre> <p><a href="https://android.googlesource.com/platform/packages/apps/Contacts+/nougat-mr1-release/src/com/android/contacts/GroupListLoader.java">https://android.googlesource.com/platform/packages/apps/Contacts+/nougat-mr1-release/src/com/android/contacts/GroupListLoader.java</a></p>

**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC
	<pre>24  * Group meta-data loader. Loads all groups or just a single group from the 25  * database (if given a {@link Uri}). 26  */ 27  public final class GroupMetaLoader extends CursorLoader { 28 29      private final static String[] COLUMNS = new String[] { 30          Groups.ACCOUNT_NAME, 31          Groups.ACCOUNT_TYPE, 32          Groups.DATA_SET, 33          Groups._ID, 34          Groups.TITLE, 35          Groups.AUTO_ADD, 36          Groups.FAVORITES, 37          Groups.GROUP_IS_READ_ONLY, 38          Groups.DELETED, 39      }; 40 41      public final static int ACCOUNT_NAME = 0; 42      public final static int ACCOUNT_TYPE = 1; 43      public final static int DATA_SET = 2; 44      public final static int GROUP_ID = 3; 45      public final static int TITLE = 4; 46      public final static int AUTO_ADD = 5; 47      public final static int FAVORITES = 6; 48      public final static int IS_READ_ONLY = 7; 49      public final static int DELETED = 8; 50 51      public GroupMetaLoader(Context context, Uri groupUri) { 52          super(context, ensureIsGroupUri(groupUri), COLUMNS, Groups.ACCOUNT_TYPE + " NOT NULL AND " 53              + Groups.ACCOUNT_NAME + " NOT NULL", null, null); 54      } 55 56      /** 57       * Ensures that this is a valid group URI. If invalid, then an exception is 58       * thrown. Otherwise, the original URI is returned. 59       */ 60      private static Uri ensureIsGroupUri(final Uri groupUri) { 61          // TODO: Fix ContactsProvider2 getType method to resolve the group Uris 62          if (groupUri == null) { 63              throw new IllegalArgumentException("Uri must not be null"); 64          } 65          if (!groupUri.toString().startsWith(Groups.CONTENT_URI.toString())) { 66              throw new IllegalArgumentException("Invalid group Uri: " + groupUri); 67          } 68          return groupUri; 69      } 70  }</pre> <p><a href="https://android.googlesource.com/platform/packages/apps/Contacts/+nougat-mr1-release/src/com/android/contacts/GroupMetaLoader.java">https://android.googlesource.com/platform/packages/apps/Contacts/+nougat-mr1-release/src/com/android/contacts/GroupMetaLoader.java</a></p>

**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC
	<pre>19  * Meta-data for a contact group. We load all groups associated with the contact's 20  * constituent accounts. 21  */ 22  public final class GroupMetaData { 23      private String mAccountName; 24      private String mAccountType; 25      private String mDataSet; 26      private long mGroupId; 27      private String mTitle; 28      private boolean mDefaultGroup; 29      private boolean mFavorites; 30 31      public GroupMetaData(String accountName, String accountType, String dataSet, long groupId, 32                          String title, boolean defaultGroup, boolean favorites) { 33          this.mAccountName = accountName; 34          this.mAccountType = accountType; 35          this.mDataSet = dataSet; 36          this.mGroupId = groupId; 37          this.mTitle = title; 38          this.mDefaultGroup = defaultGroup; 39          this.mFavorites = favorites; 40      } 41 42      public String getAccountName() { 43          return mAccountName; 44      } 45 46      public String getAccountType() { 47          return mAccountType; 48      } 49 50      public String getDataSet() { 51          return mDataSet; 52      } 53 54      public long getGroupId() { 55          return mGroupId; 56      } 57 58      public String getTitle() { 59          return mTitle; 60      } 61 62      public boolean isDefaultGroup() { 63          return mDefaultGroup; 64      } 65 66      public boolean isFavorites() { 67          return mFavorites; 68      } 69  }</pre> <p><a href="https://android.googlesource.com/platform/packages/apps/ContactsCommon/+nougat-mr1-release/src/com/android/contacts/common/GroupMetaData.java">https://android.googlesource.com/platform/packages/apps/ContactsCommon/+nougat-mr1-release/src/com/android/contacts/common/GroupMetaData.java</a></p>

**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC
	<pre>44 * Class that sends chat message via SMS. 45 * 46 * The interface emulates a blocking sending similar to making an HTTP request. 47 * It calls the SmsManager to send a (potentially multipart) message and waits 48 * on the sent status on each part. The waiting has a timeout so it won't wait 49 * forever. Once the sent status of all parts received, the call returns. 50 * A successful sending requires success status for all parts. Otherwise, we 51 * pick the highest level of failure as the error for the whole message, which 52 * is used to determine if we need to retry the sending. 53 */ 54 public class SmsSender { 55     private static final String TAG = LogUtil.BUGLE_TAG; 56 57     public static final String EXTRA_PART_ID = "part_id"; 58 59     /* 60     * A map for pending sms messages. The key is the random request UUID. 61     */ 62     private static ConcurrentHashMap&lt;Uri, SendResult&gt; sPendingMessageMap = 63         new ConcurrentHashMap&lt;Uri, SendResult&gt;(); 64 65     private static final Random RANDOM = new Random(); 66 67     // Whether we should send multipart SMS as separate messages 68     private static Boolean sSendMultipartSmsAsSeparateMessages = null; 69</pre> <p><a href="https://android.googlesource.com/platform/packages/apps/Messaging/+nougat-mr1-release/src/com/android/messaging/sms/SmsSender.java">https://android.googlesource.com/platform/packages/apps/Messaging/+nougat-mr1-release/src/com/android/messaging/sms/SmsSender.java</a></p>

**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC
	<pre>253 // Actually sending the message using SmsManager 254 private static void sendInternal(final Context context, final int subId, String dest, 255     final ArrayList&lt;String&gt; messages, final String serviceCenter, 256     final boolean requireDeliveryReport, final Uri messageUri) throws SmsException { 257     Assert.notNull(context); 258     final SmsManager smsManager = PhoneUtils.get(subId).getSmsManager(); 259     final int messageCount = messages.size(); 260     final ArrayList&lt;PendingIntent&gt; deliveryIntents = new ArrayList&lt;PendingIntent&gt;(messageCount); 261     final ArrayList&lt;PendingIntent&gt; sentIntents = new ArrayList&lt;PendingIntent&gt;(messageCount); 262     for (int i = 0; i &lt; messageCount; i++) { 263         // Make pending intents different for each message part 264         final int partId = (messageCount &lt;= 1 ? 0 : i + 1); 265         if (requireDeliveryReport &amp;&amp; (i == (messageCount - 1))) { 266             // TODO we only care about the delivery status of the last part 267             // Shall we have better tracking of delivery status of all parts? 268             deliveryIntents.add(PendingIntent.getBroadcast( 269                 context, 270                 partId, 271                 getSendStatusIntent(context, SendStatusReceiver.MESSAGE_DELIVERED_ACTION, 272                     messageUri, partId, subId), 273                 0/*flag*/)); 274         } else { 275             deliveryIntents.add(null); 276         } 277         sentIntents.add(PendingIntent.getBroadcast( 278             context, 279             partId, 280             getSendStatusIntent(context, SendStatusReceiver.MESSAGE_SENT_ACTION, 281                 messageUri, partId, subId), 282             0/*flag*/)); 283     } 284     if (sSendMultipartSmsAsSeparateMessages == null) { 285         sSendMultipartSmsAsSeparateMessages = MmsConfig.get(subId) 286             .getSendMultipartSmsAsSeparateMessages(); 287     }</pre> <p><a href="https://android.googlesource.com/platform/packages/apps/Messaging/+nougat-mr1-release/src/com/android/messaging/sms/SmsSender.java">https://android.googlesource.com/platform/packages/apps/Messaging/+nougat-mr1-release/src/com/android/messaging/sms/SmsSender.java</a></p>

**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC
	<pre>288     try { 289         if (sSendMultipartSmsAsSeparateMessages) { 290             // If multipart sms is not supported, send them as separate messages 291             for (int i = 0; i &lt; messageCount; i++) { 292                 smsManager.sendTextMessage(dest, 293                     serviceCenter, 294                     messages.get(i), 295                     sentIntents.get(i), 296                     deliveryIntents.get(i)); 297             } 298         } else { 299             smsManager.sendMultipartTextMessage( 300                 dest, serviceCenter, messages, sentIntents, deliveryIntents); 301         } 302     } catch (final Exception e) { 303         throw new SmsException("SmsSender: caught exception in sending " + e); 304     } 305 }</pre> <p><a href="https://android.googlesource.com/platform/packages/apps/Messaging/+nougat-mr1-release/src/com/android/messaging/sms/SmsSender.java">https://android.googlesource.com/platform/packages/apps/Messaging/+nougat-mr1-release/src/com/android/messaging/sms/SmsSender.java</a></p> <pre>56  * Class that receives incoming SMS messages through android.provider.Telephony.SMS_RECEIVED 57  * 58  * This class serves two purposes: 59  * - Process phone verification SMS messages 60  * - Handle SMS messages when the user has enabled us to be the default SMS app (Pre-KLP) 61  */ 62  public final class SmsReceiver extends BroadcastReceiver { 63      private static final String TAG = LogUtil.BUGLE_TAG; 64 65      private static ArrayList&lt;Pattern&gt; sIgnoreSmsPatterns; 66  }</pre> <p><a href="https://android.googlesource.com/platform/packages/apps/Messaging/+nougat-mr1-release/src/com/android/messaging/receiver/SmsReceiver.java">https://android.googlesource.com/platform/packages/apps/Messaging/+nougat-mr1-release/src/com/android/messaging/receiver/SmsReceiver.java</a></p>



**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC
	<pre> 200     public static void deliverSmsMessages(final Context context, final int subId, 201         final int errorCode, final android.telephony.SmsMessage[] messages) { 202         final ContentValues messageValues = 203             MmsUtils.parseReceivedSmsMessage(context, messages, errorCode); 204 205         LogUtil.v(TAG, "SmsReceiver.deliverSmsMessages"); 206 207         final long nowInMillis = System.currentTimeMillis(); 208         final long receivedTimestampMs = MmsUtils.getMessageDate(messages[0], nowInMillis); 209 210         messageValues.put(Sms.Inbox.DATE, receivedTimestampMs); 211         // Default to unread and unseen for us but ReceiveSmsMessageAction will override 212         // seen for the telephony db. 213         messageValues.put(Sms.Inbox.READ, 0); 214         messageValues.put(Sms.Inbox.SEEN, 0); 215         if (OsUtil.isAtLeastL_MR1()) { 216             messageValues.put(Sms.SUBSCRIPTION_ID, subId); 217         } 218 219         if (messages[0].getMessageClass() == android.telephony.SmsMessage.MessageClass.CLASS_0    220             DebugUtils.debugClassZeroSmsEnabled()) { 221             Factory.get().getUIIntents().launchClassZeroActivity(context, messageValues); 222         } else { 223             final ReceiveSmsMessageAction action = new ReceiveSmsMessageAction(messageValues); 224             action.start(); 225         } 226     } </pre> <p><a href="https://android.googlesource.com/platform/packages/apps/Messaging/+nougat-mr1-release/src/com/android/messaging/receiver/SmsReceiver.java">https://android.googlesource.com/platform/packages/apps/Messaging/+nougat-mr1-release/src/com/android/messaging/receiver/SmsReceiver.java</a></p>

**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC
	<pre>228     @Override 229     public void onReceive(final Context context, final Intent intent) { 230         LogUtil.v(TAG, "SmsReceiver.onReceive " + intent); 231         // On KLP+ we only take delivery of SMS messages in SmsDeliverReceiver. 232         if (PhoneUtils.getDefault().isSmsEnabled()) { 233             final String action = intent.getAction(); 234             if (OsUtil.isSecondaryUser() &amp;&amp; 235                 (Telephony.Sms.Intents.SMS_RECEIVED_ACTION.equals(action)    236                  // TODO: update this with the actual constant from Telephony 237                  "android.provider.Telephony.MMS_DOWNLOADED".equals(action))) { 238                 postNewMessageSecondaryUserNotification(); 239             } else if (!OsUtil.isAtLeastKLP()) { 240                 deliverSmsIntent(context, intent); 241             } 242         } 243     }</pre> <p><a href="https://android.googlesource.com/platform/packages/apps/Messaging/+/nougat-mr1-release/src/com/android/messaging/receiver/SmsReceiver.java">https://android.googlesource.com/platform/packages/apps/Messaging/+/nougat-mr1-release/src/com/android/messaging/receiver/SmsReceiver.java</a></p>

**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC
	<pre>52  * Class that sends chat message via MMS. 53  * 54  * The interface emulates a blocking send similar to making an HTTP request. 55  */ 56  public class MmsSender { 57      private static final String TAG = LogUtil.BUGLE_TAG; 58 59      /** 60       * Send an MMS message. 61       * 62       * @param context Context 63       * @param messageUri The unique URI of the message for identifying it during sending 64       * @param sendReq The SendReq PDU of the message 65       * @throws MmsFailureException 66       */ 67      public static void sendMms(final Context context, final int subId, final Uri messageUri, 68                               final SendReq sendReq, final Bundle sentIntentExtras) throws MmsFailureException { 69          sendMms(context, 70                subId, 71                messageUri, 72                null /* locationUri */, 73                sendReq, 74                true /* responseImportant */, 75                sentIntentExtras); 76      }</pre> <p><a href="https://android.googlesource.com/platform/packages/apps/Messaging/+/nougat-mr1-release/src/com/android/messaging/sms/MmsSender.java">https://android.googlesource.com/platform/packages/apps/Messaging/+/nougat-mr1-release/src/com/android/messaging/sms/MmsSender.java</a></p>

**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC
	<pre>240     * Download an MMS message. 241     * 242     * @param context Context 243     * @param contentLocation The url of the MMS message 244     * @throws MmsFailureException 245     * @throws InvalidHeaderValueException 246     */ 247     public static void downloadMms(final Context context, final int subId, 248         final String contentLocation, Bundle extras) throws MmsFailureException, 249         InvalidHeaderValueException { 250         final Uri requestUri = Uri.parse(contentLocation); 251         final Uri contentUri = MmsFileProvider.buildRawMmsUri(); 252 253         final Intent downloadedIntent = new Intent(SendStatusReceiver.MMS_DOWNLOADED_ACTION, 254             requestUri, 255             context, 256             SendStatusReceiver.class); 257         downloadedIntent.putExtra(SendMessageAction.EXTRA_CONTENT_URI, contentUri); 258         if (extras != null) { 259             downloadedIntent.putExtras(extras); 260         } 261         final PendingIntent downloadedPendingIntent = PendingIntent.getBroadcast( 262             context, 263             0 /*request code*/, 264             downloadedIntent, 265             PendingIntent.FLAG_UPDATE_CURRENT); 266 267         MmsManager.downloadMultimediaMessage(subId, context, contentLocation, contentUri, 268             downloadedPendingIntent); 269     }</pre> <p><a href="https://android.googlesource.com/platform/packages/apps/Messaging/+nougat-mr1-release/src/com/android/messaging/sms/MmsSender.java">https://android.googlesource.com/platform/packages/apps/Messaging/+nougat-mr1-release/src/com/android/messaging/sms/MmsSender.java</a></p>

**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC
	<pre> 97      * Execute an MMS HTTP request, either a POST (sending) or a GET (downloading) 98      * 99      * @param urlString The request URL, for sending it is usually the MMSC, and for downloading 100     *           it is the message URL 101     * @param pdu For POST (sending) only, the PDU to send 102     * @param method HTTP method, POST for sending and GET for downloading 103     * @param isProxySet Is there a proxy for the MMSC 104     * @param proxyHost The proxy host 105     * @param proxyPort The proxy port 106     * @param mmsConfig The MMS config to use 107     * @param userAgent The user agent header value 108     * @param uaProfUrl The UA Prof URL header value 109     * @return The HTTP response body 110     * @throws MmsHttpException For any failures 111     */ 112     public byte[] execute(String urlString, byte[] pdu, String method, boolean isProxySet, 113         String proxyHost, int proxyPort, Bundle mmsConfig, String userAgent, String uaProfUrl) 114         throws MmsHttpException { 115         Log.d(MmsService.TAG, "HTTP: " + method + " " + Utils.redactUrlForNonVerbose(urlString) 116             + (isProxySet ? (" proxy=" + proxyHost + ":" + proxyPort) : "")) 117             + ", PDU size=" + (pdu != null ? pdu.length : 0)); 118         checkMethod(method); 119         HttpURLConnection connection = null; 120         try { 121             Proxy proxy = Proxy.NO_PROXY; 122             if (isProxySet) { 123                 proxy = new Proxy(Proxy.Type.HTTP, new InetSocketAddress(proxyHost, proxyPort)); 124             } 125             final URL url = new URL(urlString); 126             // Now get the connection 127             connection = (HttpURLConnection) url.openConnection(proxy); 128             connection.setDoInput(true); 129             connection.setConnectTimeout( 130                 mmsConfig.getInt(CarrierConfigValuesLoader.CONFIG_HTTP_SOCKET_TIMEOUT, 131                     CarrierConfigValuesLoader.CONFIG_HTTP_SOCKET_TIMEOUT_DEFAULT)); </pre> <p data-bbox="506 1219 1596 1287"><a href="https://android.googlesource.com/platform/packages/apps/Messaging/+/nougat-mr1-release/src/android/support/v7/mms/MmsHttpClient.java">https://android.googlesource.com/platform/packages/apps/Messaging/+/nougat-mr1-release/src/android/support/v7/mms/MmsHttpClient.java</a></p>

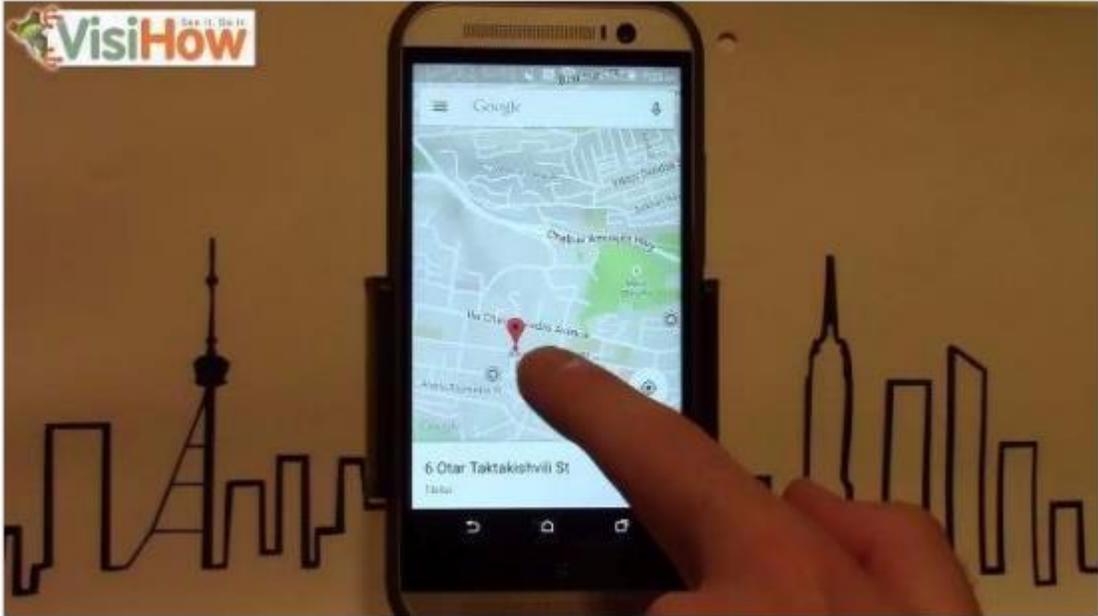
**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC
	<pre> 132 // ----- COMMON HEADERS ----- 133 // Header: Accept 134 connection.setRequestProperty(HEADER_ACCEPT, HEADER_VALUE_ACCEPT); 135 // Header: Accept-Language 136 connection.setRequestProperty( 137     HEADER_ACCEPT_LANGUAGE, getCurrentAcceptLanguage(Locale.getDefault())); 138 // Header: User-Agent 139 Log.i(MmsService.TAG, "HTTP: User-Agent=" + userAgent); 140 connection.setRequestProperty(HEADER_USER_AGENT, userAgent); 141 // Header: x-wap-profile 142 final String uaProfUrlTagName = mmsConfig.getString( 143     CarrierConfigValuesLoader.CONFIG_UA_PROF_TAG_NAME, 144     CarrierConfigValuesLoader.CONFIG_UA_PROF_TAG_NAME_DEFAULT); 145 if (uaProfUrl != null) { 146     Log.i(MmsService.TAG, "HTTP: UaProfUrl=" + uaProfUrl); 147     connection.setRequestProperty(uaProfUrlTagName, uaProfUrl); 148 } 149 // Add extra headers specified by mms_config.xml's httpparams 150 addExtraHeaders(connection, mmsConfig); 151 // Different stuff for GET and POST 152 if (METHOD_POST.equals(method)) { 153     if (pdu == null    pdu.length &lt; 1) { 154         Log.e(MmsService.TAG, "HTTP: empty pdu"); 155         throw new MmsHttpException(0/*statusCode*/, "Sending empty PDU"); 156     } 157     connection.setDoOutput(true); 158     connection.setRequestMethod(METHOD_POST); 159     if (mmsConfig.getBoolean( 160         CarrierConfigValuesLoader.CONFIG_SUPPORT_HTTP_CHARSET_HEADER, 161         CarrierConfigValuesLoader.CONFIG_SUPPORT_HTTP_CHARSET_HEADER_DEFAULT)) { 162         connection.setRequestProperty(HEADER_CONTENT_TYPE, 163             HEADER_VALUE_CONTENT_TYPE_WITH_CHARSET); 164     } else { 165         connection.setRequestProperty(HEADER_CONTENT_TYPE, 166             HEADER_VALUE_CONTENT_TYPE_WITHOUT_CHARSET); </pre> <p><a href="https://android.googlesource.com/platform/packages/apps/Messaging/+nougat-mr1-release/src/android/support/v7/mms/MmsHttpClient.java">https://android.googlesource.com/platform/packages/apps/Messaging/+nougat-mr1-release/src/android/support/v7/mms/MmsHttpClient.java</a></p>

**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

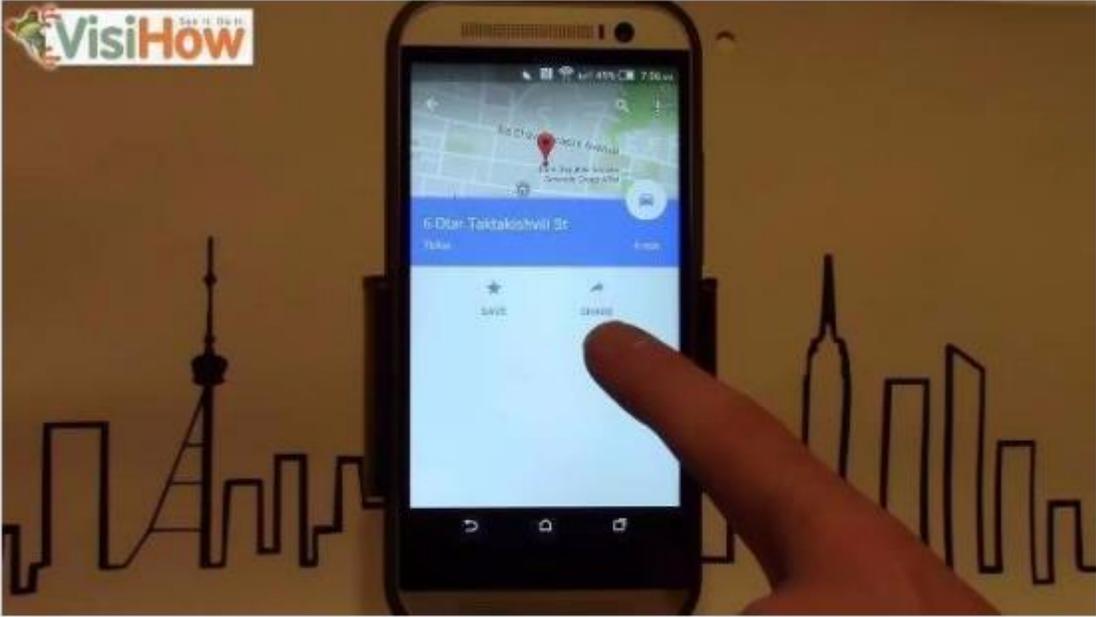
US9408055B2	HTC
	<pre>167     } 168     if (Log.isLoggable(MmsService.TAG, Log.VERBOSE)) { 169         logHttpHeaders(connection.getRequestProperties()); 170     } 171     connection.setFixedLengthStreamingMode(pdu.length); 172     // Sending request body 173     final OutputStream out = 174         new BufferedOutputStream(connection.getOutputStream()); 175     out.write(pdu); 176     out.flush(); 177     out.close(); 178 } else if (METHOD_GET.equals(method)) { 179     if (Log.isLoggable(MmsService.TAG, Log.VERBOSE)) { 180         logHttpHeaders(connection.getRequestProperties()); 181     } 182     connection.setRequestMethod(METHOD_GET); 183 } 184 // Get response 185 final int responseCode = connection.getResponseCode(); 186 final String responseMessage = connection.getResponseMessage(); 187 Log.d(MmsService.TAG, "HTTP: " + responseCode + " " + responseMessage); 188 if (Log.isLoggable(MmsService.TAG, Log.VERBOSE)) { 189     logHttpHeaders(connection.getHeaderFields()); 190 } 191 if (responseCode / 100 != 2) { 192     throw new MmsHttpException(responseCode, responseMessage); 193 } 194 final InputStream in = new BufferedInputStream(connection.getInputStream()); 195 final ByteArrayOutputStream byteOut = new ByteArrayOutputStream(); 196 final byte[] buf = new byte[4096]; 197 int count = 0; 198 while ((count = in.read(buf)) &gt; 0) { 199     byteOut.write(buf, 0, count); 200 } 201 in.close(); 202 final byte[] responseBody = byteOut.toByteArray(); 203 Log.d(MmsService.TAG, "HTTP: response size=" 204     + (responseBody != null ? responseBody.length : 0)); 205 return responseBody;</pre> <p><a href="https://android.goesource.com/platform/packages/apps/Messaging/+nougat-mr1-release/src/android/support/v7/mms/MmsHttpClient.java">https://android.goesource.com/platform/packages/apps/Messaging/+nougat-mr1-release/src/android/support/v7/mms/MmsHttpClient.java</a></p>

**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

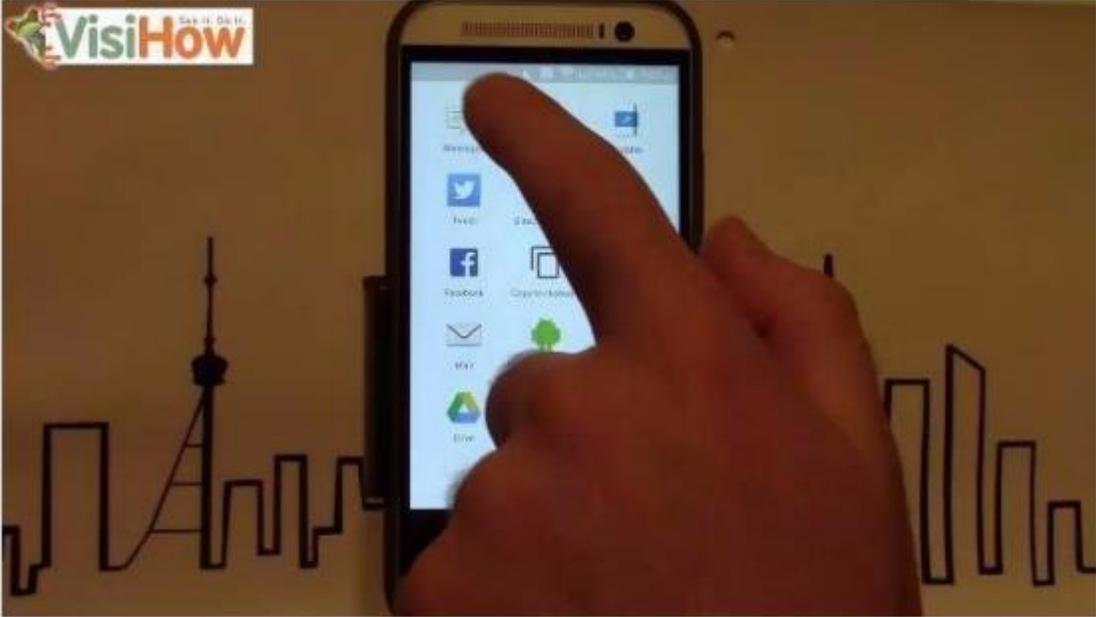
US9408055B2	HTC
<p>[54C] receiving respective IP-based responses to the SMS messages, wherein the IP-based responses to the SMS messages include location information of the respective second devices;</p>	<p>HTC infringes directly and/or indirectly by performing, inducing others to perform, and/or contributing to the performance of receiving respective IP-based responses to the SMS messages, wherein the IP-based responses to the SMS messages include location information of the respective second devices. See claims 1[C], 28[C], and 41[C], which are incorporated herein by reference in their entirety.</p> <p>For example, the HTC accused devices running Maps are configured to receive IP-based communications from the respective second devices that include location information of the second devices.</p> <h3 data-bbox="520 527 1732 630">Share Locations from Google Maps on HTC One M8</h3> <p data-bbox="527 634 829 662">Place a pin on the map.</p> <p data-bbox="527 675 1640 784">We can do this anywhere in the entire world, anywhere, on any map that we find in the Google Maps application. We are going to choose a location, press down on the screen, and hold down until we can see that a red pin has been dropped.</p> 



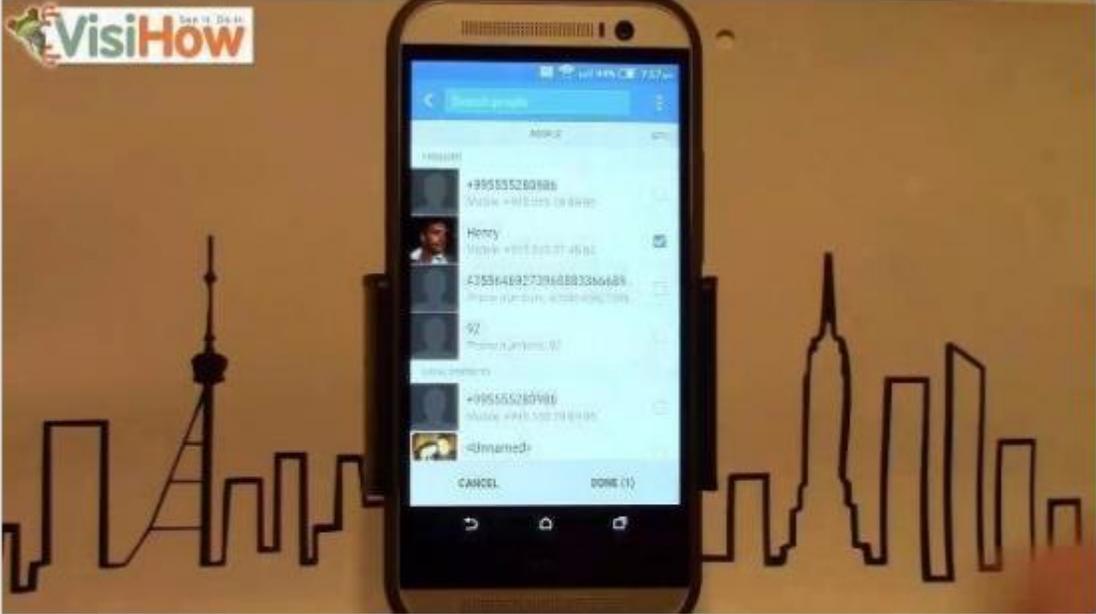
**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC
	<p data-bbox="527 277 1205 342"><b>We have two options:</b> "SAVE" and "SHARE". Press down on the word "SHARE".</p>  <p data-bbox="527 1094 1633 1247"><b>A menu appears with all of the different ways that we can share this location.</b> We can share this location via messages, Bluetooth, Twitter, Facebook, mail, Gmail and Viber. If you have other applications, such as WhatsApp, or any other chat or email applications, you will be able to use them as well and will see them here.</p>

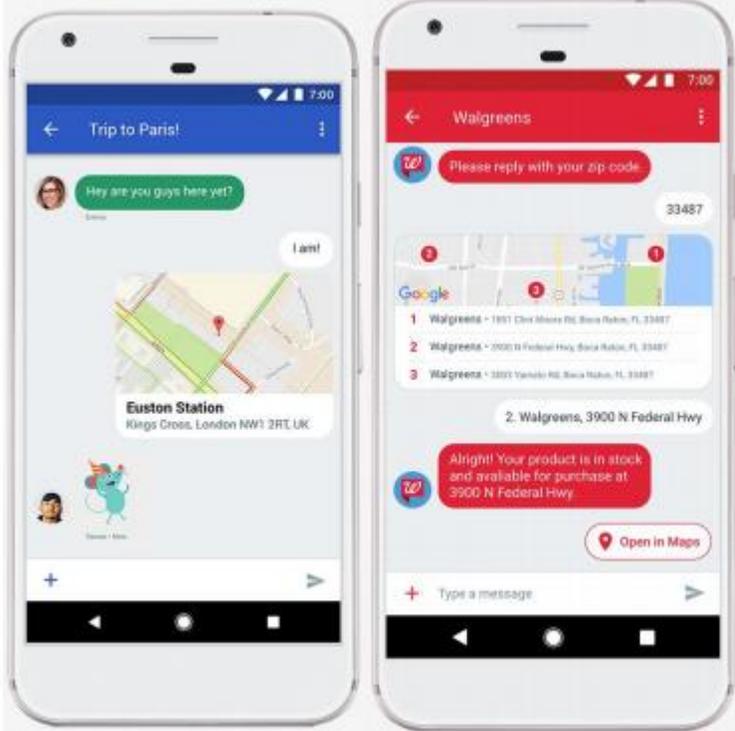
**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC
	<p data-bbox="527 237 856 261"><b>Let's choose "Messages".</b></p> <p data-bbox="527 277 1633 347">This option is located on the top left-hand corner of the screen. All options will have the same basic processes. They will ask us to send the location. Press down on the "Messages" icon.</p> 

**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC
	<p data-bbox="520 233 1260 256"><b>Press the box next to the contact who will be the recipient.</b></p> <p data-bbox="520 269 1549 292">A checkmark will appear in the box. We can select multiple contacts as well if we desire.</p>  <p data-bbox="520 1045 911 1068"><b>Press on the word "DONE (1)".</b></p> <p data-bbox="520 1084 1629 1153">This icon is located in the bottom right-hand corner of the screen. The bracketed number indicates the number of contacts we selected.</p> <p data-bbox="520 1166 1407 1192"><a href="http://visihow.com/Share_Locations_from_Google_Maps_on_HTC_One_M8">http://visihow.com/Share_Locations_from_Google_Maps_on_HTC_One_M8</a></p>

# Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products

US9408055B2	HTC
	<p data-bbox="533 240 884 280"><b>Send your location</b></p> <ol data-bbox="533 305 947 505" style="list-style-type: none"><li>1. Open the Android Messages app .</li><li>2. Open or start a conversation.</li><li>3. Tap Attach .</li><li>4. Tap Location on .</li><li>5. To send your location, tap Send .</li></ol> <p data-bbox="520 524 1539 557"><a href="https://support.google.com/pixelphone/answer/6159880?hl=en&amp;ref_topic=6211804">https://support.google.com/pixelphone/answer/6159880?hl=en&amp;ref_topic=6211804</a></p> <div data-bbox="520 594 1255 1325"></div> <p data-bbox="520 1328 1514 1360"><a href="https://blog.google/topics/rcs/delivering-rcs-messaging-android-users-worldwide/">https://blog.google/topics/rcs/delivering-rcs-messaging-android-users-worldwide/</a></p>

**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC
	<div data-bbox="533 337 915 376" data-label="Section-Header"> <p>Share a location or place</p> </div> <div data-bbox="554 418 743 443" data-label="Section-Header"> <p>Share your location</p> </div> <div data-bbox="588 472 1104 597" data-label="List-Group"> <ol style="list-style-type: none"> <li>1. On your Android phone or tablet, open the Hangouts app</li> <li>2. Open a conversation.</li> <li>3. Tap Location</li> <li>4. Tap <b>Select this location</b> &gt; <b>Select</b>.</li> </ol> </div> <div data-bbox="554 672 690 698" data-label="Section-Header"> <p>Share a place</p> </div> <div data-bbox="588 725 1104 885" data-label="List-Group"> <ol style="list-style-type: none"> <li>1. On your Android phone or tablet, open the Hangouts app</li> <li>2. Open a conversation.</li> <li>3. Tap Location &gt; Search</li> <li>4. Type in a location or address.</li> <li>5. Tap <b>Select</b>.</li> </ol> </div> <div data-bbox="514 915 1581 1016" data-label="Text"> <p><a href="https://support.google.com/hangouts/answer/3115410?visit_id=1-636271867303650973-2491837168&amp;rd=1&amp;co=GENIE.Platform%3DAndroid&amp;oco=1">https://support.google.com/hangouts/answer/3115410?visit_id=1-636271867303650973-2491837168&amp;rd=1&amp;co=GENIE.Platform%3DAndroid&amp;oco=1</a>  <a href="https://play.google.com/store/apps/details?id=com.google.android.talk&amp;hl=en">https://play.google.com/store/apps/details?id=com.google.android.talk&amp;hl=en</a></p> </div> <div data-bbox="504 1052 1919 1307" data-label="Text"> <p><b>Regarding Google Maps</b>, Google Latitude, Google Plus, Google Hangouts, Google Messages, Google Allo, Google Duo, Google Chrome, and Android Messenger, the Accused Products require a user to join the corresponding network by signing-in to the device with an identifier (e.g., Google Account). When the signed-in user enables Google Maps Share Location and Location Services on the first device, the user shares its location and the first device’s location is sent to a server. When the user enables sharing to one or more contacts (of respective devices) and the one or more contacts enable sharing their location to the user of the first device, the user of the first device receives the locations of the one or more contacts.</p> </div> <div data-bbox="504 1346 1904 1417" data-label="Text"> <p>The first device’s participation in the group is based on receiving the message from the second device, i.e. a message indicating that the second device is sharing its location.</p> </div> <div data-bbox="1346 225 1734 914" data-label="Image"> </div>

**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC
	<p>By participating in the Maps location sharing functionality, the device sends location information to a server (e.g., a network server provided by an ISP such as AT&amp;T and/or a server running Google's services). The device also receives location information from the server indicating the location of other devices that are sharing location information via Maps.</p> <p><b><u>Further regarding Google Maps</u></b>, Google Latitude, Google Plus, Google Hangouts, Google Messages, Google Allo, Google Duo, Google Chrome, and Android Messenger, the Accused Products alternatively require a user to send a request containing the identifier (e.g., Google Account). When the signed-in user enables Google Maps Share Location and Location Services on the first device, the user shares its location and the first device's location is sent to a server. When the user sends a message to another contact through Google Maps, Google Messages, and/or another means from within the Google Maps application, the message including location information are sent to a server before transmission to the intended contact. When one or more contacts enable sharing their location to the user of the first device, or alternatively send a message containing location information, or alternatively accept a request to share their location with the first user, the user of the first device receives the locations of the one or more contacts.</p> <p><b><u>Exemplary Support for Google Maps:</u></b></p>

## Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products

US9408055B2	HTC
	<p data-bbox="548 240 968 261">COMPUTER <b>ANDROID</b> IPHONE &amp; IPAD</p> <hr data-bbox="533 289 1566 293"/> <h3 data-bbox="533 347 1024 383">If they have a Google Account</h3> <ol data-bbox="533 406 1419 698" style="list-style-type: none"><li data-bbox="533 406 1220 427">1. If you haven't already, add their Gmail address to your <a href="#">Google Contacts</a>.</li><li data-bbox="533 443 1419 464">2. On your Android phone or tablet, open the Google Maps app  and sign in. <a href="#">Learn how to sign in.</a></li><li data-bbox="533 480 1031 501">3. Tap Menu ≡ &gt; <b>Location sharing</b> &gt; Add People +.</li><li data-bbox="533 518 1003 539">4. Choose how long you want to share your location.</li><li data-bbox="533 555 1140 613">5. Tap <b>Select People</b>.<ul data-bbox="569 589 1140 610" style="list-style-type: none"><li data-bbox="569 589 1140 610">• If you're asked about your contacts, give Google Maps access.</li></ul></li><li data-bbox="533 639 884 660">6. Choose who you want to share with.</li><li data-bbox="533 677 663 698">7. Tap <b>Share</b>.</li></ol> <h3 data-bbox="533 756 1110 792">If they don't have a Google Account</h3> <ol data-bbox="533 815 1560 938" style="list-style-type: none"><li data-bbox="533 815 1419 836">1. On your Android phone or tablet, open the Google Maps app  and sign in. <a href="#">Learn how to sign in.</a></li><li data-bbox="533 852 1031 873">2. Tap Menu ≡ &gt; <b>Location sharing</b> &gt; Add People +.</li><li data-bbox="533 889 1560 938">3. Tap More ... &gt; <b>Copy to clipboard</b>. People with this link can see your location for as long as you choose, up to 72 hours.</li></ol> <h3 data-bbox="533 987 869 1023">Share using another app</h3> <p data-bbox="533 1045 1205 1066">You can also share through messaging apps. Tap More ... &gt; select an app.</p> <h3 data-bbox="533 1125 741 1161">Stop sharing</h3> <ol data-bbox="533 1183 1205 1281" style="list-style-type: none"><li data-bbox="533 1183 842 1205">1. Open the Google Maps app .</li><li data-bbox="533 1221 869 1242">2. Tap Menu ≡ &gt; <b>Location sharing</b>.</li><li data-bbox="533 1258 1205 1279">3. Next to the person with whom you want to stop sharing, tap Remove ✕.</li></ol> <p data-bbox="512 1299 1701 1325"><a href="https://support.google.com/maps/answer/7326816?co=GENIE.Platform%3DAndroid&amp;oco=1">https://support.google.com/maps/answer/7326816?co=GENIE.Platform%3DAndroid&amp;oco=1</a></p>

## Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products

US9408055B2	HTC
	<h3>Share your E.T.A</h3> <p>After you start your drive, you can share your destination, estimated arrival time, and where you are on the route.</p> <ol style="list-style-type: none"><li>1. Open the Google Maps app .</li><li>2. Set a driving destination. <a href="#">Learn how to navigate to a place.</a></li><li>3. After you start navigation, tap More  &gt; <b>Share trip progress.</b></li><li>4. Choose a person from the list.</li><li>5. Tap <b>Share.</b></li><li>6. Location Sharing will stop when you reach your destination or stop navigating.</li></ol> <ul style="list-style-type: none"><li>• To stop sharing before you arrive, tap More  &gt; <b>Stop sharing.</b></li></ul> <h3>See where someone is</h3> <p>If someone shares their location with you, you can see them on the map.</p> <ol style="list-style-type: none"><li>1. Open the Google Maps app .</li><li>2. Tap Menu  &gt; <b>Location sharing.</b></li><li>3. Choose someone.</li></ol> <ul style="list-style-type: none"><li>• To see an updated location, tap on a friend's icon &gt; More  &gt; <b>Refresh.</b></li></ul> <h3>Stop seeing someone's location</h3> <ol style="list-style-type: none"><li>1. Open the Google Maps app .</li><li>2. On the map, tap their icon.</li><li>3. At the bottom, tap More .</li><li>4. To temporarily hide someone, tap <b>Hide from map.</b> You can stop hiding them at any time.</li></ol> <p><b>Note:</b> You can stop someone's location from ever appearing on your map. Learn how to <a href="#">block another person's account.</a></p> <p><a href="https://support.google.com/maps/answer/7326816?co=GENIE.Platform%3DAndroid&amp;oco=1">https://support.google.com/maps/answer/7326816?co=GENIE.Platform%3DAndroid&amp;oco=1</a></p>



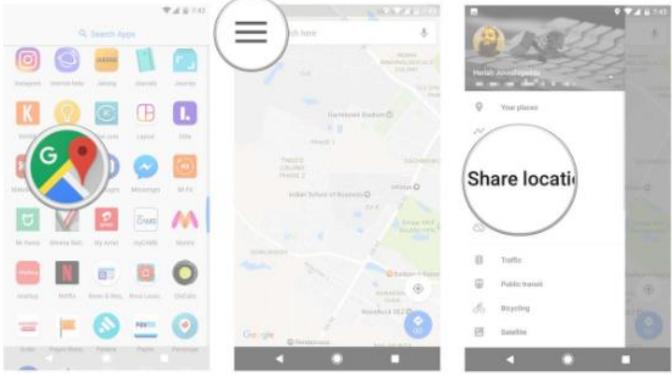
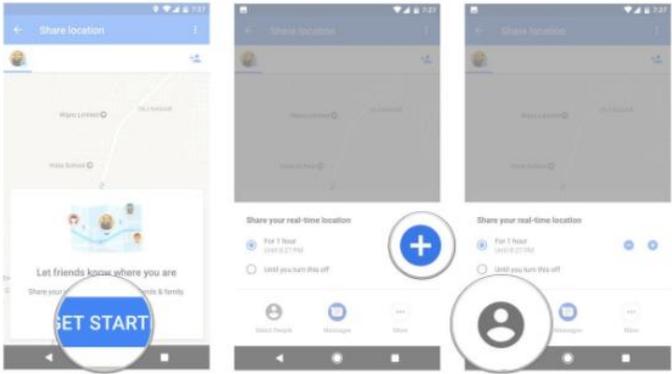
## Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products

US9408055B2	HTC
	<h3>Create a list of places</h3> <p>In Google Maps, you can create a list of places, like your favorite places or places you want to visit.</p> <p>COMPUTER <b>ANDROID</b> IPHONE &amp; IPAD</p> <hr/> <h3>Make a new list</h3> <ol style="list-style-type: none"><li>1. On your Android phone or tablet, open the Google Maps app .</li><li>2. Tap Menu  &gt; <b>Your places</b> &gt; <b>Saved</b>.</li><li>3. In the bottom right, tap Add .</li><li>4. Enter a name and description.</li><li>5. Tap <b>Save</b>.</li></ol> <h3>Save a place to a list</h3> <ol style="list-style-type: none"><li>1. Open the Google Maps app .</li><li>2. Search for a place or tap it on the map.</li><li>3. At the bottom, tap the place's name or address.</li><li>4. Tap <b>Save</b>.</li><li>5. Choose a list. To create a new list, tap <b>New list</b> .</li></ol> <h3>See your lists</h3> <ol style="list-style-type: none"><li>1. Open the Google Maps app .</li><li>2. Tap Menu  &gt; <b>Your places</b> &gt; <b>Saved</b>.</li></ol> <p><a href="https://support.google.com/maps/answer/7280933?hl=en&amp;ref_topic=7301134&amp;co=GENIE.Platform%3DAndroid&amp;oco=1">https://support.google.com/maps/answer/7280933?hl=en&amp;ref_topic=7301134&amp;co=GENIE.Platform%3DAndroid&amp;oco=1</a></p>

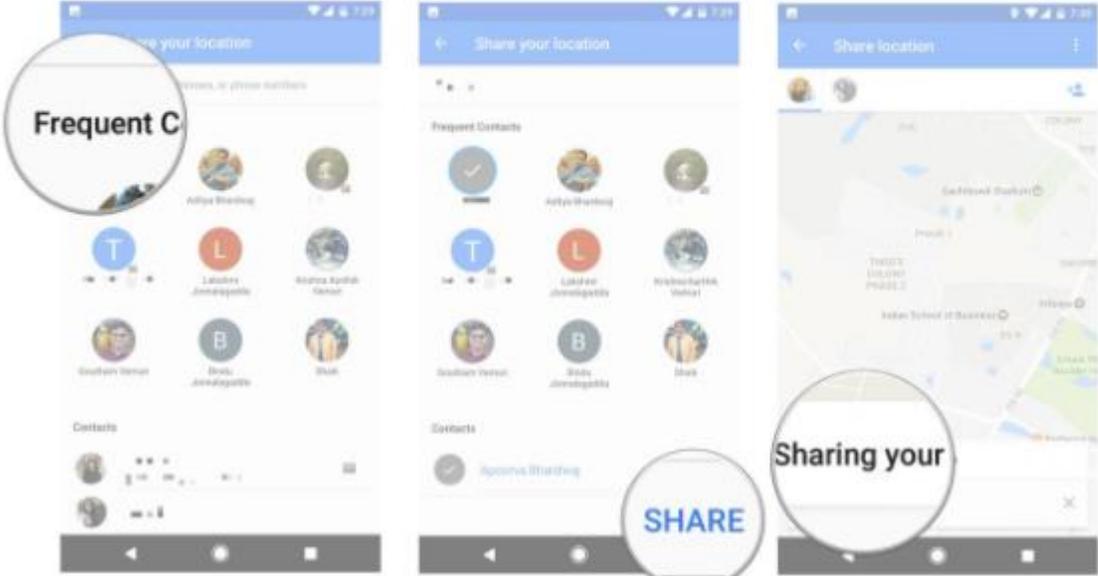
**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC
	<h2 data-bbox="541 245 877 282">Hide or share lists</h2> <p data-bbox="541 313 907 337"><b>Note:</b> You can't share starred places.</p> <ol data-bbox="554 367 1251 475" style="list-style-type: none"><li data-bbox="554 367 890 391">1. Open the Google Maps app .</li><li data-bbox="554 407 968 431">2. Tap Menu  &gt; <b>Your places</b> &gt; <b>Saved</b>.</li><li data-bbox="554 448 1251 475">3. Next to the list you want to share, tap More  &gt; choose an option:</li></ol> <ul data-bbox="583 492 1682 634" style="list-style-type: none"><li data-bbox="583 492 1440 516">• <b>Hide/Show on your map:</b> Display or hide your saved places when looking at the map.</li><li data-bbox="583 532 1058 557">• <b>Share list:</b> Allow others to see your saved list.</li><li data-bbox="583 573 1682 634">• <b>Sharing options:</b> You can make your list public, private, or shared. To let anyone with the link see your list, tap <b>Shared</b>. To let anyone find and follow your list, tap <b>Public</b>.</li></ul> <h2 data-bbox="541 699 764 737">Follow a list</h2> <p data-bbox="541 768 1728 824">If you follow a list made by someone else, their saved places will show up in Your Places. The places will also appear as suggested locations in Google Maps.</p> <h2 data-bbox="541 878 915 915">Follow a list using a link</h2> <ol data-bbox="554 938 1352 1047" style="list-style-type: none"><li data-bbox="554 938 957 963">1. Tap on the link you received to open it.</li><li data-bbox="554 979 1272 1003">2. Tap <b>Follow</b>. This list will now be added to the group of lists you follow.</li><li data-bbox="554 1019 1352 1047">3. <b>Optional:</b> To unfollow a list someone shared with you, tap the list &gt; <b>Following</b>.</li></ol> <h2 data-bbox="541 1101 924 1138">See lists made by others</h2> <p data-bbox="541 1161 1335 1185">If a user has any Google Maps lists that were made public, you can follow them.</p> <ol data-bbox="554 1216 1136 1325" style="list-style-type: none"><li data-bbox="554 1216 1136 1240">1. Tap on the name of a user whose list you want to follow.</li><li data-bbox="554 1256 680 1281">2. Tap <b>Lists</b>.</li><li data-bbox="554 1297 1136 1325">3. Tap on the list you want to follow &gt; More  &gt; <b>Follow</b>.</li></ol> <p data-bbox="512 1341 1898 1401"><a href="https://support.google.com/maps/answer/7280933?hl=en&amp;ref_topic=7301134&amp;co=GENIE.Platform%3DAndroid&amp;oco=1">https://support.google.com/maps/answer/7280933?hl=en&amp;ref_topic=7301134&amp;co=GENIE.Platform%3DAndroid&amp;oco=1</a></p>

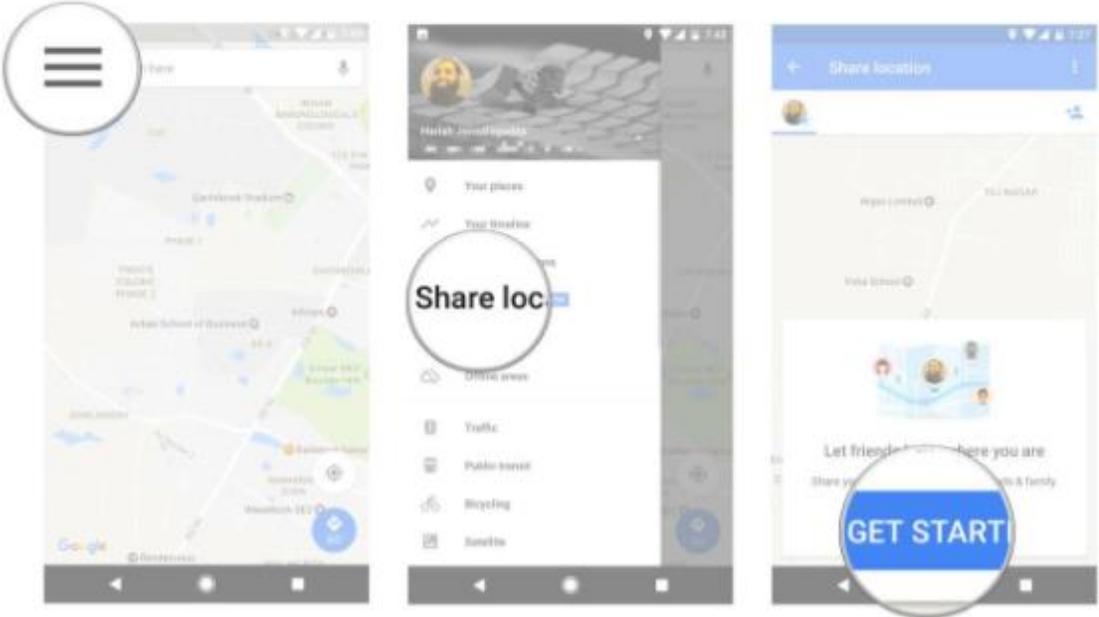
**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC
	<p data-bbox="520 240 1150 272"><b>How to share your location in Google Maps</b></p> <ol data-bbox="520 300 1134 389" style="list-style-type: none"> <li>1. Open Google Maps from the app drawer or the home screen.</li> <li>2. Tap the hamburger menu (the three horizontal lines) on the top left corner of the screen.</li> <li>3. Select Share location.</li> </ol>  <ol data-bbox="520 824 1165 933" style="list-style-type: none"> <li>4. Tap Get Started.</li> <li>5. Use the + icon to select a time period or select the <b>Until you turn this off</b> setting to share your location indefinitely.</li> <li>6. Tap Select People.</li> </ol>  <p data-bbox="520 1339 1354 1369"><a href="https://www.androidcentral.com/how-share-location-google-maps">https://www.androidcentral.com/how-share-location-google-maps</a></p>

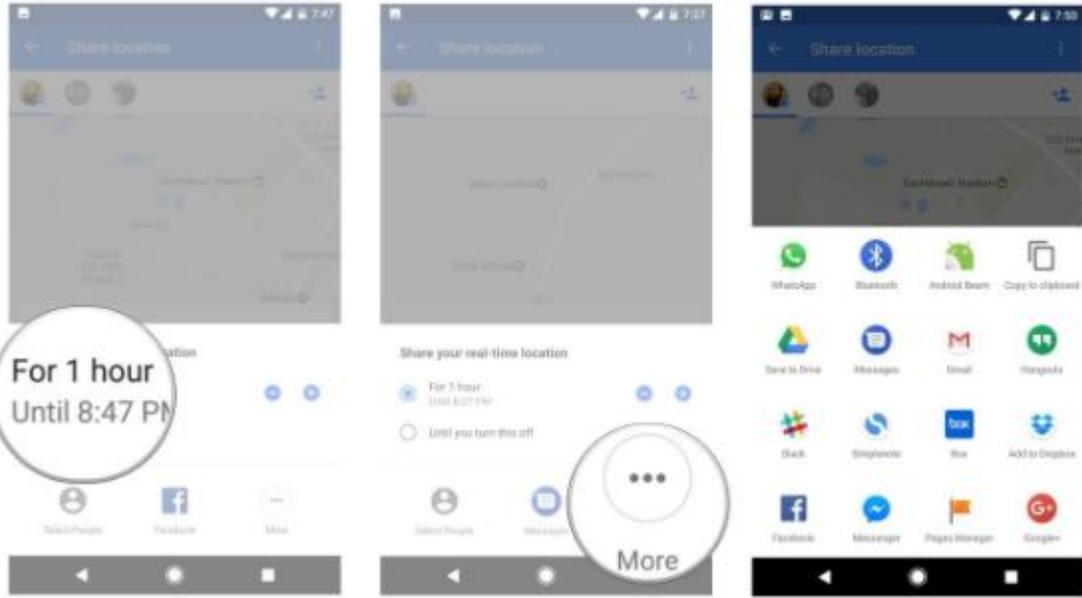
**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC
	<p data-bbox="527 253 1577 427"><b>7.</b> You'll see a list of your frequent contacts at the top, along with a full list of contacts. <b>Pick the contacts</b> by tapping their name.</p> <p data-bbox="527 342 1457 367"><b>8.</b> Once you've selected the contacts you want to share your location to, tap <b>Share</b>.</p> <p data-bbox="527 399 1419 423"><b>9.</b> You'll see a message saying that the selected contact can view your location.</p>  <p data-bbox="510 1101 1356 1133"><a href="https://www.androidcentral.com/how-share-location-google-maps">https://www.androidcentral.com/how-share-location-google-maps</a></p>

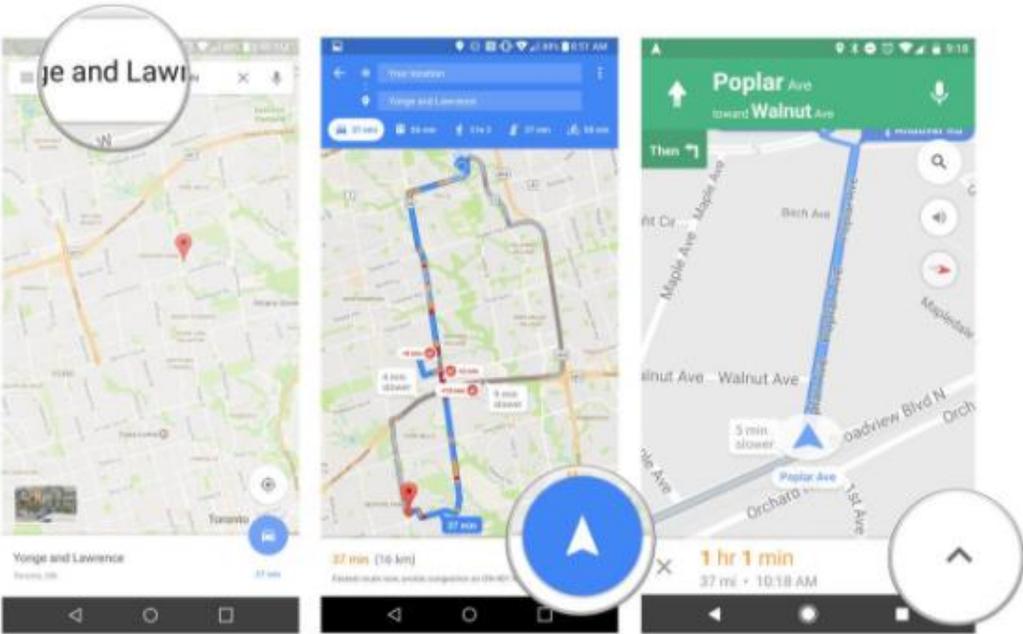
## Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products

US9408055B2	HTC
	<h3 data-bbox="520 245 1255 289">How to create a shareable link</h3> <p data-bbox="520 334 1461 362">You can also create a link and use it to share your location easily. Here's how to do it:</p> <ol data-bbox="520 412 1234 548" style="list-style-type: none"><li data-bbox="520 412 1234 440">1. Tap the hamburger menu on the top left corner of the screen.</li><li data-bbox="520 467 800 495">2. Select Share location.</li><li data-bbox="520 522 737 550">3. Tap Get Started.</li></ol>  <p data-bbox="506 1230 1356 1263"><a href="https://www.androidcentral.com/how-share-location-google-maps">https://www.androidcentral.com/how-share-location-google-maps</a></p>

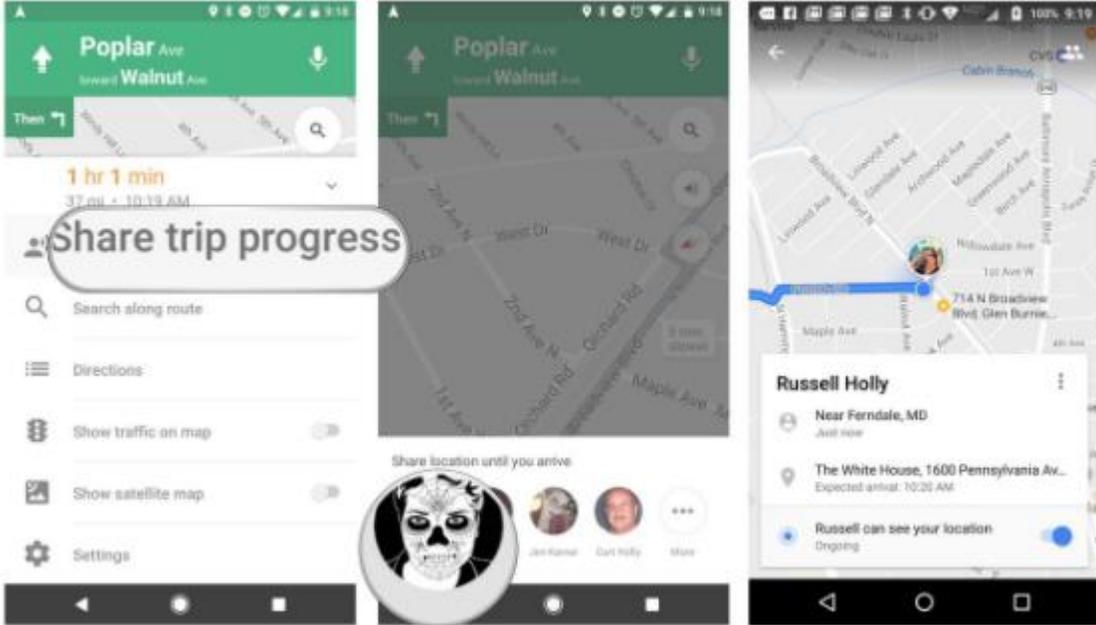
**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC
	<p data-bbox="520 245 1213 272">4. Select the <b>amount of time</b> you want to share your location.</p> <p data-bbox="520 302 680 329">5. Tap <b>More</b>.</p> <p data-bbox="520 358 1633 423">6. Select <b>your app of choice</b> to create and send a <b>unique URL</b> that broadcasts your current location. You can email it, send the link via Messenger, or even tweet it to the intended recipient.</p>  <p data-bbox="506 1081 1358 1117"><a href="https://www.androidcentral.com/how-share-location-google-maps">https://www.androidcentral.com/how-share-location-google-maps</a></p>

**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

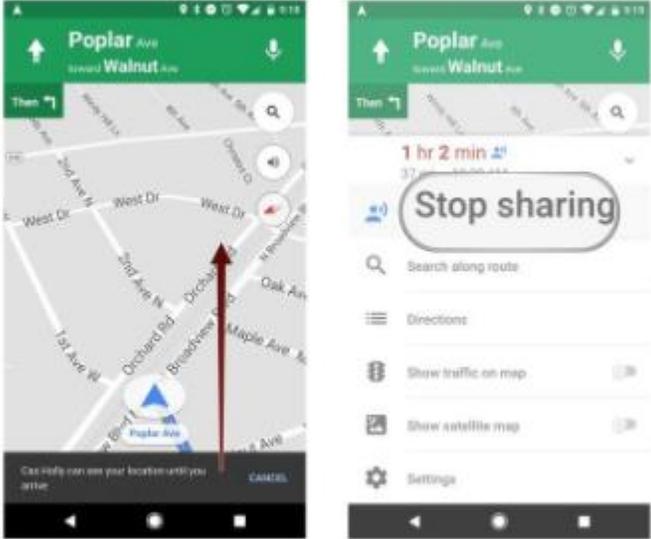
US9408055B2	HTC
	<h2 data-bbox="527 240 1428 337">How to share your navigation directions while you walk, drive or transit</h2> <p data-bbox="527 375 1556 464">One of the best ways to share your location with someone is during a drive, walk or transit. If you're meeting a friend or family member somewhere, or navigating towards their home, Google Maps lets you share your location with them for the duration of the trip. It's magic!</p> <ol data-bbox="527 513 1396 643" style="list-style-type: none"><li data-bbox="527 513 976 537">1. In the <b>search bar</b> enter your destination.</li><li data-bbox="527 561 1396 586">2. Pick your navigation type (drive, transit, walk) and press the blue <b>navigate</b> button.</li><li data-bbox="527 610 1396 634">3. Tap the arrow next to the <b>time-to-destination</b> number at the bottom of the screen.</li></ol>  <p data-bbox="512 1328 1356 1357"><a href="https://www.androidcentral.com/how-share-location-google-maps">https://www.androidcentral.com/how-share-location-google-maps</a></p>

**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC
	<p data-bbox="527 240 835 266">4. Tap Share trip progress.</p> <p data-bbox="527 297 1150 323">5. Choose one or more contacts to share trip progress.</p>  <p data-bbox="537 1029 1339 1055">You can also stop sharing your location with someone before a trip ends.</p> <p data-bbox="512 1065 1356 1091"><a href="https://www.androidcentral.com/how-share-location-google-maps">https://www.androidcentral.com/how-share-location-google-maps</a></p>



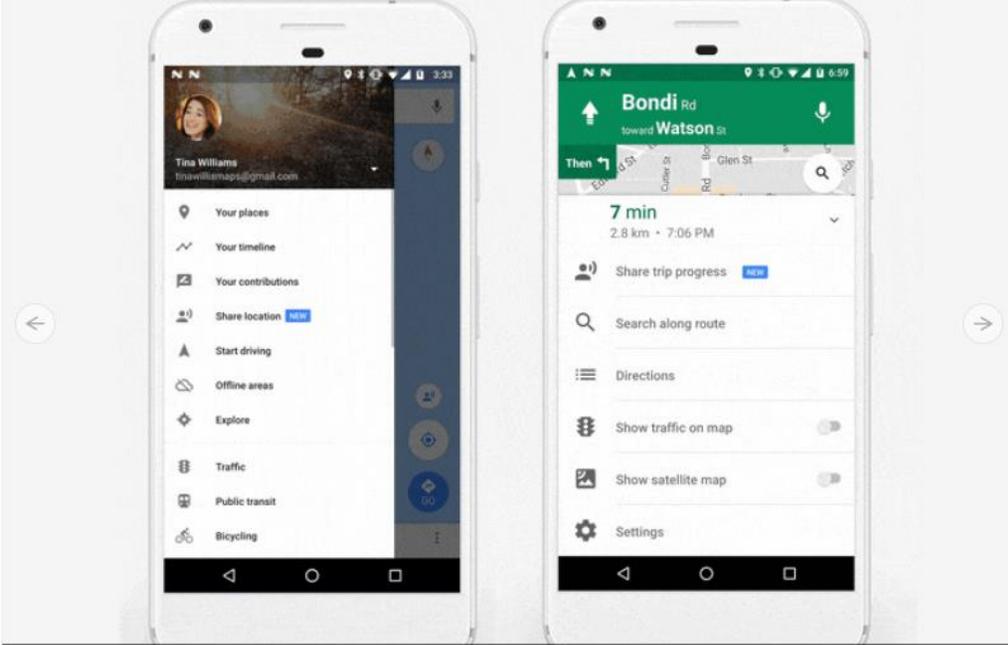
### Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products

US9408055B2	HTC
	<ol style="list-style-type: none"><li data-bbox="533 245 1470 272">1. Tap the arrow next to the <b>time-to-destination</b> number at the bottom of the screen.</li><li data-bbox="533 302 768 329">2. Tap Stop sharing.</li></ol> <div data-bbox="747 383 1398 922"></div> <p data-bbox="541 976 638 1003">That's it!</p> <p data-bbox="541 1045 1612 1073">Are you excited that location sharing is back in Google Maps? How often do you use the feature?</p> <p data-bbox="512 1084 1356 1112"><a href="https://www.androidcentral.com/how-share-location-google-maps">https://www.androidcentral.com/how-share-location-google-maps</a></p> <p data-bbox="512 1122 1419 1149">As shown below, a group may also be defined within Google Contacts.</p>

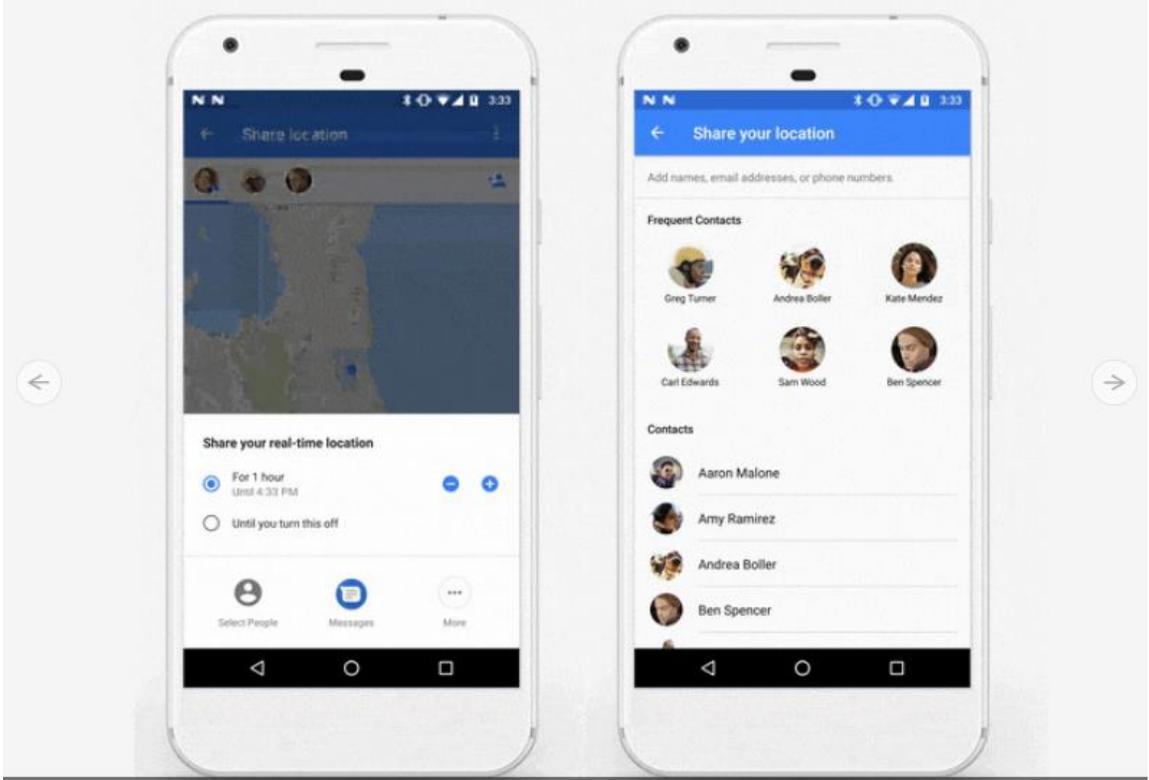
**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC
	<p><b>See your contacts</b></p> <ol style="list-style-type: none"><li>1. Open your device's Contacts app .</li><li>2. Tap Menu .</li></ol> <ul style="list-style-type: none"><li>• <b>See contacts by label:</b> Choose a label from the list.</li><li>• <b>See contacts for another account:</b> Tap Down arrow  &gt; pick an account.</li><li>• <b>See the contacts for all your accounts:</b> Choose <b>All contacts</b>. <b>Tip:</b> If you have multiple contacts with the same information, the information will be grouped into one contact.</li><li>• <b>See your Google Account contacts on the web:</b> Go to <a href="#">Google Contacts</a> .</li></ul> <p><a href="https://support.google.com/android/answer/6118731?hl=en&amp;ref_topic=6118711">https://support.google.com/android/answer/6118731?hl=en&amp;ref_topic=6118711</a></p> <p><b>Label your contacts</b></p> <p>You can group contacts together using labels.</p> <ol style="list-style-type: none"><li>1. Open your device's Contacts app .</li><li>2. Tap Menu  &gt; <b>Create label</b>.</li><li>3. Enter a label name and tap <b>Ok</b>.</li></ol> <ul style="list-style-type: none"><li>• <b>Add one contact to a label:</b> Tap Add contact  &gt; choose a contact.</li><li>• <b>Add multiple contacts to a label:</b> Tap Add contact  &gt; touch and hold a contact &gt; tap the other contacts &gt; tap <b>Add</b>.</li></ul> <p><a href="https://support.google.com/android/answer/6118731?hl=en&amp;ref_topic=6118711">https://support.google.com/android/answer/6118731?hl=en&amp;ref_topic=6118711</a></p> <p><b>Share your contacts</b></p> <ol style="list-style-type: none"><li>1. Open your device's Contacts app .</li><li>2. Tap a contact in the list.</li><li>3. Tap More  &gt; <b>Share</b>.</li><li>4. Choose how you want to share the contact.</li></ol> <p><a href="https://support.google.com/android/answer/6118731?hl=en&amp;ref_topic=6118711">https://support.google.com/android/answer/6118731?hl=en&amp;ref_topic=6118711</a></p>

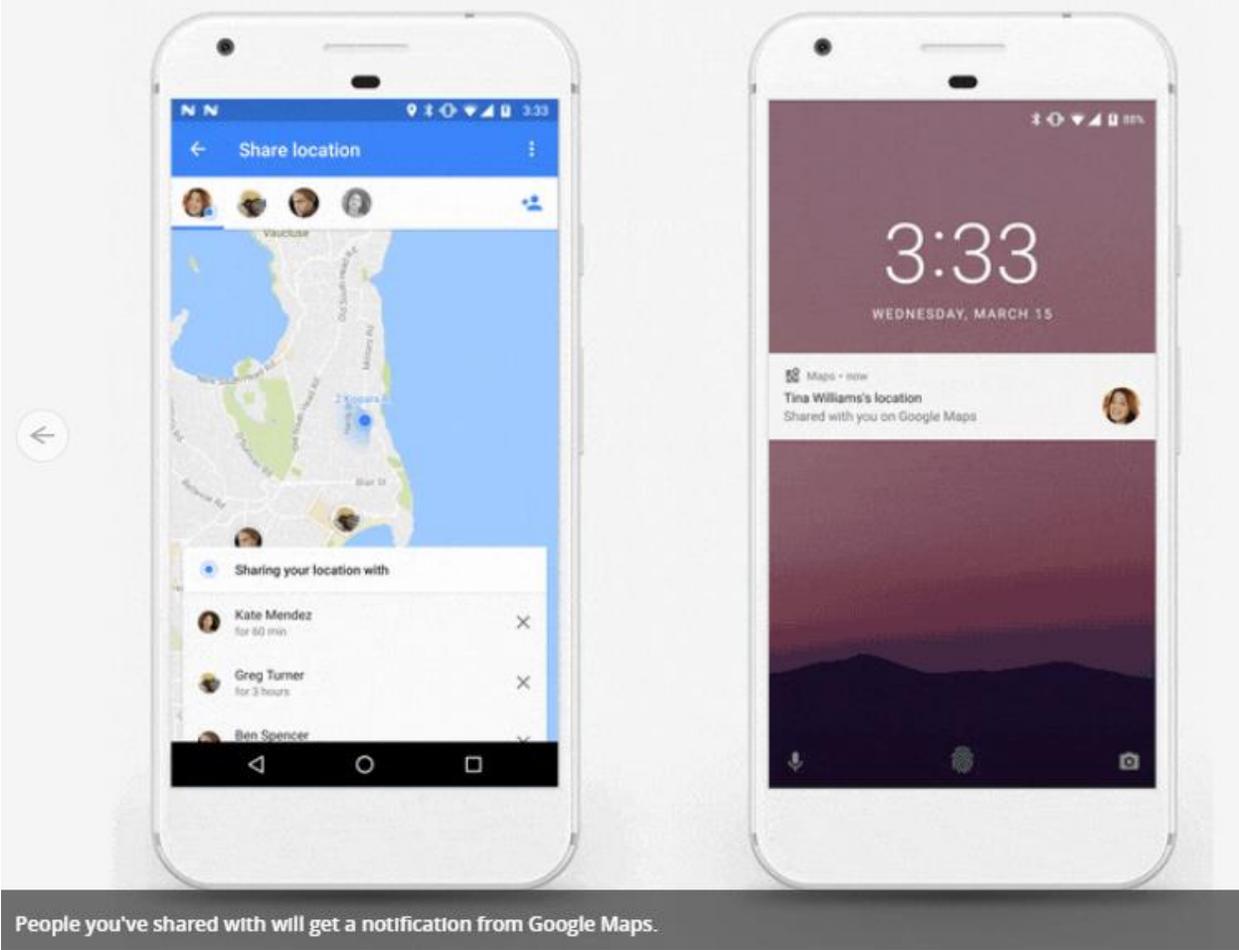
**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC
	 <p data-bbox="514 893 1522 950">Ways to start location sharing: (left) from the app navigation drawer, and (right) from the menu, you can drag up while navigating somewhere.</p> <p data-bbox="514 958 1522 990"><a href="https://arstechnica.com/gadgets/2017/03/location-sharing-finally-returns-to-google-maps/">https://arstechnica.com/gadgets/2017/03/location-sharing-finally-returns-to-google-maps/</a></p>

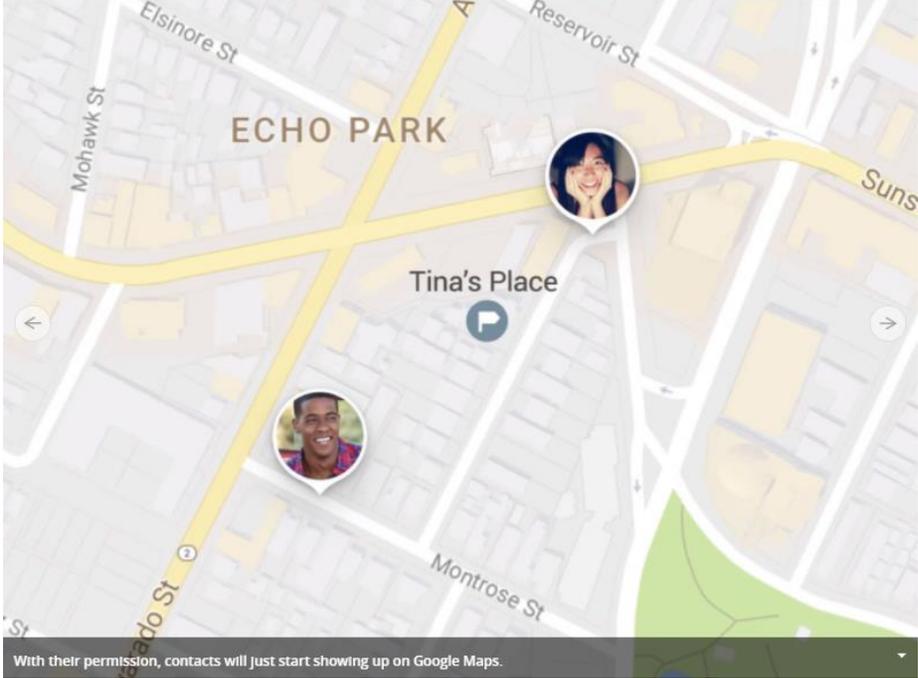
**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC
	 <p data-bbox="512 1013 1661 1062">You can pick how long you want to share your location for, and with whom you want to share it.</p> <p data-bbox="512 1062 1661 1097"><a href="https://arstechnica.com/gadgets/2017/03/location-sharing-finally-returns-to-google-maps/">https://arstechnica.com/gadgets/2017/03/location-sharing-finally-returns-to-google-maps/</a></p>

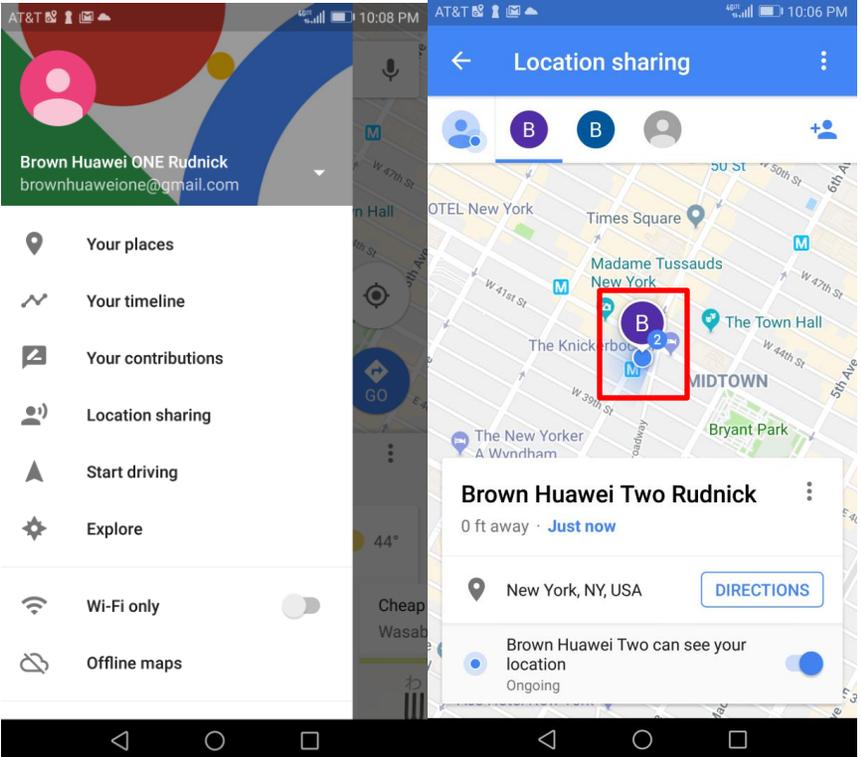
**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC
	 <p data-bbox="514 1144 1176 1177">People you've shared with will get a notification from Google Maps.</p> <p data-bbox="514 1185 1659 1226"><a href="https://arstechnica.com/gadgets/2017/03/location-sharing-finally-returns-to-google-maps/">https://arstechnica.com/gadgets/2017/03/location-sharing-finally-returns-to-google-maps/</a></p>

**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC
	 <p data-bbox="512 883 1430 911">With their permission, contacts will just start showing up on Google Maps.</p> <p data-bbox="512 915 1656 948"><a href="https://arstechnica.com/gadgets/2017/03/location-sharing-finally-returns-to-google-maps/">https://arstechnica.com/gadgets/2017/03/location-sharing-finally-returns-to-google-maps/</a></p> <p data-bbox="512 953 1024 985"><b><u>Exemplary Google Maps Screenshots</u></b></p>

**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC
	 <p>The screenshot shows an Android phone interface. On the left, a menu is open with options: 'Your places', 'Your timeline', 'Your contributions', 'Location sharing', 'Start driving', 'Explore', 'Wi-Fi only', and 'Offline maps'. The 'Location sharing' option is selected. On the right, a map of New York City is displayed with a location pin labeled 'B' highlighted by a red box. Below the map, a card for 'Brown Huawei Two Rudnick' shows '0 ft away · Just now', 'New York, NY, USA', and a 'DIRECTIONS' button. At the bottom of the card, it says 'Brown Huawei Two can see your location Ongoing' with a toggle switch.</p> <p>Location information is shared via IP-based communication resulting in map that displays location information</p> <p><b><u>Exemplary Source Code:</u></b> The above functionality is performed at least in part by the following publicly available source code and/or source code that invokes or is invoked by the following source code (or a substantially similar copy compiled and loaded onto the Accused Products by HTC). AGIS reserves the right to amend these contentions to include additional source code as discovery progresses and as additional source code is made available.</p>

**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC
	<h2 data-bbox="527 240 1016 293">Contacts Provider</h2> <p data-bbox="527 329 1472 589">The Contacts Provider is a powerful and flexible Android component that manages the device's central repository of data about people. The Contacts Provider is the source of data you see in the device's contacts application, and you can also access its data in your own application and transfer data between the device and online services. The provider accommodates a wide range of data sources and tries to manage as much data as possible for each person, with the result that its organization is complex. Because of this, the provider's API includes an extensive set of contract classes and interfaces that facilitate both data retrieval and modification.</p> <p data-bbox="527 625 835 646">This guide describes the following:</p> <ul data-bbox="527 675 1373 850" style="list-style-type: none"><li data-bbox="527 675 806 696">• The basic provider structure.</li><li data-bbox="527 725 894 747">• How to retrieve data from the provider.</li><li data-bbox="527 776 863 797">• How to modify data in the provider.</li><li data-bbox="527 826 1373 847">• How to write a sync adapter for synchronizing data from your server to the Contacts Provider.</li></ul> <p data-bbox="510 863 1486 891"><a href="https://developer.android.com/guide/topics/providers/contacts-provider.html">https://developer.android.com/guide/topics/providers/contacts-provider.html</a></p>



**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC
	<p data-bbox="531 240 657 267"><b>Overview</b></p> <p data-bbox="531 297 1608 316">ContactsContract defines an extensible database of contact-related information. Contact information is stored in a three-tier data model:</p> <ul data-bbox="531 342 1713 545" style="list-style-type: none"><li data-bbox="531 342 1713 394">• A row in the <code>ContactsContract.Data</code> table can store any kind of personal data, such as a phone number or email addresses. The set of data kinds that can be stored in this table is open-ended. There is a predefined set of common kinds, but any application can add its own data kinds.</li><li data-bbox="531 418 1713 470">• A row in the <code>ContactsContract.RawContacts</code> table represents a set of data describing a person and associated with a single account (for example, one of the user's Gmail accounts).</li><li data-bbox="531 495 1713 545">• A row in the <code>ContactsContract.Contacts</code> table represents an aggregate of one or more RawContacts presumably describing the same person. When data in or associated with the RawContacts table is changed, the affected aggregate contacts are updated as necessary.</li></ul> <p data-bbox="531 574 695 594">Other tables include:</p> <ul data-bbox="531 620 1713 899" style="list-style-type: none"><li data-bbox="531 620 1713 672">• <code>ContactsContract.Groups</code>, which contains information about raw contact groups such as Gmail contact groups. The current API does not support the notion of groups spanning multiple accounts.</li><li data-bbox="531 696 1325 716">• <code>ContactsContract.StatusUpdates</code>, which contains social status updates including IM availability.</li><li data-bbox="531 740 1493 760">• <code>ContactsContract.AggregationExceptions</code>, which is used for manual aggregation and disaggregation of raw contacts</li><li data-bbox="531 784 1325 803">• <code>ContactsContract.Settings</code>, which contains visibility and sync settings for accounts and groups.</li><li data-bbox="531 828 1346 847">• <code>ContactsContract.SyncState</code>, which contains free-form data maintained on behalf of sync adapters</li><li data-bbox="531 872 1129 891">• <code>ContactsContract.PhoneLookup</code>, which is used for quick caller-ID lookup</li></ul> <p data-bbox="510 919 1545 948"><a href="https://developer.android.com/reference/android/provider/ContactsContract.html">https://developer.android.com/reference/android/provider/ContactsContract.html</a></p> <p data-bbox="531 959 615 987"><b>Data</b></p> <p data-bbox="531 1029 1749 1170">As noted previously, the data for a raw contact is stored in a <code>ContactsContract.Data</code> row that is linked to the raw contact's <code>_ID</code> value. This allows a single raw contact to have multiple instances of the same type of data such as email addresses or phone numbers. For example, if "Thomas Higginson" for <code>emilyd@gmail.com</code> (the raw contact row for Thomas Higginson associated with the Google account <code>emilyd@gmail.com</code>) has a home email address of <code>thigg@gmail.com</code> and a work email address of <code>thomas.higginson@gmail.com</code>, the Contacts Provider stores the two email address rows and links them both to the raw contact.</p> <p data-bbox="531 1196 1749 1308">Notice that different types of data are stored in this single table. Display name, phone number, email, postal address, photo, and website detail rows are all found in the <code>ContactsContract.Data</code> table. To help manage this, the <code>ContactsContract.Data</code> table has some columns with descriptive names, and others with generic names. The contents of a descriptive-name column have the same meaning regardless of the type of data in the row, while the contents of a generic-name column have different meanings depending on the type of data.</p> <p data-bbox="510 1326 1493 1356"><a href="https://developer.android.com/guide/topics/providers/contacts-provider.html">https://developer.android.com/guide/topics/providers/contacts-provider.html</a></p>

**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC														
	<table border="1"> <thead> <tr> <th>Task</th> <th>Action</th> <th>Data</th> <th>MIME type</th> <th>Notes</th> </tr> </thead> <tbody> <tr> <td>Pick a contact from a list</td> <td><a href="#">ACTION_PICK</a></td> <td>                     One of:                     <ul style="list-style-type: none"> <li><a href="#">Contacts.CONTENT_URI</a>, which displays a list of contacts.</li> <li><a href="#">Phone.CONTENT_URI</a>, which displays a list of phone numbers for a raw contact.</li> <li><a href="#">StructuredPostal.CONTENT_URI</a>, which displays a list of postal addresses for a raw contact.</li> <li><a href="#">Email.CONTENT_URI</a>, which displays a list of email addresses for a raw contact.</li> </ul> </td> <td>Not used</td> <td>                     Displays a list of raw contacts or a list of data from a raw contact, depending on the content URI type you supply. Call <a href="#">startActivityForResult()</a>, which returns the content URI of the selected row. The form of the URI is the table's content URI with the row's <a href="#">LOOKUP_ID</a> appended to it. The device's contacts app delegates read and write permissions to this content URI for the life of your activity. See the <a href="#">Content Provider Basics</a> guide for more details.                 </td> </tr> </tbody> </table>	Task	Action	Data	MIME type	Notes	Pick a contact from a list	<a href="#">ACTION_PICK</a>	One of: <ul style="list-style-type: none"> <li><a href="#">Contacts.CONTENT_URI</a>, which displays a list of contacts.</li> <li><a href="#">Phone.CONTENT_URI</a>, which displays a list of phone numbers for a raw contact.</li> <li><a href="#">StructuredPostal.CONTENT_URI</a>, which displays a list of postal addresses for a raw contact.</li> <li><a href="#">Email.CONTENT_URI</a>, which displays a list of email addresses for a raw contact.</li> </ul>	Not used	Displays a list of raw contacts or a list of data from a raw contact, depending on the content URI type you supply. Call <a href="#">startActivityForResult()</a> , which returns the content URI of the selected row. The form of the URI is the table's content URI with the row's <a href="#">LOOKUP_ID</a> appended to it. The device's contacts app delegates read and write permissions to this content URI for the life of your activity. See the <a href="#">Content Provider Basics</a> guide for more details.				
Task	Action	Data	MIME type	Notes											
Pick a contact from a list	<a href="#">ACTION_PICK</a>	One of: <ul style="list-style-type: none"> <li><a href="#">Contacts.CONTENT_URI</a>, which displays a list of contacts.</li> <li><a href="#">Phone.CONTENT_URI</a>, which displays a list of phone numbers for a raw contact.</li> <li><a href="#">StructuredPostal.CONTENT_URI</a>, which displays a list of postal addresses for a raw contact.</li> <li><a href="#">Email.CONTENT_URI</a>, which displays a list of email addresses for a raw contact.</li> </ul>	Not used	Displays a list of raw contacts or a list of data from a raw contact, depending on the content URI type you supply. Call <a href="#">startActivityForResult()</a> , which returns the content URI of the selected row. The form of the URI is the table's content URI with the row's <a href="#">LOOKUP_ID</a> appended to it. The device's contacts app delegates read and write permissions to this content URI for the life of your activity. See the <a href="#">Content Provider Basics</a> guide for more details.											
<a href="https://developer.android.com/guide/topics/providers/contacts-provider.html">https://developer.android.com/guide/topics/providers/contacts-provider.html</a>															

**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC
	<pre>25  * Parsed form of the intent sent to the Contacts application. 26  */ 27  public class ContactsRequest { 28 29      /** Default mode: browse contacts */ 30      public static final int ACTION_DEFAULT = 10; 31 32      /** Show all contacts */ 33      public static final int ACTION_ALL_CONTACTS = 15; 34 35      /** Show all contacts with phone numbers */ 36      public static final int ACTION_CONTACTS_WITH_PHONES = 17; 37 38      /** Show contents of a specific group */ 39      public static final int ACTION_GROUP = 20; 40 41      /** Show all starred contacts */ 42      public static final int ACTION_STARRED = 30; 43 44      /** Show frequently contacted contacts */ 45      public static final int ACTION_FREQUENT = 40; 46 47      /** Show starred and the frequent */ 48      public static final int ACTION_STREQUENT = 50; 49 50      /** Show all contacts and pick them when clicking */ 51      public static final int ACTION_PICK_CONTACT = 60; 52 53      /** Show all contacts as well as the option to create a new one */ 54      public static final int ACTION_PICK_OR_CREATE_CONTACT = 70; 55 56      /** Show all contacts and pick them for edit when clicking, and allow creating a new contact */ 57      public static final int ACTION_INSERT_OR_EDIT_CONTACT = 80;</pre> <p><a href="https://android.googlesource.com/platform/packages/apps/Contacts/+nougat-mr1-release/src/com/android/contacts/list/ContactsRequest.java">https://android.googlesource.com/platform/packages/apps/Contacts/+nougat-mr1-release/src/com/android/contacts/list/ContactsRequest.java</a></p>

**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC
	<pre>59  /** Show all phone numbers and pick them when clicking */ 60  public static final int ACTION_PICK_PHONE = 90; 61 62  /** Show all postal addresses and pick them when clicking */ 63  public static final int ACTION_PICK_POSTAL = 100; 64 65  /** Show all postal addresses and pick them when clicking */ 66  public static final int ACTION_PICK_EMAIL = 105; 67 68  /** Show all contacts and create a shortcut for the picked contact */ 69  public static final int ACTION_CREATE_SHORTCUT_CONTACT = 110; 70 71  /** Show all phone numbers and create a call shortcut for the picked number */ 72  public static final int ACTION_CREATE_SHORTCUT_CALL = 120; 73 74  /** Show all phone numbers and create an SMS shortcut for the picked number */ 75  public static final int ACTION_CREATE_SHORTCUT_SMS = 130; 76 77  /** Show all contacts and activate the specified one */ 78  public static final int ACTION_VIEW_CONTACT = 140; 79 80  /** Show contacts recommended for joining with a specified target contact */ 81  public static final int ACTION_PICK_JOIN = 150;</pre> <p><a href="https://android.googlesource.com/platform/packages/apps/Contacts+/nougat-mr1-release/src/com/android/contacts/list/ContactsRequest.java">https://android.googlesource.com/platform/packages/apps/Contacts+/nougat-mr1-release/src/com/android/contacts/list/ContactsRequest.java</a></p>

**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC
	<pre>104 * Displays a list to browse contacts. 105 */ 106 public class PeopleActivity extends ContactsActivity implements 107     View.OnCreateContextMenuListener, 108     View.OnClickListener, 109     ActionBarAdapter.Listener, 110     DialogManager.DialogShowingViewActivity, 111     ContactListFilterController.ContactListFilterListener, 112     ProviderStatusListener, 113     MultiContactDeleteListener, 114     JoinContactsListener { https://android.googleusercontent.com/platform/packages/apps/Contacts+/nougat-mr1- release/src/com/android/contacts/activities/PeopleActivity.java 145 * Showing a list of Contacts. Also used for showing search results in search mode. 146 */ 147 private MultiSelectContactsListFragment mAllFragment; 148 private ContactTileListFragment mFavoritesFragment; https://android.googleusercontent.com/platform/packages/apps/Contacts+/nougat-mr1- release/src/com/android/contacts/activities/PeopleActivity.java</pre>

**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC
	<pre>458     private void configureFragments(boolean fromRequest) { 459         if (fromRequest) { 460             ContactListFilter filter = null; 461             int actionCode = mRequest.getActionCode(); 462             boolean searchMode = mRequest.isSearchMode(); 463             final int tabToOpen; 464             switch (actionCode) { 465                 case ContactsRequest.ACTION_ALL_CONTACTS: 466                     filter = ContactListFilter.createFilterWithType( 467                         ContactListFilter.FILTER_TYPE_ALL_ACCOUNTS); 468                     tabToOpen = TabState.ALL; 469                     break; 470                 case ContactsRequest.ACTION_CONTACTS_WITH_PHONES: 471                     filter = ContactListFilter.createFilterWithType( 472                         ContactListFilter.FILTER_TYPE_WITH_PHONE_NUMBERS_ONLY); 473                     tabToOpen = TabState.ALL; 474                     break; 475 476                 case ContactsRequest.ACTION_FREQUENT: 477                 case ContactsRequest.ACTION_STREQUENT: 478                 case ContactsRequest.ACTION_STARRED: 479                     tabToOpen = TabState.FAVORITES; 480                     break; 481                 case ContactsRequest.ACTION_VIEW_CONTACT: 482                     tabToOpen = TabState.ALL; 483                     break; 484                 default: 485                     tabToOpen = -1; 486                     break; 487             } </pre> <p data-bbox="506 1321 1566 1390"><a href="https://android.googlesource.com/platform/packages/apps/Contacts+/nougat-mr1-release/src/com/android/contacts/activities/PeopleActivity.java">https://android.googlesource.com/platform/packages/apps/Contacts+/nougat-mr1-release/src/com/android/contacts/activities/PeopleActivity.java</a></p>

**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC
	<pre>488         if (tabToOpen != -1) { 489             mActionBarAdapter.setCurrentTab(tabToOpen); 490         } 491 492         if (filter != null) { 493             mContactListFilterController.setContactListFilter(filter, false); 494             searchMode = false; 495         } 496 497         if (mRequest.getContactUri() != null) { 498             searchMode = false; 499         } 500 501         mActionBarAdapter.setSearchMode(searchMode); 502         configureContactListFragmentForRequest(); 503     } 504 505     configureContactListFragment(); 506 507     invalidateOptionsMenuIfNeeded(); 508 }</pre> <p data-bbox="506 1019 1570 1084"><a href="https://android.googlesource.com/platform/packages/apps/Contacts+/nougat-mr1-release/src/com/android/contacts/activities/PeopleActivity.java">https://android.googlesource.com/platform/packages/apps/Contacts+/nougat-mr1-release/src/com/android/contacts/activities/PeopleActivity.java</a></p>

**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC
	<pre>35 public class ProfileAndContactsLoader extends CursorLoader { 36 37     private boolean mLoadProfile; 38 39     private String[] mProjection; 40 41     private Uri mExtraUri; 42     private String[] mExtraProjection; 43     private String mExtraSelection; 44     private String[] mExtraSelectionArgs; 45     private boolean mMergeExtraContactsAfterPrimary; 46 47     public ProfileAndContactsLoader(Context context) { 48         super(context); 49     }</pre> <p><a href="https://android.googlesource.com/platform/packages/apps/ContactsCommon/+/nougat-mr1-release/src/com/android/contacts/common/list/ProfileAndContactsLoader.java">https://android.googlesource.com/platform/packages/apps/ContactsCommon/+/nougat-mr1-release/src/com/android/contacts/common/list/ProfileAndContactsLoader.java</a></p>



**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC
	<pre>35 public final class GroupMemberLoader extends CursorLoader { 36 37     public static class GroupEditorQuery { 38         private static final String[] PROJECTION = new String[] { 39             Data.CONTACT_ID,           // 0 40             Data.RAW_CONTACT_ID,       // 1 41             Data.DISPLAY_NAME_PRIMARY, // 2 42             Data.PHOTO_URI,            // 3 43             Data.LOOKUP_KEY,           // 4 44         }; 45 46         public static final int CONTACT_ID           = 0; 47         public static final int RAW_CONTACT_ID       = 1; 48         public static final int CONTACT_DISPLAY_NAME_PRIMARY = 2; 49         public static final int CONTACT_PHOTO_URI    = 3; 50         public static final int CONTACT_LOOKUP_KEY   = 4; 51     } 52 53     public static class GroupDetailQuery { 54         private static final String[] PROJECTION = new String[] { 55             Data.CONTACT_ID,           // 0 56             Data.PHOTO_URI,            // 1 57             Data.LOOKUP_KEY,           // 2 58             Data.DISPLAY_NAME_PRIMARY, // 3 59             Data.CONTACT_PRESENCE,     // 4 60             Data.CONTACT_STATUS,       // 5 61         }; 62 63         public static final int CONTACT_ID           = 0; 64         public static final int CONTACT_PHOTO_URI    = 1; 65         public static final int CONTACT_LOOKUP_KEY   = 2; 66         public static final int CONTACT_DISPLAY_NAME_PRIMARY = 3; 67         public static final int CONTACT_PRESENCE_STATUS = 4; 68         public static final int CONTACT_STATUS       = 5; 69     } 70 71     private final long mGroupId;</pre> <p><a href="https://android.googlesource.com/platform/packages/apps/Contacts/+nougat-mr1-release/src/com/android/contacts/GroupMemberLoader.java">https://android.googlesource.com/platform/packages/apps/Contacts/+nougat-mr1-release/src/com/android/contacts/GroupMemberLoader.java</a></p>

**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC
	<pre>24  * Group loader for the group list that includes details such as the number of contacts per group 25  * and number of groups per account. This list is sorted by account type, account name, where the 26  * group names are in alphabetical order. Note that the list excludes default, favorite, and deleted 27  * groups. 28  */ 29  public final class GroupListLoader extends CursorLoader { 30 31      private final static String[] COLUMNS = new String[] { 32          Groups.ACCOUNT_NAME, 33          Groups.ACCOUNT_TYPE, 34          Groups.DATA_SET, 35          Groups._ID, 36          Groups.TITLE, 37          Groups.SUMMARY_COUNT, 38      }; 39 40      public final static int ACCOUNT_NAME = 0; 41      public final static int ACCOUNT_TYPE = 1; 42      public final static int DATA_SET = 2; 43      public final static int GROUP_ID = 3; 44      public final static int TITLE = 4; 45      public final static int MEMBER_COUNT = 5; 46 47      private static final Uri GROUP_LIST_URI = Groups.CONTENT_SUMMARY_URI; 48 49      public GroupListLoader(Context context) { 50          super(context, GROUP_LIST_URI, COLUMNS, Groups.ACCOUNT_TYPE + " NOT NULL AND " 51              + Groups.ACCOUNT_NAME + " NOT NULL AND " + Groups.AUTO_ADD + "=0 AND " + 52              Groups.FAVORITES + "=0 AND " + Groups.DELETED + "=0", null, 53              Groups.ACCOUNT_TYPE + ", " + Groups.ACCOUNT_NAME + ", " + Groups.DATA_SET + ", " + 54              Groups.TITLE + " COLLATE LOCALIZED ASC"); 55      } 56  }</pre> <p><a href="https://android.googlesource.com/platform/packages/apps/Contacts+/nougat-mr1-release/src/com/android/contacts/GroupListLoader.java">https://android.googlesource.com/platform/packages/apps/Contacts+/nougat-mr1-release/src/com/android/contacts/GroupListLoader.java</a></p>

**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC
	<pre> 24  * Group meta-data loader. Loads all groups or just a single group from the 25  * database (if given a {@link Uri}). 26  */ 27  public final class GroupMetaLoader extends CursorLoader { 28 29      private final static String[] COLUMNS = new String[] { 30          Groups.ACCOUNT_NAME, 31          Groups.ACCOUNT_TYPE, 32          Groups.DATA_SET, 33          Groups._ID, 34          Groups.TITLE, 35          Groups.AUTO_ADD, 36          Groups.FAVORITES, 37          Groups.GROUP_IS_READ_ONLY, 38          Groups.DELETED, 39      }; 40 41      public final static int ACCOUNT_NAME = 0; 42      public final static int ACCOUNT_TYPE = 1; 43      public final static int DATA_SET = 2; 44      public final static int GROUP_ID = 3; 45      public final static int TITLE = 4; 46      public final static int AUTO_ADD = 5; 47      public final static int FAVORITES = 6; 48      public final static int IS_READ_ONLY = 7; 49      public final static int DELETED = 8; 50 51      public GroupMetaLoader(Context context, Uri groupUri) { 52          super(context, ensureIsGroupUri(groupUri), COLUMNS, Groups.ACCOUNT_TYPE + " NOT NULL AND " 53              + Groups.ACCOUNT_NAME + " NOT NULL", null, null); 54      } 55 56      /** 57       * Ensures that this is a valid group URI. If invalid, then an exception is 58       * thrown. Otherwise, the original URI is returned. 59       */ 60      private static Uri ensureIsGroupUri(final Uri groupUri) { 61          // TODO: Fix ContactsProvider2 getType method to resolve the group Uris 62          if (groupUri == null) { 63              throw new IllegalArgumentException("Uri must not be null"); 64          } 65          if (!groupUri.toString().startsWith(Groups.CONTENT_URI.toString())) { 66              throw new IllegalArgumentException("Invalid group Uri: " + groupUri); 67          } 68          return groupUri; 69      } 70  } </pre> <p><a href="https://android.googlesource.com/platform/packages/apps/Contacts/+nougat-mr1-release/src/com/android/contacts/GroupMetaLoader.java">https://android.googlesource.com/platform/packages/apps/Contacts/+nougat-mr1-release/src/com/android/contacts/GroupMetaLoader.java</a></p>

**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC
	<pre>19  * Meta-data for a contact group. We load all groups associated with the contact's 20  * constituent accounts. 21  */ 22  public final class GroupMetaData { 23      private String mAccountName; 24      private String mAccountType; 25      private String mDataSet; 26      private long mGroupId; 27      private String mTitle; 28      private boolean mDefaultGroup; 29      private boolean mFavorites; 30 31      public GroupMetaData(String accountName, String accountType, String dataSet, long groupId, 32                          String title, boolean defaultGroup, boolean favorites) { 33          this.mAccountName = accountName; 34          this.mAccountType = accountType; 35          this.mDataSet = dataSet; 36          this.mGroupId = groupId; 37          this.mTitle = title; 38          this.mDefaultGroup = defaultGroup; 39          this.mFavorites = favorites; 40      } 41 42      public String getAccountName() { 43          return mAccountName; 44      } 45 46      public String getAccountType() { 47          return mAccountType; 48      } 49 50      public String getDataSet() { 51          return mDataSet; 52      } 53 54      public long getGroupId() { 55          return mGroupId; 56      } 57 58      public String getTitle() { 59          return mTitle; 60      } 61 62      public boolean isDefaultGroup() { 63          return mDefaultGroup; 64      } 65 66      public boolean isFavorites() { 67          return mFavorites; 68      } </pre> <p><a href="https://android.googlesource.com/platform/packages/apps/ContactsCommon/+nougat-mr1-release/src/com/android/contacts/common/GroupMetaData.java">https://android.googlesource.com/platform/packages/apps/ContactsCommon/+nougat-mr1-release/src/com/android/contacts/common/GroupMetaData.java</a></p>

**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC
	<pre>44 * Class that sends chat message via SMS. 45 * 46 * The interface emulates a blocking sending similar to making an HTTP request. 47 * It calls the SmsManager to send a (potentially multipart) message and waits 48 * on the sent status on each part. The waiting has a timeout so it won't wait 49 * forever. Once the sent status of all parts received, the call returns. 50 * A successful sending requires success status for all parts. Otherwise, we 51 * pick the highest level of failure as the error for the whole message, which 52 * is used to determine if we need to retry the sending. 53 */ 54 public class SmsSender { 55     private static final String TAG = LogUtil.BUGLE_TAG; 56 57     public static final String EXTRA_PART_ID = "part_id"; 58 59     /* 60      * A map for pending sms messages. The key is the random request UUID. 61      */ 62     private static ConcurrentHashMap&lt;Uri, SendResult&gt; sPendingMessageMap = 63         new ConcurrentHashMap&lt;Uri, SendResult&gt;(); 64 65     private static final Random RANDOM = new Random(); 66 67     // Whether we should send multipart SMS as separate messages 68     private static Boolean sSendMultipartSmsAsSeparateMessages = null; 69</pre> <p><a href="https://android.googlesource.com/platform/packages/apps/Messaging/+nougat-mr1-release/src/com/android/messaging/sms/SmsSender.java">https://android.googlesource.com/platform/packages/apps/Messaging/+nougat-mr1-release/src/com/android/messaging/sms/SmsSender.java</a></p>

**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC
	<pre> 253 // Actually sending the message using SmsManager 254 private static void sendInternal(final Context context, final int subId, String dest, 255     final ArrayList&lt;String&gt; messages, final String serviceCenter, 256     final boolean requireDeliveryReport, final Uri messageUri) throws SmsException { 257     Assert.notNull(context); 258     final SmsManager smsManager = PhoneUtils.get(subId).getSmsManager(); 259     final int messageCount = messages.size(); 260     final ArrayList&lt;PendingIntent&gt; deliveryIntents = new ArrayList&lt;PendingIntent&gt;(messageCount); 261     final ArrayList&lt;PendingIntent&gt; sentIntents = new ArrayList&lt;PendingIntent&gt;(messageCount); 262     for (int i = 0; i &lt; messageCount; i++) { 263         // Make pending intents different for each message part 264         final int partId = (messageCount &lt;= 1 ? 0 : i + 1); 265         if (requireDeliveryReport &amp;&amp; (i == (messageCount - 1))) { 266             // TODO we only care about the delivery status of the last part 267             // Shall we have better tracking of delivery status of all parts? 268             deliveryIntents.add(PendingIntent.getBroadcast( 269                 context, 270                 partId, 271                 getSendStatusIntent(context, SendStatusReceiver.MESSAGE_DELIVERED_ACTION, 272                     messageUri, partId, subId), 273                 0/*flag*/)); 274         } else { 275             deliveryIntents.add(null); 276         } 277         sentIntents.add(PendingIntent.getBroadcast( 278             context, 279             partId, 280             getSendStatusIntent(context, SendStatusReceiver.MESSAGE_SENT_ACTION, 281                 messageUri, partId, subId), 282             0/*flag*/)); 283     } 284     if (sSendMultipartSmsAsSeparateMessages == null) { 285         sSendMultipartSmsAsSeparateMessages = MmsConfig.get(subId) 286             .getSendMultipartSmsAsSeparateMessages(); 287     } </pre> <p><a href="https://android.googlesource.com/platform/packages/apps/Messaging/+nougat-mr1-release/src/com/android/messaging/sms/SmsSender.java">https://android.googlesource.com/platform/packages/apps/Messaging/+nougat-mr1-release/src/com/android/messaging/sms/SmsSender.java</a></p>

**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC
	<pre>288     try { 289         if (sSendMultipartSmsAsSeparateMessages) { 290             // If multipart sms is not supported, send them as separate messages 291             for (int i = 0; i &lt; messageCount; i++) { 292                 smsManager.sendTextMessage(dest, 293                     serviceCenter, 294                     messages.get(i), 295                     sentIntents.get(i), 296                     deliveryIntents.get(i)); 297             } 298         } else { 299             smsManager.sendMultipartTextMessage( 300                 dest, serviceCenter, messages, sentIntents, deliveryIntents); 301         } 302     } catch (final Exception e) { 303         throw new SmsException("SmsSender: caught exception in sending " + e); 304     } 305 }</pre> <p><a href="https://android.googlesource.com/platform/packages/apps/Messaging/+nougat-mr1-release/src/com/android/messaging/sms/SmsSender.java">https://android.googlesource.com/platform/packages/apps/Messaging/+nougat-mr1-release/src/com/android/messaging/sms/SmsSender.java</a></p> <pre>56  * Class that receives incoming SMS messages through android.provider.Telephony.SMS_RECEIVED 57  * 58  * This class serves two purposes: 59  * - Process phone verification SMS messages 60  * - Handle SMS messages when the user has enabled us to be the default SMS app (Pre-KLP) 61  */ 62  public final class SmsReceiver extends BroadcastReceiver { 63      private static final String TAG = LogUtil.BUGLE_TAG; 64 65      private static ArrayList&lt;Pattern&gt; sIgnoreSmsPatterns; 66  }</pre> <p><a href="https://android.googlesource.com/platform/packages/apps/Messaging/+nougat-mr1-release/src/com/android/messaging/receiver/SmsReceiver.java">https://android.googlesource.com/platform/packages/apps/Messaging/+nougat-mr1-release/src/com/android/messaging/receiver/SmsReceiver.java</a></p>

**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC
	<pre>200     public static void deliverSmsMessages(final Context context, final int subId, 201         final int errorCode, final android.telephony.SmsMessage[] messages) { 202         final ContentValues messageValues = 203             MmsUtils.parseReceivedSmsMessage(context, messages, errorCode); 204 205         LogUtil.v(TAG, "SmsReceiver.deliverSmsMessages"); 206 207         final long nowInMillis = System.currentTimeMillis(); 208         final long receivedTimestampMs = MmsUtils.getMessageDate(messages[0], nowInMillis); 209 210         messageValues.put(Sms.Inbox.DATE, receivedTimestampMs); 211         // Default to unread and unseen for us but ReceiveSmsMessageAction will override 212         // seen for the telephony db. 213         messageValues.put(Sms.Inbox.READ, 0); 214         messageValues.put(Sms.Inbox.SEEN, 0); 215         if (OsUtil.isAtLeastL_MR1()) { 216             messageValues.put(Sms.SUBSCRIPTION_ID, subId); 217         } 218 219         if (messages[0].getMessageClass() == android.telephony.SmsMessage.MessageClass.CLASS_0    220             DebugUtils.debugClassZeroSmsEnabled()) { 221             Factory.get().getUIIntents().launchClassZeroActivity(context, messageValues); 222         } else { 223             final ReceiveSmsMessageAction action = new ReceiveSmsMessageAction(messageValues); 224             action.start(); 225         } 226     }</pre> <p><a href="https://android.googlesource.com/platform/packages/apps/Messaging/+nougat-mr1-release/src/com/android/messaging/receiver/SmsReceiver.java">https://android.googlesource.com/platform/packages/apps/Messaging/+nougat-mr1-release/src/com/android/messaging/receiver/SmsReceiver.java</a></p>



**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC
	<pre>228     @Override 229     public void onReceive(final Context context, final Intent intent) { 230         LogUtil.v(TAG, "SmsReceiver.onReceive " + intent); 231         // On KLP+ we only take delivery of SMS messages in SmsDeliverReceiver. 232         if (PhoneUtils.getDefault().isSmsEnabled()) { 233             final String action = intent.getAction(); 234             if (OsUtil.isSecondaryUser() &amp;&amp; 235                 (Telephony.Sms.Intents.SMS_RECEIVED_ACTION.equals(action)    236                  // TODO: update this with the actual constant from Telephony 237                  "android.provider.Telephony.MMS_DOWNLOADED".equals(action))) { 238                 postNewMessageSecondaryUserNotification(); 239             } else if (!OsUtil.isAtLeastKLP()) { 240                 deliverSmsIntent(context, intent); 241             } 242         } 243     }</pre> <p><a href="https://android.googlesource.com/platform/packages/apps/Messaging/+/nougat-mr1-release/src/com/android/messaging/receiver/SmsReceiver.java">https://android.googlesource.com/platform/packages/apps/Messaging/+/nougat-mr1-release/src/com/android/messaging/receiver/SmsReceiver.java</a></p>

**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC
	<pre>52  * Class that sends chat message via MMS. 53  * 54  * The interface emulates a blocking send similar to making an HTTP request. 55  */ 56  public class MmsSender { 57      private static final String TAG = LogUtil.BUGLE_TAG; 58 59      /** 60       * Send an MMS message. 61       * 62       * @param context Context 63       * @param messageUri The unique URI of the message for identifying it during sending 64       * @param sendReq The SendReq PDU of the message 65       * @throws MmsFailureException 66       */ 67      public static void sendMms(final Context context, final int subId, final Uri messageUri, 68                               final SendReq sendReq, final Bundle sentIntentExtras) throws MmsFailureException { 69          sendMms(context, 70                subId, 71                messageUri, 72                null /* locationUri */, 73                sendReq, 74                true /* responseImportant */, 75                sentIntentExtras); 76      }</pre> <p><a href="https://android.googlesource.com/platform/packages/apps/Messaging/+/nougat-mr1-release/src/com/android/messaging/sms/MmsSender.java">https://android.googlesource.com/platform/packages/apps/Messaging/+/nougat-mr1-release/src/com/android/messaging/sms/MmsSender.java</a></p>

**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC
	<pre>240     * Download an MMS message. 241     * 242     * @param context Context 243     * @param contentLocation The url of the MMS message 244     * @throws MmsFailureException 245     * @throws InvalidHeaderValueException 246     */ 247     public static void downloadMms(final Context context, final int subId, 248         final String contentLocation, Bundle extras) throws MmsFailureException, 249         InvalidHeaderValueException { 250         final Uri requestUri = Uri.parse(contentLocation); 251         final Uri contentUri = MmsFileProvider.buildRawMmsUri(); 252 253         final Intent downloadedIntent = new Intent(SendStatusReceiver.MMS_DOWNLOADED_ACTION, 254             requestUri, 255             context, 256             SendStatusReceiver.class); 257         downloadedIntent.putExtra(SendMessageAction.EXTRA_CONTENT_URI, contentUri); 258         if (extras != null) { 259             downloadedIntent.putExtras(extras); 260         } 261         final PendingIntent downloadedPendingIntent = PendingIntent.getBroadcast( 262             context, 263             0 /*request code*/, 264             downloadedIntent, 265             PendingIntent.FLAG_UPDATE_CURRENT); 266 267         MmsManager.downloadMultimediaMessage(subId, context, contentLocation, contentUri, 268             downloadedPendingIntent); 269     }</pre> <p><a href="https://android.googlesource.com/platform/packages/apps/Messaging/+nougat-mr1-release/src/com/android/messaging/sms/MmsSender.java">https://android.googlesource.com/platform/packages/apps/Messaging/+nougat-mr1-release/src/com/android/messaging/sms/MmsSender.java</a></p>

**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC
	<pre> 97      * Execute an MMS HTTP request, either a POST (sending) or a GET (downloading) 98      * 99      * @param urlString The request URL, for sending it is usually the MMSC, and for downloading 100     *           it is the message URL 101     * @param pdu For POST (sending) only, the PDU to send 102     * @param method HTTP method, POST for sending and GET for downloading 103     * @param isProxySet Is there a proxy for the MMSC 104     * @param proxyHost The proxy host 105     * @param proxyPort The proxy port 106     * @param mmsConfig The MMS config to use 107     * @param userAgent The user agent header value 108     * @param uaProfUrl The UA Prof URL header value 109     * @return The HTTP response body 110     * @throws MmsHttpException For any failures 111     */ 112     public byte[] execute(String urlString, byte[] pdu, String method, boolean isProxySet, 113                          String proxyHost, int proxyPort, Bundle mmsConfig, String userAgent, String uaProfUrl) 114         throws MmsHttpException { 115         Log.d(MmsService.TAG, "HTTP: " + method + " " + Utils.redactUrlForNonVerbose(urlString) 116             + (isProxySet ? (" proxy=" + proxyHost + ":" + proxyPort) : "")) 117             + ", PDU size=" + (pdu != null ? pdu.length : 0)); 118         checkMethod(method); 119         HttpURLConnection connection = null; 120         try { 121             Proxy proxy = Proxy.NO_PROXY; 122             if (isProxySet) { 123                 proxy = new Proxy(Proxy.Type.HTTP, new InetSocketAddress(proxyHost, proxyPort)); 124             } 125             final URL url = new URL(urlString); 126             // Now get the connection 127             connection = (HttpURLConnection) url.openConnection(proxy); 128             connection.setDoInput(true); 129             connection.setConnectTimeout( 130                 mmsConfig.getInt(CarrierConfigValuesLoader.CONFIG_HTTP_SOCKET_TIMEOUT, 131                     CarrierConfigValuesLoader.CONFIG_HTTP_SOCKET_TIMEOUT_DEFAULT)); </pre> <p data-bbox="506 1219 1596 1287"><a href="https://android.googlesource.com/platform/packages/apps/Messaging/+/nougat-mr1-release/src/android/support/v7/mms/MmsHttpClient.java">https://android.googlesource.com/platform/packages/apps/Messaging/+/nougat-mr1-release/src/android/support/v7/mms/MmsHttpClient.java</a></p>

**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC
	<pre> 132 // ----- COMMON HEADERS ----- 133 // Header: Accept 134 connection.setRequestProperty(HEADER_ACCEPT, HEADER_VALUE_ACCEPT); 135 // Header: Accept-Language 136 connection.setRequestProperty( 137     HEADER_ACCEPT_LANGUAGE, getCurrentAcceptLanguage(Locale.getDefault())); 138 // Header: User-Agent 139 Log.i(MmsService.TAG, "HTTP: User-Agent=" + userAgent); 140 connection.setRequestProperty(HEADER_USER_AGENT, userAgent); 141 // Header: x-wap-profile 142 final String uaProfUrlTagName = mmsConfig.getString( 143     CarrierConfigValuesLoader.CONFIG_UA_PROF_TAG_NAME, 144     CarrierConfigValuesLoader.CONFIG_UA_PROF_TAG_NAME_DEFAULT); 145 if (uaProfUrl != null) { 146     Log.i(MmsService.TAG, "HTTP: UaProfUrl=" + uaProfUrl); 147     connection.setRequestProperty(uaProfUrlTagName, uaProfUrl); 148 } 149 // Add extra headers specified by mms_config.xml's httpparams 150 addExtraHeaders(connection, mmsConfig); 151 // Different stuff for GET and POST 152 if (METHOD_POST.equals(method)) { 153     if (pdu == null    pdu.length &lt; 1) { 154         Log.e(MmsService.TAG, "HTTP: empty pdu"); 155         throw new MmsHttpException(0/*statusCode*/, "Sending empty PDU"); 156     } 157     connection.setDoOutput(true); 158     connection.setRequestMethod(METHOD_POST); 159     if (mmsConfig.getBoolean( 160         CarrierConfigValuesLoader.CONFIG_SUPPORT_HTTP_CHARSET_HEADER, 161         CarrierConfigValuesLoader.CONFIG_SUPPORT_HTTP_CHARSET_HEADER_DEFAULT)) { 162         connection.setRequestProperty(HEADER_CONTENT_TYPE, 163             HEADER_VALUE_CONTENT_TYPE_WITH_CHARSET); 164     } else { 165         connection.setRequestProperty(HEADER_CONTENT_TYPE, 166             HEADER_VALUE_CONTENT_TYPE_WITHOUT_CHARSET); </pre> <p><a href="https://android.googlesource.com/platform/packages/apps/Messaging/+nougat-mr1-release/src/android/support/v7/mms/MmsHttpClient.java">https://android.googlesource.com/platform/packages/apps/Messaging/+nougat-mr1-release/src/android/support/v7/mms/MmsHttpClient.java</a></p>

**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC
	<pre>167         } 168         if (Log.isLoggable(MmsService.TAG, Log.VERBOSE)) { 169             logHttpHeaders(connection.getRequestProperties()); 170         } 171         connection.setFixedLengthStreamingMode(pdu.length); 172         // Sending request body 173         final OutputStream out = 174             new BufferedOutputStream(connection.getOutputStream()); 175         out.write(pdu); 176         out.flush(); 177         out.close(); 178     } else if (METHOD_GET.equals(method)) { 179         if (Log.isLoggable(MmsService.TAG, Log.VERBOSE)) { 180             logHttpHeaders(connection.getRequestProperties()); 181         } 182         connection.setRequestMethod(METHOD_GET); 183     } 184     // Get response 185     final int responseCode = connection.getResponseCode(); 186     final String responseMessage = connection.getResponseMessage(); 187     Log.d(MmsService.TAG, "HTTP: " + responseCode + " " + responseMessage); 188     if (Log.isLoggable(MmsService.TAG, Log.VERBOSE)) { 189         logHttpHeaders(connection.getHeaderFields()); 190     } 191     if (responseCode / 100 != 2) { 192         throw new MmsHttpException(responseCode, responseMessage); 193     } 194     final InputStream in = new BufferedInputStream(connection.getInputStream()); 195     final ByteArrayOutputStream byteOut = new ByteArrayOutputStream(); 196     final byte[] buf = new byte[4096]; 197     int count = 0; 198     while ((count = in.read(buf)) &gt; 0) { 199         byteOut.write(buf, 0, count); 200     } 201     in.close(); 202     final byte[] responseBody = byteOut.toByteArray(); 203     Log.d(MmsService.TAG, "HTTP: response size=" 204         + (responseBody != null ? responseBody.length : 0)); 205     return responseBody;</pre> <p><a href="https://android.googlesource.com/platform/packages/apps/Messaging/+nougat-mr1-release/src/android/support/v7/mms/MmsHttpClient.java">https://android.googlesource.com/platform/packages/apps/Messaging/+nougat-mr1-release/src/android/support/v7/mms/MmsHttpClient.java</a></p>

**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC
	<pre>38  * Request to send an MMS 39  */ 40  class SendRequest extends MmsRequest { 41      // Max send response PDU size in bytes (exceeding this may cause problem with 42      // system intent delivery). 43      private static final int MAX_SEND_RESPONSE_SIZE = 1000 * 1024; 44 45      private byte[] mPduData; 46 47      SendRequest(final String locationUri, final Uri pduUri, final PendingIntent sentIntent) { 48          super(locationUri, pduUri, sentIntent); 49      } 50 51      @Override 52      protected boolean loadRequest(final Context context, final Bundle mmsConfig) { 53          mPduData = readPduFromContentUri( 54              context, 55              mPduUri, 56              mmsConfig.getInt( 57                  CarrierConfigValuesLoader.CONFIG_MAX_MESSAGE_SIZE, 58                  CarrierConfigValuesLoader.CONFIG_MAX_MESSAGE_SIZE_DEFAULT)); 59          return (mPduData != null); 60      } 61 62      @Override 63      protected boolean transferResponse(final Context context, final Intent fillIn, 64          final byte[] response) { 65          // SendConf pdus are always small and can be included in the intent 66          if (response != null &amp;&amp; fillIn != null) { 67              if (response.length &gt; MAX_SEND_RESPONSE_SIZE) { 68                  // If the response PDU is too large, it won't be able to fit in 69                  // the PendingIntent to be transferred via system IPC. 70                  return false; 71              } 72              fillIn.putExtra(SmsManager.EXTRA_MMS_DATA, response); 73          } 74          return true; 75      } }</pre>

**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC
	<p><a href="https://android.googlesource.com/platform/packages/apps/Messaging/+/nougat-mr1-release/src/android/support/v7/mms/SendRequest.java">https://android.googlesource.com/platform/packages/apps/Messaging/+/nougat-mr1-release/src/android/support/v7/mms/SendRequest.java</a></p> <pre data-bbox="533 354 1738 399">public static LocationRequest create ()</pre> <p>Create a location request with default parameters.</p> <p>Default parameters are for a block accuracy, slowly updated location. It can then be adjusted as required by the applications before passing to the <a href="#">FusedLocationProviderApi</a>.</p> <p><b>Returns</b></p> <ul style="list-style-type: none"><li>• a new location request</li></ul> <p><a href="https://developers.google.com/android/reference/com/google/android/gms/location/LocationRequest">https://developers.google.com/android/reference/com/google/android/gms/location/LocationRequest</a></p>



**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC
	<p data-bbox="533 248 1745 285">public static final int <b>PRIORITY_BALANCED_POWER_ACCURACY</b></p> <p data-bbox="533 315 1178 342">Used with <code>setPriority(int)</code> to request "block" level accuracy.</p> <p data-bbox="533 371 1644 431">Block level accuracy is considered to be about 100 meter accuracy. Using a coarse accuracy such as this often consumes less power.</p> <p data-bbox="533 456 730 483">Constant Value: 102</p> <p data-bbox="533 537 1745 574">public static final int <b>PRIORITY_HIGH_ACCURACY</b></p> <p data-bbox="533 604 1339 631">Used with <code>setPriority(int)</code> to request the most accurate locations available.</p> <p data-bbox="533 660 961 688">This will return the finest location available.</p> <p data-bbox="533 717 730 745">Constant Value: 100</p> <p data-bbox="533 799 1745 836">public static final int <b>PRIORITY_LOW_POWER</b></p> <p data-bbox="533 865 1157 893">Used with <code>setPriority(int)</code> to request "city" level accuracy.</p> <p data-bbox="533 922 1738 982">City level accuracy is considered to be about 10km accuracy. Using a coarse accuracy such as this often consumes less power.</p> <p data-bbox="533 1006 730 1034">Constant Value: 104</p> <p data-bbox="533 1049 1797 1076"><a href="https://developers.google.com/android/reference/com/google/android/gms/location/LocationRequest">https://developers.google.com/android/reference/com/google/android/gms/location/LocationRequest</a></p>

**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC
	<pre data-bbox="533 248 1751 285">public Task&lt;Location&gt; getLastLocation ()</pre> <p data-bbox="525 313 1104 339">Returns the best most recent location currently available.</p> <p data-bbox="525 371 1696 431">If a location is not available, which should happen very rarely, null will be returned. The best accuracy available while respecting the location permissions will be returned.</p> <p data-bbox="525 464 1736 524">This method provides a simplified way to get location. It is particularly well suited for applications that do not require an accurate location and that do not want to maintain extra logic for location updates.</p> <pre data-bbox="533 578 1751 615">public Task&lt;LocationAvailability&gt; getLocationAvailability ()</pre> <p data-bbox="525 647 1692 708">Returns the availability of location data. When <code>isLocationAvailable()</code> returns true, then the location returned by <code>getLastLocation()</code> will be reasonably up to date within the hints specified by the active <code>LocationRequest</code>s.</p> <p data-bbox="525 740 1472 766">If the client isn't connected to Google Play services and the request times out, null is returned.</p> <p data-bbox="525 799 1673 859">Note it's always possible for <code>getLastLocation()</code> to return null even when this method returns true (e.g. location settings were disabled between calls).</p> <p data-bbox="510 867 1900 927"><a href="https://developers.google.com/android/reference/com/google/android/gms/location/FusedLocationProviderClient">https://developers.google.com/android/reference/com/google/android/gms/location/FusedLocationProviderClient</a></p>

**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC						
	<pre data-bbox="527 240 1747 326">public Task&lt;Void&gt; requestLocationUpdates (LocationRequest request,     LocationCallback callback, Looper looper)</pre> <p data-bbox="520 354 1272 380">Requests location updates with a callback on the specified Looper thread.</p> <p data-bbox="520 410 1686 472">This method is suited for the foreground use cases. For background use cases, the <code>PendingIntent</code> version of the method is recommended, see <code>requestLocationUpdates(LocationRequest, PendingIntent)</code>.</p> <p data-bbox="520 503 1371 529">Any previous LocationRequests registered on this LocationListener will be replaced.</p> <p data-bbox="520 560 1686 654">This call will keep the Google Play services connection active, so make sure to call <code>removeLocationUpdates(LocationCallback)</code> when you no longer need it, otherwise you lose the benefits of the automatic connection management.</p> <p data-bbox="520 685 1745 711">Callbacks for <code>LocationCallback</code> will be made on the specified thread, which must already be a prepared looper thread.</p> <p data-bbox="541 735 667 761"><b>Parameters</b></p> <table border="1" data-bbox="520 792 1747 1003"> <tbody> <tr> <td data-bbox="527 800 625 857"><b>request</b></td> <td data-bbox="636 800 1738 857">The location request for the updates.</td> </tr> <tr> <td data-bbox="527 865 625 922"><b>callback</b></td> <td data-bbox="636 865 1738 922">The callback for the location updates.</td> </tr> <tr> <td data-bbox="527 930 625 1003"><b>looper</b></td> <td data-bbox="636 930 1738 1003">The Looper object whose message queue will be used to implement the callback mechanism, or null to make callbacks on the calling thread.</td> </tr> </tbody> </table> <p data-bbox="512 1023 1902 1084"><a href="https://developers.google.com/android/reference/com/google/android/gms/location/FusedLocationProviderClient">https://developers.google.com/android/reference/com/google/android/gms/location/FusedLocationProviderClient</a></p>	<b>request</b>	The location request for the updates.	<b>callback</b>	The callback for the location updates.	<b>looper</b>	The Looper object whose message queue will be used to implement the callback mechanism, or null to make callbacks on the calling thread.
<b>request</b>	The location request for the updates.						
<b>callback</b>	The callback for the location updates.						
<b>looper</b>	The Looper object whose message queue will be used to implement the callback mechanism, or null to make callbacks on the calling thread.						

**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC				
	<pre data-bbox="533 240 1740 321">public Task&lt;Void&gt; requestLocationUpdates (LocationRequest request, PendingIntent callbackIntent)</pre> <p data-bbox="525 354 1268 378">Requests location updates with a callback on the specified PendingIntent.</p> <p data-bbox="525 410 1732 537">This method is suited for the background use cases, more specifically for receiving location updates, even when the app has been killed by the system. In order to do so, use a <code>PendingIntent</code> for a started service. For foreground use cases, the <code>LocationCallback</code> version of the method is recommended, see <code>requestLocationUpdates(LocationRequest, LocationCallback, Looper)</code>.</p> <p data-bbox="525 570 1724 630">Any previously registered requests that have the same PendingIntent (as defined by <code>equals(Object)</code>) will be replaced by this request.</p> <p data-bbox="525 662 1728 755">Both <code>LocationResult</code> and <code>LocationAvailability</code> are sent to the given PendingIntent. You can extract data from an Intent using <code>hasResult(Intent)</code>, <code>extractResult(Intent)</code>, <code>hasLocationAvailability(Intent)</code>, and <code>extractLocationAvailability(Intent)</code>.</p> <p data-bbox="543 781 667 805"><b>Parameters</b></p> <table border="1" data-bbox="525 833 1740 971"> <tbody> <tr> <td data-bbox="525 833 835 902"><code>request</code></td> <td data-bbox="835 833 1740 902">The location request for the updates.</td> </tr> <tr> <td data-bbox="525 902 835 971"><code>callbackIntent</code></td> <td data-bbox="835 902 1740 971">A pending intent to be sent for each location update.</td> </tr> </tbody> </table> <p data-bbox="543 997 630 1021"><b>Returns</b></p> <ul data-bbox="552 1044 1358 1068" style="list-style-type: none"> <li>• a Task for the call, check <code>isSuccessful()</code> to determine if it was successful.</li> </ul> <p data-bbox="512 1081 1898 1144"><a href="https://developers.google.com/android/reference/com/google/android/gms/location/FusedLocationProviderClient">https://developers.google.com/android/reference/com/google/android/gms/location/FusedLocationProviderClient</a></p>	<code>request</code>	The location request for the updates.	<code>callbackIntent</code>	A pending intent to be sent for each location update.
<code>request</code>	The location request for the updates.				
<code>callbackIntent</code>	A pending intent to be sent for each location update.				

**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC						
	<p data-bbox="533 245 1738 277"><code>public void onLocationAvailability (LocationAvailability locationAvailability)</code></p> <p data-bbox="533 310 1171 334">Called when there is a change in the availability of location data.</p> <p data-bbox="533 367 1738 561">When <code>isLocationAvailable()</code> returns <code>false</code> you can assume that location will not be returned in <code>onLocationResult(LocationResult)</code> until something changes in the device's settings or environment. Even when <code>isLocationAvailable()</code> returns <code>true</code> the <code>onLocationResult(LocationResult)</code> may not always be called regularly, however the device location is known and both the most recently delivered location and <code>getLastLocation(GoogleApiClient)</code> will be reasonably up to date given the hints specified by the active <code>LocationRequest</code>s.</p> <p data-bbox="548 586 667 610"><b>Parameters</b></p> <table border="1" data-bbox="533 643 1738 708"> <tr> <td data-bbox="533 651 961 708"><code>locationAvailability</code></td> <td data-bbox="968 651 1738 708">The current status of location availability.</td> </tr> </table> <p data-bbox="533 756 1738 789"><code>public void onLocationResult (LocationResult result)</code></p> <p data-bbox="533 821 1052 846">Called when device location information is available.</p> <p data-bbox="533 878 1661 943">The most recent location returned by <code>getLastLocation()</code> is not guaranteed to be immediately fresh, but will be reasonably up to date given the hints specified by the active <code>LocationRequest</code>s.</p> <p data-bbox="548 967 667 992"><b>Parameters</b></p> <table border="1" data-bbox="533 1024 1738 1089"> <tr> <td data-bbox="533 1032 772 1089"><code>result</code></td> <td data-bbox="779 1032 1738 1089">The latest location result available.</td> </tr> </table> <p data-bbox="512 1105 1801 1130"><a href="https://developers.google.com/android/reference/com/google/android/gms/location/LocationCallback">https://developers.google.com/android/reference/com/google/android/gms/location/LocationCallback</a></p> <p data-bbox="533 1146 1738 1179"><code>public abstract void onLocationChanged (Location location)</code></p> <p data-bbox="533 1211 915 1235">Called when the location has changed.</p> <p data-bbox="548 1260 667 1284"><b>Parameters</b></p> <table border="1" data-bbox="533 1317 1738 1382"> <tr> <td data-bbox="533 1325 926 1382"><code>location</code></td> <td data-bbox="932 1325 1738 1382">The updated location.</td> </tr> </table>	<code>locationAvailability</code>	The current status of location availability.	<code>result</code>	The latest location result available.	<code>location</code>	The updated location.
<code>locationAvailability</code>	The current status of location availability.						
<code>result</code>	The latest location result available.						
<code>location</code>	The updated location.						

**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC		
	<p><a href="https://developers.google.com/android/reference/com/google/android/gms/location/LocationListener">https://developers.google.com/android/reference/com/google/android/gms/location/LocationListener</a></p> <p>Public Constructors</p> <hr/> <p>public <b>MapView</b> (<a href="#">Context</a> context)</p> <hr/> <p>public <b>MapView</b> (<a href="#">Context</a> context, <a href="#">AttributeSet</a> attrs)</p> <hr/> <p>public <b>MapView</b> (<a href="#">Context</a> context, <a href="#">AttributeSet</a> attrs, int defStyleAttr)</p> <hr/> <p>public <b>MapView</b> (<a href="#">Context</a> context, <a href="#">GoogleMapOptions</a> options)</p> <p><a href="https://developers.google.com/android/reference/com/google/android/gms/maps/MapView">https://developers.google.com/android/reference/com/google/android/gms/maps/MapView</a></p> <p>public void <b>getMapAsync</b> (<a href="#">OnMapReadyCallback</a> callback)</p> <p>Returns a non-null instance of the <a href="#">GoogleMap</a> , ready to be used.</p> <p>Note that:</p> <ul style="list-style-type: none"> <li>• This method must be called from the main thread.</li> <li>• The callback will be executed in the main thread.</li> <li>• In the case where Google Play services is not installed on the user's device, the callback will not be triggered until the user installs it.</li> <li>• The <a href="#">GoogleMap</a> object provided by the callback is non-null.</li> </ul> <p><b>Parameters</b></p> <table border="1"> <tr> <td data-bbox="527 1203 688 1263">callback</td> <td data-bbox="695 1203 1738 1263">The callback object that will be triggered when the map is ready to be used.</td> </tr> </table> <hr/> <p>public final void <b>onCreate</b> (<a href="#">Bundle</a> savedInstanceState)</p> <p>You must call this method from the parent Activity/Fragment's corresponding method.</p>	callback	The callback object that will be triggered when the map is ready to be used.
callback	The callback object that will be triggered when the map is ready to be used.		

**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC
<p>[54D] transmitting IP-based messages including a location of the first device to the respective second devices;</p>	<p><a href="https://developers.google.com/android/reference/com/google/android/gms/maps/MapView">https://developers.google.com/android/reference/com/google/android/gms/maps/MapView</a></p> <p>HTC infringes directly and/or indirectly by performing, inducing others to perform, and/or contributing to the performance of transmitting IP-based messages including a location of the first device to the respective second devices. See claims 1[D], 28[D], and 41[D], which are incorporated herein by reference in their entirety.</p> <p>For example, users send their location to a server and receive the location of other devices with whom the location is being shared. To send a location to the network, a user enables location service which enables the device to determine and send its location. If location service is already enabled, the device sends its location to the server as needed by the application (e.g. Google Maps). If location service is not enabled, the application will ask the user to enable location service in order to continue with full functionality, which includes using the device's location. Google Maps applications receive the location of other devices when those devices have location service enabled while using the same respective application. Android Device Manager and Google Maps use the received locations to display those locations on the map, indicating the locations of other devices.</p> <p>See, e.g., location sharing including corresponding code described above with regard to limitation [1C].</p> <p>Using Google Maps, a user enables location services to send its location the network, but the user can also choose to share its location, as shown below. Again, each device that participates is able to see the location of the other device using Google Maps' share your location feature. For example:</p>

**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC
	<p data-bbox="527 233 1304 277"><b>HTC One V™ – Google Location Service &amp; GPS</b></p> <p data-bbox="527 321 1717 428">Google Maps lets you track your current location, view real-time traffic situations, and receive detailed directions to your destination. It also provides a search tool where you can locate a place of interest or an address on a vector or aerial map, or view locations in street level.</p> <p data-bbox="527 472 898 500"><b>Turning on Location Services</b></p> <div data-bbox="527 516 1717 954"> </div> <ol data-bbox="527 980 1717 1154" style="list-style-type: none"> <li>1. From the Home Screen, slide the <b>Notifications</b> panel open.</li> <li>2. In the top right corner, tap <b>Settings</b>.</li> <li>3. Tap <b>Location</b>.</li> <li>4. Make your selection by tapping <b>Google's location service</b>, <b>Use GPS satellites</b>, or both. <b>Note:</b> You will need to accept the location consent terms and conditions.</li> </ol>



# Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products

US9408055B2	HTC
	<ol style="list-style-type: none"><li>1. If you haven't already, add their Gmail address to your <a href="#">Google Contacts</a>.</li><li>2. On your Android phone or tablet, open the Google Maps app and sign in. <a href="#">Learn how to sign in.</a></li><li>3. Tap the Menu  &gt; <b>Share location</b> &gt; Add People .</li><li>4. Choose how long you want to share your location.</li><li>5. Tap <b>Select People</b>.<ul style="list-style-type: none"><li>▪ If you're asked about your contacts, give Google Maps access.</li></ul></li><li>6. Choose who you want to share with.</li><li>7. Tap <b>Share</b>.</li></ol> <p><a href="https://support.google.com/plus/answer/3302509?co=GENIE.Platform%3DAndroid&amp;hl=en">https://support.google.com/plus/answer/3302509?co=GENIE.Platform%3DAndroid&amp;hl=en</a></p>  <p><a href="https://arstechnica.com/gadgets/2017/03/location-sharing-finally-returns-to-google-maps/">https://arstechnica.com/gadgets/2017/03/location-sharing-finally-returns-to-google-maps/</a></p>

**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC
	 <p><a href="https://arstechnica.com/gadgets/2017/03/location-sharing-finally-returns-to-google-maps/">https://arstechnica.com/gadgets/2017/03/location-sharing-finally-returns-to-google-maps/</a></p> <p>Below are exemplary methods used by Google applications to obtain, send, and receive locations.</p>

**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC
	<p data-bbox="527 233 982 261">The Google Play services Location API</p> <p data-bbox="527 298 1356 347">The Google Play services <a href="#">Location API</a> is the preferred method for adding location awareness to your Android application. It includes functionality that lets you:</p> <ul data-bbox="552 371 1184 505" style="list-style-type: none"> <li>• Determine the device location.</li> <li>• Listen for location changes.</li> <li>• Determine the mode of transportation, if the device is moving.</li> <li>• Create and monitor predefined geographical regions, known as geofences.</li> </ul> <p data-bbox="527 531 1360 662">The location APIs make it easy for you to build power efficient, location-aware applications. Like the Google Maps Android API, the Location API is distributed as part of the Google Play services SDK. For more information on the Location API, please refer to the Android training class <a href="#">Making Your App Location Aware</a> or the <a href="#">Location API Reference</a>. Code examples are included as part of the Google Play services SDK.</p> <p data-bbox="527 667 1331 695"><a href="https://developers.google.com/maps/documentation/android-api/location">https://developers.google.com/maps/documentation/android-api/location</a></p> <div data-bbox="527 776 1339 1177"> <p>The diagram illustrates the architecture of the Google API Client. On the left, a box labeled 'Device' contains 'Your app' (represented by an Android robot icon) and the 'Google Play services library'. Inside the library is the 'Google API Client'. On the right, a dashed box labeled 'Google Play services' contains three service boxes: 'Games service', 'Drive service', and 'Other services'. Bidirectional arrows connect 'Your app' to the 'Google API Client', and the 'Google API Client' to each of the three service boxes.</p> </div> <p data-bbox="527 1193 1331 1268">Figure 1: An illustration showing how the Google API Client provides an interface for connecting and making calls to any of the available Google Play services such as Google Play Games and Google Drive.</p> <p data-bbox="527 1273 1247 1300"><a href="https://developers.google.com/android/guides/api-client#Starting">https://developers.google.com/android/guides/api-client#Starting</a></p>

**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC
	<h3 data-bbox="527 233 919 261">Get the Last Known Location</h3> <p data-bbox="527 289 1545 386">Once you have connected to Google Play services and the location services API, you can get the last known location of a user's device. When your app is connected to these you can use the fused location provider's <code>getLastLocation()</code> method to retrieve the device location. The precision of the location returned by this call is determined by the permission setting you put in your app manifest, as described in the <a href="#">Specify App Permissions</a> section of this document.</p> <p data-bbox="527 407 1545 480">To request the last known location, call the <code>getLastLocation()</code> method, passing it your instance of the <code>GoogleApiClient</code> object. Do this in the <code>onConnected()</code> callback provided by Google API Client, which is called when the client is ready. The following code snippet illustrates the request and a simple handling of the response:</p> <pre data-bbox="527 493 1545 781">public class MainActivity extends ActionBarActivity implements     ConnectionCallbacks, OnConnectionFailedListener {     ...     @Override     public void onConnected(Bundle connectionHint) {         mLastLocation = LocationServices.FusedLocationApi.getLastLocation(             mGoogleApiClient);         if (mLastLocation != null) {             mLatitudeText.setText(String.valueOf(mLastLocation.getLatitude()));             mLongitudeText.setText(String.valueOf(mLastLocation.getLongitude()));         }     } }</pre> <p data-bbox="527 797 1545 842">The <code>getLastLocation()</code> method returns a <code>Location</code> object from which you can retrieve the latitude and longitude coordinates of a geographic location. The location object returned may be null in rare cases when the location is not available.</p> <p data-bbox="527 849 1289 878"><a href="https://developer.android.com/training/location/retrieve-current.html">https://developer.android.com/training/location/retrieve-current.html</a></p>

**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC
	<p>Determining the user's current location</p> <hr/> <p>The Geolocation API offers a simple, "one-shot" method to obtain the user's location: <code>getCurrentPosition()</code>. A call to this method asynchronously reports on the user's current location.</p> <pre> window.onload = function() {   var startPos;   var geoSuccess = function(position) {     startPos = position;     document.getElementById('startLat').innerHTML = startPos.coords.latitude;     document.getElementById('startLon').innerHTML = startPos.coords.longitude;   };   navigator.geolocation.getCurrentPosition(geoSuccess); }; </pre> <p>If this is the first time that an application on this domain has requested permissions, the browser typically checks for user consent. Depending on the browser, there may also be preferences to always allow—or disallow—permission lookups, in which case the confirmation process is bypassed.</p> <p>Depending on the location device your browser is using, the position object might actually contain a lot more than just latitude and longitude; for example, it might include an altitude or a direction. You can't tell what extra information that location system uses until it actually returns the data.</p> <p><a href="https://developers.google.com/web/fundamentals/native-hardware/user-location/">https://developers.google.com/web/fundamentals/native-hardware/user-location/</a></p>
<p>[54E] transmitting an IP-based text message to at least one of the second devices via a cellular communications network;</p>	<p>HTC infringes directly and/or indirectly by performing, inducing others to perform, and/or contributing to the performance of transmitting an IP-based text message to at least one of the second devices via a cellular communications network. See claims 1[A], 8, 12, 28[A], 41[A], and 54[A], which are incorporated herein by reference in their entirety.</p> <p>For example, the Accused products run Android Messages which access contact information for second users using respective second devices.</p> <p>Upon information and belief, the Accused Products are forms of cellular devices or PDAs in that the functionality of a PDA has been subsumed into smartphones, tablets, and portable media players having the functionalities of a PDA that include cellular transceivers to enable cellular communications. To the extent that it is necessary, AGIS submits that the Accused Products meet the claim limitation “the first device is a</p>

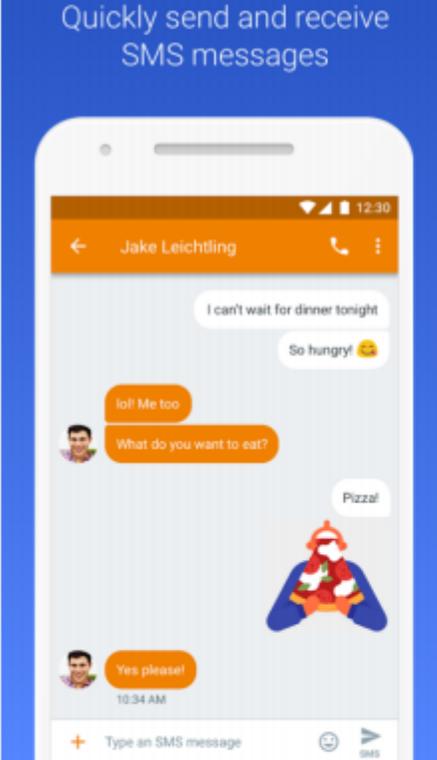
**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC
	<p>personal digital assistant (PDA) or a cellular phone” under the doctrine of equivalents. For example, U.S. Cellular which is one of the largest mobile cellular service providers in the United States and categorizes HTC Android phones as smartphones for which U.S. Cellular provides cellular services.</p> <p><a href="https://www.uscellular.com/uscellular/cell-phones/showPhoneDetails.jsp?productId=prod1060051">https://www.uscellular.com/uscellular/cell-phones/showPhoneDetails.jsp?productId=prod1060051</a></p> <p><b>Send &amp; receive text messages in Android Messages</b></p> <p>You can send and receive text messages with friends and contacts on Android Messages.</p> <p><b>Start a conversation</b></p> <ol style="list-style-type: none"> <li>1. Open the Android Messages app .</li> <li>2. Tap Compose .</li> <li>3. In "To," enter the names, phone numbers, or email addresses that you'd like to message. You can also pick from your top contacts or your whole contact list.</li> <li>4. Tap Next .</li> </ol> <p><a href="https://support.google.com/nexus/answer/6080324?hl=en&amp;ref_topic=6080329">https://support.google.com/nexus/answer/6080324?hl=en&amp;ref_topic=6080329</a></p> <p>For example, the Accused Products are known to use IP-based communication over wireless or data connections. Both Android Messages and Google Hangouts utilize SMS messages, including group messages from one device to several devices, to send an SMS message, with additional information, to a contact. U.S. Cellular which is one of the largest mobile cellular service providers in the United States and categorizes HTC Android phones as smartphones for which U.S. Cellular provides cellular communications services to enable the IP-based communications over its cellular network.</p> <p><a href="https://www.uscellular.com/uscellular/cell-phones/showPhoneDetails.jsp?productId=prod1060051">https://www.uscellular.com/uscellular/cell-phones/showPhoneDetails.jsp?productId=prod1060051</a></p>

**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC
	<p>Android Messages makes it easy to communicate with anyone by using SMS, MMS, and more. Stay in touch with friends and family, send group texts, and share your favorite pictures, videos, audio messages.</p> <p>· <b>Enhanced features:</b> On supported carriers, you can send messages over Wi-Fi or your data network, see when friends have read your message, and more.</p> <p><a href="https://play.google.com/store/apps/details?id=com.google.android.apps.messaging&amp;hl=en">https://play.google.com/store/apps/details?id=com.google.android.apps.messaging&amp;hl=en</a></p>

**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC
	 <p><a href="https://play.google.com/store/apps/details?id=com.google.android.apps.messaging&amp;hl=en">https://play.google.com/store/apps/details?id=com.google.android.apps.messaging&amp;hl=en</a></p> <h3>Get started with Hangouts</h3> <p>You can use Hangouts to:</p> <ul style="list-style-type: none"><li>• Start a chat conversation or video call.</li><li>• Make phone calls using Wi-Fi or data.</li><li>• Send text messages with your <a href="#">Google Voice</a> or <a href="#">Project Fi</a> phone number.</li></ul> <p>Hangouts sync automatically across devices. If you start a Hangout on your computer, you can continue your chat on another device, like your phone.</p> <p><a href="https://support.google.com/hangouts/answer/2944865?hl=en&amp;ref_topic=6386410">https://support.google.com/hangouts/answer/2944865?hl=en&amp;ref_topic=6386410</a></p>



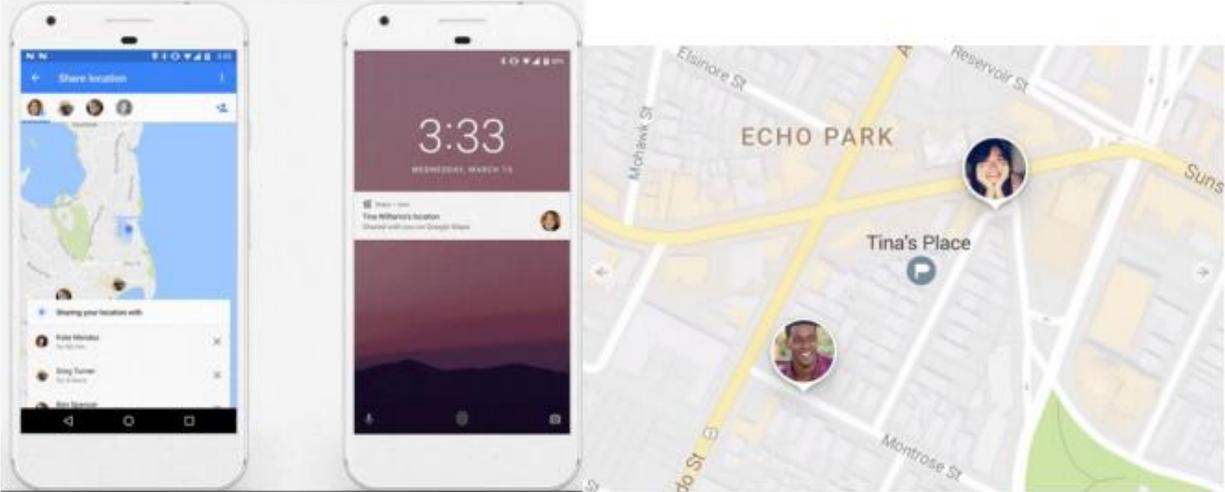
**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC
<p>[54F] presenting, via an interactive display of the first device, an interactive map and a plurality of user selectable symbols corresponding to the plurality of second devices, wherein the symbols are positioned on the map at respective positions corresponding to the respective locations of the second devices;</p>	<p>HTC infringes directly and/or indirectly by performing, inducing others to perform, and/or contributing to the performance of presenting, via an interactive display of the first device, an interactive map and a plurality of user selectable symbols corresponding to the plurality of second devices, wherein the symbols are positioned on the map at respective positions corresponding to the respective locations of the second devices. See claims 1[E], 28[E], and 41[E], which are incorporated herein by reference in their entirety.</p> <p>For example, the Accused Products use Android Device Manager, and Google Maps to display an interface with a map and symbols representing devices.</p> <p>Using Android Device Manager, the user is presented with a map that appears to be based on or imported from Google Maps. The map is interactive because the user can pan, zoom, and make selections, among other interactivities. Depending on the number of devices linked to the Google Account, Android Device Manager places symbols on the map and in a list, where those symbols represent the linked devices and their positions. The symbols are selectable by touching the display. The map is georeferenced, at least because its contents, including the symbols, are placed and spaced according to geographical coordinates.</p> <p><b><u>Regarding Google Maps</u></b>, Google Latitude, Google Plus, Google Hangouts, Google Messages, Google Allo, Google Duo, Google Chrome, and Android Messenger, the Accused Products display, to the user on the display of the first device, a map with one or more symbols corresponding to one or more second users (or second devices corresponding to the second users). The map is interactive because the user may control the display of the map, e.g., pan, zoom, and/or effect change to the map in an otherwise interactive manner. The map is georeferenced for at least the reason that one or more symbols are associated with spatial locations, i.e., coordinates. The symbols are user-selectable because a user may touch the display to select the user or device associated with the symbol. For example, the user may input a touch selection directly on or near the portion of the display corresponding to the symbol's coordinates on the map to effect a selection of the user or device.</p> <p><b><u>Exemplary Support for Google Maps:</u></b></p> <p>Using Google Maps and its location sharing feature, the user is presented with a map that is interactive because the user can pan, zoom, and make selections, among other interactivities. Depending on how many other devices or Google Accounts are sharing their locations, Google Maps places symbols on the map and</p>

**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC
	<p>in a list, where those symbols represent the linked devices and their positions. The symbols are selectable by touching the display. The map is georeferenced, at least because its contents, including the symbols, are placed and spaced according to geographical coordinates.</p> <p><b>Location settings</b></p> <hr/> <p><b>Turning location services on or off</b></p> <hr/> <p>In order to find your location on HTC One, you need to enable location sources.</p> <ol style="list-style-type: none"> <li>1. Go to Settings, and then tap <b>Location</b>.</li> <li>2. Tap the <b>On/Off</b> switch to turn location services on and off.</li> <li>3. Under Location sources, select the location mode you want. For example, for a better estimate of your location, choose <b>High accuracy</b>. To save battery power, choose <b>Battery saving</b>.</li> </ol> <p>Turning off a location source (for example GPS) means no applications on HTC One will collect your location data through that location source. However, third party applications may collect — and HTC One may continue to provide — location data through other sources, including through Wi-Fi and signal triangulation.</p> <p><b>Google Maps</b></p> <hr/> <p><b>About Google Maps</b></p> <hr/> <p>Google Maps™ lets you track your current location, view real-time traffic situations, and receive detailed directions to your destination.</p> <p>It also provides a search tool where you can locate places of interest or an address on a map, or view locations at street level.</p>

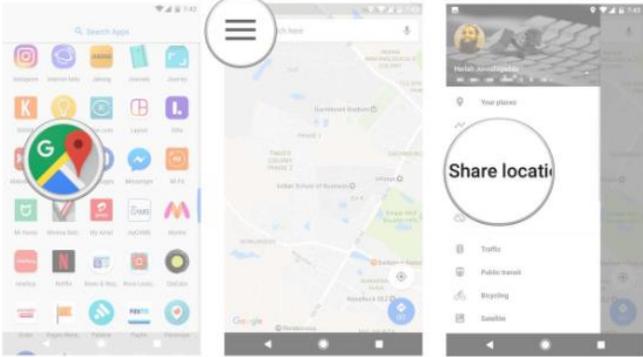
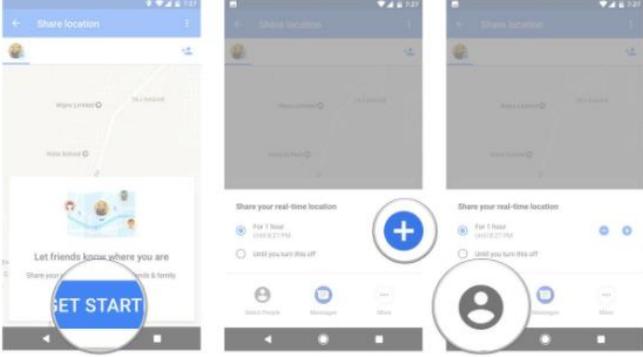
# Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products

US9408055B2	HTC
	 <p><a href="https://arstechnica.com/gadgets/2017/03/location-sharing-finally-returns-to-google-maps/">https://arstechnica.com/gadgets/2017/03/location-sharing-finally-returns-to-google-maps/</a></p> <h3>See where someone is</h3> <p>If someone shares their location with you, you can see them on the map.</p> <ol style="list-style-type: none"><li>1. Open the Google Maps app</li><li>2. Tap the Menu ≡ &gt; <b>Share location</b>.</li><li>3. Choose someone.</li></ol> <p>• To see an updated location, tap on a friend's icon &gt; More ≡ &gt; <b>Refresh</b>.</p> <h3>Stop seeing someone's location</h3> <ol style="list-style-type: none"><li>1. Open the Google Maps app</li><li>2. On the map, tap their icon.</li><li>3. At the bottom, tap More ≡</li><li>4. To temporarily hide someone, tap <b>Hide from map</b>. You can stop hiding them at any time.</li></ol> <p><b>Note:</b> You can stop someone's location from ever appearing on your map. <a href="#">Learn how to block another person's account.</a></p> <p><a href="https://support.google.com/maps/answer/7326816?hl=en&amp;ref_topic=3092425&amp;co=GENIE.Platform%3DAndroid&amp;oc=1">https://support.google.com/maps/answer/7326816?hl=en&amp;ref_topic=3092425&amp;co=GENIE.Platform%3DAndroid&amp;oc=1</a></p>

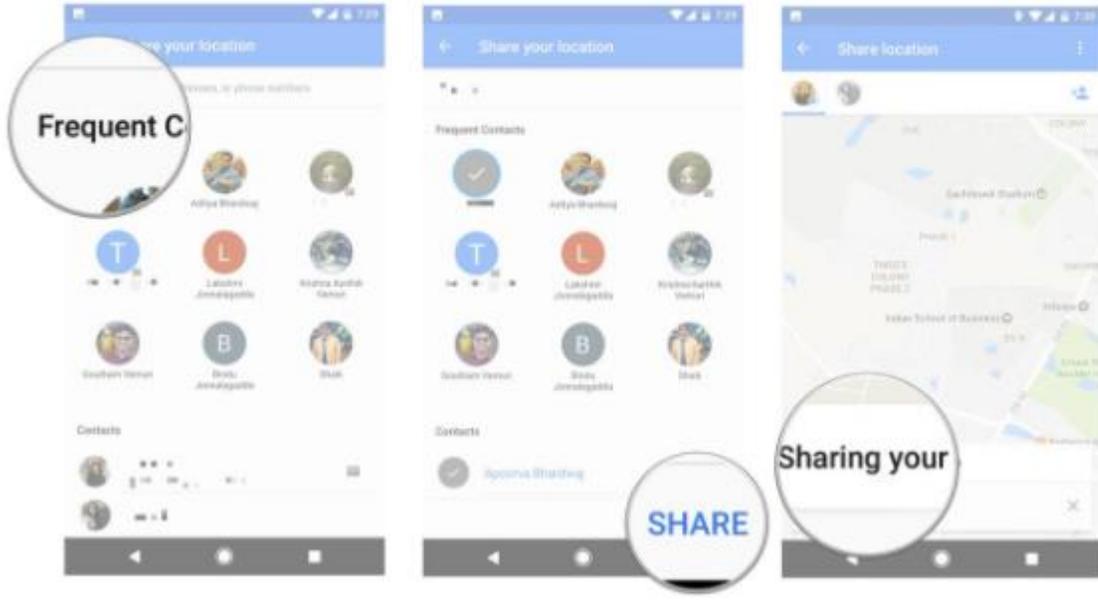
## Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products

US9408055B2	HTC
	<h3>Share your E.T.A</h3> <p>After you start your drive, you can share your destination, estimated arrival time, and where you are on the route.</p> <ol style="list-style-type: none"><li>1. Open the Google Maps app .</li><li>2. Set a driving destination. <a href="#">Learn how to navigate to a place.</a></li><li>3. After you start navigation, tap More ^ &gt; <b>Share trip progress.</b></li><li>4. Choose a person from the list.</li><li>5. Tap <b>Share.</b></li><li>6. Location Sharing will stop when you reach your destination or stop navigating.</li></ol> <ul style="list-style-type: none"><li>• To stop sharing before you arrive, tap More ^ &gt; <b>Stop sharing.</b></li></ul> <h3>See where someone is</h3> <p>If someone shares their location with you, you can see them on the map.</p> <ol style="list-style-type: none"><li>1. Open the Google Maps app .</li><li>2. Tap Menu ≡ &gt; <b>Location sharing.</b></li><li>3. Choose someone.</li></ol> <ul style="list-style-type: none"><li>• To see an updated location, tap on a friend's icon &gt; More ⋮ &gt; <b>Refresh.</b></li></ul> <h3>Stop seeing someone's location</h3> <ol style="list-style-type: none"><li>1. Open the Google Maps app .</li><li>2. On the map, tap their icon.</li><li>3. At the bottom, tap More ^ .</li><li>4. To temporarily hide someone, tap <b>Hide from map.</b> You can stop hiding them at any time.</li></ol> <p><b>Note:</b> You can stop someone's location from ever appearing on your map. Learn how to <a href="#">block another person's account.</a></p> <p><u><a href="https://support.google.com/maps/answer/7326816?co=GENIE.Platform%3DAndroid&amp;oco=1">https://support.google.com/maps/answer/7326816?co=GENIE.Platform%3DAndroid&amp;oco=1</a></u></p>

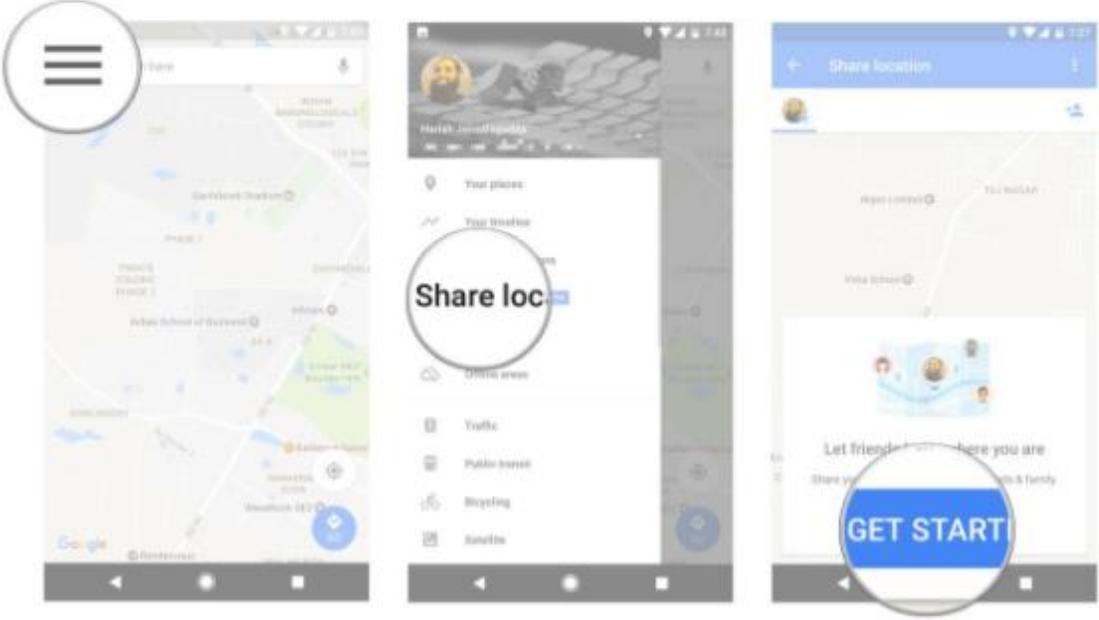
# Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products

US9408055B2	HTC
	<p data-bbox="520 240 1129 267"><b>How to share your location in Google Maps</b></p> <ol data-bbox="520 298 1129 381" style="list-style-type: none"><li>1. Open Google Maps from the app drawer or the home screen.</li><li>2. Tap the <b>hamburger menu</b> (the three horizontal lines) on the top left corner of the screen.</li><li>3. Select <b>Share location</b>.</li></ol>  <ol data-bbox="520 808 1129 907" style="list-style-type: none"><li>4. Tap <b>Get Started</b>.</li><li>5. Use the <b>+</b> icon to select a time period or select the <b>Until you turn this off</b> setting to share your location indefinitely.</li><li>6. Tap <b>Select People</b>.</li></ol>  <p data-bbox="520 1305 1409 1333"><a href="https://www.androidcentral.com/how-share-location-google-maps">https://www.androidcentral.com/how-share-location-google-maps</a></p>

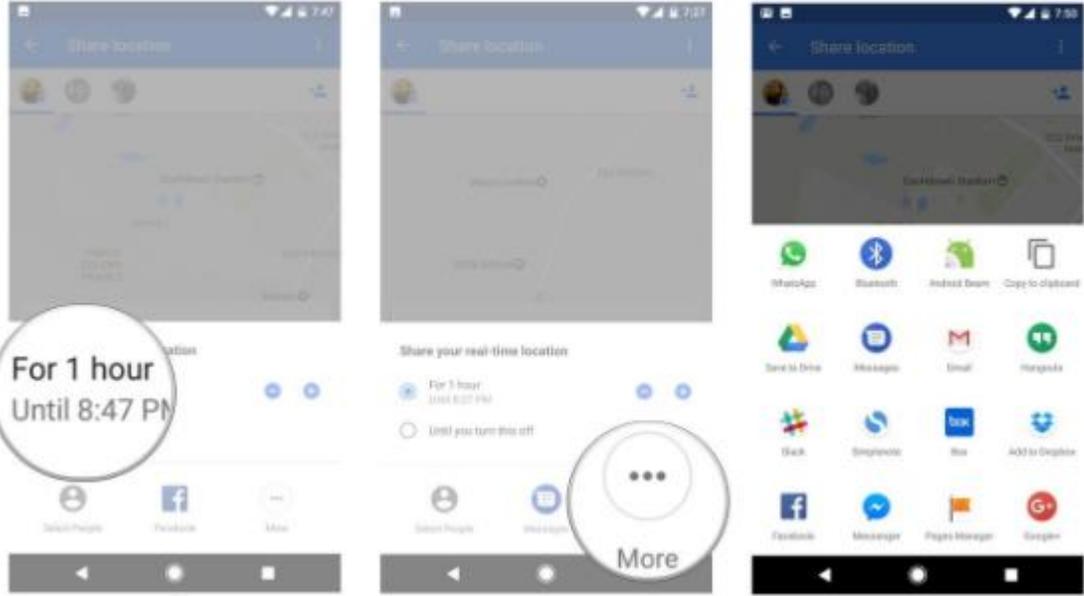
**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC
	<p data-bbox="527 250 1577 310">7. You'll see a list of your frequent contacts at the top, along with a full list of contacts. Pick the contacts by tapping their name.</p> <p data-bbox="527 337 1457 367">8. Once you've selected the contacts you want to share your location to, tap <b>Share</b>.</p> <p data-bbox="527 394 1419 423">9. You'll see a message saying that the selected contact can view your location.</p>  <p data-bbox="510 1105 1409 1138"><a href="https://www.androidcentral.com/how-share-location-google-maps">https://www.androidcentral.com/how-share-location-google-maps</a></p>

# Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products

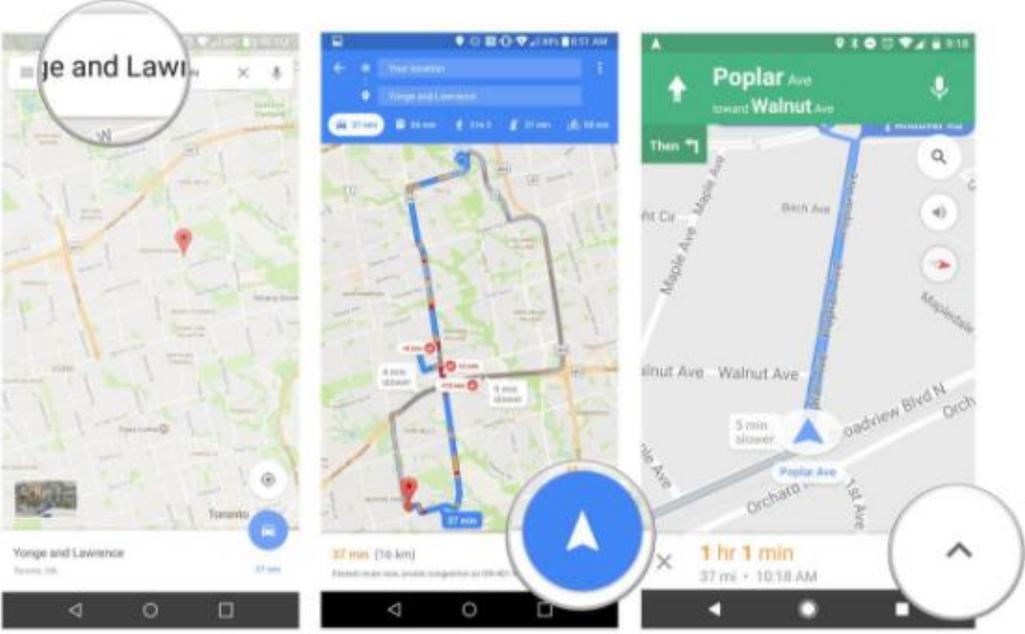
US9408055B2	HTC
	<h2 data-bbox="520 245 1255 293">How to create a shareable link</h2> <p data-bbox="520 334 1461 363">You can also create a link and use it to share your location easily. Here's how to do it:</p> <ol data-bbox="520 412 1234 553" style="list-style-type: none"><li data-bbox="520 412 1234 441">1. Tap the hamburger menu on the top left corner of the screen.</li><li data-bbox="520 469 800 498">2. Select Share location.</li><li data-bbox="520 526 737 555">3. Tap Get Started.</li></ol>  <p data-bbox="510 1239 1409 1273"><a href="https://www.androidcentral.com/how-share-location-google-maps">https://www.androidcentral.com/how-share-location-google-maps</a></p>

**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

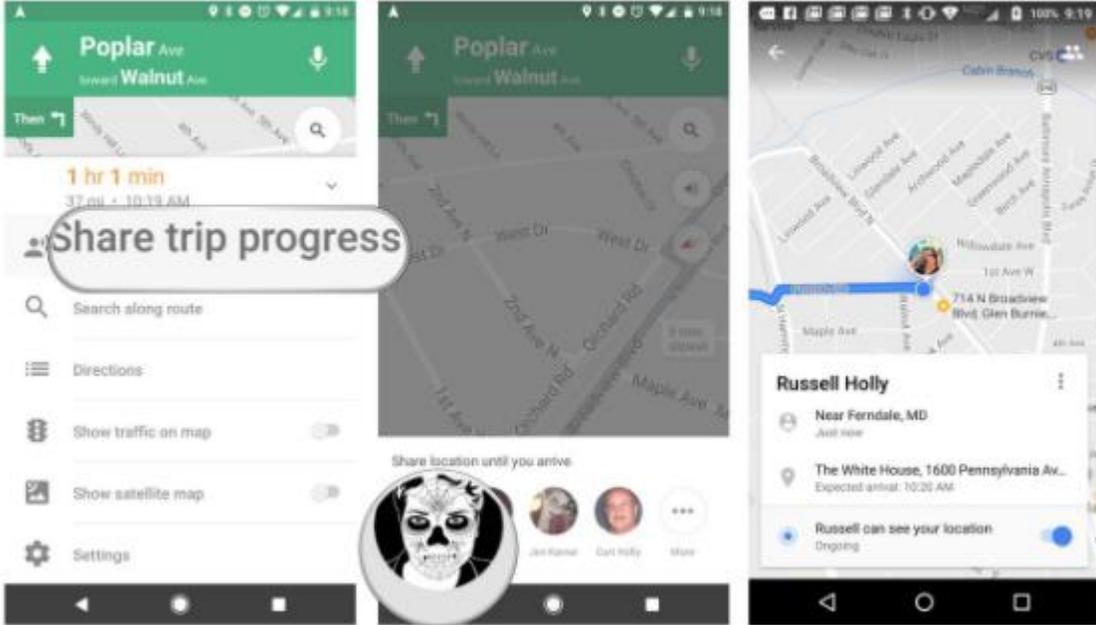
US9408055B2	HTC
	<p data-bbox="520 245 1213 272">4. Select the <b>amount of time</b> you want to share your location.</p> <p data-bbox="520 302 680 329">5. Tap <b>More</b>.</p> <p data-bbox="520 358 1633 418">6. Select <b>your app of choice</b> to create and send a <b>unique URL</b> that broadcasts your current location. You can email it, send the link via Messenger, or even tweet it to the intended recipient.</p>  <p data-bbox="510 1089 1409 1122"><a href="https://www.androidcentral.com/how-share-location-google-maps">https://www.androidcentral.com/how-share-location-google-maps</a></p>



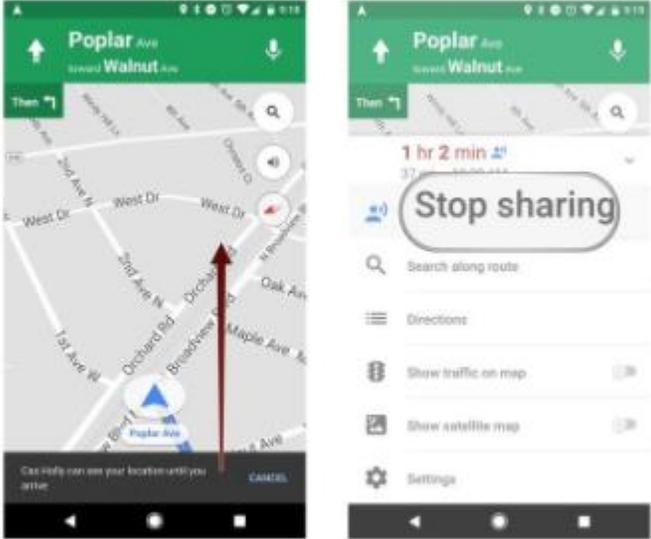
**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC
	<p data-bbox="527 240 1428 337"><b>How to share your navigation directions while you walk, drive or transit</b></p> <p data-bbox="527 375 1556 464">One of the best ways to share your location with someone is during a drive, walk or transit. If you're meeting a friend or family member somewhere, or navigating towards their home, Google Maps lets you share your location with them for the duration of the trip. It's magic!</p> <ol data-bbox="527 513 1396 646" style="list-style-type: none"><li>1. In the <b>search bar</b> enter your destination.</li><li>2. Pick your navigation type (drive, transit, walk) and press the blue <b>navigate</b> button.</li><li>3. Tap the arrow next to the <b>time-to-destination</b> number at the bottom of the screen.</li></ol>  <p data-bbox="512 1333 1409 1365"><a href="https://www.androidcentral.com/how-share-location-google-maps">https://www.androidcentral.com/how-share-location-google-maps</a></p>

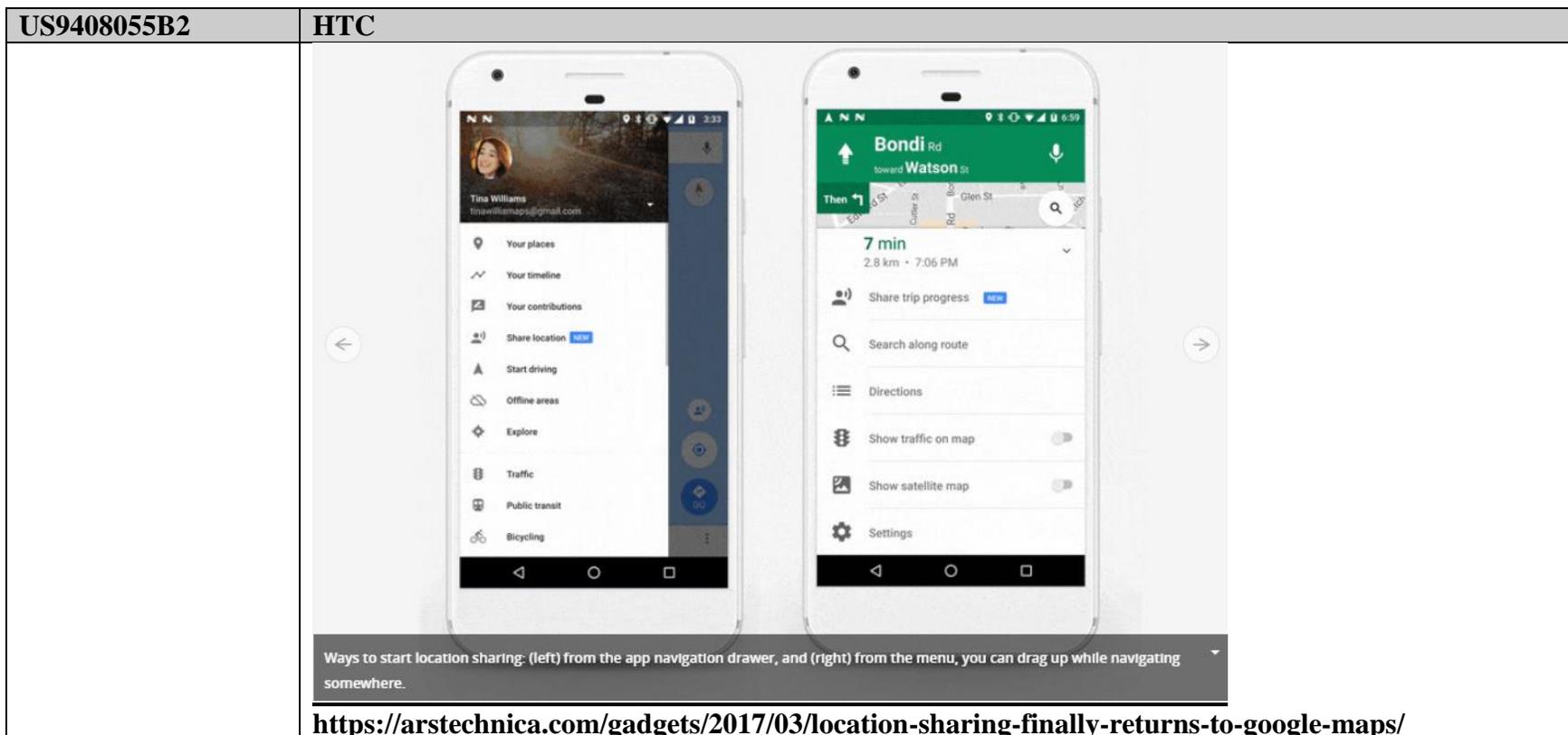
**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC
	<p data-bbox="527 240 835 267">4. Tap Share trip progress.</p> <p data-bbox="527 297 1150 324">5. Choose one or more contacts to share trip progress.</p>  <p data-bbox="537 1027 1339 1055">You can also stop sharing your location with someone before a trip ends.</p> <p data-bbox="512 1070 1409 1101"><a href="https://www.androidcentral.com/how-share-location-google-maps">https://www.androidcentral.com/how-share-location-google-maps</a></p>

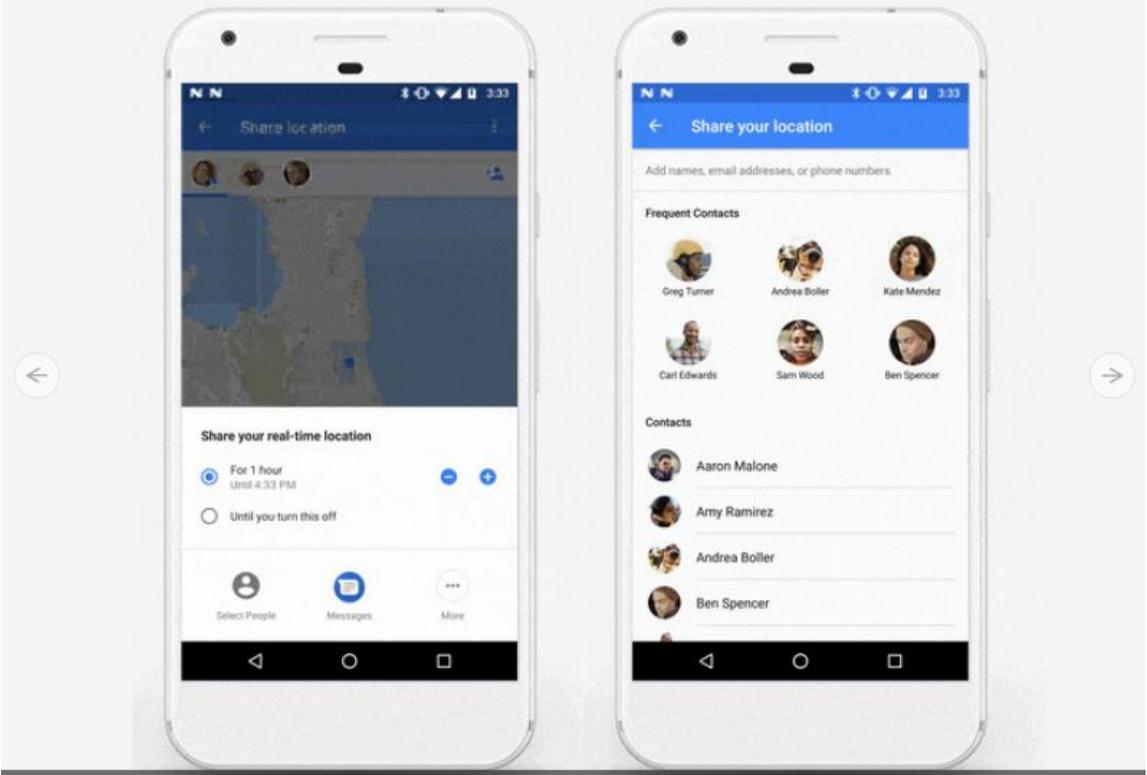
**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC
	<ol style="list-style-type: none"><li data-bbox="533 245 1465 272">1. Tap the arrow next to the <b>time-to-destination</b> number at the bottom of the screen.</li><li data-bbox="533 302 768 329">2. Tap Stop sharing.</li></ol> <div data-bbox="747 383 1398 922"></div> <p data-bbox="541 976 638 1003">That's it!</p> <p data-bbox="541 1045 1608 1073">Are you excited that location sharing is back in Google Maps? How often do you use the feature?</p> <p data-bbox="512 1092 1409 1120"><u><a href="https://www.androidcentral.com/how-share-location-google-maps">https://www.androidcentral.com/how-share-location-google-maps</a></u></p>

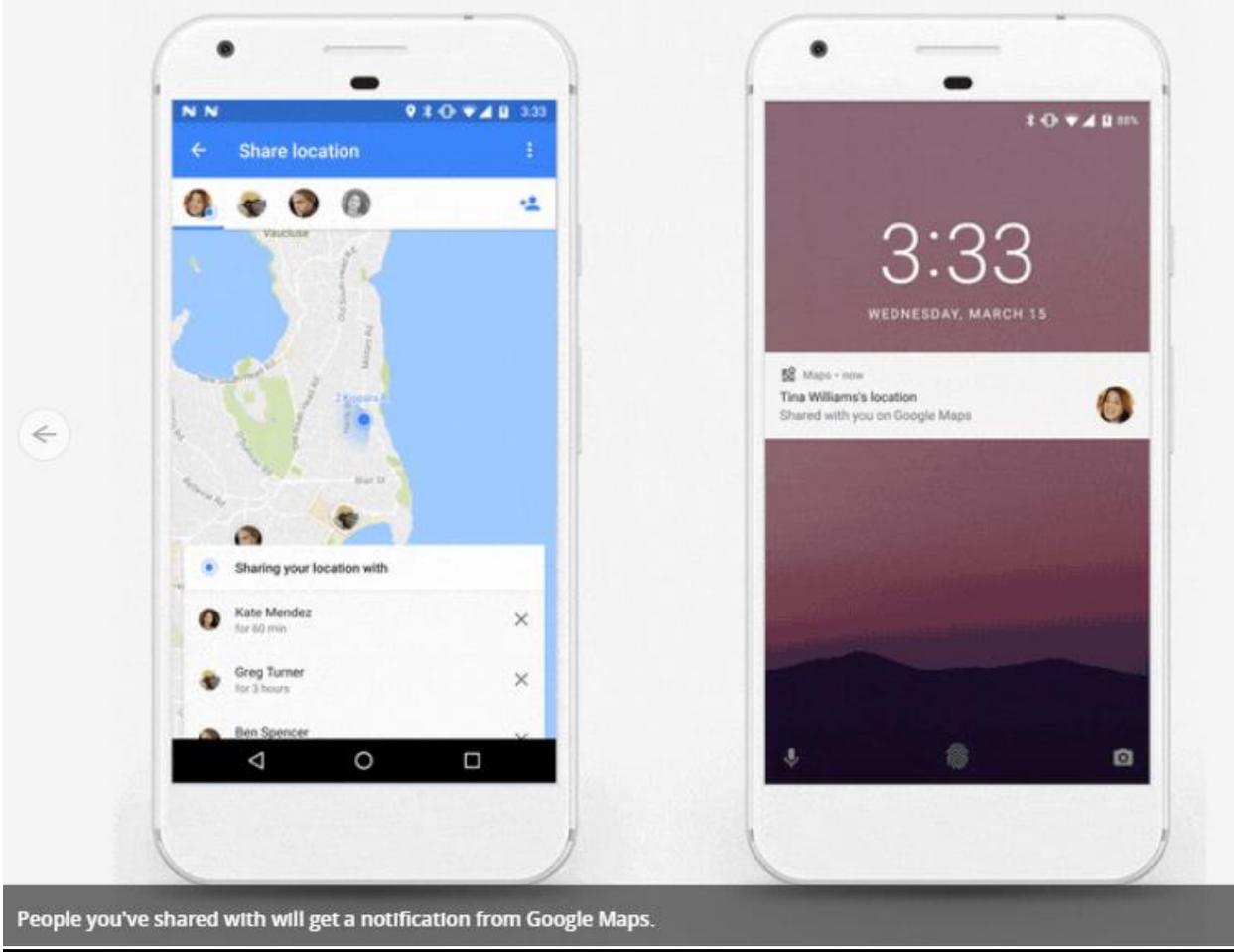
**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**



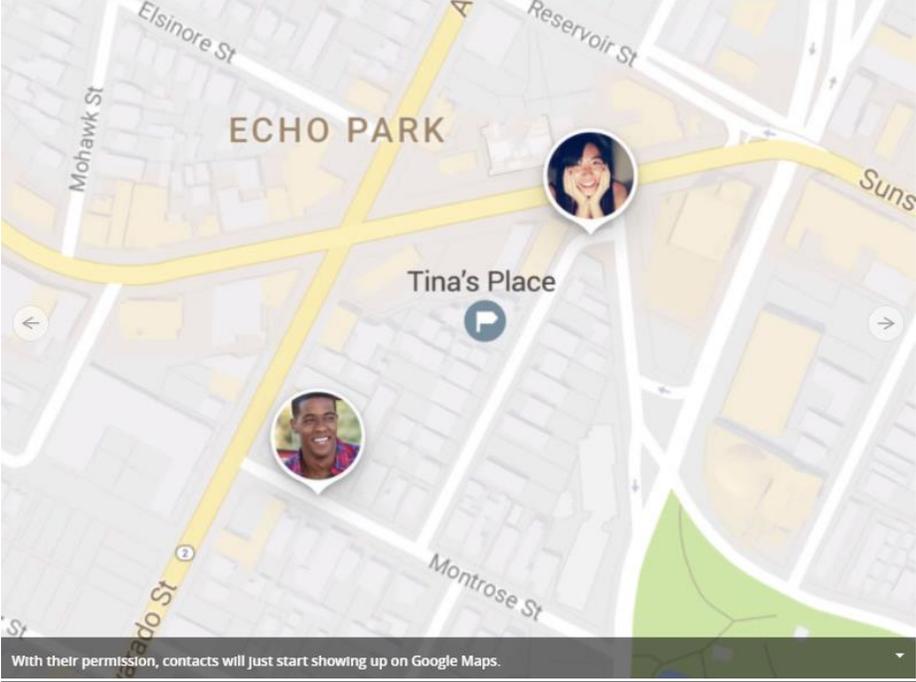
**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC
	 <p data-bbox="520 1024 1654 1063">You can pick how long you want to share your location for, and with whom you want to share it.</p> <p data-bbox="514 1068 1724 1102"><a href="https://arstechnica.com/gadgets/2017/03/location-sharing-finally-returns-to-google-maps/">https://arstechnica.com/gadgets/2017/03/location-sharing-finally-returns-to-google-maps/</a></p>

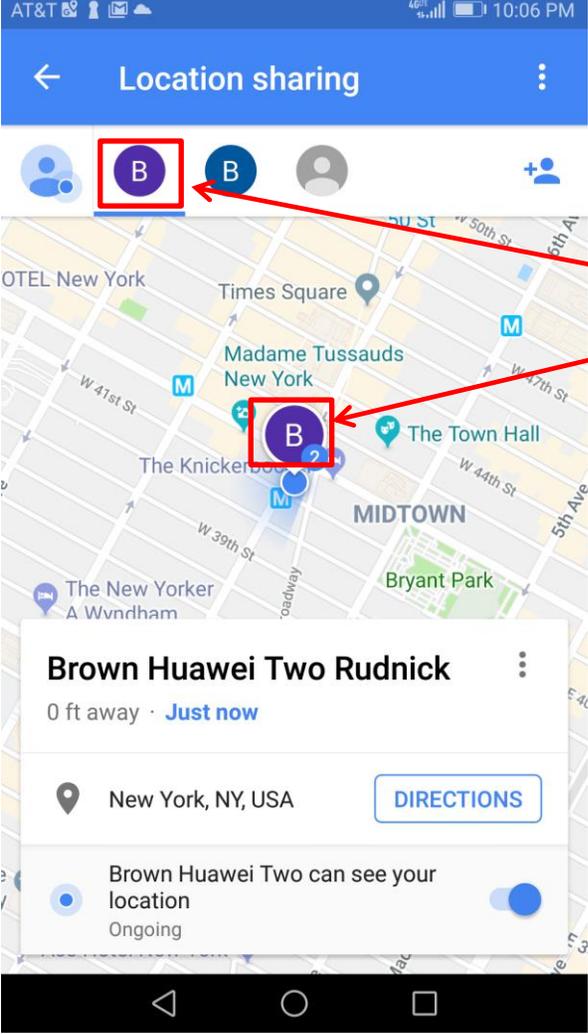
**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC
	 <p data-bbox="512 1143 1176 1170">People you've shared with will get a notification from Google Maps.</p> <p data-bbox="512 1192 1724 1227"><a href="https://arstechnica.com/gadgets/2017/03/location-sharing-finally-returns-to-google-maps/">https://arstechnica.com/gadgets/2017/03/location-sharing-finally-returns-to-google-maps/</a></p>

**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC
	 <p data-bbox="520 883 1016 902">With their permission, contacts will just start showing up on Google Maps.</p> <p data-bbox="514 922 1717 951"><a href="https://arstechnica.com/gadgets/2017/03/location-sharing-finally-returns-to-google-maps/">https://arstechnica.com/gadgets/2017/03/location-sharing-finally-returns-to-google-maps/</a></p> <p data-bbox="514 992 1031 1024"><b><u>Exemplary Google Maps Screenshots:</u></b></p>

**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC
	 <p>The screenshot shows an Android mobile interface for "Location sharing". At the top, there's a blue header with a back arrow, the text "Location sharing", and a menu icon. Below the header is a row of user icons: a group icon, a purple circle with a white 'B' (highlighted with a red square), another purple circle with a white 'B', a grey person icon, and a plus sign. Below this is a map of Midtown Manhattan, New York, with several location pins. One pin, labeled with a purple circle and white 'B', is also highlighted with a red square. A red arrow points from a text box on the right to this pin. Below the map is a card for the selected location: "Brown Huawei Two Rudnick", "0 ft away · Just now", "New York, NY, USA" with a "DIRECTIONS" button, and a toggle switch for "Brown Huawei Two can see your location" which is currently turned on (blue).</p> <p data-bbox="1323 467 1646 639">Exemplary User Selectable Symbols</p> <p data-bbox="514 1356 856 1388"><b><u>Exemplary Source Code:</u></b></p> <p data-bbox="514 1393 1879 1424">The above functionality is performed at least in part by the following publicly available source code and/or</p>



**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC
	<p>source code that invokes or is invoked by the following source code (or a substantially similar copy compiled and loaded onto the Accused Products by HTC). AGIS reserves the right to supplement these contentions with additional source code as discovery progresses and as additional source code is made available.</p> <pre data-bbox="533 428 1738 472">public static LocationRequest create ()</pre> <p>Create a location request with default parameters.</p> <p>Default parameters are for a block accuracy, slowly updated location. It can then be adjusted as required by the applications before passing to the <code>FusedLocationProviderApi</code>.</p> <p><b>Returns</b></p> <ul data-bbox="554 683 814 711" style="list-style-type: none"><li>• a new location request</li></ul> <hr data-bbox="512 722 1759 725"/> <p><a data-bbox="512 732 1871 761" href="https://developers.google.com/android/reference/com/google/android/gms/location/LocationRequest">https://developers.google.com/android/reference/com/google/android/gms/location/LocationRequest</a></p>

**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC
	<p data-bbox="533 250 1745 282">public static final int <b>PRIORITY_BALANCED_POWER_ACCURACY</b></p> <p data-bbox="533 315 1178 342">Used with <code>setPriority(int)</code> to request "block" level accuracy.</p> <p data-bbox="533 375 1644 431">Block level accuracy is considered to be about 100 meter accuracy. Using a coarse accuracy such as this often consumes less power.</p> <p data-bbox="533 456 730 483">Constant Value: 102</p> <p data-bbox="533 540 1745 573">public static final int <b>PRIORITY_HIGH_ACCURACY</b></p> <p data-bbox="533 605 1335 633">Used with <code>setPriority(int)</code> to request the most accurate locations available.</p> <p data-bbox="533 657 961 685">This will return the finest location available.</p> <p data-bbox="533 709 730 737">Constant Value: 100</p> <p data-bbox="533 794 1745 826">public static final int <b>PRIORITY_LOW_POWER</b></p> <p data-bbox="533 859 1157 886">Used with <code>setPriority(int)</code> to request "city" level accuracy.</p> <p data-bbox="533 911 1738 976">City level accuracy is considered to be about 10km accuracy. Using a coarse accuracy such as this often consumes less power.</p> <p data-bbox="533 1000 730 1027">Constant Value: 104</p> <p data-bbox="516 1052 1866 1084"><u><a href="https://developers.google.com/android/reference/com/google/android/gms/location/LocationRequest">https://developers.google.com/android/reference/com/google/android/gms/location/LocationRequest</a></u></p>

**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC
	<pre data-bbox="533 248 1749 285">public Task&lt;Location&gt; getLastLocation ()</pre> <p data-bbox="527 315 1104 337">Returns the best most recent location currently available.</p> <p data-bbox="527 371 1696 430">If a location is not available, which should happen very rarely, null will be returned. The best accuracy available while respecting the location permissions will be returned.</p> <p data-bbox="527 464 1736 522">This method provides a simplified way to get location. It is particularly well suited for applications that do not require an accurate location and that do not want to maintain extra logic for location updates.</p> <pre data-bbox="533 578 1749 615">public Task&lt;LocationAvailability&gt; getLocationAvailability ()</pre> <p data-bbox="527 646 1692 704">Returns the availability of location data. When <code>isLocationAvailable()</code> returns true, then the location returned by <code>getLastLocation()</code> will be reasonably up to date within the hints specified by the active <code>LocationRequest</code>s.</p> <p data-bbox="527 738 1472 761">If the client isn't connected to Google Play services and the request times out, null is returned.</p> <p data-bbox="527 795 1673 854">Note it's always possible for <code>getLastLocation()</code> to return null even when this method returns true (e.g. location settings were disabled between calls).</p> <hr data-bbox="512 865 1749 867"/> <p data-bbox="512 873 1902 932"><a href="https://developers.google.com/android/reference/com/google/android/gms/location/FusedLocationProviderClient">https://developers.google.com/android/reference/com/google/android/gms/location/FusedLocationProviderClient</a></p>

**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC						
	<p data-bbox="527 245 1745 326"> <code>public Task&lt;Void&gt; requestLocationUpdates (LocationRequest request, LocationCallback callback, Looper looper)</code> </p> <p data-bbox="527 354 1272 378">Requests location updates with a callback on the specified Looper thread.</p> <p data-bbox="527 412 1686 472">This method is suited for the foreground use cases. For background use cases, the <code>PendingIntent</code> version of the method is recommended, see <code>requestLocationUpdates(LocationRequest, PendingIntent)</code>.</p> <p data-bbox="527 506 1371 531">Any previous LocationRequests registered on this LocationListener will be replaced.</p> <p data-bbox="527 565 1686 656">This call will keep the Google Play services connection active, so make sure to call <code>removeLocationUpdates(LocationCallback)</code> when you no longer need it, otherwise you lose the benefits of the automatic connection management.</p> <p data-bbox="527 690 1745 714">Callbacks for <code>LocationCallback</code> will be made on the specified thread, which must already be a prepared looper thread.</p> <p data-bbox="527 740 663 764"><b>Parameters</b></p> <table border="1" data-bbox="527 792 1745 1008"> <tbody> <tr> <td data-bbox="527 800 625 857"><b>request</b></td> <td data-bbox="632 800 1745 857">The location request for the updates.</td> </tr> <tr> <td data-bbox="527 862 625 919"><b>callback</b></td> <td data-bbox="632 862 1745 919">The callback for the location updates.</td> </tr> <tr> <td data-bbox="527 924 625 1005"><b>looper</b></td> <td data-bbox="632 924 1745 1005">The Looper object whose message queue will be used to implement the callback mechanism, or null to make callbacks on the calling thread.</td> </tr> </tbody> </table> <p data-bbox="527 1032 1902 1092"> <a href="https://developers.google.com/android/reference/com/google/android/gms/location/FusedLocationProviderClient">https://developers.google.com/android/reference/com/google/android/gms/location/FusedLocationProviderClient</a> </p>	<b>request</b>	The location request for the updates.	<b>callback</b>	The callback for the location updates.	<b>looper</b>	The Looper object whose message queue will be used to implement the callback mechanism, or null to make callbacks on the calling thread.
<b>request</b>	The location request for the updates.						
<b>callback</b>	The callback for the location updates.						
<b>looper</b>	The Looper object whose message queue will be used to implement the callback mechanism, or null to make callbacks on the calling thread.						

**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC				
	<pre data-bbox="533 240 1740 321">public Task&lt;Void&gt; requestLocationUpdates (LocationRequest request, PendingIntent callbackIntent)</pre> <p data-bbox="525 354 1268 378">Requests location updates with a callback on the specified PendingIntent.</p> <p data-bbox="525 410 1732 540">This method is suited for the background use cases, more specifically for receiving location updates, even when the app has been killed by the system. In order to do so, use a <code>PendingIntent</code> for a started service. For foreground use cases, the <code>LocationCallback</code> version of the method is recommended, see <code>requestLocationUpdates(LocationRequest, LocationCallback, Looper)</code>.</p> <p data-bbox="525 573 1724 630">Any previously registered requests that have the same PendingIntent (as defined by <code>equals(Object)</code>) will be replaced by this request.</p> <p data-bbox="525 662 1728 751">Both <code>LocationResult</code> and <code>LocationAvailability</code> are sent to the given PendingIntent. You can extract data from an Intent using <code>hasResult(Intent)</code>, <code>extractResult(Intent)</code>, <code>hasLocationAvailability(Intent)</code>, and <code>extractLocationAvailability(Intent)</code>.</p> <p data-bbox="543 784 667 808"><b>Parameters</b></p> <table border="1" data-bbox="525 833 1740 971"> <tbody> <tr> <td data-bbox="525 833 835 898"><code>request</code></td> <td data-bbox="835 833 1740 898">The location request for the updates.</td> </tr> <tr> <td data-bbox="525 898 835 971"><code>callbackIntent</code></td> <td data-bbox="835 898 1740 971">A pending intent to be sent for each location update.</td> </tr> </tbody> </table> <p data-bbox="543 995 630 1019"><b>Returns</b></p> <ul data-bbox="554 1044 1360 1068" style="list-style-type: none"> <li>• a Task for the call, check <code>isSuccessful()</code> to determine if it was successful.</li> </ul> <p data-bbox="512 1092 1902 1151"><a href="https://developers.google.com/android/reference/com/google/android/gms/location/FusedLocationProviderClient">https://developers.google.com/android/reference/com/google/android/gms/location/FusedLocationProviderClient</a></p>	<code>request</code>	The location request for the updates.	<code>callbackIntent</code>	A pending intent to be sent for each location update.
<code>request</code>	The location request for the updates.				
<code>callbackIntent</code>	A pending intent to be sent for each location update.				

**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC						
	<p data-bbox="533 245 1738 277"><code>public void onLocationAvailability (LocationAvailability locationAvailability)</code></p> <p data-bbox="533 310 1171 334">Called when there is a change in the availability of location data.</p> <p data-bbox="533 367 1738 561">When <code>isLocationAvailable()</code> returns <code>false</code> you can assume that location will not be returned in <code>onLocationResult(LocationResult)</code> until something changes in the device's settings or environment. Even when <code>isLocationAvailable()</code> returns <code>true</code> the <code>onLocationResult(LocationResult)</code> may not always be called regularly, however the device location is known and both the most recently delivered location and <code>getLastLocation(GoogleApiClient)</code> will be reasonably up to date given the hints specified by the active <code>LocationRequest</code>s.</p> <p data-bbox="533 586 667 610"><b>Parameters</b></p> <table border="1" data-bbox="533 643 1738 708"> <tr> <td data-bbox="533 651 961 708"><code>locationAvailability</code></td> <td data-bbox="968 651 1738 708">The current status of location availability.</td> </tr> </table> <p data-bbox="533 756 1738 789"><code>public void onLocationResult (LocationResult result)</code></p> <p data-bbox="533 821 1052 846">Called when device location information is available.</p> <p data-bbox="533 878 1661 943">The most recent location returned by <code>getLastLocation()</code> is not guaranteed to be immediately fresh, but will be reasonably up to date given the hints specified by the active <code>LocationRequest</code>s.</p> <p data-bbox="533 967 667 992"><b>Parameters</b></p> <table border="1" data-bbox="533 1024 1738 1089"> <tr> <td data-bbox="533 1032 768 1089"><code>result</code></td> <td data-bbox="774 1032 1738 1089">The latest location result available.</td> </tr> </table> <p data-bbox="512 1105 1881 1138"><a href="https://developers.google.com/android/reference/com/google/android/gms/location/LocationCallback">https://developers.google.com/android/reference/com/google/android/gms/location/LocationCallback</a></p> <p data-bbox="533 1154 1738 1187"><code>public abstract void onLocationChanged (Location location)</code></p> <p data-bbox="533 1219 915 1243">Called when the location has changed.</p> <p data-bbox="533 1268 667 1292"><b>Parameters</b></p> <table border="1" data-bbox="533 1325 1738 1390"> <tr> <td data-bbox="533 1333 926 1390"><code>location</code></td> <td data-bbox="932 1333 1738 1390">The updated location.</td> </tr> </table>	<code>locationAvailability</code>	The current status of location availability.	<code>result</code>	The latest location result available.	<code>location</code>	The updated location.
<code>locationAvailability</code>	The current status of location availability.						
<code>result</code>	The latest location result available.						
<code>location</code>	The updated location.						

**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC
	<p data-bbox="516 235 1873 264"><a href="https://developers.google.com/android/reference/com/google/android/gms/location/LocationListener">https://developers.google.com/android/reference/com/google/android/gms/location/LocationListener</a></p> <p data-bbox="516 316 814 345">Public Constructors</p> <hr data-bbox="516 358 1743 362"/> <p data-bbox="533 414 924 443">public <b>MapView</b> (<a href="#">Context</a> context)</p> <p data-bbox="533 505 1129 534">public <b>MapView</b> (<a href="#">Context</a> context, <a href="#">AttributeSet</a> attrs)</p> <p data-bbox="533 596 1268 625">public <b>MapView</b> (<a href="#">Context</a> context, <a href="#">AttributeSet</a> attrs, int defStyle)</p> <p data-bbox="533 686 1247 716">public <b>MapView</b> (<a href="#">Context</a> context, <a href="#">GoogleMapOptions</a> options)</p> <hr data-bbox="516 745 1743 748"/> <p data-bbox="516 755 1734 784"><a href="https://developers.google.com/android/reference/com/google/android/gms/maps/MapView">https://developers.google.com/android/reference/com/google/android/gms/maps/MapView</a></p>

**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

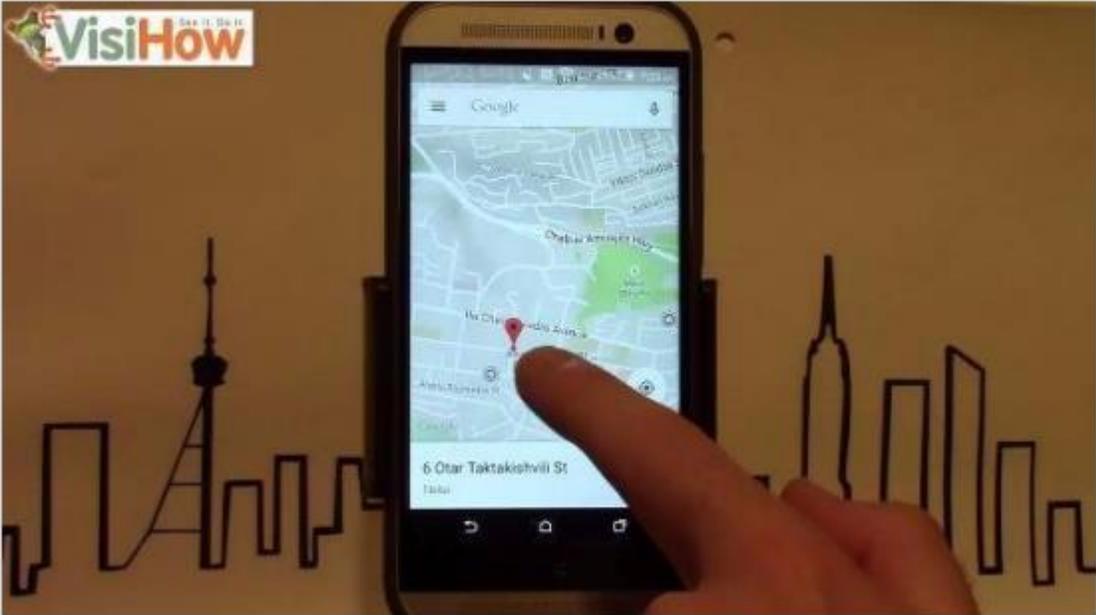
US9408055B2	HTC		
	<p><code>public void <b>getMapAsync</b> (<a href="#">OnMapReadyCallback</a> callback)</code></p> <p>Returns a non-null instance of the <a href="#">GoogleMap</a>, ready to be used.</p> <p>Note that:</p> <ul style="list-style-type: none"> <li>• This method must be called from the main thread.</li> <li>• The callback will be executed in the main thread.</li> <li>• In the case where Google Play services is not installed on the user's device, the callback will not be triggered until the user installs it.</li> <li>• The <a href="#">GoogleMap</a> object provided by the callback is non-null.</li> </ul> <p><b>Parameters</b></p> <table border="1" data-bbox="527 688 1738 753"> <tr> <td data-bbox="527 688 695 753"><b>callback</b></td> <td data-bbox="701 688 1738 753">The callback object that will be triggered when the map is ready to be used.</td> </tr> </table> <p><code>public final void <b>onCreate</b> (<a href="#">Bundle</a> savedInstanceState)</code></p> <p>You must call this method from the parent Activity/Fragment's corresponding method.</p> <hr/> <p><a href="https://developers.google.com/android/reference/com/google/android/gms/maps/MapView">https://developers.google.com/android/reference/com/google/android/gms/maps/MapView</a></p>	<b>callback</b>	The callback object that will be triggered when the map is ready to be used.
<b>callback</b>	The callback object that will be triggered when the map is ready to be used.		
<p>[54G] identifying user interaction with the interactive display selecting one or more of the user-selectable symbols corresponding to one or more of the second devices and user interaction with the display specifying an</p>	<p>HTC infringes directly and/or indirectly by performing, inducing others to perform, and/or contributing to the performance of identifying user interaction with the interactive display selecting one or more of the user-selectable symbols corresponding to one or more of the second devices and user interaction with the display specifying an action and, based thereon, sending data to the one or more second devices. See claims 1[F], 28[F], and 41[F], which are incorporated herein by reference in their entirety.</p> <p><b>Regarding Google Maps</b>, Google Latitude, Google Plus, Google Hangouts, Google Messages, and Android Messenger, the Accused Products are configured to allow a user of the first device to interact with the display, to select a user, contact, or device, and to select an action to be performed, such as: sending a message, initiating a call, initiating a data conference, sharing a location, stop sharing a location, block a user from location sharing, sending a location, requesting a location, or sending other data. When an action is</p>		



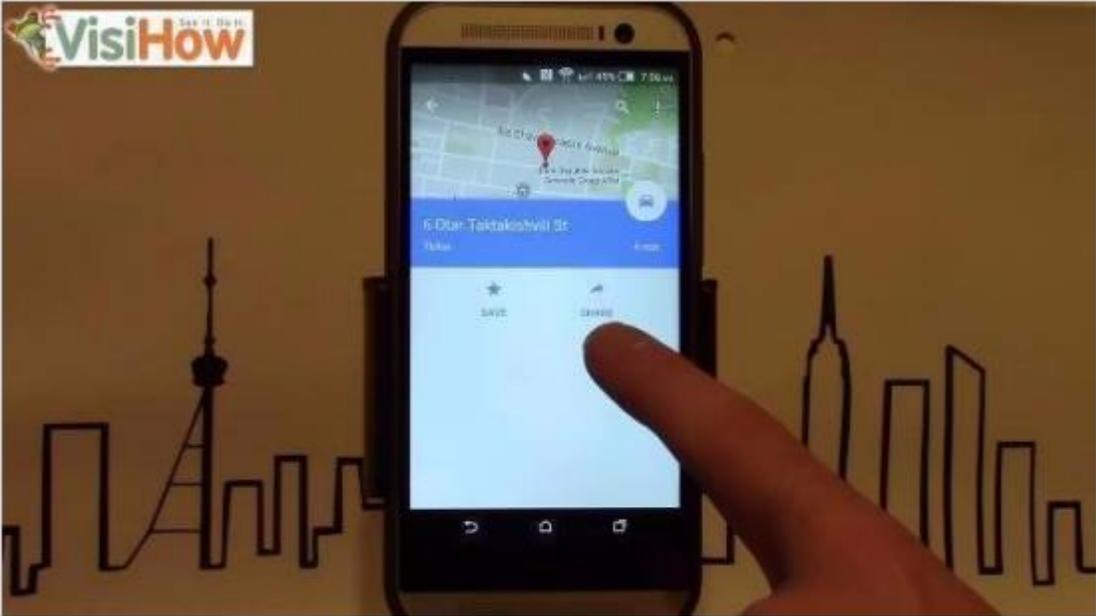
**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

<b>US9408055B2</b>	<b>HTC</b>
action and, based thereon, sending data to the one or more second devices;	<p>specified, data is sent from the first device to the second device via a server.</p> <p><b><u>Exemplary Support for Google Maps:</u></b></p> <p>Using Google Maps, a user may choose a symbol and send data to that device. For example, a user who is already sharing her location with another user can stop sharing by making a selection resulting in the second device no longer displaying the first device's location. Additionally, a user can share an ETA message with another user or send another user a link in a message to share her location. Additionally, a user who is sharing a location until she arrives can make a selection to stop her location from showing on the second device.</p> <p><b>Share Locations from Google Maps on HTC One M8</b></p>

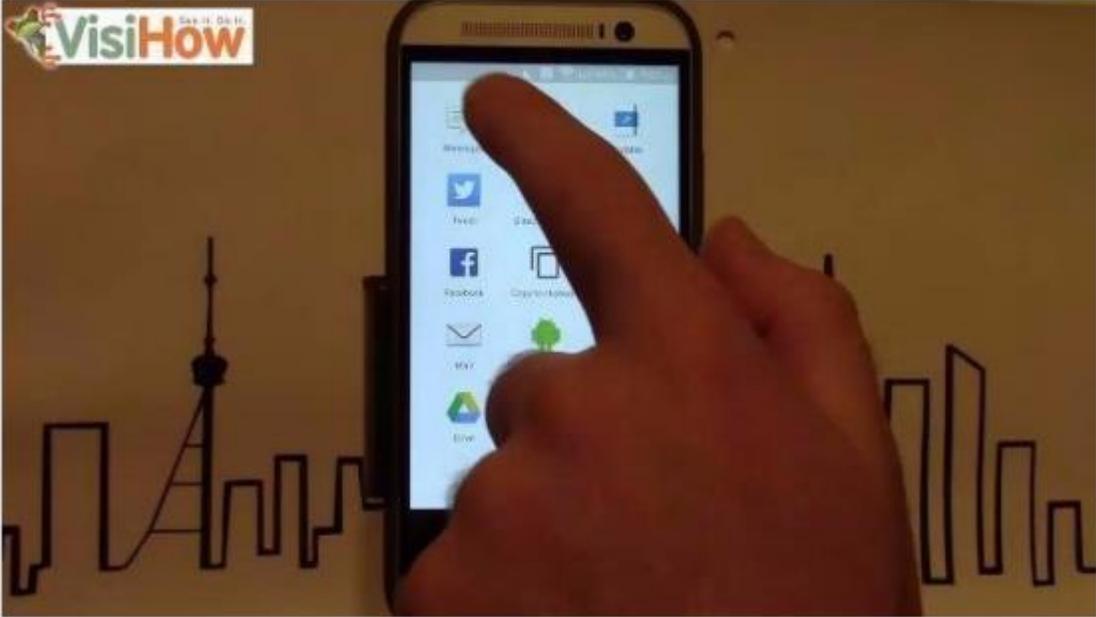
**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC
	<p data-bbox="531 228 827 256"><b>Place a pin on the map.</b></p> <p data-bbox="531 269 1638 380">We can do this anywhere in the entire world, anywhere, on any map that we find in the Google Maps application. We are going to choose a location, press down on the screen, and hold down until we can see that a red pin has been dropped.</p> 

**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC
	<p data-bbox="527 240 1207 305"><b>We have two options:</b> "SAVE" and "SHARE". Press down on the word "SHARE".</p>  <p data-bbox="527 1057 1633 1208"><b>A menu appears with all of the different ways that we can share this location.</b> We can share this location via messages, Bluetooth, Twitter, Facebook, mail, Gmail and Viber. If you have other applications, such as WhatsApp, or any other chat or email applications, you will be able to use them as well and will see them here.</p>

**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC
	<p data-bbox="527 237 856 261"><b>Let's choose "Messages".</b></p> <p data-bbox="527 277 1633 342">This option is located on the top left-hand corner of the screen. All options will have the same basic processes. They will ask us to send the location. Press down on the "Messages" icon.</p> 

**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC
	<p data-bbox="520 233 1260 256">Press the box next to the contact who will be the recipient.</p> <p data-bbox="520 269 1549 292">A checkmark will appear in the box. We can select multiple contacts as well if we desire.</p>  <p data-bbox="520 1045 911 1068">Press on the word "DONE (1)".</p> <p data-bbox="520 1081 1629 1153">This icon is located in the bottom right-hand corner of the screen. The bracketed number indicates the number of contacts we selected.</p> <p data-bbox="520 1166 1407 1188"><a href="http://visihow.com/Share_Locations_from_Google_Maps_on_HTC_One_M8">http://visihow.com/Share_Locations_from_Google_Maps_on_HTC_One_M8</a></p>

## Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products

US9408055B2	HTC
	<p data-bbox="548 277 968 298">COMPUTER <b>ANDROID</b> IPHONE &amp; IPAD</p> <hr data-bbox="533 326 1568 329"/> <h3 data-bbox="533 383 1024 420">If they have a Google Account</h3> <ol data-bbox="533 443 1419 732" style="list-style-type: none"><li data-bbox="533 443 1220 464">1. If you haven't already, add their Gmail address to your <a href="#">Google Contacts</a>.</li><li data-bbox="533 480 1419 501">2. On your Android phone or tablet, open the Google Maps app  and sign in. <a href="#">Learn how to sign in.</a></li><li data-bbox="533 518 1031 539">3. Tap Menu ≡ &gt; <b>Location sharing</b> &gt; Add People +.</li><li data-bbox="533 555 1003 576">4. Choose how long you want to share your location.</li><li data-bbox="533 592 1140 647">5. Tap <b>Select People</b>.<ul data-bbox="569 626 1140 647" style="list-style-type: none"><li data-bbox="569 626 1140 647">• If you're asked about your contacts, give Google Maps access.</li></ul></li><li data-bbox="533 680 884 701">6. Choose who you want to share with.</li><li data-bbox="533 717 663 738">7. Tap <b>Share</b>.</li></ol> <h3 data-bbox="533 794 1110 831">If they don't have a Google Account</h3> <ol data-bbox="533 854 1560 976" style="list-style-type: none"><li data-bbox="533 854 1419 875">1. On your Android phone or tablet, open the Google Maps app  and sign in. <a href="#">Learn how to sign in.</a></li><li data-bbox="533 891 1031 912">2. Tap Menu ≡ &gt; <b>Location sharing</b> &gt; Add People +.</li><li data-bbox="533 928 1560 976">3. Tap More ... &gt; <b>Copy to clipboard</b>. People with this link can see your location for as long as you choose, up to 72 hours.</li></ol> <h3 data-bbox="533 1027 869 1065">Share using another app</h3> <p data-bbox="533 1079 1205 1101">You can also share through messaging apps. Tap More ... &gt; select an app.</p> <h3 data-bbox="533 1162 741 1200">Stop sharing</h3> <ol data-bbox="533 1222 1205 1317" style="list-style-type: none"><li data-bbox="533 1222 842 1243">1. Open the Google Maps app .</li><li data-bbox="533 1260 869 1281">2. Tap Menu ≡ &gt; <b>Location sharing</b>.</li><li data-bbox="533 1297 1205 1318">3. Next to the person with whom you want to stop sharing, tap Remove ✕.</li></ol> <p data-bbox="512 1333 1703 1365"><a href="https://support.google.com/maps/answer/7326816?co=GENIE.Platform%3DAndroid&amp;oco=1">https://support.google.com/maps/answer/7326816?co=GENIE.Platform%3DAndroid&amp;oco=1</a></p>

## Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products

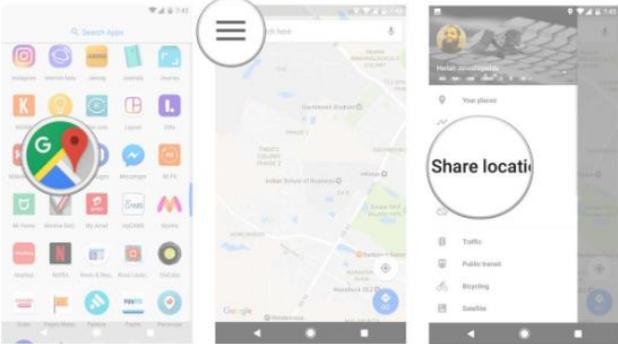
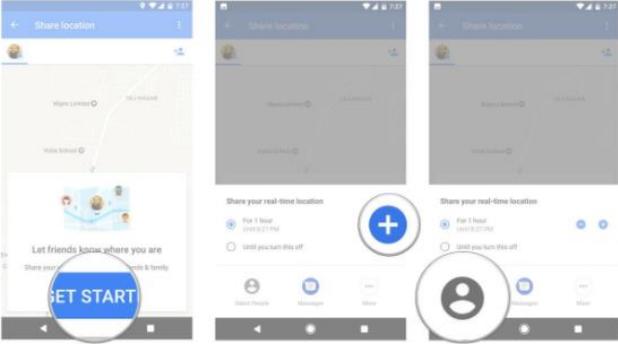
US9408055B2	HTC
	<h3>Share your E.T.A</h3> <p>After you start your drive, you can share your destination, estimated arrival time, and where you are on the route.</p> <ol style="list-style-type: none"><li>1. Open the Google Maps app .</li><li>2. Set a driving destination. <a href="#">Learn how to navigate to a place.</a></li><li>3. After you start navigation, tap More  &gt; <b>Share trip progress.</b></li><li>4. Choose a person from the list.</li><li>5. Tap <b>Share.</b></li><li>6. Location Sharing will stop when you reach your destination or stop navigating.</li></ol> <ul style="list-style-type: none"><li>• To stop sharing before you arrive, tap More  &gt; <b>Stop sharing.</b></li></ul> <h3>See where someone is</h3> <p>If someone shares their location with you, you can see them on the map.</p> <ol style="list-style-type: none"><li>1. Open the Google Maps app .</li><li>2. Tap Menu  &gt; <b>Location sharing.</b></li><li>3. Choose someone.</li></ol> <ul style="list-style-type: none"><li>• To see an updated location, tap on a friend's icon &gt; More  &gt; <b>Refresh.</b></li></ul> <h3>Stop seeing someone's location</h3> <ol style="list-style-type: none"><li>1. Open the Google Maps app .</li><li>2. On the map, tap their icon.</li><li>3. At the bottom, tap More .</li><li>4. To temporarily hide someone, tap <b>Hide from map.</b> You can stop hiding them at any time.</li></ol> <p><b>Note:</b> You can stop someone's location from ever appearing on your map. <a href="#">Learn how to block another person's account.</a></p> <p><a href="https://support.google.com/maps/answer/7326816?co=GENIE.Platform%3DAndroid&amp;oco=1">https://support.google.com/maps/answer/7326816?co=GENIE.Platform%3DAndroid&amp;oco=1</a></p>

## Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products

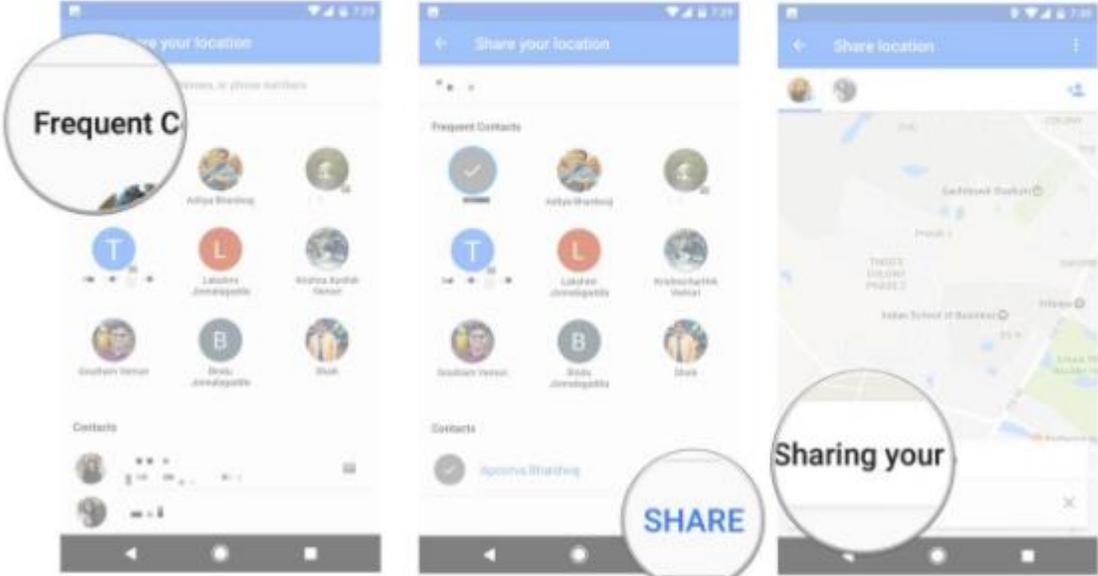
US9408055B2	HTC
	<h3 data-bbox="541 282 877 321">Hide or share lists</h3> <p data-bbox="541 349 907 373"><b>Note:</b> You can't share starred places.</p> <ol data-bbox="554 402 1251 511" style="list-style-type: none"><li data-bbox="554 402 890 430">1. Open the Google Maps app .</li><li data-bbox="554 443 966 470">2. Tap Menu  &gt; <b>Your places</b> &gt; <b>Saved</b>.</li><li data-bbox="554 483 1251 511">3. Next to the list you want to share, tap More  &gt; choose an option:</li></ol> <ul data-bbox="583 524 1680 669" style="list-style-type: none"><li data-bbox="583 524 1440 552">• <b>Hide/Show on your map:</b> Display or hide your saved places when looking at the map.</li><li data-bbox="583 565 1054 592">• <b>Share list:</b> Allow others to see your saved list.</li><li data-bbox="583 605 1680 669">• <b>Sharing options:</b> You can make your list public, private, or shared. To let anyone with the link see your list, tap <b>Shared</b>. To let anyone find and follow your list, tap <b>Public</b>.</li></ul> <h3 data-bbox="541 735 764 774">Follow a list</h3> <p data-bbox="541 802 1726 862">If you follow a list made by someone else, their saved places will show up in Your Places. The places will also appear as suggested locations in Google Maps.</p> <h3 data-bbox="541 914 915 953">Follow a list using a link</h3> <ol data-bbox="554 974 1352 1083" style="list-style-type: none"><li data-bbox="554 974 957 1002">1. Tap on the link you received to open it.</li><li data-bbox="554 1015 1268 1042">2. Tap <b>Follow</b>. This list will now be added to the group of lists you follow.</li><li data-bbox="554 1055 1352 1083">3. <b>Optional:</b> To unfollow a list someone shared with you, tap the list &gt; <b>Following</b>.</li></ol> <h3 data-bbox="541 1135 924 1174">See lists made by others</h3> <p data-bbox="541 1195 1331 1219">If a user has any Google Maps lists that were made public, you can follow them.</p> <ol data-bbox="554 1248 1134 1357" style="list-style-type: none"><li data-bbox="554 1248 1134 1276">1. Tap on the name of a user whose list you want to follow.</li><li data-bbox="554 1289 676 1317">2. Tap <b>Lists</b>.</li><li data-bbox="554 1330 1134 1357">3. Tap on the list you want to follow &gt; More  &gt; <b>Follow</b>.</li></ol> <p data-bbox="512 1370 1898 1403"><a href="https://support.google.com/maps/answer/7280933?hl=en&amp;ref_topic=7301134&amp;co=GENIE.Platform%3DAn">https://support.google.com/maps/answer/7280933?hl=en&amp;ref_topic=7301134&amp;co=GENIE.Platform%3DAn</a></p>



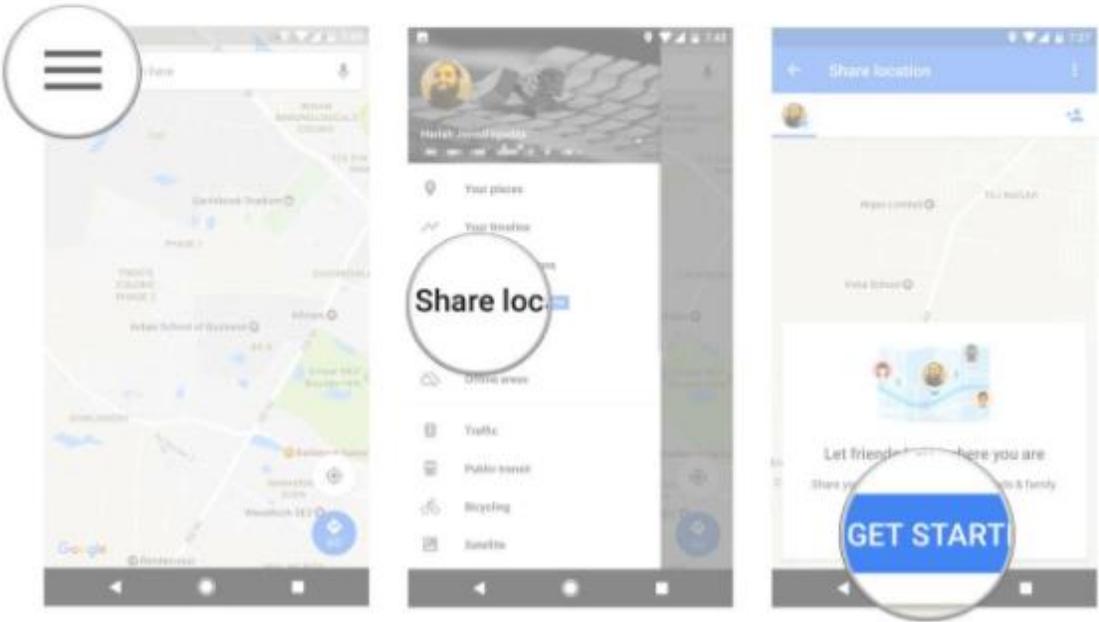
# Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products

US9408055B2	HTC
	<p data-bbox="512 233 684 261">droid&amp;oco=1</p> <h3 data-bbox="512 310 1104 342">How to share your location in Google Maps</h3> <ol data-bbox="512 367 1087 451" style="list-style-type: none"><li>1. Open Google Maps from the app drawer or the home screen.</li><li>2. Tap the <b>hamburger menu</b> (the three horizontal lines) on the top left corner of the screen.</li><li>3. Select <b>Share location</b>.</li></ol>  <ol data-bbox="512 854 1117 954" style="list-style-type: none"><li>4. Tap <b>Get Started</b>.</li><li>5. Use the <b>+</b> icon to select a time period or select the <b>Until you turn this off</b> setting to share your location indefinitely.</li><li>6. Tap <b>Select People</b>.</li></ol>  <p data-bbox="512 1333 1356 1365"><a href="https://www.androidcentral.com/how-share-location-google-maps">https://www.androidcentral.com/how-share-location-google-maps</a></p>

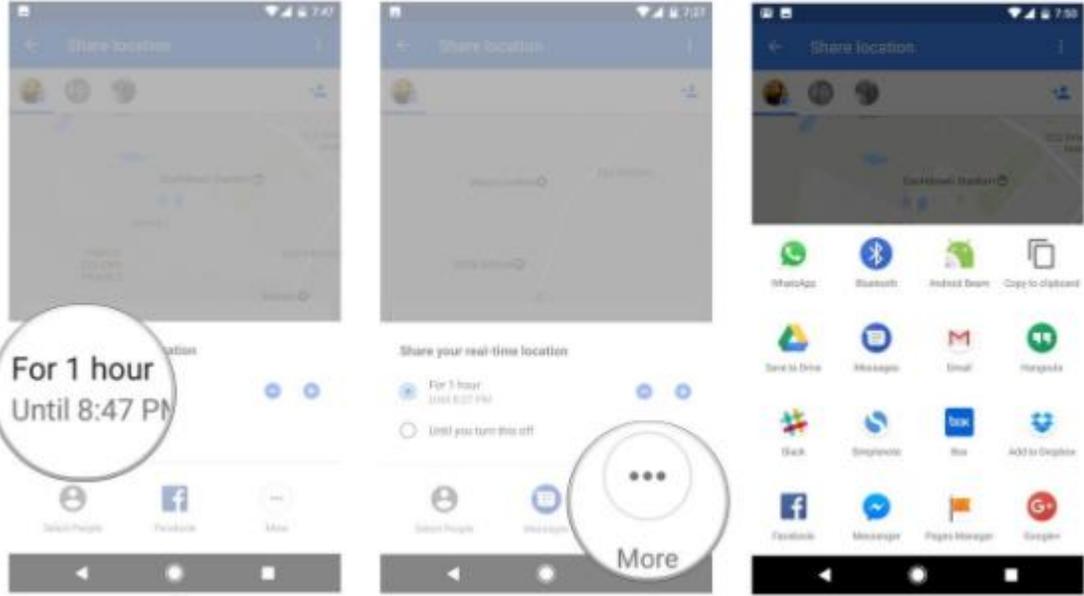
**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC
	<p data-bbox="527 253 1577 427"><b>7.</b> You'll see a list of your frequent contacts at the top, along with a full list of contacts. <b>Pick the contacts</b> by tapping their name.</p> <p data-bbox="527 342 1457 367"><b>8.</b> Once you've selected the contacts you want to share your location to, tap <b>Share</b>.</p> <p data-bbox="527 399 1419 423"><b>9.</b> You'll see a message saying that the selected contact can view your location.</p>  <p data-bbox="510 1101 1356 1133"><a href="https://www.androidcentral.com/how-share-location-google-maps">https://www.androidcentral.com/how-share-location-google-maps</a></p>

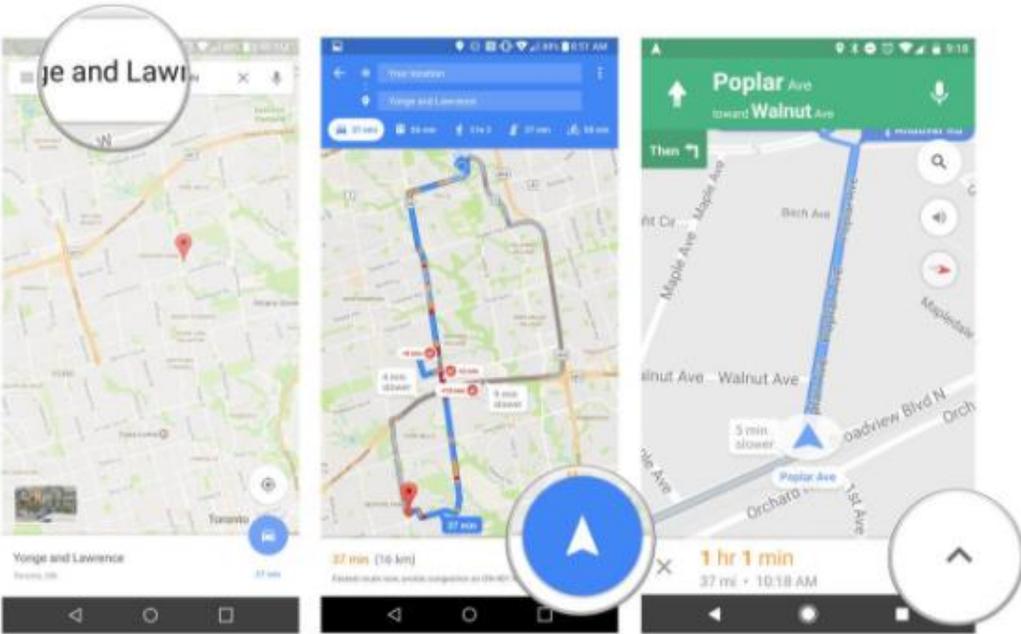
## Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products

US9408055B2	HTC
	<h3 data-bbox="520 240 1255 289">How to create a shareable link</h3> <p data-bbox="520 329 1461 358">You can also create a link and use it to share your location easily. Here's how to do it:</p> <ol data-bbox="520 407 1234 548" style="list-style-type: none"><li data-bbox="520 407 1234 436">1. Tap the hamburger menu on the top left corner of the screen.</li><li data-bbox="520 464 800 493">2. Select Share location.</li><li data-bbox="520 521 737 550">3. Tap Get Started.</li></ol>  <p data-bbox="510 1230 1356 1263"><a href="https://www.androidcentral.com/how-share-location-google-maps">https://www.androidcentral.com/how-share-location-google-maps</a></p>

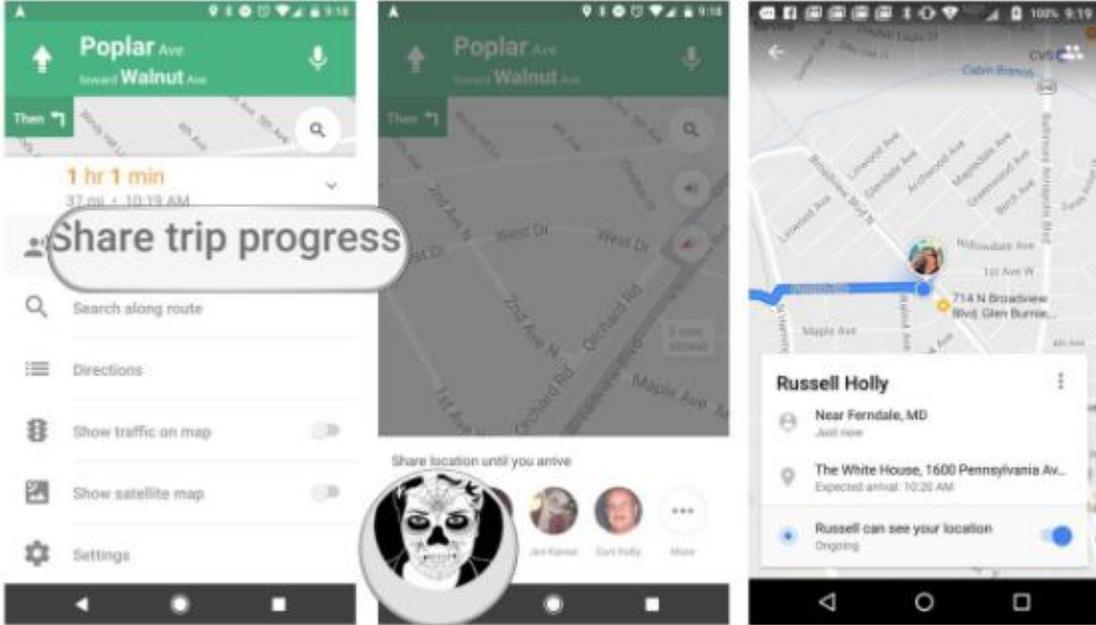
**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC
	<p data-bbox="520 245 1213 272">4. Select the <b>amount of time</b> you want to share your location.</p> <p data-bbox="520 302 680 329">5. Tap <b>More</b>.</p> <p data-bbox="520 358 1633 418">6. Select <b>your app of choice</b> to create and send a <b>unique URL</b> that broadcasts your current location. You can email it, send the link via Messenger, or even tweet it to the intended recipient.</p>  <p data-bbox="506 1084 1360 1117"><a href="https://www.androidcentral.com/how-share-location-google-maps">https://www.androidcentral.com/how-share-location-google-maps</a></p>

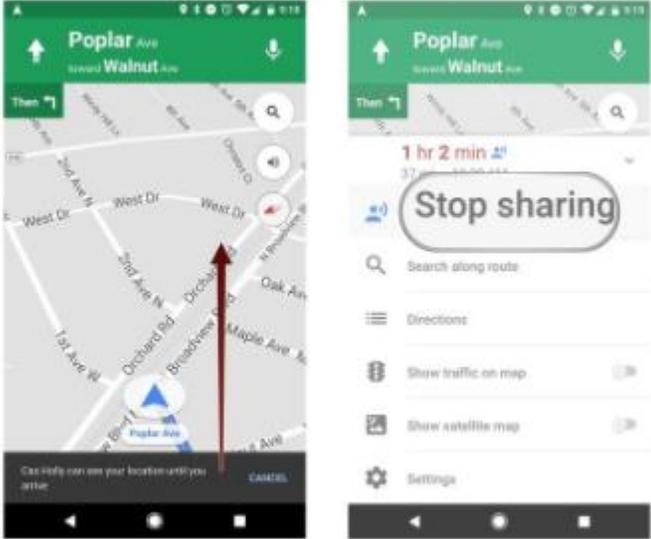
**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC
	<h2 data-bbox="527 240 1423 337">How to share your navigation directions while you walk, drive or transit</h2> <p data-bbox="527 375 1556 464">One of the best ways to share your location with someone is during a drive, walk or transit. If you're meeting a friend or family member somewhere, or navigating towards their home, Google Maps lets you share your location with them for the duration of the trip. It's magic!</p> <ol data-bbox="527 513 1394 643" style="list-style-type: none"><li data-bbox="527 513 974 537">1. In the <b>search bar</b> enter your destination.</li><li data-bbox="527 561 1394 586">2. Pick your navigation type (drive, transit, walk) and press the blue <b>navigate</b> button.</li><li data-bbox="527 610 1394 634">3. Tap the arrow next to the <b>time-to-destination</b> number at the bottom of the screen.</li></ol>  <p data-bbox="512 1328 1356 1360"><a href="https://www.androidcentral.com/how-share-location-google-maps">https://www.androidcentral.com/how-share-location-google-maps</a></p>

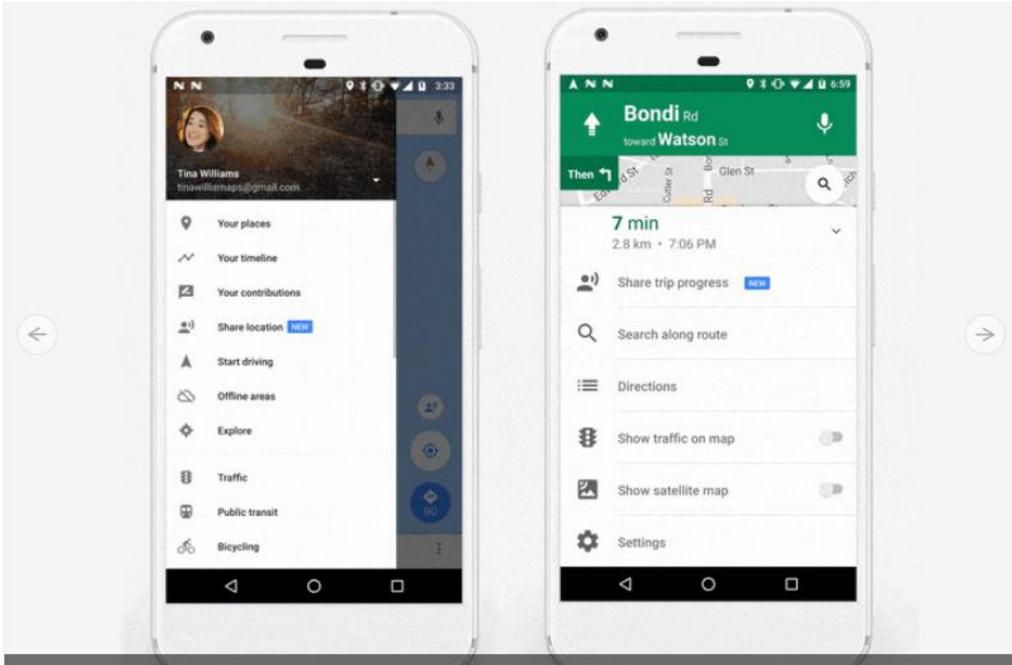
### Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products

US9408055B2	HTC
	<p data-bbox="527 277 835 305">4. Tap Share trip progress.</p> <p data-bbox="527 334 1150 362">5. Choose one or more contacts to share trip progress.</p>  <p data-bbox="527 1065 1360 1133">You can also stop sharing your location with someone before a trip ends. <a href="https://www.androidcentral.com/how-share-location-google-maps">https://www.androidcentral.com/how-share-location-google-maps</a></p>

**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

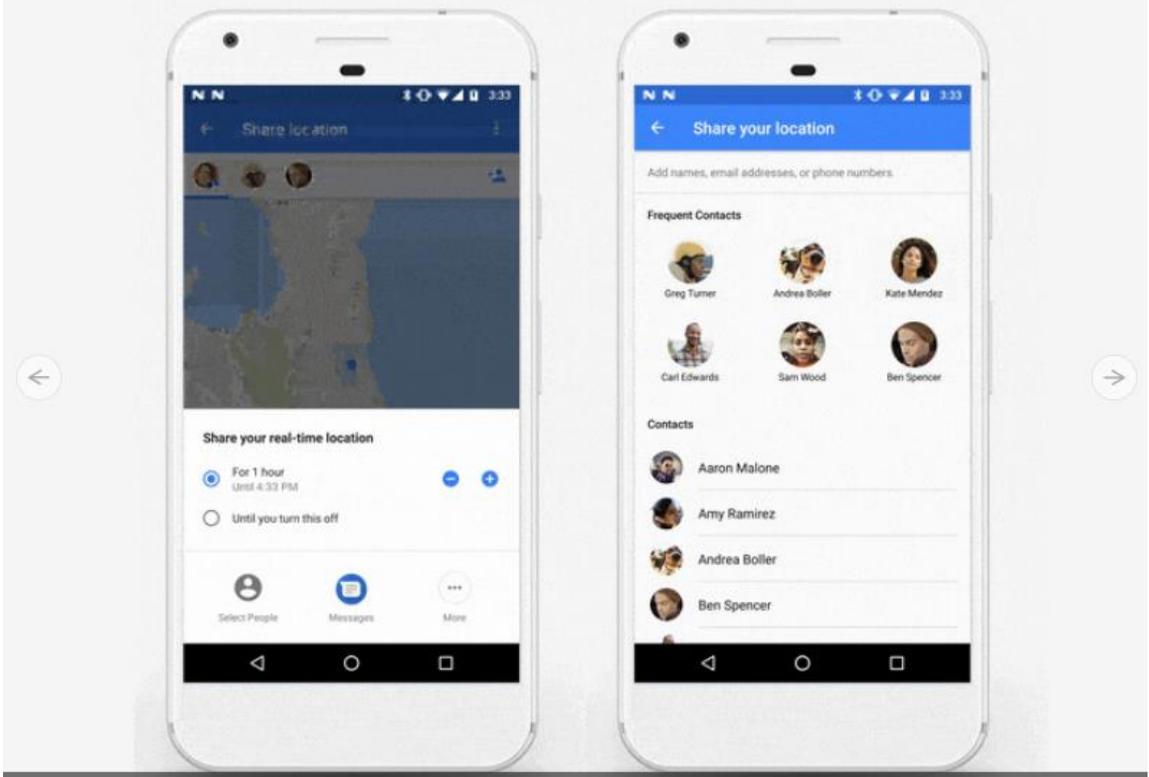
US9408055B2	HTC
	<ol style="list-style-type: none"><li data-bbox="533 245 1465 272">1. Tap the arrow next to the <b>time-to-destination</b> number at the bottom of the screen.</li><li data-bbox="533 302 768 329">2. Tap <b>Stop sharing</b>.</li></ol> <div data-bbox="747 383 1398 922"></div> <p data-bbox="541 976 636 1003">That's it!</p> <p data-bbox="541 1045 1608 1073">Are you excited that location sharing is back in Google Maps? How often do you use the feature?</p> <p data-bbox="512 1084 1356 1112"><a href="https://www.androidcentral.com/how-share-location-google-maps">https://www.androidcentral.com/how-share-location-google-maps</a></p> <p data-bbox="512 1195 1419 1222">As shown below, a group may also be defined within Google Contacts.</p>

## Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products

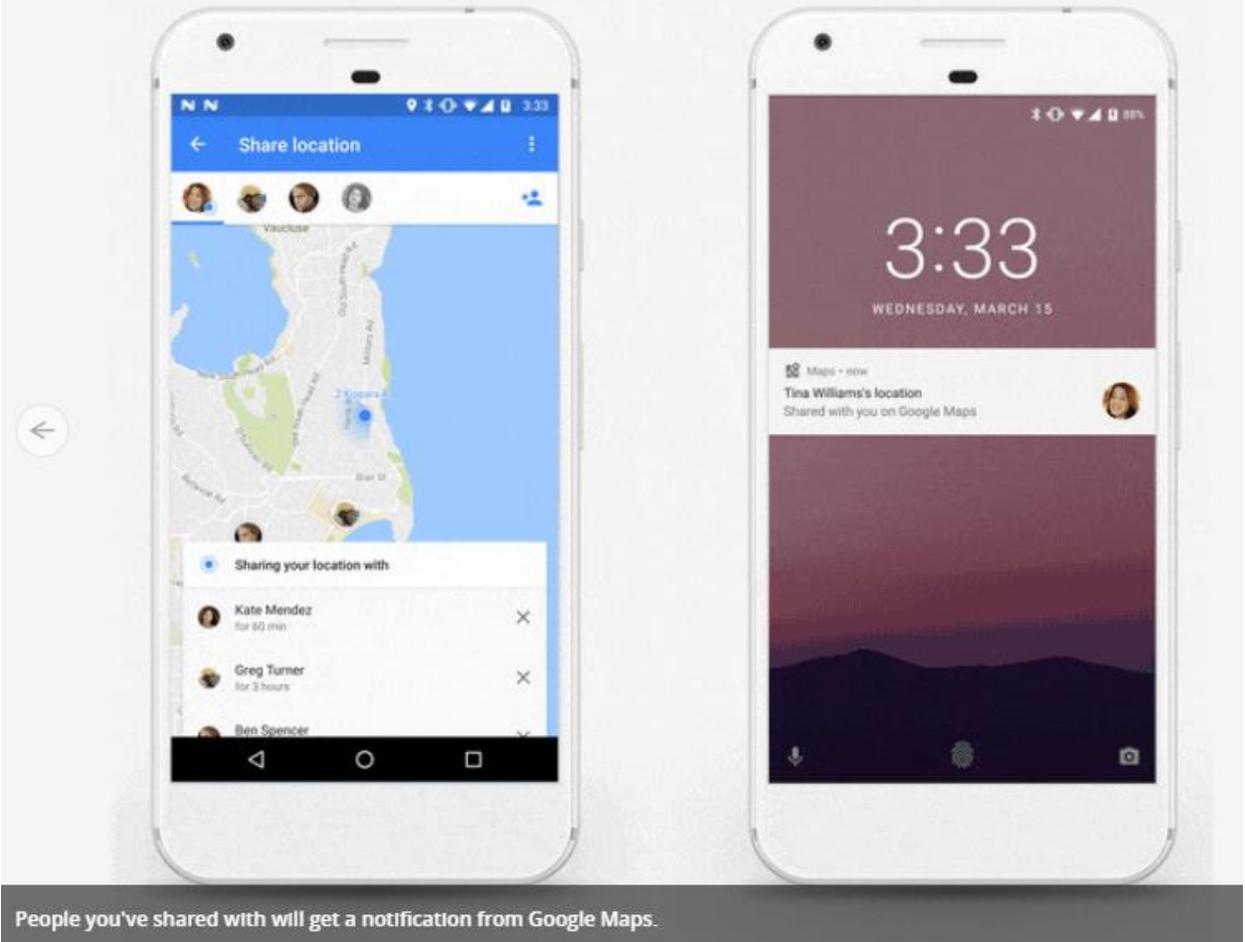
US9408055B2	HTC
	<p data-bbox="548 245 940 289"><b>Share your contacts</b></p> <ol data-bbox="562 318 1045 477" style="list-style-type: none"><li>1. Open your device's Contacts app .</li><li>2. Tap a contact in the list.</li><li>3. Tap More  &gt; <b>Share</b>.</li><li>4. Choose how you want to share the contact.</li></ol> <p data-bbox="512 496 1535 529"><a href="https://support.google.com/android/answer/6118731?hl=en&amp;ref_topic=6118711">https://support.google.com/android/answer/6118731?hl=en&amp;ref_topic=6118711</a></p>  <p data-bbox="520 1230 1516 1289">Ways to start location sharing: (left) from the app navigation drawer, and (right) from the menu, you can drag up while navigating somewhere.</p> <p data-bbox="512 1295 1654 1325"><a href="https://arstechnica.com/gadgets/2017/03/location-sharing-finally-returns-to-google-maps/">https://arstechnica.com/gadgets/2017/03/location-sharing-finally-returns-to-google-maps/</a></p>



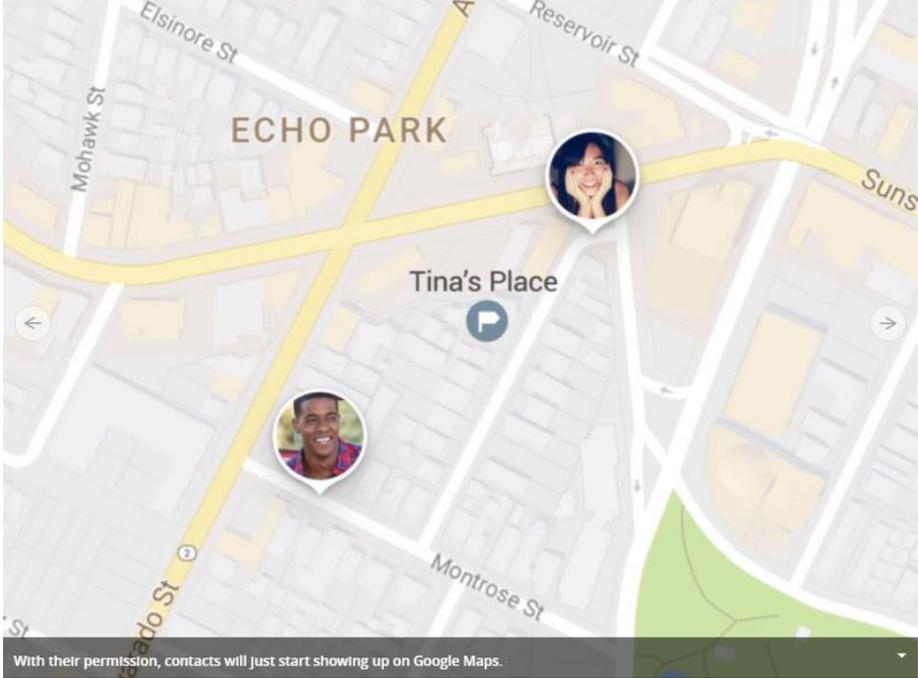
**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC
	 <p data-bbox="512 1024 1661 1060">You can pick how long you want to share your location for, and with whom you want to share it.</p> <p data-bbox="512 1065 1661 1096"><a href="https://arstechnica.com/gadgets/2017/03/location-sharing-finally-returns-to-google-maps/">https://arstechnica.com/gadgets/2017/03/location-sharing-finally-returns-to-google-maps/</a></p>

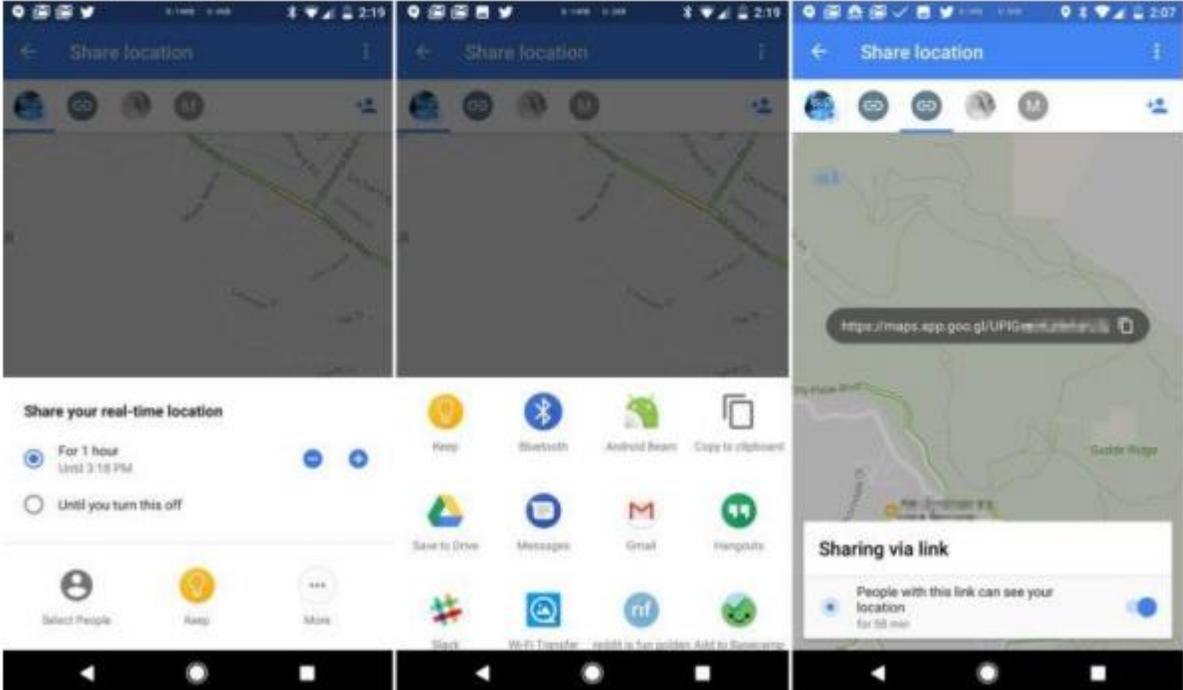
**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC
	 <p data-bbox="514 1144 1176 1177">People you've shared with will get a notification from Google Maps.</p> <p data-bbox="514 1185 1659 1226"><a href="https://arstechnica.com/gadgets/2017/03/location-sharing-finally-returns-to-google-maps/">https://arstechnica.com/gadgets/2017/03/location-sharing-finally-returns-to-google-maps/</a></p>

**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC
	 <p data-bbox="512 883 1430 911">With their permission, contacts will just start showing up on Google Maps.</p> <p data-bbox="512 915 1656 946"><a href="https://arstechnica.com/gadgets/2017/03/location-sharing-finally-returns-to-google-maps/">https://arstechnica.com/gadgets/2017/03/location-sharing-finally-returns-to-google-maps/</a></p> <h3 data-bbox="512 1019 779 1068">Stop sharing</h3> <ol data-bbox="512 1094 1352 1219" style="list-style-type: none"><li>1. Open the Google Maps app 📍.</li><li>2. Tap the Menu ☰ &gt; Share location.</li><li>3. Next to the person with whom you want to stop sharing, tap Remove ✕.</li></ol>

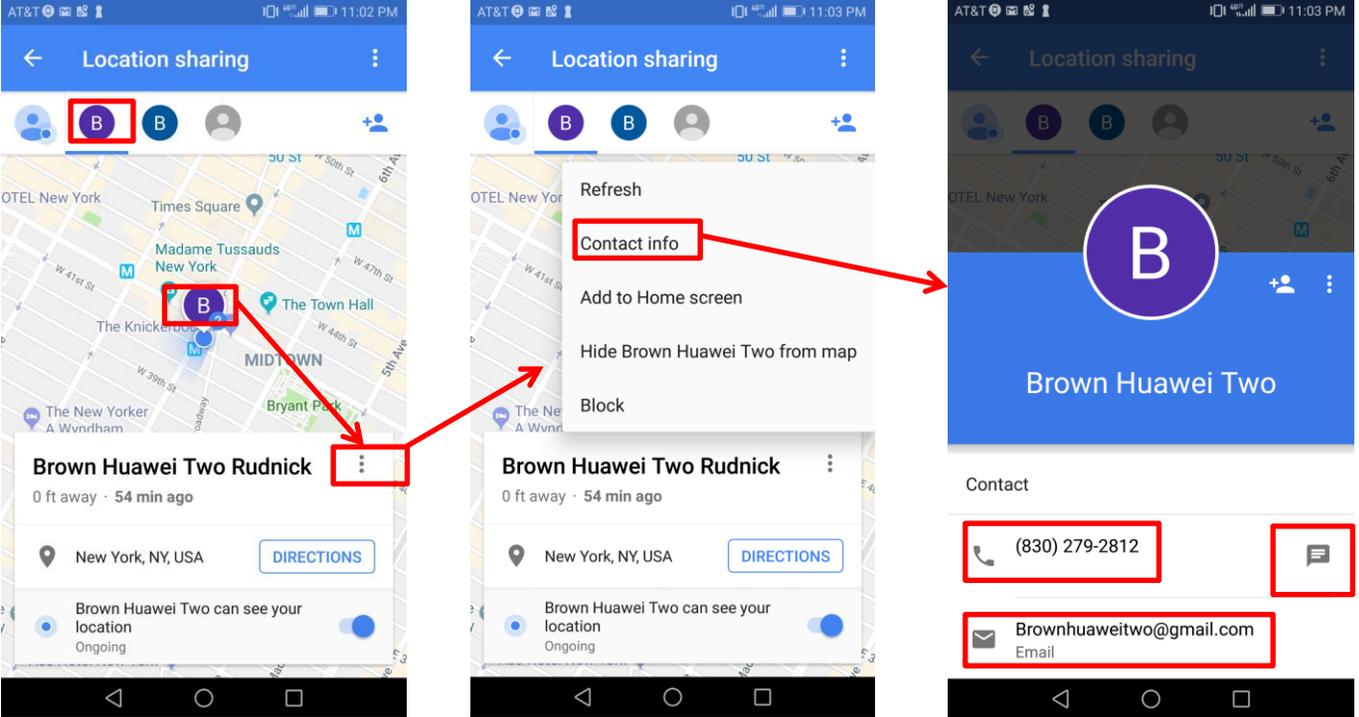
**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC
	<h3 data-bbox="541 233 856 272">Share your E.T.A</h3> <p data-bbox="541 302 1703 326">After you start your drive, you can share your destination, estimated arrival time, and where you are on the route.</p> <ol data-bbox="552 358 1388 602" style="list-style-type: none"><li>1. Open the Google Maps app .</li><li>2. Set a driving destination. <a href="#">Learn how to navigate to a place.</a></li><li>3. After you start navigation, tap <b>More</b>  <b>&gt; Share trip progress.</b></li><li>4. Choose a person from the list.</li><li>5. Tap <b>Share.</b></li><li>6. Location Sharing will stop when you reach your destination or stop navigating.</li></ol> <p data-bbox="552 634 1230 659">• To stop sharing before you arrive, tap <b>More</b>  <b>&gt; Stop sharing.</b></p> 

**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC
	 <p><b><u>Exemplary Maps Screenshots:</u></b></p>

**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC
	 <p><b>Exemplary Source Code:</b></p> <p>The above functionality is performed at least in part by the following publicly available source code and/or source code that invokes or is invoked by the following source code (or a substantially similar copy compiled and loaded onto the Accused Products by HTC). AGIS reserves the right to supplement these contentions with additional source code as discovery progresses and as additional source code is made available. AGIS reserves the right to supplement these contentions with additional source code as discovery progresses and as additional source code is made available.</p>

**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC
	<pre>44 * Class that sends chat message via SMS. 45 * 46 * The interface emulates a blocking sending similar to making an HTTP request. 47 * It calls the SmsManager to send a (potentially multipart) message and waits 48 * on the sent status on each part. The waiting has a timeout so it won't wait 49 * forever. Once the sent status of all parts received, the call returns. 50 * A successful sending requires success status for all parts. Otherwise, we 51 * pick the highest level of failure as the error for the whole message, which 52 * is used to determine if we need to retry the sending. 53 */ 54 public class SmsSender { 55     private static final String TAG = LogUtil.BUGLE_TAG; 56 57     public static final String EXTRA_PART_ID = "part_id"; 58 59     /* 60     * A map for pending sms messages. The key is the random request UUID. 61     */ 62     private static ConcurrentHashMap&lt;Uri, SendResult&gt; sPendingMessageMap = 63         new ConcurrentHashMap&lt;Uri, SendResult&gt;(); 64 65     private static final Random RANDOM = new Random(); 66 67     // Whether we should send multipart SMS as separate messages 68     private static Boolean sSendMultipartSmsAsSeparateMessages = null; 69</pre> <p><a href="https://android.googlesource.com/platform/packages/apps/Messaging/+nougat-mr1-release/src/com/android/messaging/sms/SmsSender.java">https://android.googlesource.com/platform/packages/apps/Messaging/+nougat-mr1-release/src/com/android/messaging/sms/SmsSender.java</a></p>

**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC
	<pre> 253 // Actually sending the message using SmsManager 254 private static void sendInternal(final Context context, final int subId, String dest, 255     final ArrayList&lt;String&gt; messages, final String serviceCenter, 256     final boolean requireDeliveryReport, final Uri messageUri) throws SmsException { 257     Assert.notNull(context); 258     final SmsManager smsManager = PhoneUtils.get(subId).getSmsManager(); 259     final int messageCount = messages.size(); 260     final ArrayList&lt;PendingIntent&gt; deliveryIntents = new ArrayList&lt;PendingIntent&gt;(messageCount); 261     final ArrayList&lt;PendingIntent&gt; sentIntents = new ArrayList&lt;PendingIntent&gt;(messageCount); 262     for (int i = 0; i &lt; messageCount; i++) { 263         // Make pending intents different for each message part 264         final int partId = (messageCount &lt;= 1 ? 0 : i + 1); 265         if (requireDeliveryReport &amp;&amp; (i == (messageCount - 1))) { 266             // TODO we only care about the delivery status of the last part 267             // Shall we have better tracking of delivery status of all parts? 268             deliveryIntents.add(PendingIntent.getBroadcast( 269                 context, 270                 partId, 271                 getSendStatusIntent(context, SendStatusReceiver.MESSAGE_DELIVERED_ACTION, 272                     messageUri, partId, subId), 273                 0/*flag*/)); 274         } else { 275             deliveryIntents.add(null); 276         } 277         sentIntents.add(PendingIntent.getBroadcast( 278             context, 279             partId, 280             getSendStatusIntent(context, SendStatusReceiver.MESSAGE_SENT_ACTION, 281                 messageUri, partId, subId), 282             0/*flag*/)); 283     } 284     if (sSendMultipartSmsAsSeparateMessages == null) { 285         sSendMultipartSmsAsSeparateMessages = MmsConfig.get(subId) 286             .getSendMultipartSmsAsSeparateMessages(); 287     } </pre> <p><a href="https://android.googlesource.com/platform/packages/apps/Messaging/+nougat-mr1-release/src/com/android/messaging/sms/SmsSender.java">https://android.googlesource.com/platform/packages/apps/Messaging/+nougat-mr1-release/src/com/android/messaging/sms/SmsSender.java</a></p>



**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC
	<pre> 288     try { 289         if (sSendMultipartSmsAsSeparateMessages) { 290             // If multipart sms is not supported, send them as separate messages 291             for (int i = 0; i &lt; messageCount; i++) { 292                 smsManager.sendTextMessage(dest, 293                     serviceCenter, 294                     messages.get(i), 295                     sentIntents.get(i), 296                     deliveryIntents.get(i)); 297             } 298         } else { 299             smsManager.sendMultipartTextMessage( 300                 dest, serviceCenter, messages, sentIntents, deliveryIntents); 301         } 302     } catch (final Exception e) { 303         throw new SmsException("SmsSender: caught exception in sending " + e); 304     } 305 } </pre> <p><a href="https://android.googlesource.com/platform/packages/apps/Messaging/+nougat-mr1-release/src/com/android/messaging/sms/SmsSender.java">https://android.googlesource.com/platform/packages/apps/Messaging/+nougat-mr1-release/src/com/android/messaging/sms/SmsSender.java</a></p> <pre> 56  * Class that receives incoming SMS messages through android.provider.Telephony.SMS_RECEIVED 57  * 58  * This class serves two purposes: 59  * - Process phone verification SMS messages 60  * - Handle SMS messages when the user has enabled us to be the default SMS app (Pre-KLP) 61  */ 62  public final class SmsReceiver extends BroadcastReceiver { 63      private static final String TAG = LogUtil.BUGLE_TAG; 64 65      private static ArrayList&lt;Pattern&gt; sIgnoreSmsPatterns; 66 </pre> <p><a href="https://android.googlesource.com/platform/packages/apps/Messaging/+nougat-mr1-release/src/com/android/messaging/receiver/SmsReceiver.java">https://android.googlesource.com/platform/packages/apps/Messaging/+nougat-mr1-release/src/com/android/messaging/receiver/SmsReceiver.java</a></p>

**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC
	<pre>200 public static void deliverSmsMessages(final Context context, final int subId, 201     final int errorCode, final android.telephony.SmsMessage[] messages) { 202     final ContentValues messageValues = 203         MmsUtils.parseReceivedSmsMessage(context, messages, errorCode); 204 205     LogUtil.v(TAG, "SmsReceiver.deliverSmsMessages"); 206 207     final long nowInMillis = System.currentTimeMillis(); 208     final long receivedTimestampMs = MmsUtils.getMessageDate(messages[0], nowInMillis); 209 210     messageValues.put(Sms.Inbox.DATE, receivedTimestampMs); 211     // Default to unread and unseen for us but ReceiveSmsMessageAction will override 212     // seen for the telephony db. 213     messageValues.put(Sms.Inbox.READ, 0); 214     messageValues.put(Sms.Inbox.SEEN, 0); 215     if (OsUtil.isAtLeastL_MR1()) { 216         messageValues.put(Sms.SUBSCRIPTION_ID, subId); 217     } 218 219     if (messages[0].getMessageClass() == android.telephony.SmsMessage.MessageClass.CLASS_0    220         DebugUtils.debugClassZeroSmsEnabled()) { 221         Factory.get().getUIIntents().launchClassZeroActivity(context, messageValues); 222     } else { 223         final ReceiveSmsMessageAction action = new ReceiveSmsMessageAction(messageValues); 224         action.start(); 225     } 226 }</pre> <p><a href="https://android.googlesource.com/platform/packages/apps/Messaging/+nougat-mr1-release/src/com/android/messaging/receiver/SmsReceiver.java">https://android.googlesource.com/platform/packages/apps/Messaging/+nougat-mr1-release/src/com/android/messaging/receiver/SmsReceiver.java</a></p>

**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC
	<pre>228     @Override 229     public void onReceive(final Context context, final Intent intent) { 230         LogUtil.v(TAG, "SmsReceiver.onReceive " + intent); 231         // On KLP+ we only take delivery of SMS messages in SmsDeliverReceiver. 232         if (PhoneUtils.getDefault().isSmsEnabled()) { 233             final String action = intent.getAction(); 234             if (OsUtil.isSecondaryUser() &amp;&amp; 235                 (Telephony.Sms.Intents.SMS_RECEIVED_ACTION.equals(action)    236                  // TODO: update this with the actual constant from Telephony 237                  "android.provider.Telephony.MMS_DOWNLOADED".equals(action))) { 238                 postNewMessageSecondaryUserNotification(); 239             } else if (!OsUtil.isAtLeastKLP()) { 240                 deliverSmsIntent(context, intent); 241             } 242         } 243     }</pre> <p><a href="https://android.googlesource.com/platform/packages/apps/Messaging/+/nougat-mr1-release/src/com/android/messaging/receiver/SmsReceiver.java">https://android.googlesource.com/platform/packages/apps/Messaging/+/nougat-mr1-release/src/com/android/messaging/receiver/SmsReceiver.java</a></p>

**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC
	<pre>52  * Class that sends chat message via MMS. 53  * 54  * The interface emulates a blocking send similar to making an HTTP request. 55  */ 56  public class MmsSender { 57      private static final String TAG = LogUtil.BUGLE_TAG; 58 59      /** 60       * Send an MMS message. 61       * 62       * @param context Context 63       * @param messageUri The unique URI of the message for identifying it during sending 64       * @param sendReq The SendReq PDU of the message 65       * @throws MmsFailureException 66       */ 67      public static void sendMms(final Context context, final int subId, final Uri messageUri, 68                               final SendReq sendReq, final Bundle sentIntentExtras) throws MmsFailureException { 69          sendMms(context, 70                subId, 71                messageUri, 72                null /* locationUri */, 73                sendReq, 74                true /* responseImportant */, 75                sentIntentExtras); 76      }</pre> <p><a href="https://android.googlesource.com/platform/packages/apps/Messaging/+/nougat-mr1-release/src/com/android/messaging/sms/MmsSender.java">https://android.googlesource.com/platform/packages/apps/Messaging/+/nougat-mr1-release/src/com/android/messaging/sms/MmsSender.java</a></p>

**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC
	<pre>240     * Download an MMS message. 241     * 242     * @param context Context 243     * @param contentLocation The url of the MMS message 244     * @throws MmsFailureException 245     * @throws InvalidHeaderValueException 246     */ 247     public static void downloadMms(final Context context, final int subId, 248         final String contentLocation, Bundle extras) throws MmsFailureException, 249         InvalidHeaderValueException { 250         final Uri requestUri = Uri.parse(contentLocation); 251         final Uri contentUri = MmsFileProvider.buildRawMmsUri(); 252 253         final Intent downloadedIntent = new Intent(SendStatusReceiver.MMS_DOWNLOADED_ACTION, 254             requestUri, 255             context, 256             SendStatusReceiver.class); 257         downloadedIntent.putExtra(SendMessageAction.EXTRA_CONTENT_URI, contentUri); 258         if (extras != null) { 259             downloadedIntent.putExtras(extras); 260         } 261         final PendingIntent downloadedPendingIntent = PendingIntent.getBroadcast( 262             context, 263             0 /*request code*/, 264             downloadedIntent, 265             PendingIntent.FLAG_UPDATE_CURRENT); 266 267         MmsManager.downloadMultimediaMessage(subId, context, contentLocation, contentUri, 268             downloadedPendingIntent); 269     }</pre> <p><a href="https://android.googlesource.com/platform/packages/apps/Messaging/+nougat-mr1-release/src/com/android/messaging/sms/MmsSender.java">https://android.googlesource.com/platform/packages/apps/Messaging/+nougat-mr1-release/src/com/android/messaging/sms/MmsSender.java</a></p>

**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC
	<pre> 97      * Execute an MMS HTTP request, either a POST (sending) or a GET (downloading) 98      * 99      * @param urlString The request URL, for sending it is usually the MMSC, and for downloading 100     *           it is the message URL 101     * @param pdu For POST (sending) only, the PDU to send 102     * @param method HTTP method, POST for sending and GET for downloading 103     * @param isProxySet Is there a proxy for the MMSC 104     * @param proxyHost The proxy host 105     * @param proxyPort The proxy port 106     * @param mmsConfig The MMS config to use 107     * @param userAgent The user agent header value 108     * @param uaProfUrl The UA Prof URL header value 109     * @return The HTTP response body 110     * @throws MmsHttpException For any failures 111     */ 112     public byte[] execute(String urlString, byte[] pdu, String method, boolean isProxySet, 113                          String proxyHost, int proxyPort, Bundle mmsConfig, String userAgent, String uaProfUrl) 114         throws MmsHttpException { 115         Log.d(MmsService.TAG, "HTTP: " + method + " " + Utils.redactUrlForNonVerbose(urlString) 116             + (isProxySet ? (" proxy=" + proxyHost + ":" + proxyPort) : "")) 117             + ", PDU size=" + (pdu != null ? pdu.length : 0)); 118         checkMethod(method); 119         HttpURLConnection connection = null; 120         try { 121             Proxy proxy = Proxy.NO_PROXY; 122             if (isProxySet) { 123                 proxy = new Proxy(Proxy.Type.HTTP, new InetSocketAddress(proxyHost, proxyPort)); 124             } 125             final URL url = new URL(urlString); 126             // Now get the connection 127             connection = (HttpURLConnection) url.openConnection(proxy); 128             connection.setDoInput(true); 129             connection.setConnectTimeout( 130                 mmsConfig.getInt(CarrierConfigValuesLoader.CONFIG_HTTP_SOCKET_TIMEOUT, 131                     CarrierConfigValuesLoader.CONFIG_HTTP_SOCKET_TIMEOUT_DEFAULT)); </pre> <p data-bbox="506 1219 1596 1287"><a href="https://android.googlesource.com/platform/packages/apps/Messaging/+/nougat-mr1-release/src/android/support/v7/mms/MmsHttpClient.java">https://android.googlesource.com/platform/packages/apps/Messaging/+/nougat-mr1-release/src/android/support/v7/mms/MmsHttpClient.java</a></p>

**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC
	<pre> 132 // ----- COMMON HEADERS ----- 133 // Header: Accept 134 connection.setRequestProperty(HEADER_ACCEPT, HEADER_VALUE_ACCEPT); 135 // Header: Accept-Language 136 connection.setRequestProperty( 137     HEADER_ACCEPT_LANGUAGE, getCurrentAcceptLanguage(Locale.getDefault())); 138 // Header: User-Agent 139 Log.i(MmsService.TAG, "HTTP: User-Agent=" + userAgent); 140 connection.setRequestProperty(HEADER_USER_AGENT, userAgent); 141 // Header: x-wap-profile 142 final String uaProfUrlTagName = mmsConfig.getString( 143     CarrierConfigValuesLoader.CONFIG_UA_PROF_TAG_NAME, 144     CarrierConfigValuesLoader.CONFIG_UA_PROF_TAG_NAME_DEFAULT); 145 if (uaProfUrl != null) { 146     Log.i(MmsService.TAG, "HTTP: UaProfUrl=" + uaProfUrl); 147     connection.setRequestProperty(uaProfUrlTagName, uaProfUrl); 148 } 149 // Add extra headers specified by mms_config.xml's httpparams 150 addExtraHeaders(connection, mmsConfig); 151 // Different stuff for GET and POST 152 if (METHOD_POST.equals(method)) { 153     if (pdu == null    pdu.length &lt; 1) { 154         Log.e(MmsService.TAG, "HTTP: empty pdu"); 155         throw new MmsHttpException(0/*statusCode*/, "Sending empty PDU"); 156     } 157     connection.setDoOutput(true); 158     connection.setRequestMethod(METHOD_POST); 159     if (mmsConfig.getBoolean( 160         CarrierConfigValuesLoader.CONFIG_SUPPORT_HTTP_CHARSET_HEADER, 161         CarrierConfigValuesLoader.CONFIG_SUPPORT_HTTP_CHARSET_HEADER_DEFAULT)) { 162         connection.setRequestProperty(HEADER_CONTENT_TYPE, 163             HEADER_VALUE_CONTENT_TYPE_WITH_CHARSET); 164     } else { 165         connection.setRequestProperty(HEADER_CONTENT_TYPE, 166             HEADER_VALUE_CONTENT_TYPE_WITHOUT_CHARSET); </pre> <p data-bbox="506 1256 1596 1321"><a href="https://android.googlesource.com/platform/packages/apps/Messaging/+nougat-mr1-release/src/android/support/v7/mms/MmsHttpClient.java">https://android.googlesource.com/platform/packages/apps/Messaging/+nougat-mr1-release/src/android/support/v7/mms/MmsHttpClient.java</a></p>

**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC
	<pre>167     } 168     if (Log.isLoggable(MmsService.TAG, Log.VERBOSE)) { 169         logHttpHeaders(connection.getRequestProperties()); 170     } 171     connection.setFixedLengthStreamingMode(pdu.length); 172     // Sending request body 173     final OutputStream out = 174         new BufferedOutputStream(connection.getOutputStream()); 175     out.write(pdu); 176     out.flush(); 177     out.close(); 178 } else if (METHOD_GET.equals(method)) { 179     if (Log.isLoggable(MmsService.TAG, Log.VERBOSE)) { 180         logHttpHeaders(connection.getRequestProperties()); 181     } 182     connection.setRequestMethod(METHOD_GET); 183 } 184 // Get response 185 final int responseCode = connection.getResponseCode(); 186 final String responseMessage = connection.getResponseMessage(); 187 Log.d(MmsService.TAG, "HTTP: " + responseCode + " " + responseMessage); 188 if (Log.isLoggable(MmsService.TAG, Log.VERBOSE)) { 189     logHttpHeaders(connection.getHeaderFields()); 190 } 191 if (responseCode / 100 != 2) { 192     throw new MmsHttpException(responseCode, responseMessage); 193 } 194 final InputStream in = new BufferedInputStream(connection.getInputStream()); 195 final ByteArrayOutputStream byteOut = new ByteArrayOutputStream(); 196 final byte[] buf = new byte[4096]; 197 int count = 0; 198 while ((count = in.read(buf)) &gt; 0) { 199     byteOut.write(buf, 0, count); 200 } 201 in.close(); 202 final byte[] responseBody = byteOut.toByteArray(); 203 Log.d(MmsService.TAG, "HTTP: response size=" 204     + (responseBody != null ? responseBody.length : 0)); 205 return responseBody;</pre> <p><a href="https://android.googlesource.com/platform/packages/apps/Messaging/+nougat-mr1-release/src/android/support/v7/mms/MmsHttpClient.java">https://android.googlesource.com/platform/packages/apps/Messaging/+nougat-mr1-release/src/android/support/v7/mms/MmsHttpClient.java</a></p>



**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC
	<pre>38  * Request to send an MMS 39  */ 40  class SendRequest extends MmsRequest { 41      // Max send response PDU size in bytes (exceeding this may cause problem with 42      // system intent delivery). 43      private static final int MAX_SEND_RESPONSE_SIZE = 1000 * 1024; 44 45      private byte[] mPduData; 46 47      SendRequest(final String locationUri, final Uri pduUri, final PendingIntent sentIntent) { 48          super(locationUri, pduUri, sentIntent); 49      } 50 51      @Override 52      protected boolean loadRequest(final Context context, final Bundle mmsConfig) { 53          mPduData = readPduFromContentUri( 54              context, 55              mPduUri, 56              mmsConfig.getInt( 57                  CarrierConfigValuesLoader.CONFIG_MAX_MESSAGE_SIZE, 58                  CarrierConfigValuesLoader.CONFIG_MAX_MESSAGE_SIZE_DEFAULT)); 59          return (mPduData != null); 60      } 61 62      @Override 63      protected boolean transferResponse(final Context context, final Intent fillIn, 64          final byte[] response) { 65          // SendConf pdus are always small and can be included in the intent 66          if (response != null &amp;&amp; fillIn != null) { 67              if (response.length &gt; MAX_SEND_RESPONSE_SIZE) { 68                  // If the response PDU is too large, it won't be able to fit in 69                  // the PendingIntent to be transferred via system IPC. 70                  return false; 71              } 72              fillIn.putExtra(SmsManager.EXTRA_MMS_DATA, response); 73          } 74          return true; 75      } }</pre>

**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC
	<p><a href="https://android.googlesource.com/platform/packages/apps/Messaging/+/nougat-mr1-release/src/android/support/v7/mms/SendRequest.java">https://android.googlesource.com/platform/packages/apps/Messaging/+/nougat-mr1-release/src/android/support/v7/mms/SendRequest.java</a></p> <pre data-bbox="533 354 1738 397">public static LocationRequest create ()</pre> <p>Create a location request with default parameters.</p> <p>Default parameters are for a block accuracy, slowly updated location. It can then be adjusted as required by the applications before passing to the <code>FusedLocationProviderApi</code>.</p> <p><b>Returns</b></p> <ul data-bbox="554 613 814 636" style="list-style-type: none"> <li>• a new location request</li> </ul> <p><a href="https://developers.google.com/android/reference/com/google/android/gms/location/LocationRequest">https://developers.google.com/android/reference/com/google/android/gms/location/LocationRequest</a></p>

**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC
	<p><code>public static final int PRIORITY_BALANCED_POWER_ACCURACY</code></p> <p>Used with <code>setPriority(int)</code> to request "block" level accuracy.</p> <p>Block level accuracy is considered to be about 100 meter accuracy. Using a coarse accuracy such as this often consumes less power.</p> <p>Constant Value: 102</p> <hr/> <p><code>public static final int PRIORITY_HIGH_ACCURACY</code></p> <p>Used with <code>setPriority(int)</code> to request the most accurate locations available.</p> <p>This will return the finest location available.</p> <p>Constant Value: 100</p> <hr/> <p><code>public static final int PRIORITY_LOW_POWER</code></p> <p>Used with <code>setPriority(int)</code> to request "city" level accuracy.</p> <p>City level accuracy is considered to be about 10km accuracy. Using a coarse accuracy such as this often consumes less power.</p> <p>Constant Value: 104</p> <p><a href="https://developers.google.com/android/reference/com/google/android/gms/location/LocationRequest">https://developers.google.com/android/reference/com/google/android/gms/location/LocationRequest</a></p>

**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC
	<pre data-bbox="533 248 1749 285">public Task&lt;Location&gt; getLastLocation ()</pre> <p data-bbox="527 315 1104 337">Returns the best most recent location currently available.</p> <p data-bbox="527 371 1696 430">If a location is not available, which should happen very rarely, null will be returned. The best accuracy available while respecting the location permissions will be returned.</p> <p data-bbox="527 464 1736 522">This method provides a simplified way to get location. It is particularly well suited for applications that do not require an accurate location and that do not want to maintain extra logic for location updates.</p> <pre data-bbox="533 578 1749 615">public Task&lt;LocationAvailability&gt; getLocationAvailability ()</pre> <p data-bbox="527 646 1692 704">Returns the availability of location data. When <code>isLocationAvailable()</code> returns true, then the location returned by <code>getLastLocation()</code> will be reasonably up to date within the hints specified by the active <code>LocationRequest</code>s.</p> <p data-bbox="527 738 1472 761">If the client isn't connected to Google Play services and the request times out, null is returned.</p> <p data-bbox="527 795 1673 854">Note it's always possible for <code>getLastLocation()</code> to return null even when this method returns true (e.g. location settings were disabled between calls).</p> <p data-bbox="512 865 1900 930"><a href="https://developers.google.com/android/reference/com/google/android/gms/location/FusedLocationProviderClient">https://developers.google.com/android/reference/com/google/android/gms/location/FusedLocationProviderClient</a></p>

**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC						
	<p data-bbox="527 245 1749 326"> <code>public Task&lt;Void&gt; requestLocationUpdates (LocationRequest request, LocationCallback callback, Looper looper)</code> </p> <p data-bbox="527 354 1272 378">Requests location updates with a callback on the specified Looper thread.</p> <p data-bbox="527 410 1686 472">This method is suited for the foreground use cases. For background use cases, the <code>PendingIntent</code> version of the method is recommended, see <code>requestLocationUpdates(LocationRequest, PendingIntent)</code>.</p> <p data-bbox="527 505 1371 529">Any previous LocationRequests registered on this LocationListener will be replaced.</p> <p data-bbox="527 561 1686 656">This call will keep the Google Play services connection active, so make sure to call <code>removeLocationUpdates(LocationCallback)</code> when you no longer need it, otherwise you lose the benefits of the automatic connection management.</p> <p data-bbox="527 688 1745 712">Callbacks for <code>LocationCallback</code> will be made on the specified thread, which must already be a prepared looper thread.</p> <p data-bbox="527 737 667 761"><b>Parameters</b></p> <table border="1" data-bbox="527 789 1749 1008"> <tbody> <tr> <td data-bbox="527 797 625 854"><b>request</b></td> <td data-bbox="632 797 1749 854">The location request for the updates.</td> </tr> <tr> <td data-bbox="527 862 625 919"><b>callback</b></td> <td data-bbox="632 862 1749 919">The callback for the location updates.</td> </tr> <tr> <td data-bbox="527 927 625 1000"><b>looper</b></td> <td data-bbox="632 927 1749 1000">The Looper object whose message queue will be used to implement the callback mechanism, or null to make callbacks on the calling thread.</td> </tr> </tbody> </table> <p data-bbox="527 1024 1902 1089"> <a href="https://developers.google.com/android/reference/com/google/android/gms/location/FusedLocationProviderClient">https://developers.google.com/android/reference/com/google/android/gms/location/FusedLocationProviderClient</a> </p>	<b>request</b>	The location request for the updates.	<b>callback</b>	The callback for the location updates.	<b>looper</b>	The Looper object whose message queue will be used to implement the callback mechanism, or null to make callbacks on the calling thread.
<b>request</b>	The location request for the updates.						
<b>callback</b>	The callback for the location updates.						
<b>looper</b>	The Looper object whose message queue will be used to implement the callback mechanism, or null to make callbacks on the calling thread.						

**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC				
	<p data-bbox="533 240 1740 321"><code>public Task&lt;Void&gt; requestLocationUpdates (LocationRequest request, PendingIntent callbackIntent)</code></p> <p data-bbox="533 354 1268 378">Requests location updates with a callback on the specified PendingIntent.</p> <p data-bbox="533 410 1734 537">This method is suited for the background use cases, more specifically for receiving location updates, even when the app has been killed by the system. In order to do so, use a <code>PendingIntent</code> for a started service. For foreground use cases, the <code>LocationCallback</code> version of the method is recommended, see <code>requestLocationUpdates(LocationRequest, LocationCallback, Looper)</code>.</p> <p data-bbox="533 570 1724 630">Any previously registered requests that have the same PendingIntent (as defined by <code>equals(Object)</code>) will be replaced by this request.</p> <p data-bbox="533 662 1730 755">Both <code>LocationResult</code> and <code>LocationAvailability</code> are sent to the given PendingIntent. You can extract data from an Intent using <code>hasResult(Intent)</code>, <code>extractResult(Intent)</code>, <code>hasLocationAvailability(Intent)</code>, and <code>extractLocationAvailability(Intent)</code>.</p> <p data-bbox="533 781 667 805"><b>Parameters</b></p> <table border="1" data-bbox="533 833 1740 971"> <tbody> <tr> <td data-bbox="533 833 835 902"><code>request</code></td> <td data-bbox="835 833 1740 902">The location request for the updates.</td> </tr> <tr> <td data-bbox="533 902 835 971"><code>callbackIntent</code></td> <td data-bbox="835 902 1740 971">A pending intent to be sent for each location update.</td> </tr> </tbody> </table> <p data-bbox="533 997 632 1021"><b>Returns</b></p> <ul data-bbox="533 1045 1360 1070" style="list-style-type: none"> <li>• a Task for the call, check <code>isSuccessful()</code> to determine if it was successful.</li> </ul> <p data-bbox="533 1078 1900 1144"><a href="https://developers.google.com/android/reference/com/google/android/gms/location/FusedLocationProviderClient">https://developers.google.com/android/reference/com/google/android/gms/location/FusedLocationProviderClient</a></p>	<code>request</code>	The location request for the updates.	<code>callbackIntent</code>	A pending intent to be sent for each location update.
<code>request</code>	The location request for the updates.				
<code>callbackIntent</code>	A pending intent to be sent for each location update.				

**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC						
	<p><code>public void onLocationAvailability (LocationAvailability locationAvailability)</code></p> <p>Called when there is a change in the availability of location data.</p> <p>When <code>isLocationAvailable()</code> returns <code>false</code> you can assume that location will not be returned in <code>onLocationResult(LocationResult)</code> until something changes in the device's settings or environment. Even when <code>isLocationAvailable()</code> returns <code>true</code> the <code>onLocationResult(LocationResult)</code> may not always be called regularly, however the device location is known and both the most recently delivered location and <code>getLastLocation(GoogleApiClient)</code> will be reasonably up to date given the hints specified by the active <code>LocationRequest</code>s.</p> <p><b>Parameters</b></p> <table border="1" data-bbox="527 639 1738 704"> <tr> <td><code>locationAvailability</code></td> <td>The current status of location availability.</td> </tr> </table> <p><code>public void onLocationResult (LocationResult result)</code></p> <p>Called when device location information is available.</p> <p>The most recent location returned by <code>getLastLocation()</code> is not guaranteed to be immediately fresh, but will be reasonably up to date given the hints specified by the active <code>LocationRequest</code>s.</p> <p><b>Parameters</b></p> <table border="1" data-bbox="527 1019 1738 1084"> <tr> <td><code>result</code></td> <td>The latest location result available.</td> </tr> </table> <p><a href="https://developers.google.com/android/reference/com/google/android/gms/location/LocationCallback">https://developers.google.com/android/reference/com/google/android/gms/location/LocationCallback</a></p> <p><code>public abstract void onLocationChanged (Location location)</code></p> <p>Called when the location has changed.</p> <p><b>Parameters</b></p> <table border="1" data-bbox="527 1318 1738 1383"> <tr> <td><code>location</code></td> <td>The updated location.</td> </tr> </table>	<code>locationAvailability</code>	The current status of location availability.	<code>result</code>	The latest location result available.	<code>location</code>	The updated location.
<code>locationAvailability</code>	The current status of location availability.						
<code>result</code>	The latest location result available.						
<code>location</code>	The updated location.						

**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

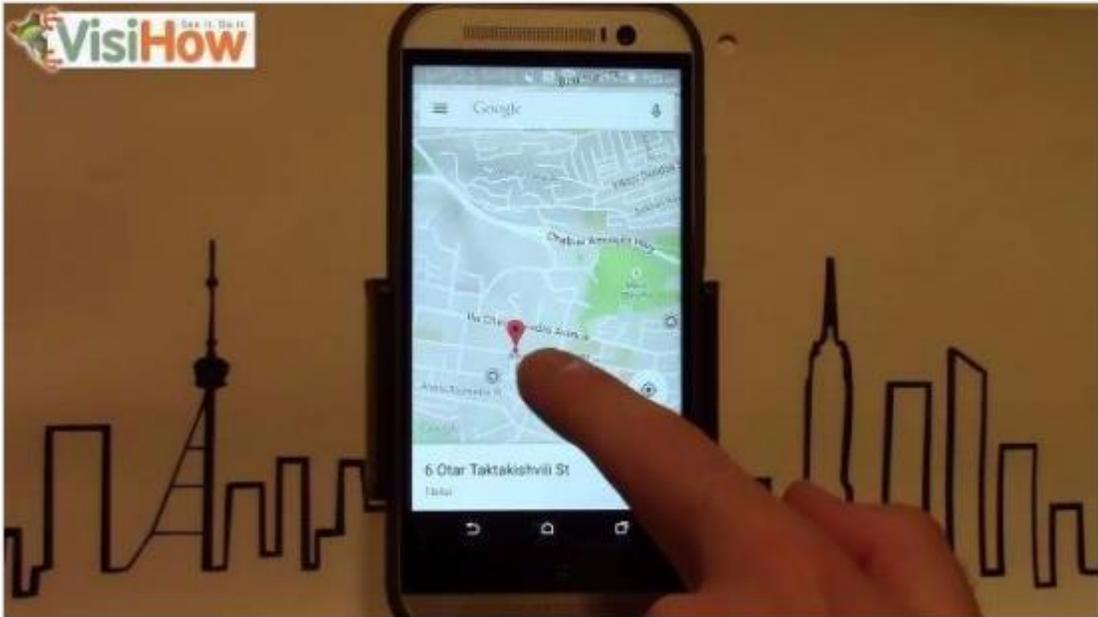
US9408055B2	HTC
	<p><a href="https://developers.google.com/android/reference/com/google/android/gms/location/LocationListener">https://developers.google.com/android/reference/com/google/android/gms/location/LocationListener</a></p> <p>Public Constructors</p> <hr/> <p>public <b>MapView</b> (<a href="#">Context</a> context)</p> <p>public <b>MapView</b> (<a href="#">Context</a> context, <a href="#">AttributeSet</a> attrs)</p> <p>public <b>MapView</b> (<a href="#">Context</a> context, <a href="#">AttributeSet</a> attrs, int defStyle)</p> <p>public <b>MapView</b> (<a href="#">Context</a> context, <a href="#">GoogleMapOptions</a> options)</p> <p><a href="https://developers.google.com/android/reference/com/google/android/gms/maps/MapView">https://developers.google.com/android/reference/com/google/android/gms/maps/MapView</a></p>



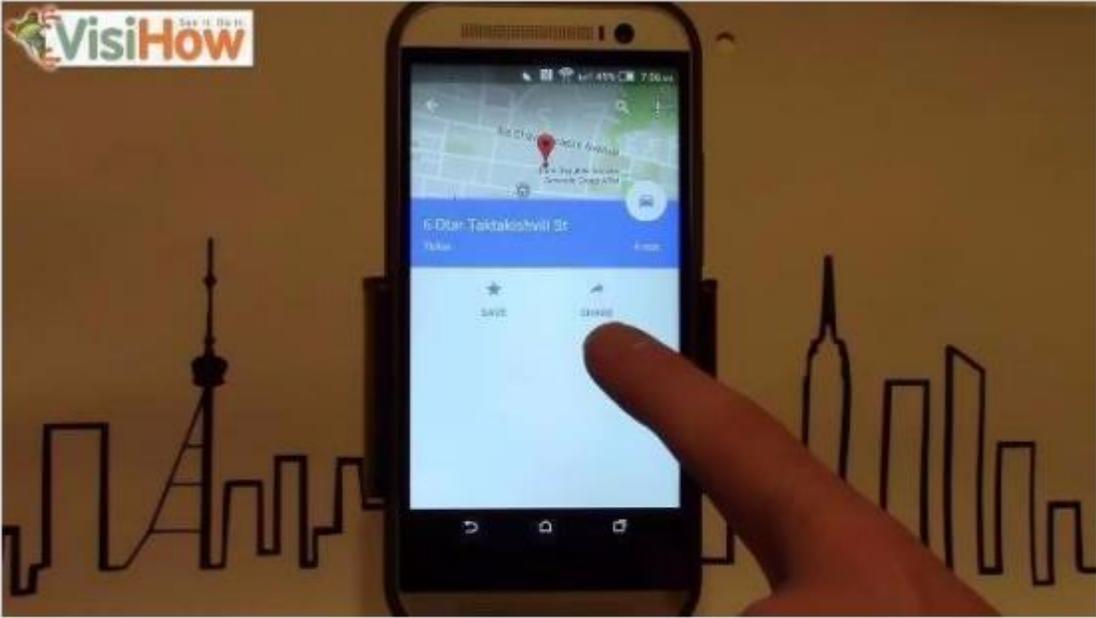
**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC		
	<pre>public void getMapAsync (OnMapReadyCallback callback)</pre> <p>Returns a non-null instance of the <code>GoogleMap</code>, ready to be used.</p> <p>Note that:</p> <ul style="list-style-type: none"> <li>• This method must be called from the main thread.</li> <li>• The callback will be executed in the main thread.</li> <li>• In the case where Google Play services is not installed on the user's device, the callback will not be triggered until the user installs it.</li> <li>• The <code>GoogleMap</code> object provided by the callback is non-null.</li> </ul> <p><b>Parameters</b></p> <table border="1"> <tr> <td data-bbox="527 688 690 753"><b>callback</b></td> <td data-bbox="690 688 1738 753">The callback object that will be triggered when the map is ready to be used.</td> </tr> </table> <pre>public final void onCreate (Bundle savedInstanceState)</pre> <p>You must call this method from the parent Activity/Fragment's corresponding method.</p> <p><a href="https://developers.google.com/android/reference/com/google/android/gms/maps/MapView">https://developers.google.com/android/reference/com/google/android/gms/maps/MapView</a></p>	<b>callback</b>	The callback object that will be triggered when the map is ready to be used.
<b>callback</b>	The callback object that will be triggered when the map is ready to be used.		
<p>[54H] receiving user input via user interaction with the interactive display of the first device, the user input specifying a location and a symbol corresponding to an entity other than the first device and the second devices;</p>	<p>HTC infringes directly and/or indirectly by performing, inducing others to perform, and/or contributing to the performance of receiving user input via user interaction with the interactive display of the first device, the user input specifying a location and a symbol corresponding to an entity other than the first device and the second devices. See claims 1[G], 28[G], and 41[G], which are incorporated herein by reference in their entirety.</p> <p><b>Share Locations from Google Maps on HTC One M8</b></p>		

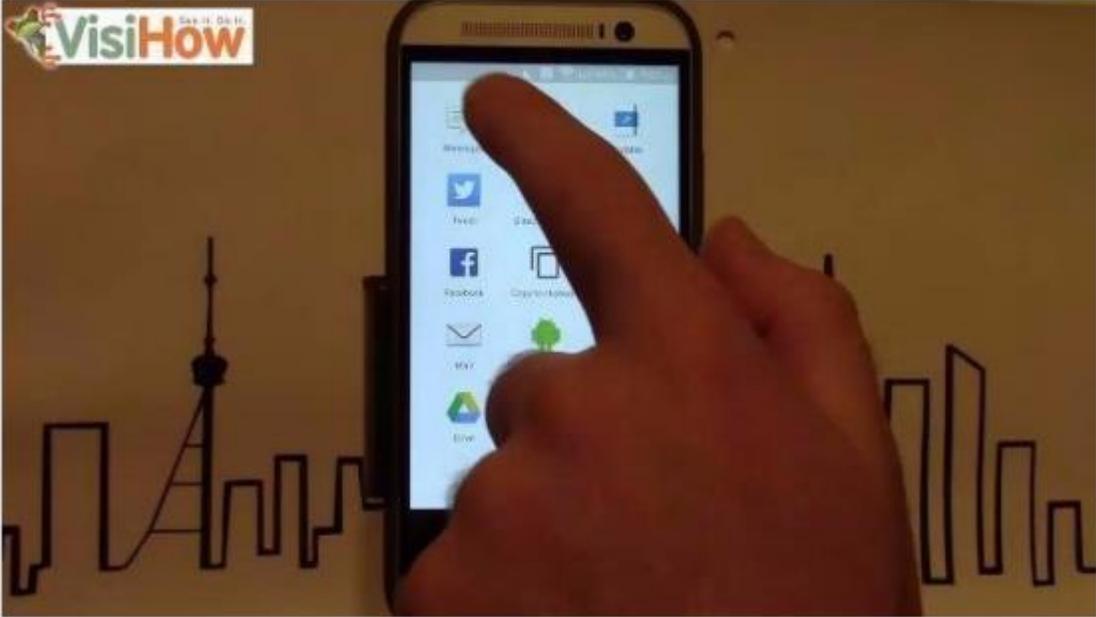
**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC
	<p data-bbox="531 228 827 256"><b>Place a pin on the map.</b></p> <p data-bbox="531 269 1638 380">We can do this anywhere in the entire world, anywhere, on any map that we find in the Google Maps application. We are going to choose a location, press down on the screen, and hold down until we can see that a red pin has been dropped.</p> 

**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC
	<p data-bbox="527 240 1207 305"><b>We have two options:</b> "SAVE" and "SHARE". Press down on the word "SHARE".</p>  <hr/> <p data-bbox="527 1057 1633 1206"><b>A menu appears with all of the different ways that we can share this location.</b> We can share this location via messages, Bluetooth, Twitter, Facebook, mail, Gmail and Viber. If you have other applications, such as WhatsApp, or any other chat or email applications, you will be able to use them as well and will see them here.</p>

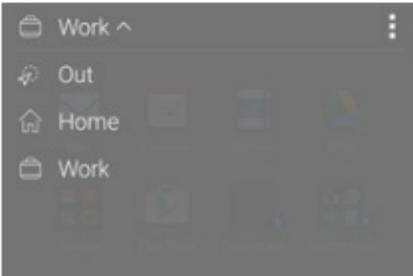
**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC
	<p data-bbox="527 237 856 264"><b>Let's choose "Messages".</b></p> <p data-bbox="527 277 1629 347">This option is located on the top left-hand corner of the screen. All options will have the same basic processes. They will ask us to send the location. Press down on the "Messages" icon.</p> 

**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC
	<p data-bbox="520 233 1260 256"><b>Press the box next to the contact who will be the recipient.</b></p> <p data-bbox="520 269 1549 292">A checkmark will appear in the box. We can select multiple contacts as well if we desire.</p>  <p data-bbox="520 1045 911 1068"><b>Press on the word "DONE (1)".</b></p> <p data-bbox="520 1084 1629 1153">This icon is located in the bottom right-hand corner of the screen. The bracketed number indicates the number of contacts we selected.</p> <p data-bbox="520 1166 1407 1188"><a href="http://visihow.com/Share_Locations_from_Google_Maps_on_HTC_One_M8">http://visihow.com/Share_Locations_from_Google_Maps_on_HTC_One_M8</a></p>

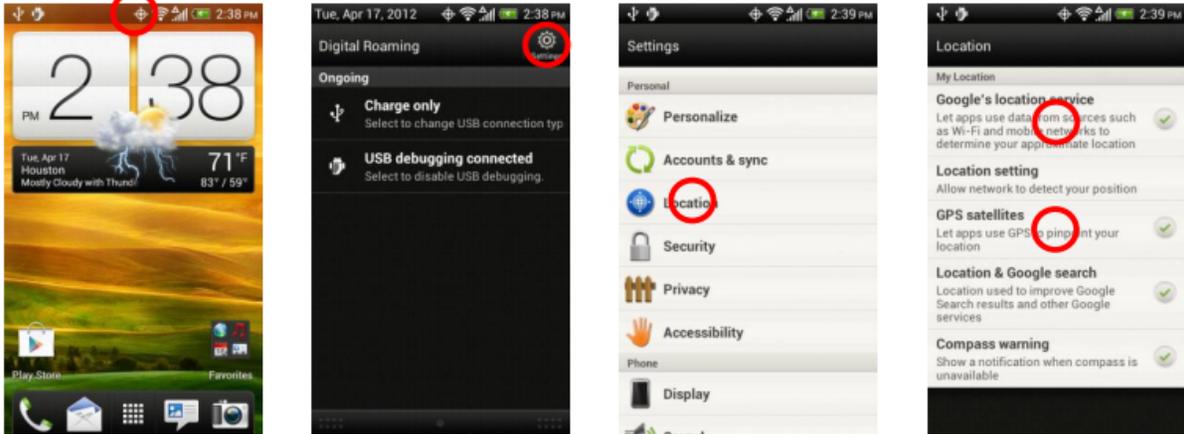
## Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products

US9408055B2	HTC
	<h3 data-bbox="520 261 1115 310">Manually switching locations</h3> <p data-bbox="520 358 1409 467">The HTC Sense Home widget automatically changes locations based on where you are. You can also manually change the location in the HTC Sense Home widget.</p> <p data-bbox="520 505 1398 613">For the HTC Sense Home widget to change locations automatically, you need to make sure that location services is turned on. See <a href="#">Turning location services on or off</a>.</p> <ol data-bbox="583 651 1388 776" style="list-style-type: none"><li>1. On your Home screen, slide right or left until you see the HTC Sense Home widget.</li><li>2. Tap , and then tap the location you want.</li></ol>  <p data-bbox="527 1101 1409 1128">HTC One (M8) - Manually switching locations - Support   HTC United States</p>

**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC
	<h2 data-bbox="548 269 1331 318">Setting your home and work locations</h2> <p data-bbox="548 367 1415 435">In the HTC Sense Home widget, set your home and work locations based on your address, Wi-Fi network, or both.</p> <p data-bbox="548 475 1425 621">You can associate multiple addresses and Wi-Fi networks to each of these locations. Using your set addresses or Wi-Fi networks, the HTC Sense Home widget will be able to determine where you are and display the appropriate apps.</p> <ol data-bbox="621 662 1457 1146" style="list-style-type: none"><li data-bbox="621 662 1423 727">1. On the Home screen, swipe right or left until you see the HTC Sense Home widget.</li><li data-bbox="621 751 972 784">2. Tap  &gt;  &gt; <b>Set locations</b>.</li><li data-bbox="621 816 1062 849">3. Choose the location you want to set.</li><li data-bbox="621 865 1457 1092">4. Tap  and do one of the following:<ul data-bbox="701 930 1457 1092" style="list-style-type: none"><li data-bbox="701 930 1444 995">• Tap <b>Address</b> and then enter your street address or select it on the map.</li><li data-bbox="701 1027 1457 1092">• Tap <b>Wi-Fi network</b> and select one or more Wi-Fi networks you want to associate with the location.</li></ul></li><li data-bbox="621 1117 1457 1146">5. When you've finished setting your home and work locations, press .</li></ol> <p data-bbox="520 1182 1503 1214">HTC One (M8) - Setting your home and work locations - Support   HTC United States</p>

## Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products

US9408055B2	HTC
	<p data-bbox="533 233 1304 272"><b>HTC One V™ – Google Location Service &amp; GPS</b></p> <p data-bbox="533 318 1717 423">Google Maps lets you track your current location, view real-time traffic situations, and receive detailed directions to your destination. It also provides a search tool where you can locate a place of interest or an address on a vector or aerial map, or view locations in street level.</p> <p data-bbox="533 469 905 496"><b>Turning on Location Services</b></p> <div data-bbox="533 509 1717 943"></div> <ol data-bbox="533 976 1717 1073" style="list-style-type: none"><li>1. From the Home Screen, slide the <b>Notifications</b> panel open.</li><li>2. In the top right corner, tap <b>Settings</b>.</li><li>3. Tap <b>Location</b>.</li><li>4. Make your selection by tapping <b>Google's location service</b>, <b>Use GPS satellites</b>, or both.</li></ol> <p data-bbox="1461 1078 1717 1146"><b>Note:</b> You will need to accept the location consent terms and conditions.</p> <p data-bbox="512 1203 1906 1414"><b>Regarding Google Maps,</b> Google Latitude, Google Plus, Google Hangouts, Google Messages, and Android Messenger, the Accused Products are configured to allow a user of the first device to interact with the display, to select a user, contact, or device, and to select an action to be performed, such as: sending a message, initiating a call, initiating a data conference, sharing a location, stop sharing a location, block a user from location sharing, sending a location, requesting a location, or sending other data. When an action is specified, data is sent from the first device to the second device via a server. In an example, using Google</p>



**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC
	<p>Maps, a user can interact with the display to specify a location that does not correspond to the first or second devices. Alternatively, a user can share an ETA which includes a route, where the route is not the location of any of the first or second devices.</p> <p>Selection with Markers: <a href="https://developers.google.com/maps/documentation/android-api/marker">https://developers.google.com/maps/documentation/android-api/marker</a></p> <p>Queries with GeoTagging database: <a href="https://developers.google.com/android/reference/com/google/android/gms/location/places/GeoDataClient">https://developers.google.com/android/reference/com/google/android/gms/location/places/GeoDataClient</a></p>

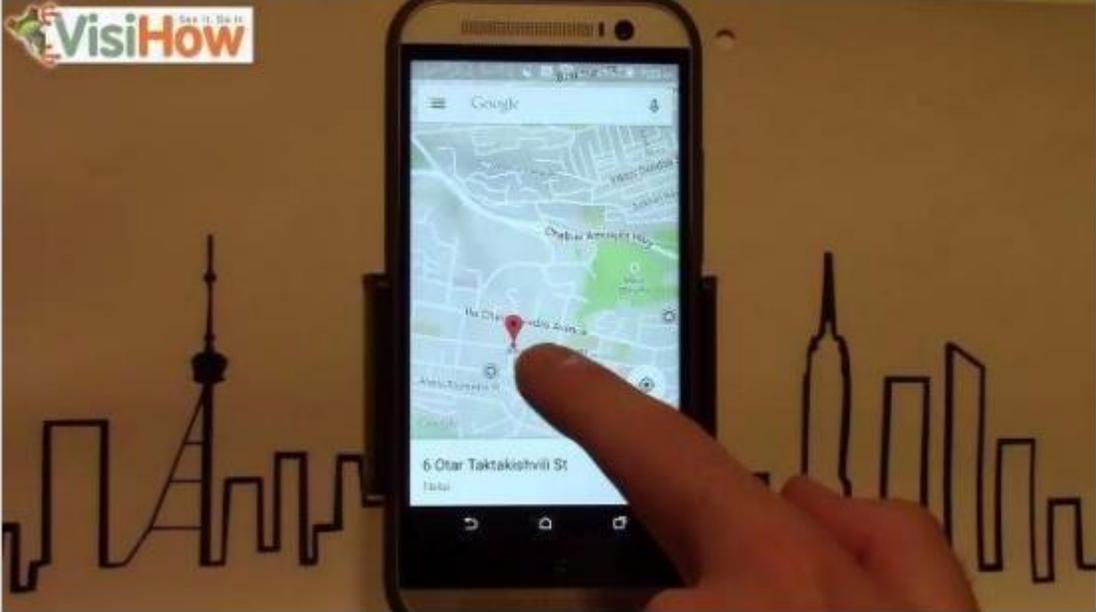
## Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products

US9408055B2	HTC
	<h3 data-bbox="541 240 1255 289">Embed a map or share a location</h3> <p data-bbox="541 310 1518 386">On your computer, you can embed a map, Street View image, driving directions, or search into your website or blog. On your computer, phone, or tablet, you can share a location or map with others over email, Google+, Facebook, Twitter, or text.</p> <p data-bbox="556 462 947 483"><a href="#">ANDROID</a> <a href="#">COMPUTER</a> <a href="#">IPHONE &amp; IPAD</a></p> <hr data-bbox="541 508 1528 511"/> <h3 data-bbox="541 560 850 592">Share a map or location</h3> <ol data-bbox="550 609 1228 771" style="list-style-type: none"><li>1. Open the Google Maps app .</li><li>2. Search for a place. Or find a place on the map then touch and hold to drop a pin.</li><li>3. At the bottom, tap the place's name or address.</li><li>4. Tap Share . If you don't see this, tap More  &gt; <b>Share</b>.</li><li>5. Select an app. It'll send a link that shows the place in Google Maps.</li></ol> <p data-bbox="520 800 1640 833"><a href="https://support.google.com/maps/answer/144361?co=GENIE.Platform%3DAndroid&amp;hl=en">https://support.google.com/maps/answer/144361?co=GENIE.Platform%3DAndroid&amp;hl=en</a></p> <h3 data-bbox="541 893 735 925">Share your E.T.A</h3> <p data-bbox="541 941 1461 963">After you start your drive, you can share your destination, estimated arrival time, and where you are on the route.</p> <ol data-bbox="550 982 1207 1169" style="list-style-type: none"><li>1. Open the Google Maps app .</li><li>2. Set a driving destination. <a href="#">Learn how to navigate to a place.</a></li><li>3. After you start navigation, tap <b>More</b>  &gt; <b>Share trip progress</b>.</li><li>4. Choose a person from the list.</li><li>5. Tap <b>Share</b>.</li><li>6. Location Sharing will stop when you reach your destination or stop navigating.</li></ol> <p data-bbox="541 1188 1083 1209">• To stop sharing before you arrive, tap <b>More</b>  &gt; <b>Stop sharing</b>.</p> <p data-bbox="510 1214 1698 1247"><a href="https://support.google.com/maps/answer/7326816?co=GENIE.Platform%3DAndroid&amp;hl=en">https://support.google.com/maps/answer/7326816?co=GENIE.Platform%3DAndroid&amp;hl=en</a></p> <p data-bbox="510 1287 1520 1320">Markers (adding location information to the link associated with the database):</p> <div data-bbox="510 1360 1913 1395" style="background-color: black; height: 20px;"></div>

**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC
	<pre>static final LatLng PERTH = new LatLng(-31.90, 115.86); Marker perth = mMap.addMarker(new MarkerOptions()     .position(PERTH)     .draggable(true));</pre>
<p>[54I] and based on the user input, adding the user-specified symbol to the interactive display at a position on the interactive map corresponding to the user-specified location, and transmitting the user-specified symbol and location to the second devices for addition of the user-specified symbol to respective interactive displays of the second devices at respective positions on respective interactive maps corresponding to the user-specified location.</p>	<p>HTC infringes directly and/or indirectly by performing, inducing others to perform, and/or contributing to the performance of [] based on the user input, adding the user-specified symbol to the interactive display at a position on the interactive map corresponding to the user-specified location, and transmitting the user-specified symbol and location to the second devices for addition of the user-specified symbol to respective interactive displays of the second devices at respective positions on respective interactive maps corresponding to the user-specified location. See claims 1[H], 28[H], and 41[H], which are incorporated herein by reference in their entirety.</p> <p>A user can interact with the display to specify a location that does not correspond to the first or second devices. A user can drop a symbol pin on the specified location. A user can then share that location and transmit the location to one or more second devices using Android Messages, Google Hangouts, or another application.</p> <p>Alternatively, a user can share an ETA which includes a route, where the route is not the location of any of the first or second devices. Again, this route can be shared with users over Android Messages, Google Hangouts, or another application.</p> <p><b>Share Locations from Google Maps on HTC One M8</b></p>

**Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products**

US9408055B2	HTC
	<p data-bbox="531 233 827 256"><b>Place a pin on the map.</b></p> <p data-bbox="531 272 1640 380">We can do this anywhere in the entire world, anywhere, on any map that we find in the Google Maps application. We are going to choose a location, press down on the screen, and hold down until we can see that a red pin has been dropped.</p>  <p data-bbox="512 1068 743 1091">Placing a Marker:</p> <p data-bbox="512 1107 1430 1130"><a href="https://developers.google.com/maps/documentation/android-api/marker">https://developers.google.com/maps/documentation/android-api/marker</a></p> <p data-bbox="512 1179 1083 1201">based on queries with GeoTagging database:</p> <p data-bbox="512 1218 1860 1240"><a href="https://developers.google.com/android/reference/com/google/android/gms/location/places/GeoDataClient">https://developers.google.com/android/reference/com/google/android/gms/location/places/GeoDataClient</a></p>

## Exhibit B for U.S. Patent No. 9,408,055 Against HTC Accused Products

US9408055B2	HTC
	<h3 data-bbox="541 233 1241 280">Embed a map or share a location</h3> <p data-bbox="541 302 1499 375">On your computer, you can embed a map, Street View image, driving directions, or search into your website or blog. On your computer, phone, or tablet, you can share a location or map with others over email, Google+, Facebook, Twitter, or text.</p> <p data-bbox="554 451 936 470"><a href="#">ANDROID</a> <a href="#">COMPUTER</a> <a href="#">IPHONE &amp; IPAD</a></p> <hr data-bbox="541 495 1499 498"/> <h3 data-bbox="541 548 842 576">Share a map or location</h3> <ol data-bbox="548 597 1213 753" style="list-style-type: none"><li>1. Open the Google Maps app .</li><li>2. Search for a place. Or find a place on the map then touch and hold to drop a pin.</li><li>3. At the bottom, tap the place's name or address.</li><li>4. Tap Share . If you don't see this, tap More  &gt; <b>Share</b>.</li><li>5. Select an app. It'll send a link that shows the place in Google Maps.</li></ol> <p data-bbox="520 786 1633 813"><a href="https://support.google.com/maps/answer/144361?co=GENIE.Platform%3DAndroid&amp;hl=en">https://support.google.com/maps/answer/144361?co=GENIE.Platform%3DAndroid&amp;hl=en</a></p> <h3 data-bbox="527 857 768 885">Share your E.T.A</h3> <p data-bbox="527 915 1667 940">After you start your drive, you can share your destination, estimated arrival time, and where you are on the route.</p> <ol data-bbox="533 967 1356 1192" style="list-style-type: none"><li>1. Open the Google Maps app .</li><li>2. Set a driving destination. <a href="#">Learn how to navigate to a place.</a></li><li>3. After you start navigation, tap <b>More</b>  &gt; <b>Share trip progress</b>.</li><li>4. Choose a person from the list.</li><li>5. Tap <b>Share</b>.</li><li>6. Location Sharing will stop when you reach your destination or stop navigating.</li></ol> <ul data-bbox="533 1219 1199 1243" style="list-style-type: none"><li>• To stop sharing before you arrive, tap <b>More</b>  &gt; <b>Stop sharing</b>.</li></ul>

**Exhibit C for U.S. Patent No. 9,445,251 Against HTC Accused Products**

In these Infringement Contentions, AGIS Software Development LLC (“AGIS”) contends that at least the following claims of U.S. Patent No. 9,445,251 (the “251 Patent”) identified below are infringed by the Accused Products (*e.g.*, phones and tablets) which are manufactured, sold, offered for sale, and/or used by HTC Corporation (“HTC”).

The Accused Products comprise HTC products running the Android mobile operating system and manufactured, used, or sold during and after 2011. For example, the Accused Products comprise the following Android-based phones and tablets: 10, 10 evo, 10 Lifestyle, 2125 / 2100 (Faraday), 3125 / Smartflip / 8500 (Star Trek), 5800 / Fusion / S720, 7 Mozart, 7 Pro, 7 Surround, 7 Trophy, 8125 / 8100 / MDA (USA) / K-JAM / P4300 (Wizard), 8XT, Amaze 4G, Aria, Arrive, Arrive / 7 Pro (CDMA), Bolt, Butterfly, Butterfly 2, Butterfly 3, Butterfly S, ChaCha, Dash / S620 / S621 (Excalibur), Dash 3G / Snap (GSM), Desire, Desire (CDMA), Desire / Desire 601 (CDMA), Desire 10 Compact, Desire 10 Lifestyle, Desire 10 Pro, Desire 200, Desire 210 dual sim, Desire 300, Desire 310, Desire 310 dual sim, Desire 320, Desire 326G dual sim, Desire 400 dual sim, Desire 500, Desire 501, Desire 501 dual sim, Desire 510, Desire 510 (CDMA), Desire 510 (GSM), Desire 516 dual sim, Desire 520, Desire 526, Desire 526 (CDMA), Desire 526G+ dual sim, Desire 530, Desire 555, Desire 600 dual sim, Desire 601, Desire 601 dual sim, Desire 610, Desire 610 (GSM), Desire 612, Desire 612 (CDMA), Desire 616 dual sim, Desire 620, Desire 620G dual sim, Desire 625, Desire 626, Desire 626 (CDMA), Desire 626 (GSM), Desire 626 (USA), Desire 626G+, Desire 626s, Desire 626s (CDMA), Desire 626s (GSM), Desire 628, Desire 630, Desire 650, Desire 700, Desire 700 dual sim, Desire 728 dual sim, Desire 728 Ultra, Desire 816, Desire 816 dual sim, Desire 816G dual sim, Desire 820, Desire 820 dual sim, Desire 820G+ dual sim, Desire 820q dual sim, Desire 820s dual sim, Desire 825, Desire 826 dual sim, Desire 828 dual sim, Desire 830, Desire C, Desire C (CDMA), Desire Eye, Desire HD, Desire L, Desire P, Desire Q, Desire S, Desire SV, Desire U, Desire V, Desire VC, Desire VT, Desire X, Desire XC, Desire Z, Dream, DROID DNA, DROID ERIS, Droid Incredible, DROID Incredible 2, DROID Incredible 4G LTE, EVO 3D, EVO 3D CDMA, Evo 4G, Evo 4G LTE, Evo 4G+, EVO Design 4G, EVO Design 4G / Hero S (CDMA), EVO Shift 4G, EVO V 4G / EVO 3D (CDMA), EVO View 4G, Explorer, First, Flyer, Flyer Wi-Fi, Freestyle, Fuze / Touch Pro (GSM), G1, G2, Glacier, Gratia, HD mini, HD2, HD7, HD7 / HD7S, HD7S, Hero, Hero (CDMA), Hero S, Imagio, Incredible S, Inspire 4G, J, JAMin / S200 (Prophet), Jetstream, Lead, Legend, Magic, MAX 4G, MDA Compact / xda II mini / JAM (Magician), Merge, Mogul / XV6800 / PPC6800 / P4000, myTouch 3G / Magic, myTouch 3G Slide, myTouch 4G, myTouch 4G Slide, One, One (E8), One (E8) CDMA, One (M7 / CDMA), One (M7 / GSM), One (M8 Eye), One (M8), One (M8) (CDMA), One (M8) (GSM), One (M8) CDMA, One (M8) dual sim, One A9, One A9s, One Dual Sim, One E9, One E9+, One M8s, One M9, One M9 (CDMA), One M9 (GSM), One M9 Prime Camera, One M9+, One M9+ Supreme Camera, One M9s, One Max, One max (CDMA), One ME, One mini, One mini 2, One mini 2 (GSM), One Remix, One Remix / One mini 2 (CDMA), One S, One S C2, One S9, One SC, One ST, One SV, One SV CDMA, One V, One VX, One X, One X

**Exhibit C for U.S. Patent No. 9,445,251 Against HTC Accused Products**

AT&T, One X+, One X10, One X9, One XC, One XL, Ozone, Ozone XV6175, Panache, Paradise, Prime, Pure, Pure / Touch Diamond2, Radar, Raider 4G, Rezound, Rhyme, Rhyme CDMA, Rider, S710 (Vox), S730, S740, Salsa, Schubert, SDA (USA) / SP5m (Tornado), Sensation, Sensation 4G, Sensation XE, Sensation XL, Shadow, Shadow (2009), Smart, Snap, Snap S511 (CDMA), SP3i / SDA (Europe) (Feeler), SPV C550 (Hurricane), SPV E200 / XPhone (Voyager), Status, Surround, Tattoo, ThunderBolt, ThunderBolt 4G, Tilt 8925 / TyTN II, Tilt2, Titan, Titan II, Touch (CDMA) / XV6900, Touch 3G, Touch Cruise, Touch Cruise 09, Touch Diamond (CDMA), Touch Diamond2, Touch Diamond2 CDMA, Touch Dual, Touch HD, Touch HD T8285, Touch Pro, Touch Pro (CDMA), Touch Pro2, Touch Pro2 (CDMA), Touch Pro2 (GSM) / Tilt 2, Touch Viva, Touch2, Trophy, Trophy (CDMA), TyTN / 8525 / JasJam (Hermes), U Play, U Ultra, U Ultra , U11, U11, U11 Eyes, U11 Life, U11 Plus, U11+, Velocity 4G, Vivid, Wildfire, Wildfire (CDMA), Wildfire S, Wildfire S (CDMA), Wildfire S (GSM), Wing / P4350 (Herald), xda II / MDA II, FLYER, JETSTREAM, FLYER WI-FI, EVO View 4G, FLYER CDMA, and any variants thereof. AGIS reserves the right to amend this list of Accused Products as discovery progresses. For example, AGIS reviewed Android-based products from multiple Android-based handset manufacturers, including two HTC phones (serial numbers FA73J1500645, FA73E1500899) which are available for inspection at HTC's request. For example, the Accused Products comprise HTC products, including but not limited to the phones and tablets as described herein, running the following versions (and all intervening updates and sub-versions) of the Android mobile operating system: Android 2.3, 4.0, 4.1, 4.2, 4.3, 4.4, 5.0, 5.1, 6.0, 7.0, 7.1, 8.0, and 8.1. For example, the Accused Products comprise HTC products, including but not limited to the phones and tablets as described herein, running any versions of the following Android-based applications and/or software: Android Device Manager, Find My Phone, Find My Device, Google Latitude, Google Plus, Google Hangouts, Google Maps, Google Assistant, Google Search, Google Messages, Android Messenger, Google Allo, Google Duo, Gmail, and Google Chrome. For example, the Accused Products comprise HTC products, including but not limited to the phones and tablets described herein, participating in any networks and/or services related to the execution and/or use of the Android mobile operating system versions and Android-based applications and/or software described herein.

AGIS does not concede that any claims of the '251 Patent that are not listed below are not infringed by the identified products. Moreover, the citations to certain documents and other information below are intended to be exemplary only and in no way foreclose AGIS from citing or relying on additional documents, information, source code, and/or testimony at a later time. These contentions are preliminary in nature, and an analysis of HTC's products, internal documentation, source code, and/or testimony from relevant witnesses may more fully and accurately describe the infringing features of its accused products. Accordingly, AGIS reserves the right to supplement, correct, modify, and/or amend these contentions once such additional information is made available to AGIS. Furthermore, AGIS reserves the right to supplement, correct, modify, and/or amend these contentions as discovery in this case

**Exhibit C for U.S. Patent No. 9,445,251 Against HTC Accused Products**

progresses; in view of the Court’s claim construction order(s); in view of any positions taken by HTC, including but not limited to positions on claim construction, invalidity, and/or non-infringement; and in connection with the preparation and exchange of expert reports.

<b>US9445251B2</b>	<b>HTC</b>
<p>1[P]. A computer-implemented method comprising:</p>	<p><b>HTC</b> infringes directly and/or indirectly by performing, inducing others to perform, and/or contributing to the performance of: a computer-implemented method [of claim 1].</p> <p>The Accused Products meet the claim limitations by providing device-location tracking features such as those features described below. For example, the Accused Products meet the claim limitations because they are pre-installed with Android mobile operating systems containing code for providing device-location tracking features as provided in the claims limitations herein. For example, the Accused Products run applications and/or software that run within the Android mobile operating system and that use components of the Android mobile operating system to provide device-location tracking features. Upon information and belief, in addition to the components and features of the Android mobile operating system itself, the following applications and/or software run within the Android mobile operating system and use components of the Android mobile operating system to provide device-location tracking features: Android Device Manager, Find My Device, Google Latitude, Google Plus, Google Hangouts (including Allo and Duo), Google Maps, Google Chrome, Google Messages, and Android Messenger.</p> <p><b><u>Find My Device (also known as “Android Device Manager”)</u></b></p> <p>Android Device Manager is the predecessor to Find My Device and has been available as a standard, pre-installed feature since 2013 and downloadable as a software application. The current iteration, Find My Device, often called the “new and improved Android Device Manager” or “rebranded Android Device Manager” is now part of the standard Google Play Protect suite, which is “built in and enabled on all devices,” <i>i.e.</i>, the Accused Products running Android OS. Upon information and belief, the Find My Device method also uses and/or works in conjunction with functionalities associated with Google Maps, Google Messages, Android Messenger, Google Chrome, Location Access, and other features, which come pre-installed on the Accused Products. For the purposes of avoiding needlessly presenting cumulative and duplicative evidence, AGIS sets forth the Find My Device feature of the Accused Products as representative of this first exemplary method. AGIS reserves the right to supplement these contentions to the extent that defendant requires additional information in accordance with P.R. 3-1 and for any other reason. In the cases</p>



**Exhibit C for U.S. Patent No. 9,445,251 Against HTC Accused Products**

US9445251B2	HTC
	<p>where Find My Device may not be pre-installed as a stand-alone “app,” the functionality is available within the Google Chrome browser, which is preinstalled on all Android devices. Find My Device below refers to both the native app as well as the Chrome browser version, which appears to be identical to the native app.</p> <p><i>See, e.g.</i>, <a href="https://www.androidcentral.com/find-my-device">https://www.androidcentral.com/find-my-device</a>;  <a href="https://support.google.com/android/answer/6160491?hl=en">https://support.google.com/android/answer/6160491?hl=en</a>; <a href="https://android.googleblog.com/2013/08/find-your-lost-phone-with-android.html">https://android.googleblog.com/2013/08/find-your-lost-phone-with-android.html</a>;  <a href="https://play.google.com/store/apps/details?id=com.google.android.apps.adm&amp;hl=en">https://play.google.com/store/apps/details?id=com.google.android.apps.adm&amp;hl=en</a>;  <a href="https://www.blog.google/products/android/google-play-protect/">https://www.blog.google/products/android/google-play-protect/</a></p> <p><b><u>Google Maps Share Location</u></b></p> <p>Share Location is currently included as a standard feature on the Accused Devices operating as a feature of Google Maps. Google Maps is a pre-installed software application in Android OS. The Accused Devices have included the Share Location functionalities since 2009 as part of Google Latitude, which was an opt-in feature for Google Maps on Android OS-based mobile devices, such as the Accused Products. Share Location functionalities were briefly shifted from Latitude for Google Maps to Google Plus and Google Hangouts, until reappearing as a standard feature in Google Maps. Upon information and belief, the Share Location method also uses and/or works in conjunction with functionalities associated with Google Maps, Google Messages, Android Messenger, Location Access, and other features, which are pre-installed on the Accused Products. For the purposes of these contentions, AGIS sets forth Google Maps’ Share Location feature of the Accused Products as representative of this exemplary software. AGIS reserves the right to supplement these contentions to the extent that defendant requires additional information in accordance with P.R. 3-1 and for any other reason.</p> <p><i>See, e.g.</i>, <a href="https://techcrunch.com/2017/03/22/google-maps-now-lets-you-share-your-location-with-friends-and-family-for-a-specific-period-of-time/">https://techcrunch.com/2017/03/22/google-maps-now-lets-you-share-your-location-with-friends-and-family-for-a-specific-period-of-time/</a>; <a href="https://googleblog.blogspot.com/2009/02/see-where-your-friends-are-with-google.html">https://googleblog.blogspot.com/2009/02/see-where-your-friends-are-with-google.html</a>; <a href="https://googleblog.blogspot.jp/2013/07/a-new-google-maps-app-for-smartphones.html">https://googleblog.blogspot.jp/2013/07/a-new-google-maps-app-for-smartphones.html</a>; <a href="http://googleplusproject.blogspot.com/2013/05/google-for-android-42.html">http://googleplusproject.blogspot.com/2013/05/google-for-android-42.html</a>;  <a href="https://googleblog.blogspot.com/2013/10/google-hangouts-and-photos-save-some.html">https://googleblog.blogspot.com/2013/10/google-hangouts-and-photos-save-some.html</a></p>

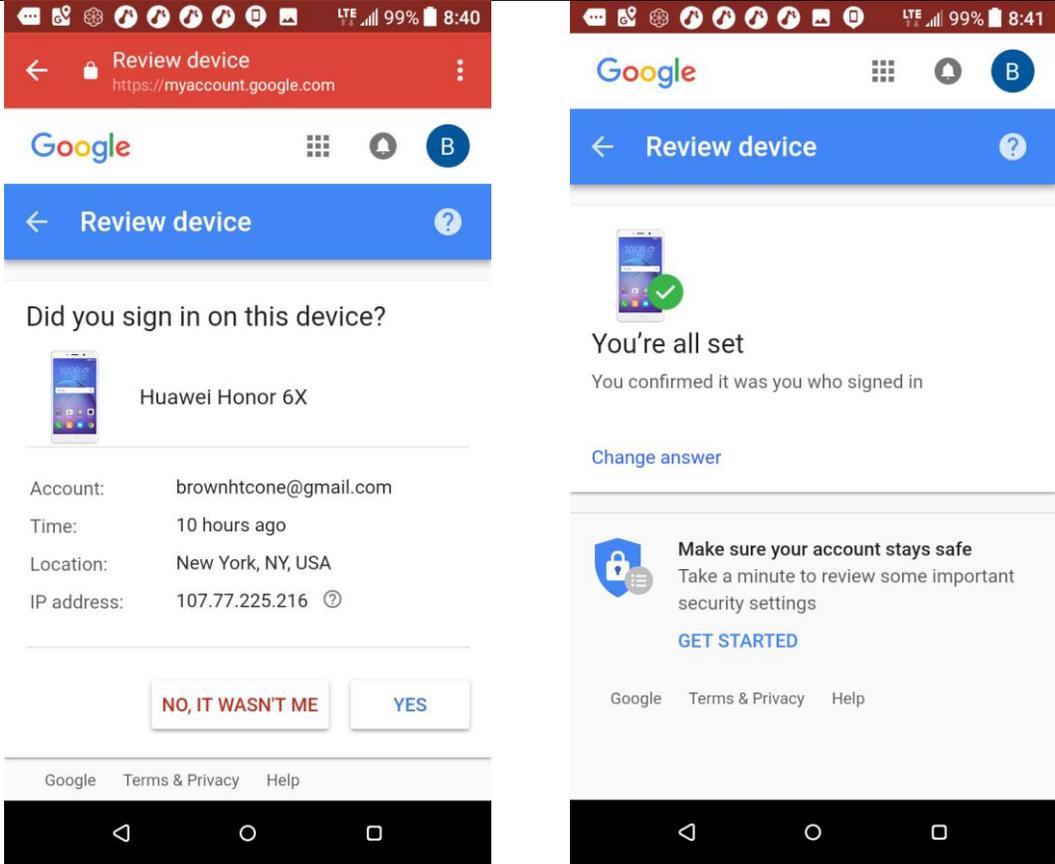
**Exhibit C for U.S. Patent No. 9,445,251 Against HTC Accused Products**

US9445251B2	HTC
	<p><b>Control within reach, even when your device isn't</b></p> <p>One of the biggest security risks you're likely to face is simply losing your phone. To help in these times of need, we're launching <a href="#">Find My Device</a> as part of Google Play Protect. With Find My Device you can locate, ring, lock and erase your Android devices—phones, tablets, and even watches. This feature is built in and enabled on all devices; visit <a href="https://android.com/find">android.com/find</a> or check out <a href="#">the app</a>.</p> <p><i>See, e.g.,</i> <a href="https://www.blog.google/products/android/google-play-protect/">https://www.blog.google/products/android/google-play-protect/</a></p>

**Exhibit C for U.S. Patent No. 9,445,251 Against HTC Accused Products**

<b>US9445251B2</b>	<b>HTC</b>
<p>[1A] with a first device, receiving a message from a second device, wherein the message relates to joining a group;</p>	<p>HTC infringes directly and/or indirectly by performing, inducing others to perform, and/or contributing to the performance of: with a first device, receiving a message from a second device, wherein the message relates to joining a group.</p> <p>The Accused Products practice receiving a message from a second device, wherein the message relates to joining a group. Each Accused Product (“first device”) is programmed to receive messages from other devices (second device). The Accused Products are capable of forming groups as set forth below.</p> <p>For example, an Accused Product is joined to a group (e.g. Google services such as the Play and Play Protect services) using a Google ID (e.g, an email address). Alternatively, any device with a browser or a device-location application can sign on to a Google service. The Google ID corresponds to one or more groups. These groups include, for example, the Google network, “friends” or “families” as defined by association with device-location features and applications (e.g., Find My Device, Hangouts (including Allo and Duo), Google+, Maps, and Chrome.)</p> <p>In establishing these groups, each Accused Product receives messages from other devices, e.g. other Accused Products, related to forming a group.</p> <p><b><u>Find My Device (also known as “Android Device Manager”)</u></b></p> <p>The Accused Products require a user to join the corresponding network by: signing-in to the device with an identifier (e.g., Google Account) or linking the device to the identifier by remote means. The message received by the first device relates to the second device joining into a group with the first device.</p>

**Exhibit C for U.S. Patent No. 9,445,251 Against HTC Accused Products**

US9445251B2	HTC
	 <p>e.g., above, the device receives a message indicating that the second device has joined the group, i.e. the google account.</p> <p><b><u>Google Maps Share Location</u></b></p> <p>Google Latitude, Google Plus, Google Hangouts, Google Messages, and Android Messenger, Google Duo, Google Allo, and Google Chrome, the Accused Products require a user to join the corresponding network by</p>

**Exhibit C for U.S. Patent No. 9,445,251 Against HTC Accused Products**

US9445251B2	HTC
	<p>signing-in to the device with an identifier (e.g., Google Account). The sign-in process takes place within the Google Maps software on the Accused Product or by navigating to maps.google.com within the Google Chrome browser on the Accused Product. Alternatively, the sign-in process may partially or completely take place using credentials already provided when the user associates a Google Account with the Accused Product, e.g., during initial setup of the Accused Product. Subject to discovery, one or more additional or substitute identifiers may correspond to the group. The sign-in process involves a user entering its Google Account and additional authentication data on the interface of the Accused Product and sending a message containing the Google Account and additional authentication data over a network to members of a group. The group comprises the multiple identifiers, individuals, profiles, and/or devices associated with the group.</p> <p>Further regarding Google Maps, Google Latitude, Google Plus, Google Hangouts, Google Messages, and Android Messenger, Google Duo, Google Allo, and Google Chrome, the Accused Products alternatively require a user to send a request containing the identifier (e.g., Google Account). Subject to discovery, additional identifiers may be assigned or used to correspond to the group. The request may be an invitation or message that associates a Google Account with one or more Google Accounts for the purposes of sharing locations within the group. The group comprises the multiple identifiers, individuals, profiles, and/or devices associated with the group</p> <p><b><u>Exemplary Support for Find My Device:</u></b></p>

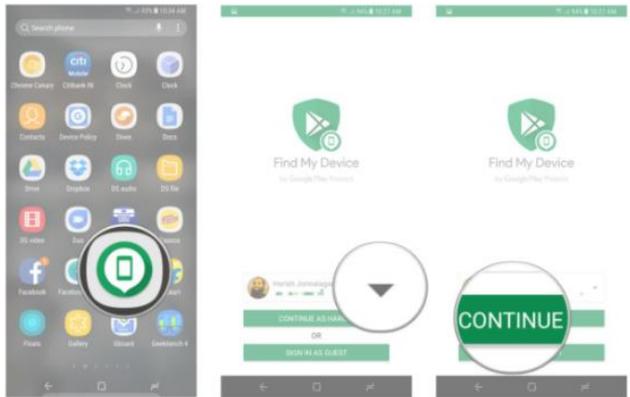
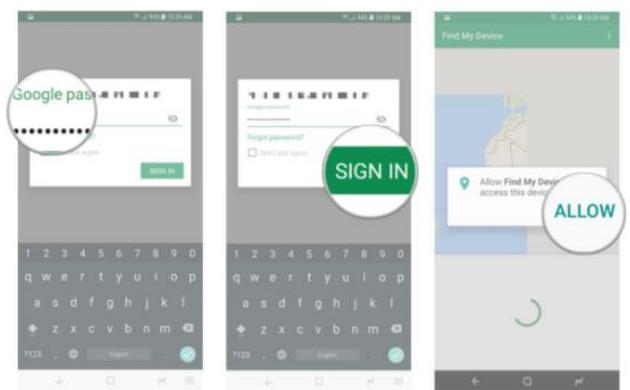
**Exhibit C for U.S. Patent No. 9,445,251 Against HTC Accused Products**

US9445251B2	HTC
	<p data-bbox="506 248 1329 289"><b>How to enable Find My Device on your phone</b></p> <p data-bbox="506 326 1528 444">In newer Android phones, the Find My Device service is already located conveniently in your Settings app, but if you can't find it you can always <a href="#">download Find My Device from the Google Play Store</a>. This locating service has essentially amalgamated with Google to make finding your phone easier. There are just a couple of things you'll need to activate.</p> <ol data-bbox="506 492 821 626" style="list-style-type: none"> <li data-bbox="506 492 716 516">1. Launch Settings.</li> <li data-bbox="506 545 674 570">2. Tap Security.</li> <li data-bbox="506 599 821 623">3. Tap Device Administration.</li> </ol> <div data-bbox="541 659 1570 1190"> <p>The first screenshot shows the home screen with the Settings app icon circled. The second screenshot shows the Settings app with the Security option circled. The third screenshot shows the Security settings with the Device Administration option circled.</p> </div> <ol data-bbox="506 1247 1226 1271" style="list-style-type: none"> <li data-bbox="506 1247 1226 1271">4. Tap Find My Device so that a checkmark appears in the checkbox.</li> </ol>

**Exhibit C for U.S. Patent No. 9,445,251 Against HTC Accused Products**

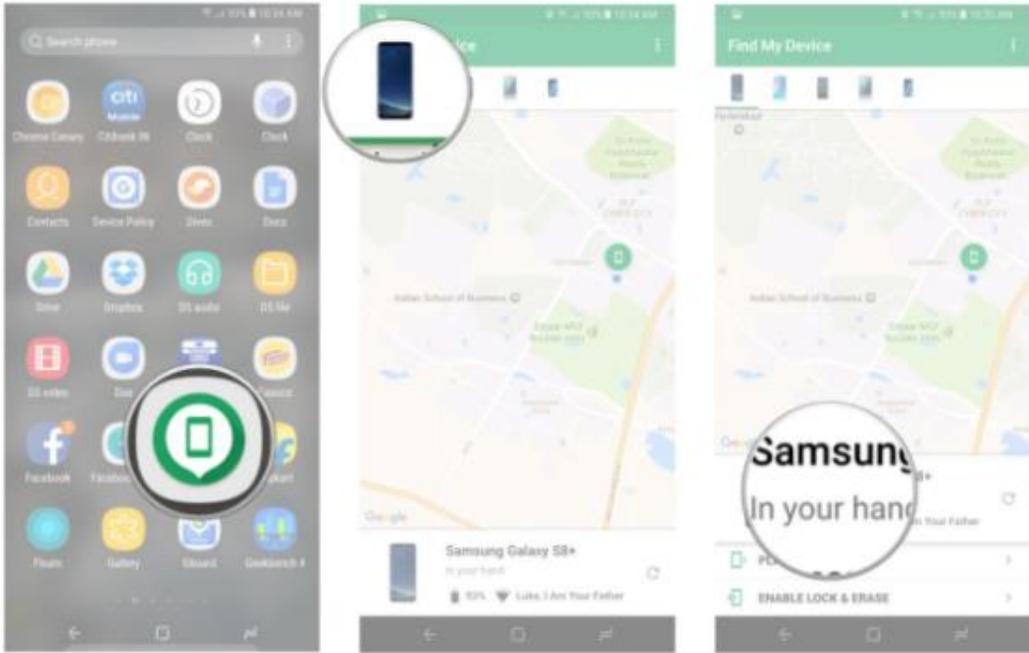
US9445251B2	HTC
	<p data-bbox="499 235 1249 267"><a href="https://www.androidcentral.com/how-track-android-phone">https://www.androidcentral.com/how-track-android-phone</a></p> <p data-bbox="520 310 1167 345"><b>How to locate your phone with Google</b></p> <p data-bbox="520 378 1482 431">Should you happen to lose your phone, you can locate its whereabouts by logging into your Google account from any computer or even from another phone.</p> <ol data-bbox="520 477 1241 602" style="list-style-type: none"> <li>1. Launch a <b>web browser</b> from a phone, tablet, or computer.</li> <li>2. Navigate to <b>Google</b> if it is not your default search engine or home page.</li> <li>3. Type <b>find my phone android</b> in the Google search bar.</li> </ol> <div data-bbox="550 634 1514 1198"> <p>The image contains three sequential screenshots from a mobile device. The first screenshot shows an Android home screen with various app icons; a Chrome browser icon is circled in red. The second screenshot shows a Google search page with the word 'google' entered in the search bar, which is circled in red. The third screenshot shows search results for 'find my phone android', with the word 'android' in the search bar circled in red.</p> </div> <ol data-bbox="520 1252 1514 1385" style="list-style-type: none"> <li>4. Tap on <b>Find My Device</b> (usually the first option in the search).</li> <li>5. Enter your <b>email address</b> and <b>password</b> just as though you were checking your email. If you have 2-step verification set up on your Google account (and you most certainly should), you'll need to complete that process as well.</li> </ol>

# Exhibit C for U.S. Patent No. 9,445,251 Against HTC Accused Products

US9445251B2	HTC
	<p data-bbox="499 235 1249 267"><a href="https://www.androidcentral.com/how-track-android-phone">https://www.androidcentral.com/how-track-android-phone</a></p> <ol data-bbox="514 300 913 389" style="list-style-type: none"><li>1. Open <b>Find Device</b> from your home screen or app drawer.</li><li>2. Select the <b>Google account</b> you want to use the service with.</li><li>3. Hit the <b>Continue as</b> button.</li></ol>  <ol data-bbox="514 836 766 917" style="list-style-type: none"><li>4. Enter your <b>Google account password</b>.</li><li>5. Tap <b>Sign in</b>.</li><li>6. Give <b>location access</b> to the service.</li></ol> 



**Exhibit C for U.S. Patent No. 9,445,251 Against HTC Accused Products**

US9445251B2	HTC
	<p data-bbox="499 235 1121 267"><a href="https://www.androidcentral.com/find-my-device">https://www.androidcentral.com/find-my-device</a></p> <p data-bbox="514 316 1533 406">Once you're signed in to Find My Device, you'll see a map with your current location as well as the make and model of your phone, and two options — Play Sound, and Enable Lock &amp; Erase. Hitting the latter option will allow you to start using the Lock and Erase functions.</p> <p data-bbox="514 446 1549 500">If you've signed into more than one phone, you can select a particular device by browsing the list at the top of the screen.</p> <ol data-bbox="514 544 1234 682" style="list-style-type: none"> <li>1. Open <b>Find My Device</b> from your home screen or app drawer.</li> <li>2. Select your phone from the list of devices at the top of the screen.</li> <li>3. See if your phone is <b>discoverable</b>.</li> </ol> 

**Exhibit C for U.S. Patent No. 9,445,251 Against HTC Accused Products**

US9445251B2	HTC
	<p data-bbox="499 235 1121 267"><a href="https://www.androidcentral.com/find-my-device">https://www.androidcentral.com/find-my-device</a></p> <h2 data-bbox="512 305 1650 362">How to locate your phone over the internet</h2> <p data-bbox="512 402 1659 545">If you've lost your phone, you can remotely locate it through the <a href="#">Find My Device website</a>. You'll need to sign in to the Google account that was used to set up Find My Device. It takes a few seconds, but the service should be able to track your phone. Alternatively, you can also do a Google search for "find my phone" to locate your handset.</p> <ol data-bbox="504 602 997 756" style="list-style-type: none"> <li>1. Head to the <a href="#">Find My Device website</a>.</li> <li>2. Sign in to your Google account.</li> <li>3. Check if your device is visible.</li> </ol> <div data-bbox="548 797 1759 1252"> </div> <p data-bbox="499 1312 1121 1344"><a href="https://www.androidcentral.com/find-my-device">https://www.androidcentral.com/find-my-device</a></p>

# Exhibit C for U.S. Patent No. 9,445,251 Against HTC Accused Products

US9445251B2	HTC
	<p data-bbox="499 305 1016 334"><b><u>Exemplary Support for Google Maps:</u></b></p> <p data-bbox="531 378 888 394">COMPUTER   <b>ANDROID</b>   IPHONE &amp; IPAD</p> <hr data-bbox="520 418 1402 422"/> <h3 data-bbox="520 467 936 496">If they have a Google Account</h3> <ol data-bbox="527 516 1272 764" style="list-style-type: none"><li>1. If you haven't already, add their Gmail address to your <a href="#">Google Contacts</a>.</li><li>2. On your Android phone or tablet, open the Google Maps app and sign in. <a href="#">Learn how to sign in.</a></li><li>3. Tap Menu ≡ &gt; <b>Location sharing</b> &gt; Add People +.</li><li>4. Choose how long you want to share your location.</li><li>5. Tap <b>Select People</b>.<ul data-bbox="548 675 1035 695" style="list-style-type: none"><li>• If you're asked about your contacts, give Google Maps access.</li></ul></li><li>6. Choose who you want to share with.</li><li>7. Tap <b>Share</b>.</li></ol> <h3 data-bbox="520 818 1010 847">If they don't have a Google Account</h3> <ol data-bbox="527 867 1394 976" style="list-style-type: none"><li>1. On your Android phone or tablet, open the Google Maps app and sign in. <a href="#">Learn how to sign in.</a></li><li>2. Tap Menu ≡ &gt; <b>Location sharing</b> &gt; Add People +.</li><li>3. Tap More ... &gt; <b>Copy to clipboard</b>. People with this link can see your location for as long as you choose, up to 72 hours.</li></ol> <h3 data-bbox="520 1016 806 1045">Share using another app</h3> <p data-bbox="520 1062 1094 1081">You can also share through messaging apps. Tap More ... &gt; select an app.</p> <h3 data-bbox="520 1133 695 1162">Stop sharing</h3> <ol data-bbox="527 1182 1094 1268" style="list-style-type: none"><li>1. Open the Google Maps app.</li><li>2. Tap Menu ≡ &gt; <b>Location sharing</b>.</li><li>3. Next to the person with whom you want to stop sharing, tap Remove ×.</li></ol> <p data-bbox="499 1312 1688 1341"><a href="https://support.google.com/maps/answer/7326816?co=GENIE.Platform%3DAndroid&amp;oco=1">https://support.google.com/maps/answer/7326816?co=GENIE.Platform%3DAndroid&amp;oco=1</a></p>

## Exhibit C for U.S. Patent No. 9,445,251 Against HTC Accused Products

US9445251B2	HTC
	<h3 data-bbox="520 326 810 362">Share your E.T.A</h3> <p data-bbox="520 391 1598 410">After you start your drive, you can share your destination, estimated arrival time, and where you are on the route.</p> <ol data-bbox="531 444 1304 667" style="list-style-type: none"><li data-bbox="531 444 856 464">1. Open the Google Maps app .</li><li data-bbox="531 483 1115 503">2. Set a driving destination. <a href="#">Learn how to navigate to a place.</a></li><li data-bbox="531 522 1157 542">3. After you start navigation, tap More ^ &gt; <b>Share trip progress.</b></li><li data-bbox="531 561 846 581">4. Choose a person from the list.</li><li data-bbox="531 600 659 620">5. Tap <b>Share.</b></li><li data-bbox="531 639 1304 659">6. Location Sharing will stop when you reach your destination or stop navigating.</li></ol> <ul data-bbox="531 695 1157 714" style="list-style-type: none"><li data-bbox="531 695 1157 714">• To stop sharing before you arrive, tap More ^ &gt; <b>Stop sharing.</b></li></ul> <h3 data-bbox="520 789 921 824">See where someone is</h3> <p data-bbox="520 854 1220 873">If someone shares their location with you, you can see them on the map.</p> <ol data-bbox="531 907 888 1008" style="list-style-type: none"><li data-bbox="531 907 856 927">1. Open the Google Maps app .</li><li data-bbox="531 946 888 966">2. Tap Menu ≡ &gt; <b>Location sharing.</b></li><li data-bbox="531 985 730 1005">3. Choose someone.</li></ol> <ul data-bbox="531 1040 1251 1060" style="list-style-type: none"><li data-bbox="531 1040 1251 1060">• To see an updated location, tap on a friend's icon &gt; More ⋮ &gt; <b>Refresh.</b></li></ul> <h3 data-bbox="520 1118 995 1154">Stop seeing someone's location</h3> <ol data-bbox="531 1174 1402 1317" style="list-style-type: none"><li data-bbox="531 1174 856 1193">1. Open the Google Maps app .</li><li data-bbox="531 1213 810 1232">2. On the map, tap their icon.</li><li data-bbox="531 1252 825 1271">3. At the bottom, tap More ^ .</li><li data-bbox="531 1291 1402 1310">4. To temporarily hide someone, tap <b>Hide from map.</b> You can stop hiding them at any time.</li></ol> <p data-bbox="520 1349 1671 1369"><b>Note:</b> You can stop someone's location from ever appearing on your map. Learn how to <a href="#">block another person's account.</a></p>

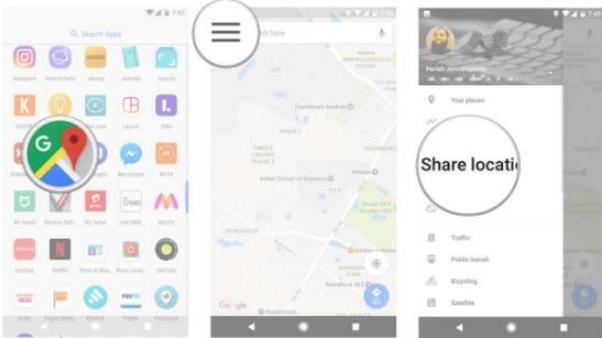
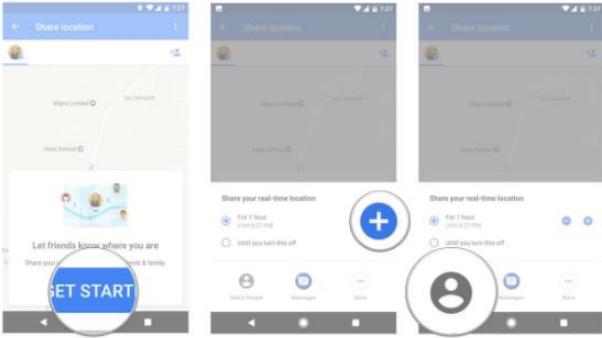
## Exhibit C for U.S. Patent No. 9,445,251 Against HTC Accused Products

US9445251B2	HTC
	<p data-bbox="499 235 1690 267"><a href="https://support.google.com/maps/answer/7326816?co=GENIE.Platform%3DAndroid&amp;oco=1">https://support.google.com/maps/answer/7326816?co=GENIE.Platform%3DAndroid&amp;oco=1</a></p> <h3 data-bbox="514 308 987 357">Create a list of places</h3> <p data-bbox="514 373 1333 397">In Google Maps, you can create a list of places, like your favorite places or places you want to visit.</p> <p data-bbox="525 470 924 495">COMPUTER <b>ANDROID</b> IPHONE &amp; IPAD</p> <hr data-bbox="514 511 1354 519"/> <h3 data-bbox="514 568 756 609">Make a new list</h3> <ol data-bbox="514 625 1081 787" style="list-style-type: none"><li>1. On your Android phone or tablet, open the Google Maps app .</li><li>2. Tap Menu ≡ &gt; <b>Your places</b> &gt; <b>Saved</b>.</li><li>3. In the bottom right, tap Add +.</li><li>4. Enter a name and description.</li><li>5. Tap <b>Save</b>.</li></ol> <h3 data-bbox="514 836 829 876">Save a place to a list</h3> <ol data-bbox="514 893 966 1055" style="list-style-type: none"><li>1. Open the Google Maps app .</li><li>2. Search for a place or tap it on the map.</li><li>3. At the bottom, tap the place's name or address.</li><li>4. Tap <b>Save</b>.</li><li>5. Choose a list. To create a new list, tap <b>New list</b> +.</li></ol> <h3 data-bbox="514 1112 724 1153">See your lists</h3> <ol data-bbox="514 1169 871 1226" style="list-style-type: none"><li>1. Open the Google Maps app .</li><li>2. Tap Menu ≡ &gt; <b>Your places</b> &gt; <b>Saved</b>.</li></ol> <p data-bbox="499 1266 1900 1339"><a href="https://support.google.com/maps/answer/7280933?hl=en&amp;ref_topic=7301134&amp;co=GENIE.Platform%3DAndroid&amp;oco=1">https://support.google.com/maps/answer/7280933?hl=en&amp;ref_topic=7301134&amp;co=GENIE.Platform%3DAndroid&amp;oco=1</a></p>

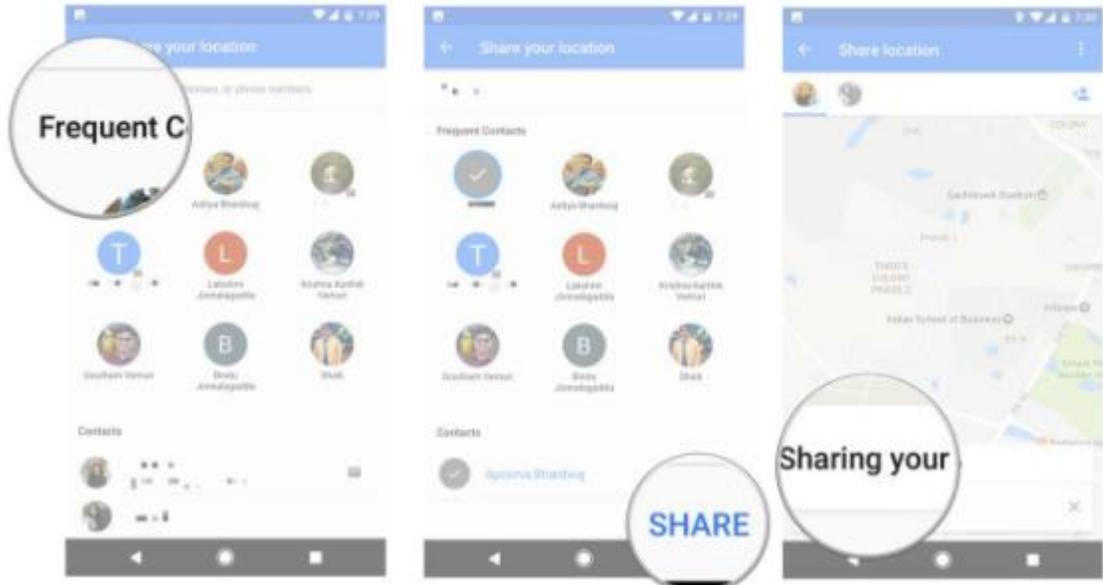
## Exhibit C for U.S. Patent No. 9,445,251 Against HTC Accused Products

US9445251B2	HTC
	<h3 data-bbox="531 248 865 289">Hide or share lists</h3> <p data-bbox="531 315 894 337"><b>Note:</b> You can't share starred places.</p> <ol data-bbox="531 370 1241 477" style="list-style-type: none"><li>1. Open the Google Maps app .</li><li>2. Tap Menu  &gt; <b>Your places</b> &gt; <b>Saved</b>.</li><li>3. Next to the list you want to share, tap More  &gt; choose an option:</li></ol> <ul data-bbox="573 493 1671 634" style="list-style-type: none"><li>• <b>Hide/Show on your map:</b> Display or hide your saved places when looking at the map.</li><li>• <b>Share list:</b> Allow others to see your saved list.</li><li>• <b>Sharing options:</b> You can make your list public, private, or shared. To let anyone with the link see your list, tap <b>Shared</b>. To let anyone find and follow your list, tap <b>Public</b>.</li></ul> <h3 data-bbox="531 703 751 743">Follow a list</h3> <p data-bbox="531 769 1713 824">If you follow a list made by someone else, their saved places will show up in Your Places. The places will also appear as suggested locations in Google Maps.</p> <h3 data-bbox="531 881 898 922">Follow a list using a link</h3> <ol data-bbox="531 941 1339 1049" style="list-style-type: none"><li>1. Tap on the link you received to open it.</li><li>2. Tap <b>Follow</b>. This list will now be added to the group of lists you follow.</li><li>3. <b>Optional:</b> To unfollow a list someone shared with you, tap the list &gt; <b>Following</b>.</li></ol> <h3 data-bbox="531 1101 909 1141">See lists made by others</h3> <p data-bbox="531 1161 1318 1183">If a user has any Google Maps lists that were made public, you can follow them.</p> <ol data-bbox="531 1216 1119 1323" style="list-style-type: none"><li>1. Tap on the name of a user whose list you want to follow.</li><li>2. Tap <b>Lists</b>.</li><li>3. Tap on the list you want to follow &gt; More  &gt; <b>Follow</b>.</li></ol> <p data-bbox="499 1372 1902 1404"><a href="https://support.google.com/maps/answer/7280933?hl=en&amp;ref_topic=7301134&amp;co=GENIE.Platform%3DAnd">https://support.google.com/maps/answer/7280933?hl=en&amp;ref_topic=7301134&amp;co=GENIE.Platform%3DAnd</a></p>

# Exhibit C for U.S. Patent No. 9,445,251 Against HTC Accused Products

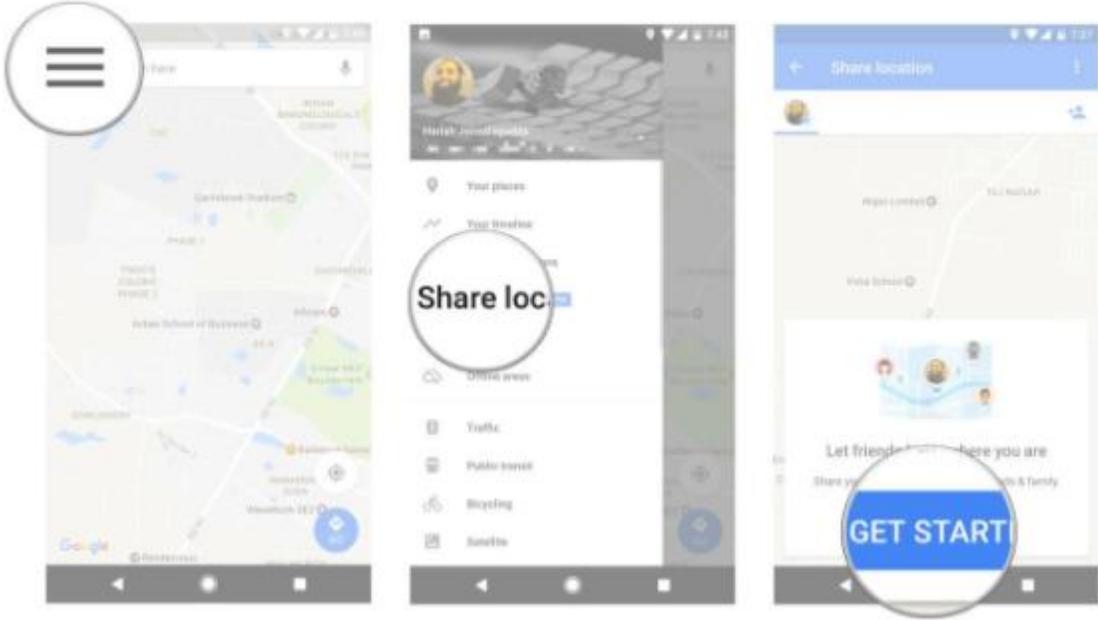
US9445251B2	HTC
	<p data-bbox="499 235 655 261">roid&amp;oco=1</p> <h3 data-bbox="506 305 1075 334">How to share your location in Google Maps</h3> <ol data-bbox="506 362 1058 440" style="list-style-type: none"><li data-bbox="506 362 898 378">1. Open <b>Google Maps</b> from the app drawer or the home screen.</li><li data-bbox="506 391 1058 407">2. Tap the <b>hamburger menu</b> (the three horizontal lines) on the top left corner of the screen.</li><li data-bbox="506 420 659 436">3. Select <b>Share location</b>.</li></ol>  <ol data-bbox="506 837 1087 932" style="list-style-type: none"><li data-bbox="506 837 625 854">4. Tap <b>Get Started</b>.</li><li data-bbox="506 867 1087 899">5. Use the <b>+</b> icon to select a time period or select the <b>Until you turn this off</b> setting to share your location indefinitely.</li><li data-bbox="506 912 638 928">6. Tap <b>Select People</b>.</li></ol>  <p data-bbox="499 1333 1346 1365"><a href="https://www.androidcentral.com/how-share-location-google-maps">https://www.androidcentral.com/how-share-location-google-maps</a></p>

### Exhibit C for U.S. Patent No. 9,445,251 Against HTC Accused Products

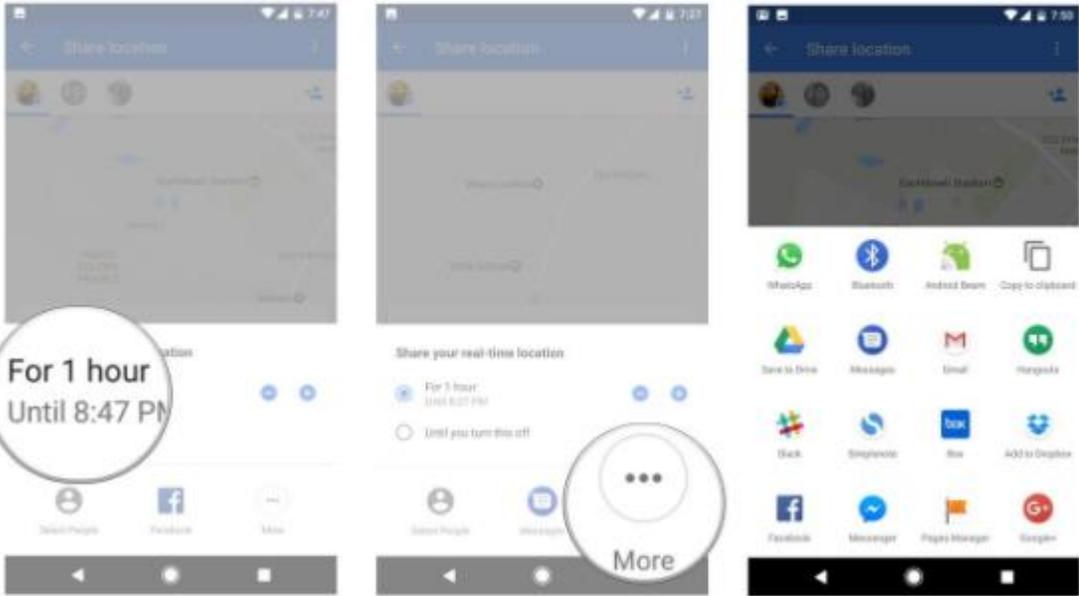
US9445251B2	HTC
	<p data-bbox="514 251 1564 430"><b>7.</b> You'll see a list of your frequent contacts at the top, along with a full list of contacts. <b>Pick the contacts</b> by tapping their name.</p> <p data-bbox="514 341 1438 365"><b>8.</b> Once you've selected the contacts you want to share your location to, tap <b>Share</b>.</p> <p data-bbox="514 397 1407 422"><b>9.</b> You'll see a message saying that the selected contact can view your location.</p>  <p data-bbox="493 1096 1344 1128"><a href="https://www.androidcentral.com/how-share-location-google-maps">https://www.androidcentral.com/how-share-location-google-maps</a></p>



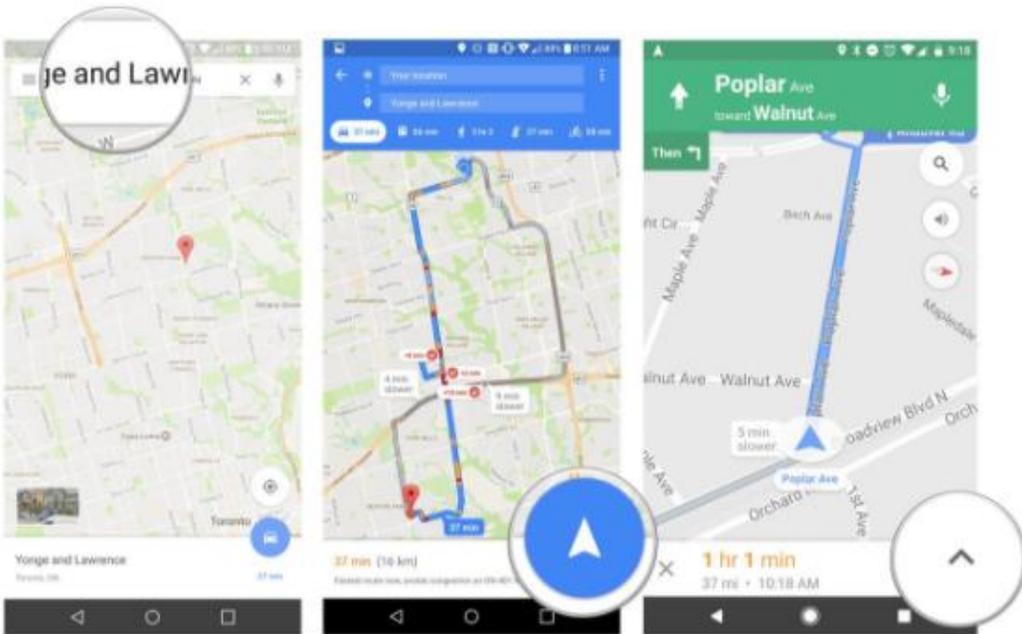
## Exhibit C for U.S. Patent No. 9,445,251 Against HTC Accused Products

US9445251B2	HTC
	<h3 data-bbox="510 245 1245 293">How to create a shareable link</h3> <p data-bbox="510 334 1451 363">You can also create a link and use it to share your location easily. Here's how to do it:</p> <ol data-bbox="510 412 1224 553" style="list-style-type: none"><li data-bbox="510 412 1224 441">1. Tap the <b>hamburger menu</b> on the top left corner of the screen.</li><li data-bbox="510 469 789 498">2. Select <b>Share location</b>.</li><li data-bbox="510 526 726 555">3. Tap <b>Get Started</b>.</li></ol>  <p data-bbox="499 1268 1346 1297"><a href="https://www.androidcentral.com/how-share-location-google-maps">https://www.androidcentral.com/how-share-location-google-maps</a></p>

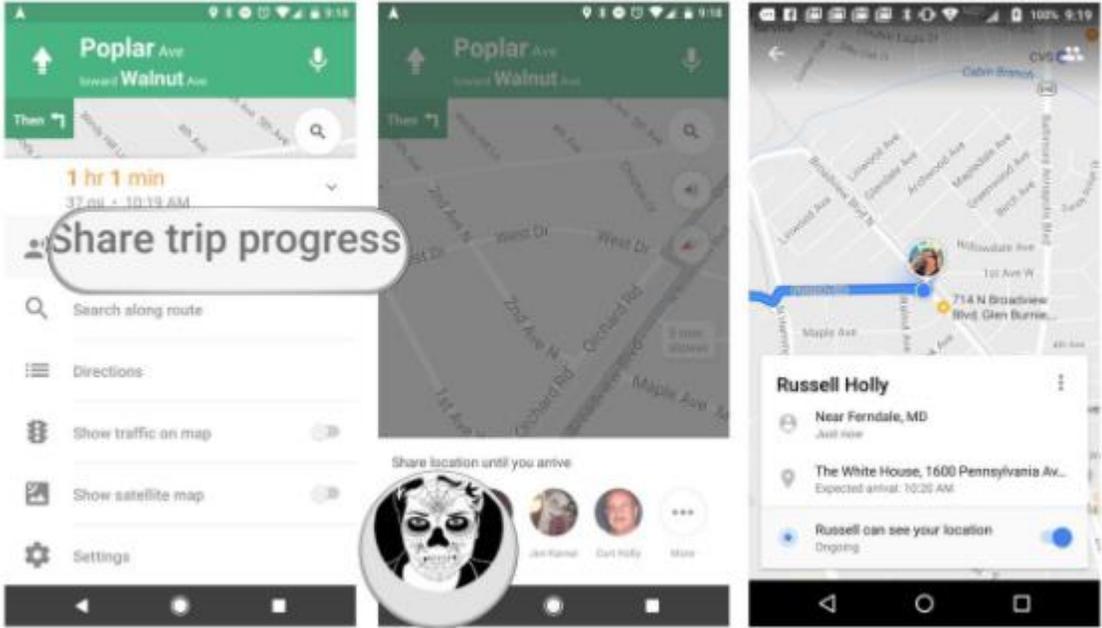
### Exhibit C for U.S. Patent No. 9,445,251 Against HTC Accused Products

US9445251B2	HTC
	<p data-bbox="510 245 1199 272">4. Select the amount of time you want to share your location.</p> <p data-bbox="510 302 667 329">5. Tap More.</p> <p data-bbox="510 358 1619 423">6. Select your app of choice to create and send a unique URL that broadcasts your current location. You can email it, send the link via Messenger, or even tweet it to the intended recipient.</p>  <p data-bbox="499 1117 1346 1149"><a href="https://www.androidcentral.com/how-share-location-google-maps">https://www.androidcentral.com/how-share-location-google-maps</a></p>

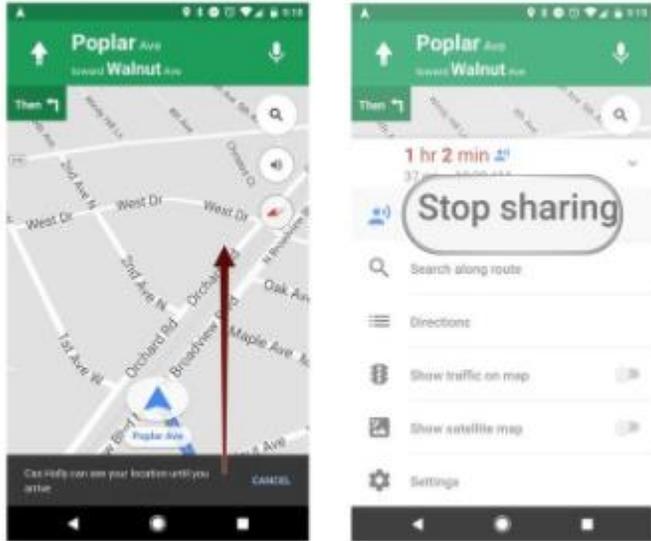
**Exhibit C for U.S. Patent No. 9,445,251 Against HTC Accused Products**

US9445251B2	HTC
	<p data-bbox="514 240 1417 337"><b>How to share your navigation directions while you walk, drive or transit</b></p> <p data-bbox="514 375 1543 464">One of the best ways to share your location with someone is during a drive, walk or transit. If you're meeting a friend or family member somewhere, or navigating towards their home, Google Maps lets you share your location with them for the duration of the trip. It's magic!</p> <ol data-bbox="514 513 1381 646" style="list-style-type: none"><li>1. In the <b>search bar</b> enter your destination.</li><li>2. Pick your navigation type (drive, transit, walk) and press the blue <b>navigate button</b>.</li><li>3. Tap the arrow next to the <b>time-to-destination</b> number at the bottom of the screen.</li></ol>  <p data-bbox="499 1360 1346 1393"><a href="https://www.androidcentral.com/how-share-location-google-maps">https://www.androidcentral.com/how-share-location-google-maps</a></p>

### Exhibit C for U.S. Patent No. 9,445,251 Against HTC Accused Products

US9445251B2	HTC
	<p data-bbox="514 378 825 407">4. Tap Share trip progress.</p> <p data-bbox="514 435 1136 464">5. Choose one or more contacts to share trip progress.</p>  <p data-bbox="520 1166 1329 1195">You can also stop sharing your location with someone before a trip ends.</p> <p data-bbox="499 1235 1346 1265"><a href="https://www.androidcentral.com/how-share-location-google-maps">https://www.androidcentral.com/how-share-location-google-maps</a></p>

## Exhibit C for U.S. Patent No. 9,445,251 Against HTC Accused Products

US9445251B2	HTC
	<ol style="list-style-type: none"><li data-bbox="520 313 1457 342">1. Tap the arrow next to the <b>time-to-destination</b> number at the bottom of the screen.</li><li data-bbox="520 370 758 399">2. Tap <b>Stop sharing</b>.</li></ol> <div data-bbox="732 451 1383 992"></div> <p data-bbox="527 1045 625 1075">That's it!</p> <p data-bbox="527 1114 1598 1143">Are you excited that location sharing is back in Google Maps? How often do you use the feature?</p> <p data-bbox="499 1187 1346 1219"><a href="https://www.androidcentral.com/how-share-location-google-maps">https://www.androidcentral.com/how-share-location-google-maps</a></p>

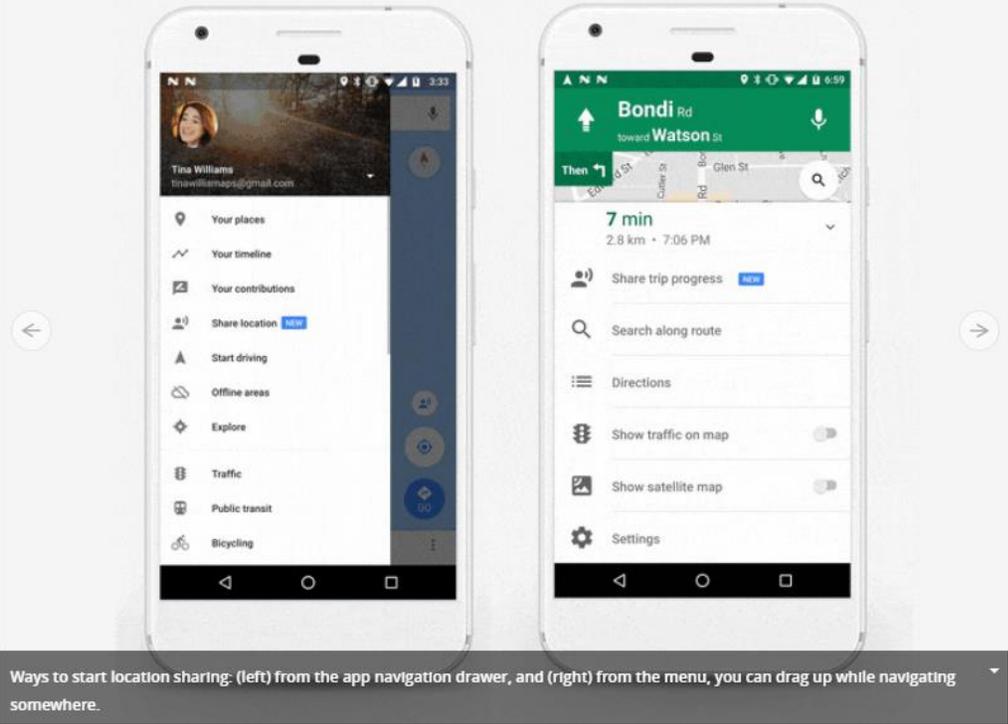
## Exhibit C for U.S. Patent No. 9,445,251 Against HTC Accused Products

US9445251B2	HTC
	<p>As shown below, a group may also be defined within Google Contacts.</p> <h3>See your contacts</h3> <ol style="list-style-type: none"><li>1. Open your device's Contacts app .</li><li>2. Tap Menu .</li></ol> <ul style="list-style-type: none"><li>• <b>See contacts by label:</b> Choose a label from the list.</li><li>• <b>See contacts for another account:</b> Tap Down arrow  &gt; pick an account.</li><li>• <b>See the contacts for all your accounts:</b> Choose <b>All contacts</b>.</li></ul> <p><b>Tip:</b> If you have multiple contacts with the same information, the information will be grouped into one contact.</p> <ul style="list-style-type: none"><li>• <b>See your Google Account contacts on the web:</b> Go to <a href="#">Google Contacts</a> .</li></ul> <p><a href="https://support.google.com/android/answer/6118731?hl=en&amp;ref_topic=6118711">https://support.google.com/android/answer/6118731?hl=en&amp;ref_topic=6118711</a></p> <h3>Label your contacts</h3> <p>You can group contacts together using labels.</p> <ol style="list-style-type: none"><li>1. Open your device's Contacts app .</li><li>2. Tap Menu  &gt; <b>Create label</b>.</li><li>3. Enter a label name and tap <b>Ok</b>.</li></ol> <ul style="list-style-type: none"><li>• <b>Add one contact to a label:</b> Tap Add contact  &gt; choose a contact.</li><li>• <b>Add multiple contacts to a label:</b> Tap Add contact  &gt; touch and hold a contact &gt; tap the other contacts &gt; tap <b>Add</b>.</li></ul> <p><a href="https://support.google.com/android/answer/6118731?hl=en&amp;ref_topic=6118711">https://support.google.com/android/answer/6118731?hl=en&amp;ref_topic=6118711</a></p>

**Exhibit C for U.S. Patent No. 9,445,251 Against HTC Accused Products**

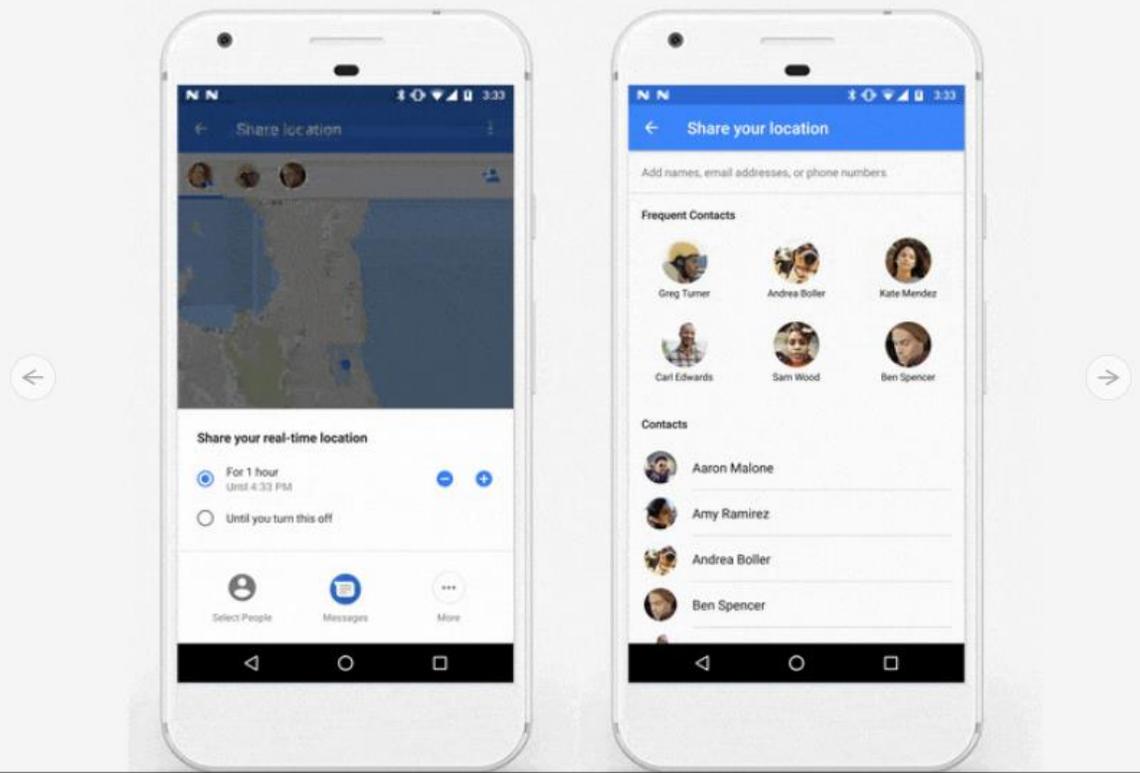
US9445251B2	HTC
	<p data-bbox="533 316 932 360"><b>Share your contacts</b></p> <ol data-bbox="548 386 1031 545" style="list-style-type: none"><li data-bbox="548 386 961 415">1. Open your device's Contacts app .</li><li data-bbox="548 431 827 461">2. Tap a contact in the list.</li><li data-bbox="548 477 821 506">3. Tap More  &gt; <b>Share</b>.</li><li data-bbox="548 522 1031 552">4. Choose how you want to share the contact.</li></ol> <p data-bbox="499 597 1520 630"><a href="https://support.google.com/android/answer/6118731?hl=en&amp;ref_topic=6118711">https://support.google.com/android/answer/6118731?hl=en&amp;ref_topic=6118711</a></p>

# Exhibit C for U.S. Patent No. 9,445,251 Against HTC Accused Products

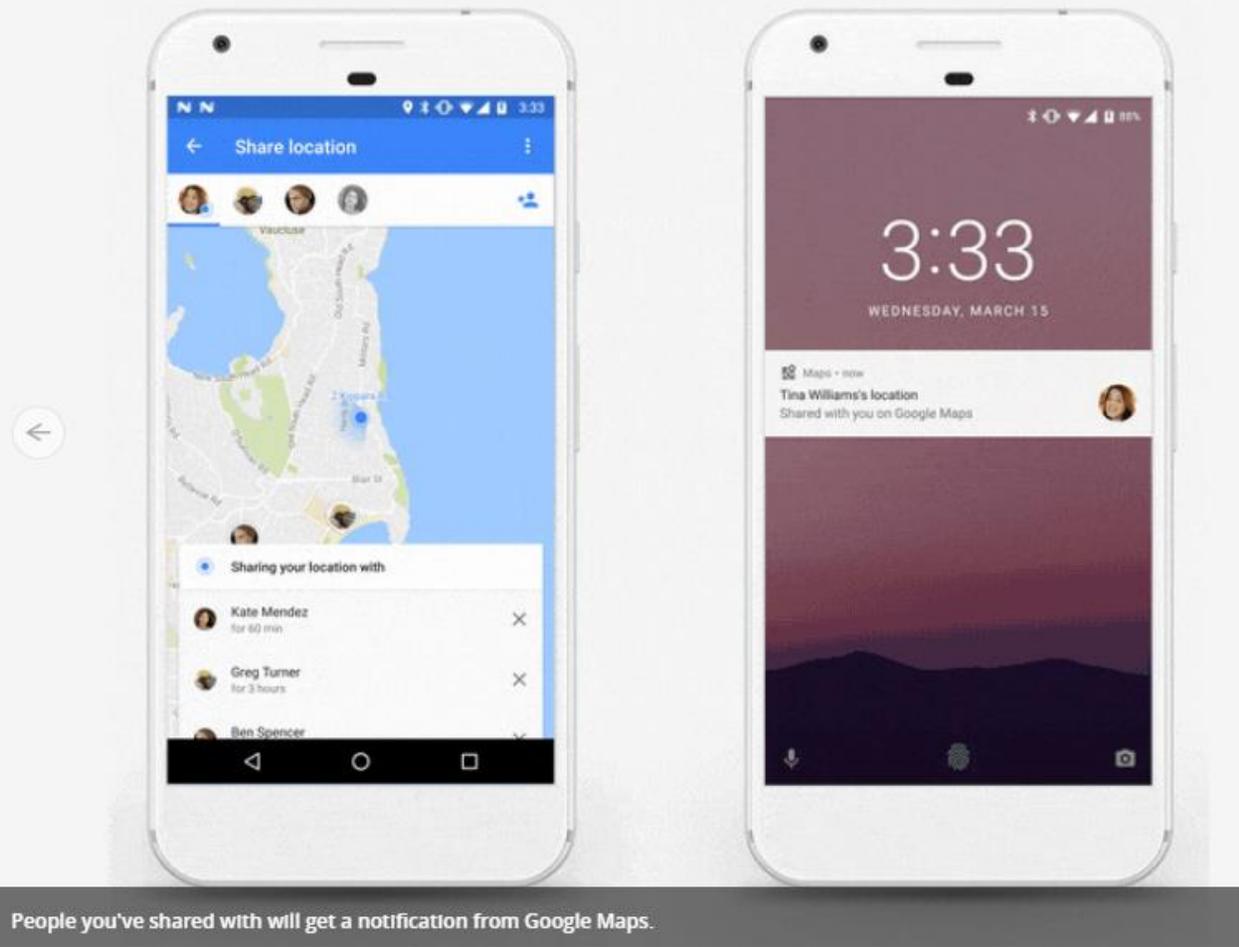
US9445251B2	HTC
	 <p>Ways to start location sharing: (left) from the app navigation drawer, and (right) from the menu, you can drag up while navigating somewhere.</p> <p><a href="https://arstechnica.com/gadgets/2017/03/location-sharing-finally-returns-to-google-maps/">https://arstechnica.com/gadgets/2017/03/location-sharing-finally-returns-to-google-maps/</a></p>



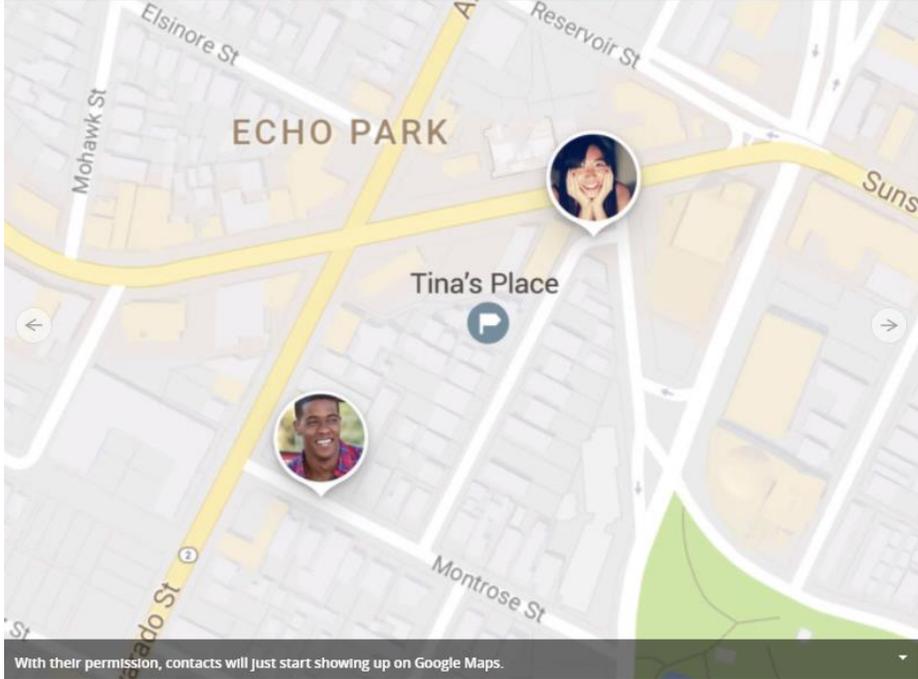
**Exhibit C for U.S. Patent No. 9,445,251 Against HTC Accused Products**

US9445251B2	HTC
	 <p data-bbox="506 1024 1646 1062">You can pick how long you want to share your location for, and with whom you want to share it.</p> <p data-bbox="499 1097 1646 1135"><a href="https://arstechnica.com/gadgets/2017/03/location-sharing-finally-returns-to-google-maps/">https://arstechnica.com/gadgets/2017/03/location-sharing-finally-returns-to-google-maps/</a></p>

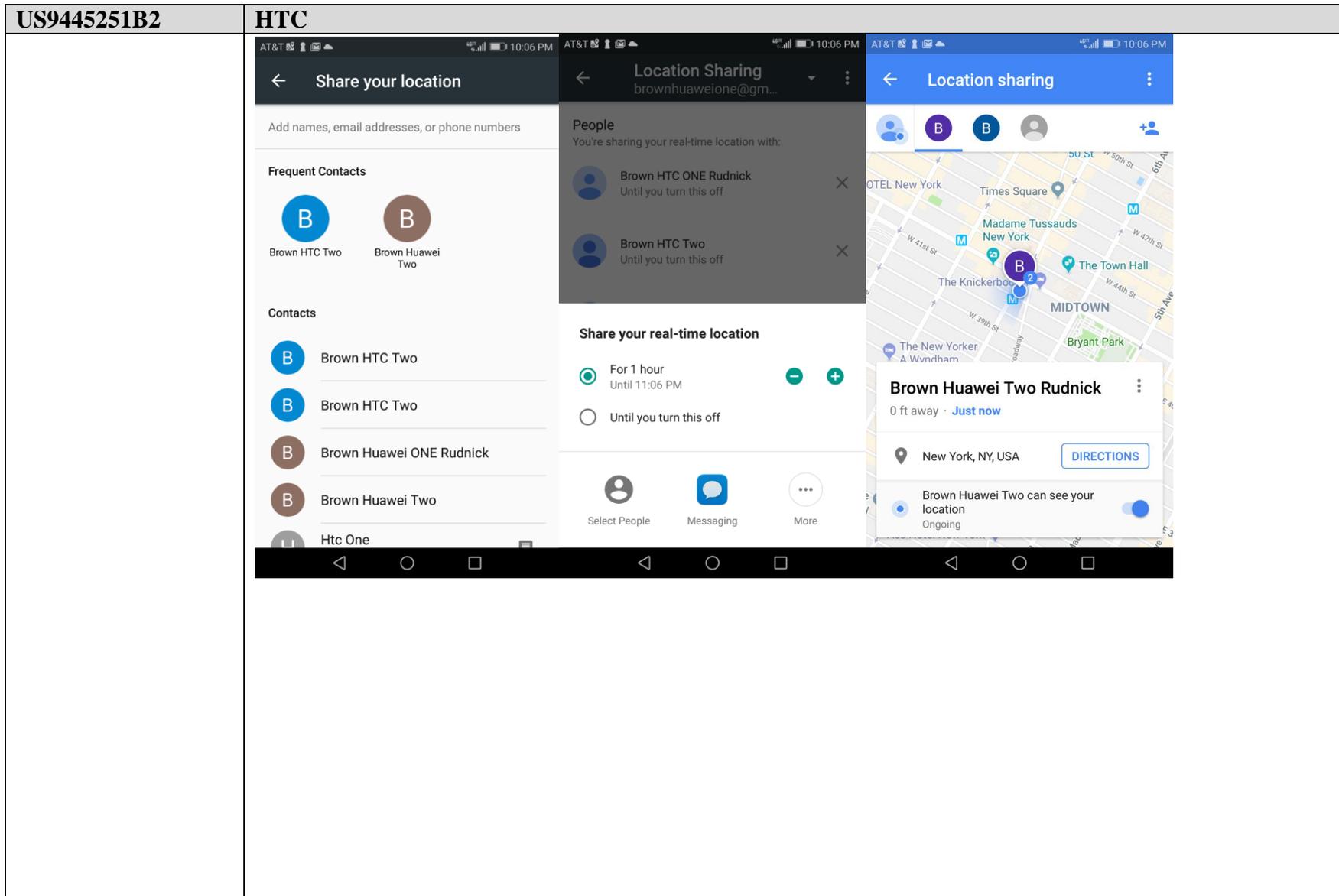
**Exhibit C for U.S. Patent No. 9,445,251 Against HTC Accused Products**

US9445251B2	HTC
	 <p data-bbox="506 1146 1745 1187">People you've shared with will get a notification from Google Maps.</p> <p data-bbox="499 1222 1646 1256"><a href="https://arstechnica.com/gadgets/2017/03/location-sharing-finally-returns-to-google-maps/">https://arstechnica.com/gadgets/2017/03/location-sharing-finally-returns-to-google-maps/</a></p>

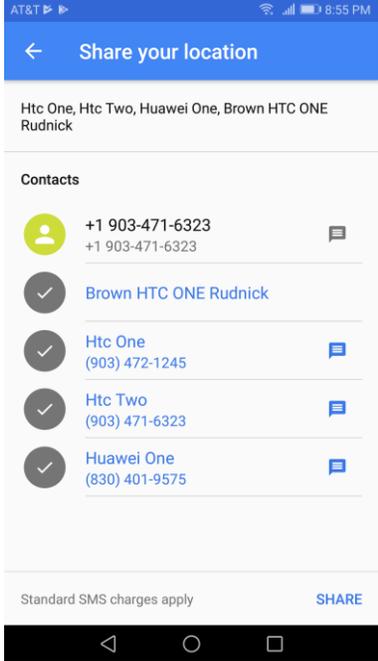
**Exhibit C for U.S. Patent No. 9,445,251 Against HTC Accused Products**

US9445251B2	HTC
	 <p>The screenshot shows a Google Maps view of the Echo Park neighborhood in Los Angeles. The map is centered on the intersection of Montrose St and a street labeled 'A'. Two circular location pins are overlaid on the map: one for a woman (Tina) and one for a man. A blue location pin labeled 'Tina's Place' is also visible. Street names include Elsinore St, Reservoir St, Mohawk St, Montrose St, and Sun. A text overlay at the bottom of the map reads: 'With their permission, contacts will just start showing up on Google Maps.'</p> <p><a href="https://arstechnica.com/gadgets/2017/03/location-sharing-finally-returns-to-google-maps/">https://arstechnica.com/gadgets/2017/03/location-sharing-finally-returns-to-google-maps/</a></p> <p><b><u>Exemplary Google Maps Screenshots:</u></b></p>

**Exhibit C for U.S. Patent No. 9,445,251 Against HTC Accused Products**



**Exhibit C for U.S. Patent No. 9,445,251 Against HTC Accused Products**

US9445251B2	HTC
	 <p><b>Exemplary Source Code:</b></p> <p>The above functionality is performed at least in part by the following publicly available source code and/or source code that invokes or is invoked by the following source code (or a substantially similar copy compiled and loaded onto the Accused Products by HTC): AGIS reserves the right to amend these contentions to include additional source code as discovery progresses and as additional source code is made available.</p>

**Exhibit C for U.S. Patent No. 9,445,251 Against HTC Accused Products**

US9445251B2	HTC
	<h2 data-bbox="514 240 1003 293">Contacts Provider</h2> <p data-bbox="514 329 1461 591">The Contacts Provider is a powerful and flexible Android component that manages the device's central repository of data about people. The Contacts Provider is the source of data you see in the device's contacts application, and you can also access its data in your own application and transfer data between the device and online services. The provider accommodates a wide range of data sources and tries to manage as much data as possible for each person, with the result that its organization is complex. Because of this, the provider's API includes an extensive set of contract classes and interfaces that facilitate both data retrieval and modification.</p> <p data-bbox="514 626 823 647">This guide describes the following:</p> <ul data-bbox="514 678 1360 852" style="list-style-type: none"><li data-bbox="514 678 793 699">• The basic provider structure.</li><li data-bbox="514 729 884 750">• How to retrieve data from the provider.</li><li data-bbox="514 779 852 800">• How to modify data in the provider.</li><li data-bbox="514 829 1360 850">• How to write a sync adapter for synchronizing data from your server to the Contacts Provider.</li></ul> <p data-bbox="499 898 1476 927"><a href="https://developer.android.com/guide/topics/providers/contacts-provider.html">https://developer.android.com/guide/topics/providers/contacts-provider.html</a></p>

**Exhibit C for U.S. Patent No. 9,445,251 Against HTC Accused Products**

US9445251B2	HTC
	<p data-bbox="520 245 646 269"><b>Overview</b></p> <p data-bbox="520 298 1598 318">ContactsContract defines an extensible database of contact-related information. Contact information is stored in a three-tier data model:</p> <ul data-bbox="520 342 1703 548" style="list-style-type: none"> <li data-bbox="520 342 1703 394">• A row in the <code>ContactsContract.Data</code> table can store any kind of personal data, such as a phone number or email addresses. The set of data kinds that can be stored in this table is open-ended. There is a predefined set of common kinds, but any application can add its own data kinds.</li> <li data-bbox="520 418 1703 470">• A row in the <code>ContactsContract.RawContacts</code> table represents a set of data describing a person and associated with a single account (for example, one of the user's Gmail accounts).</li> <li data-bbox="520 495 1703 548">• A row in the <code>ContactsContract.Contacts</code> table represents an aggregate of one or more RawContacts presumably describing the same person. When data in or associated with the RawContacts table is changed, the affected aggregate contacts are updated as necessary.</li> </ul> <p data-bbox="520 573 684 592">Other tables include:</p> <ul data-bbox="520 621 1703 906" style="list-style-type: none"> <li data-bbox="520 621 1703 673">• <code>ContactsContract.Groups</code>, which contains information about raw contact groups such as Gmail contact groups. The current API does not support the notion of groups spanning multiple accounts.</li> <li data-bbox="520 698 1314 717">• <code>ContactsContract.StatusUpdates</code>, which contains social status updates including IM availability.</li> <li data-bbox="520 742 1482 761">• <code>ContactsContract.AggregationExceptions</code>, which is used for manual aggregation and disaggregation of raw contacts</li> <li data-bbox="520 786 1314 805">• <code>ContactsContract.Settings</code>, which contains visibility and sync settings for accounts and groups.</li> <li data-bbox="520 829 1335 849">• <code>ContactsContract.SyncState</code>, which contains free-form data maintained on behalf of sync adapters</li> <li data-bbox="520 873 1119 893">• <code>ContactsContract.PhoneLookup</code>, which is used for quick caller-ID lookup</li> </ul> <p data-bbox="499 954 1530 984"><a href="https://developer.android.com/reference/android/provider/ContactsContract.html">https://developer.android.com/reference/android/provider/ContactsContract.html</a></p> <p data-bbox="520 1024 604 1049"><b>Data</b></p> <p data-bbox="520 1094 1734 1235">As noted previously, the data for a raw contact is stored in a <code>ContactsContract.Data</code> row that is linked to the raw contact's <code>_ID</code> value. This allows a single raw contact to have multiple instances of the same type of data such as email addresses or phone numbers. For example, if "Thomas Higginson" for <code>emilyd@gmail.com</code> (the raw contact row for Thomas Higginson associated with the Google account <code>emilyd@gmail.com</code>) has a home email address of <code>thigg@gmail.com</code> and a work email address of <code>thomas.higginson@gmail.com</code>, the Contacts Provider stores the two email address rows and links them both to the raw contact.</p> <p data-bbox="520 1260 1734 1377">Notice that different types of data are stored in this single table. Display name, phone number, email, postal address, photo, and website detail rows are all found in the <code>ContactsContract.Data</code> table. To help manage this, the <code>ContactsContract.Data</code> table has some columns with descriptive names, and others with generic names. The contents of a descriptive-name column have the same meaning regardless of the type of data in the row, while the contents of a generic-name column have different meanings depending on the type of data.</p>

**Exhibit C for U.S. Patent No. 9,445,251 Against HTC Accused Products**

US9445251B2	HTC				
	<a href="https://developer.android.com/guide/topics/providers/contacts-provider.html">https://developer.android.com/guide/topics/providers/contacts-provider.html</a>				
Task	Action	Data	MIME type	Notes	
Pick a contact from a list	<code>ACTION_PICK</code>	One of: <ul style="list-style-type: none"> <li>• <code>Contacts.CONTENT_URI</code>, which displays a list of contacts.</li> <li>• <code>Phone.CONTENT_URI</code>, which displays a list of phone numbers for a raw contact.</li> <li>• <code>StructuredPostal.CONTENT_URI</code>, which displays a list of postal addresses for a raw contact.</li> <li>• <code>Email.CONTENT_URI</code>, which displays a list of email addresses for a raw contact.</li> </ul>	Not used	Displays a list of raw contacts or a list of data from a raw contact, depending on the content URI type you supply. Call <code>startActivityForResult()</code> , which returns the content URI of the selected row. The form of the URI is the table's content URI with the row's <code>LOOKUP_ID</code> appended to it. The device's contacts app delegates read and write permissions to this content URI for the life of your activity. See the <a href="#">Content Provider Basics</a> guide for more details.	
	<a href="https://developer.android.com/guide/topics/providers/contacts-provider.html">https://developer.android.com/guide/topics/providers/contacts-provider.html</a>				



**Exhibit C for U.S. Patent No. 9,445,251 Against HTC Accused Products**

US9445251B2	HTC
	<pre>25  * Parsed form of the intent sent to the Contacts application. 26  */ 27  public class ContactsRequest { 28 29      /** Default mode: browse contacts */ 30      public static final int ACTION_DEFAULT = 10; 31 32      /** Show all contacts */ 33      public static final int ACTION_ALL_CONTACTS = 15; 34 35      /** Show all contacts with phone numbers */ 36      public static final int ACTION_CONTACTS_WITH_PHONES = 17; 37 38      /** Show contents of a specific group */ 39      public static final int ACTION_GROUP = 20; 40 41      /** Show all starred contacts */ 42      public static final int ACTION_STARRED = 30; 43 44      /** Show frequently contacted contacts */ 45      public static final int ACTION_FREQUENT = 40; 46 47      /** Show starred and the frequent */ 48      public static final int ACTION_STREQUENT = 50; 49 50      /** Show all contacts and pick them when clicking */ 51      public static final int ACTION_PICK_CONTACT = 60; 52 53      /** Show all contacts as well as the option to create a new one */ 54      public static final int ACTION_PICK_OR_CREATE_CONTACT = 70; 55 56      /** Show all contacts and pick them for edit when clicking, and allow creating a new contact */ 57      public static final int ACTION_INSERT_OR_EDIT_CONTACT = 80;</pre> <p><a href="https://android.goesource.com/platform/packages/apps/Contacts/+/nougat-mr1-release/src/com/android/contacts/list/ContactsRequest.java">https://android.goesource.com/platform/packages/apps/Contacts/+/nougat-mr1-release/src/com/android/contacts/list/ContactsRequest.java</a></p>

**Exhibit C for U.S. Patent No. 9,445,251 Against HTC Accused Products**

US9445251B2	HTC
	<pre>59     /** Show all phone numbers and pick them when clicking */ 60     public static final int ACTION_PICK_PHONE = 90; 61 62     /** Show all postal addresses and pick them when clicking */ 63     public static final int ACTION_PICK_POSTAL = 100; 64 65     /** Show all postal addresses and pick them when clicking */ 66     public static final int ACTION_PICK_EMAIL = 105; 67 68     /** Show all contacts and create a shortcut for the picked contact */ 69     public static final int ACTION_CREATE_SHORTCUT_CONTACT = 110; 70 71     /** Show all phone numbers and create a call shortcut for the picked number */ 72     public static final int ACTION_CREATE_SHORTCUT_CALL = 120; 73 74     /** Show all phone numbers and create an SMS shortcut for the picked number */ 75     public static final int ACTION_CREATE_SHORTCUT_SMS = 130; 76 77     /** Show all contacts and activate the specified one */ 78     public static final int ACTION_VIEW_CONTACT = 140; 79 80     /** Show contacts recommended for joining with a specified target contact */ 81     public static final int ACTION_PICK_JOIN = 150;</pre> <p><a href="https://android.googlesource.com/platform/packages/apps/Contacts/+/nougat-mr1-release/src/com/android/contacts/list/ContactsRequest.java">https://android.googlesource.com/platform/packages/apps/Contacts/+/nougat-mr1-release/src/com/android/contacts/list/ContactsRequest.java</a></p>

**Exhibit C for U.S. Patent No. 9,445,251 Against HTC Accused Products**

US9445251B2	HTC
	<pre> 104  * Displays a list to browse contacts. 105  */ 106  public class PeopleActivity extends ContactsActivity implements 107      View.OnCreateContextMenuListener, 108      View.OnClickListener, 109      ActionBarAdapter.Listener, 110      DialogManager.DialogShowingViewActivity, 111      ContactListFilterController.ContactListFilterListener, 112      ProviderStatusListener, 113      MultiContactDeleteListener, 114      JoinContactsListener { </pre> <p data-bbox="499 695 1556 764"><a href="https://android.googlesource.com/platform/packages/apps/Contacts/+/nougat-mr1-release/src/com/android/contacts/activities/PeopleActivity.java">https://android.googlesource.com/platform/packages/apps/Contacts/+/nougat-mr1-release/src/com/android/contacts/activities/PeopleActivity.java</a></p> <pre> 145  * Showing a list of Contacts. Also used for showing search results in search mode. 146  */ 147  private MultiSelectContactsListFragment mAllFragment; 148  private ContactTileListFragment mFavoritesFragment; </pre> <p data-bbox="499 976 1556 1045"><a href="https://android.googlesource.com/platform/packages/apps/Contacts/+/nougat-mr1-release/src/com/android/contacts/activities/PeopleActivity.java">https://android.googlesource.com/platform/packages/apps/Contacts/+/nougat-mr1-release/src/com/android/contacts/activities/PeopleActivity.java</a></p>

**Exhibit C for U.S. Patent No. 9,445,251 Against HTC Accused Products**

US9445251B2	HTC
	<pre>458     private void configureFragments(boolean fromRequest) { 459         if (fromRequest) { 460             ContactListFilter filter = null; 461             int actionCode = mRequest.getActionCode(); 462             boolean searchMode = mRequest.isSearchMode(); 463             final int tabToOpen; 464             switch (actionCode) { 465                 case ContactsRequest.ACTION_ALL_CONTACTS: 466                     filter = ContactListFilter.createFilterWithType( 467                         ContactListFilter.FILTER_TYPE_ALL_ACCOUNTS); 468                     tabToOpen = TabState.ALL; 469                     break; 470                 case ContactsRequest.ACTION_CONTACTS_WITH_PHONES: 471                     filter = ContactListFilter.createFilterWithType( 472                         ContactListFilter.FILTER_TYPE_WITH_PHONE_NUMBERS_ONLY); 473                     tabToOpen = TabState.ALL; 474                     break; 475 476                 case ContactsRequest.ACTION_FREQUENT: 477                 case ContactsRequest.ACTION_STREQUENT: 478                 case ContactsRequest.ACTION_STARRED: 479                     tabToOpen = TabState.FAVORITES; 480                     break; 481                 case ContactsRequest.ACTION_VIEW_CONTACT: 482                     tabToOpen = TabState.ALL; 483                     break; 484                 default: 485                     tabToOpen = -1; 486                     break; 487             } </pre> <p><a href="https://android.googlesource.com/platform/packages/apps/Contacts/+nougat-mr1-">https://android.googlesource.com/platform/packages/apps/Contacts/+nougat-mr1-</a></p>

**Exhibit C for U.S. Patent No. 9,445,251 Against HTC Accused Products**

US9445251B2	HTC
	<pre>release/src/com/android/contacts/activities/PeopleActivity.java  488         if (tabToOpen != -1) { 489             mActionBarAdapter.setCurrentTab(tabToOpen); 490         } 491 492         if (filter != null) { 493             mContactListFilterController.setContactListFilter(filter, false); 494             searchMode = false; 495         } 496 497         if (mRequest.getContactUri() != null) { 498             searchMode = false; 499         } 500 501         mActionBarAdapter.setSearchMode(searchMode); 502         configureContactListFragmentForRequest(); 503     } 504 505     configureContactListFragment(); 506 507     invalidateOptionsMenuIfNeeded(); 508 }</pre> <p><a href="https://android.goesource.com/platform/packages/apps/Contacts/+/nougat-mr1-release/src/com/android/contacts/activities/PeopleActivity.java">https://android.goesource.com/platform/packages/apps/Contacts/+/nougat-mr1-release/src/com/android/contacts/activities/PeopleActivity.java</a></p>

**Exhibit C for U.S. Patent No. 9,445,251 Against HTC Accused Products**

US9445251B2	HTC
	<pre data-bbox="512 237 1444 792">35 public class ProfileAndContactsLoader extends CursorLoader { 36 37     private boolean mLoadProfile; 38 39     private String[] mProjection; 40 41     private Uri mExtraUri; 42     private String[] mExtraProjection; 43     private String mExtraSelection; 44     private String[] mExtraSelectionArgs; 45     private boolean mMergeExtraContactsAfterPrimary; 46 47     public ProfileAndContactsLoader(Context context) { 48         super(context); 49     }</pre> <p data-bbox="499 841 1675 914"><a href="https://android.googlesource.com/platform/packages/apps/ContactsCommon/+nougat-mr1-release/src/com/android/contacts/common/list/ProfileAndContactsLoader.java">https://android.googlesource.com/platform/packages/apps/ContactsCommon/+nougat-mr1-release/src/com/android/contacts/common/list/ProfileAndContactsLoader.java</a></p>

**Exhibit C for U.S. Patent No. 9,445,251 Against HTC Accused Products**

US9445251B2	HTC
	<pre>35 public final class GroupMemberLoader extends CursorLoader { 36 37     public static class GroupEditorQuery { 38         private static final String[] PROJECTION = new String[] { 39             Data.CONTACT_ID,           // 0 40             Data.RAW_CONTACT_ID,       // 1 41             Data.DISPLAY_NAME_PRIMARY, // 2 42             Data.PHOTO_URI,           // 3 43             Data.LOOKUP_KEY,          // 4 44         }; 45 46         public static final int CONTACT_ID           = 0; 47         public static final int RAW_CONTACT_ID       = 1; 48         public static final int CONTACT_DISPLAY_NAME_PRIMARY = 2; 49         public static final int CONTACT_PHOTO_URI    = 3; 50         public static final int CONTACT_LOOKUP_KEY   = 4; 51     } 52 53     public static class GroupDetailQuery { 54         private static final String[] PROJECTION = new String[] { 55             Data.CONTACT_ID,           // 0 56             Data.PHOTO_URI,           // 1 57             Data.LOOKUP_KEY,          // 2 58             Data.DISPLAY_NAME_PRIMARY, // 3 59             Data.CONTACT_PRESENCE,     // 4 60             Data.CONTACT_STATUS,      // 5 61         }; 62 63         public static final int CONTACT_ID           = 0; 64         public static final int CONTACT_PHOTO_URI    = 1; 65         public static final int CONTACT_LOOKUP_KEY   = 2; 66         public static final int CONTACT_DISPLAY_NAME_PRIMARY = 3; 67         public static final int CONTACT_PRESENCE_STATUS = 4; 68         public static final int CONTACT_STATUS      = 5; 69     } 70 71     private final long mGroupId;</pre> <p><a href="https://android.googlesource.com/platform/packages/apps/Contacts/+/nougat-mr1-release/src/com/android/contacts/GroupMemberLoader.java">https://android.googlesource.com/platform/packages/apps/Contacts/+/nougat-mr1-release/src/com/android/contacts/GroupMemberLoader.java</a></p>

**Exhibit C for U.S. Patent No. 9,445,251 Against HTC Accused Products**

US9445251B2	HTC
	<pre>24 * Group loader for the group list that includes details such as the number of contacts per group 25 * and number of groups per account. This list is sorted by account type, account name, where the 26 * group names are in alphabetical order. Note that the list excludes default, favorite, and deleted 27 * groups. 28 */ 29 public final class GroupListLoader extends CursorLoader { 30 31     private final static String[] COLUMNS = new String[] { 32         Groups.ACCOUNT_NAME, 33         Groups.ACCOUNT_TYPE, 34         Groups.DATA_SET, 35         Groups._ID, 36         Groups.TITLE, 37         Groups.SUMMARY_COUNT, 38     }; 39 40     public final static int ACCOUNT_NAME = 0; 41     public final static int ACCOUNT_TYPE = 1; 42     public final static int DATA_SET = 2; 43     public final static int GROUP_ID = 3; 44     public final static int TITLE = 4; 45     public final static int MEMBER_COUNT = 5; 46 47     private static final Uri GROUP_LIST_URI = Groups.CONTENT_SUMMARY_URI; 48 49     public GroupListLoader(Context context) { 50         super(context, GROUP_LIST_URI, COLUMNS, Groups.ACCOUNT_TYPE + " NOT NULL AND " 51             + Groups.ACCOUNT_NAME + " NOT NULL AND " + Groups.AUTO_ADD + "=0 AND " + 52             Groups.FAVORITES + "=0 AND " + Groups.DELETED + "=0", null, 53             Groups.ACCOUNT_TYPE + ", " + Groups.ACCOUNT_NAME + ", " + Groups.DATA_SET + ", " + 54             Groups.TITLE + " COLLATE LOCALIZED ASC"); 55     } 56 }</pre> <p><a href="https://android.googlesource.com/platform/packages/apps/Contacts+/nougat-mr1-release/src/com/android/contacts/GroupListLoader.java">https://android.googlesource.com/platform/packages/apps/Contacts+/nougat-mr1-release/src/com/android/contacts/GroupListLoader.java</a></p>



**Exhibit C for U.S. Patent No. 9,445,251 Against HTC Accused Products**

US9445251B2	HTC
	<pre>24 * Group meta-data loader. Loads all groups or just a single group from the 25 * database (if given a {@link Uri}). 26 */ 27 public final class GroupMetaDataLoader extends CursorLoader { 28 29     private final static String[] COLUMNS = new String[] { 30         Groups.ACCOUNT_NAME, 31         Groups.ACCOUNT_TYPE, 32         Groups.DATA_SET, 33         Groups._ID, 34         Groups.TITLE, 35         Groups.AUTO_ADD, 36         Groups.FAVORITES, 37         Groups.GROUP_IS_READ_ONLY, 38         Groups.DELETED, 39     }; 40 41     public final static int ACCOUNT_NAME = 0; 42     public final static int ACCOUNT_TYPE = 1; 43     public final static int DATA_SET = 2; 44     public final static int GROUP_ID = 3; 45     public final static int TITLE = 4; 46     public final static int AUTO_ADD = 5; 47     public final static int FAVORITES = 6; 48     public final static int IS_READ_ONLY = 7; 49     public final static int DELETED = 8; 50 51     public GroupMetaDataLoader(Context context, Uri groupUri) { 52         super(context, ensureIsGroupUri(groupUri), COLUMNS, Groups.ACCOUNT_TYPE + " NOT NULL AND " 53             + Groups.ACCOUNT_NAME + " NOT NULL", null, null); 54     } 55 56     /** 57     * Ensures that this is a valid group URI. If invalid, then an exception is 58     * thrown. Otherwise, the original URI is returned. 59     */ 60     private static Uri ensureIsGroupUri(final Uri groupUri) { 61         // TODO: Fix ContactsProvider2 getType method to resolve the group Uri 62         if (groupUri == null) { 63             throw new IllegalArgumentException("Uri must not be null"); 64         } 65         if (!groupUri.toString().startsWith(Groups.CONTENT_URI.toString())) { 66             throw new IllegalArgumentException("Invalid group Uri: " + groupUri); 67         } 68         return groupUri; 69     } 70 }</pre> <p><a href="https://android.googlesource.com/platform/packages/apps/Contacts/+/nougat-mr1-release/src/com/android/contacts/GroupMetaDataLoader.java">https://android.googlesource.com/platform/packages/apps/Contacts/+/nougat-mr1-release/src/com/android/contacts/GroupMetaDataLoader.java</a></p>

**Exhibit C for U.S. Patent No. 9,445,251 Against HTC Accused Products**

US9445251B2	HTC

**Exhibit C for U.S. Patent No. 9,445,251 Against HTC Accused Products**

US9445251B2	HTC
	<pre>19  * Meta-data for a contact group. We load all groups associated with the contact's 20  * constituent accounts. 21  */ 22  public final class GroupMetaData { 23      private String mAccountName; 24      private String mAccountType; 25      private String mDataSet; 26      private long mGroupId; 27      private String mTitle; 28      private boolean mDefaultGroup; 29      private boolean mFavorites; 30 31      public GroupMetaData(String accountName, String accountType, String dataSet, long groupId, 32                          String title, boolean defaultGroup, boolean favorites) { 33          this.mAccountName = accountName; 34          this.mAccountType = accountType; 35          this.mDataSet = dataSet; 36          this.mGroupId = groupId; 37          this.mTitle = title; 38          this.mDefaultGroup = defaultGroup; 39          this.mFavorites = favorites; 40      } 41 42      public String getAccountName() { 43          return mAccountName; 44      } 45 46      public String getAccountType() { 47          return mAccountType; 48      } 49 50      public String getDataSet() { 51          return mDataSet; 52      } 53 54      public long getGroupId() { 55          return mGroupId; 56      } 57 58      public String getTitle() { 59          return mTitle; 60      } 61 62      public boolean isDefaultGroup() { 63          return mDefaultGroup; 64      } 65 66      public boolean isFavorites() { 67          return mFavorites; 68      } 69  }</pre> <p><a href="https://android.googlesource.com/platform/packages/apps/ContactsCommon/+nougat-mr1-">https://android.googlesource.com/platform/packages/apps/ContactsCommon/+nougat-mr1-</a></p>

**Exhibit C for U.S. Patent No. 9,445,251 Against HTC Accused Products**

US9445251B2	HTC
	<p data-bbox="499 237 1318 266"><a href="https://android.googlesource.com/platform/packages/apps/Messaging/+nougat-mr1-release/src/com/android/contacts/common/GroupMetaData.java">release/src/com/android/contacts/common/GroupMetaData.java</a></p> <pre data-bbox="520 513 1591 1333">44 * Class that sends chat message via SMS. 45 * 46 * The interface emulates a blocking sending similar to making an HTTP request. 47 * It calls the SmsManager to send a (potentially multipart) message and waits 48 * on the sent status on each part. The waiting has a timeout so it won't wait 49 * forever. Once the sent status of all parts received, the call returns. 50 * A successful sending requires success status for all parts. Otherwise, we 51 * pick the highest level of failure as the error for the whole message, which 52 * is used to determine if we need to retry the sending. 53 */ 54 public class SmsSender { 55     private static final String TAG = LogUtil.BUGLE_TAG; 56 57     public static final String EXTRA_PART_ID = "part_id"; 58 59     /* 60     * A map for pending sms messages. The key is the random request UUID. 61     */ 62     private static ConcurrentHashMap&lt;Uri, SendResult&gt; sPendingMessageMap = 63         new ConcurrentHashMap&lt;Uri, SendResult&gt;(); 64 65     private static final Random RANDOM = new Random(); 66 67     // Whether we should send multipart SMS as separate messages 68     private static Boolean sSendMultipartSmsAsSeparateMessages = null; 69</pre> <p data-bbox="499 1377 1583 1406"><a href="https://android.googlesource.com/platform/packages/apps/Messaging/+nougat-mr1-">https://android.googlesource.com/platform/packages/apps/Messaging/+nougat-mr1-</a></p>

**Exhibit C for U.S. Patent No. 9,445,251 Against HTC Accused Products**

US9445251B2	HTC
	<p data-bbox="499 237 1220 269"><a href="https://android.googlesource.com/platform/packages/apps/Messaging/+/nougat-mr1-release/src/com/android/messaging/sms/SmsSender.java">release/src/com/android/messaging/sms/SmsSender.java</a></p> <pre data-bbox="514 310 1732 1308">253     // Actually sending the message using SmsManager 254     private static void sendInternal(final Context context, final int subId, String dest, 255         final ArrayList&lt;String&gt; messages, final String serviceCenter, 256         final boolean requireDeliveryReport, final Uri messageUri) throws SmsException { 257     Assert.notNull(context); 258     final SmsManager smsManager = PhoneUtils.get(subId).getSmsManager(); 259     final int messageCount = messages.size(); 260     final ArrayList&lt;PendingIntent&gt; deliveryIntents = new ArrayList&lt;PendingIntent&gt;(messageCount); 261     final ArrayList&lt;PendingIntent&gt; sentIntents = new ArrayList&lt;PendingIntent&gt;(messageCount); 262     for (int i = 0; i &lt; messageCount; i++) { 263         // Make pending intents different for each message part 264         final int partId = (messageCount &lt;= 1 ? 0 : i + 1); 265         if (requireDeliveryReport &amp;&amp; (i == (messageCount - 1))) { 266             // TODO we only care about the delivery status of the last part 267             // Shall we have better tracking of delivery status of all parts? 268             deliveryIntents.add(PendingIntent.getBroadcast( 269                 context, 270                 partId, 271                 getSendStatusIntent(context, SendStatusReceiver.MESSAGE_DELIVERED_ACTION, 272                     messageUri, partId, subId), 273                 0/*flag*/)); 274         } else { 275             deliveryIntents.add(null); 276         } 277         sentIntents.add(PendingIntent.getBroadcast( 278             context, 279             partId, 280             getSendStatusIntent(context, SendStatusReceiver.MESSAGE_SENT_ACTION, 281                 messageUri, partId, subId), 282             0/*flag*/)); 283     } 284     if (sSendMultipartSmsAsSeparateMessages == null) { 285         sSendMultipartSmsAsSeparateMessages = MmsConfig.get(subId) 286             .getSendMultipartSmsAsSeparateMessages(); 287     }</pre> <p data-bbox="499 1352 1583 1385"><a href="https://android.googlesource.com/platform/packages/apps/Messaging/+/nougat-mr1-">https://android.googlesource.com/platform/packages/apps/Messaging/+/nougat-mr1-</a></p>

**Exhibit C for U.S. Patent No. 9,445,251 Against HTC Accused Products**

US9445251B2	HTC
	<p data-bbox="499 235 1222 263"><a href="https://android.googlesource.com/platform/packages/apps/Messaging/+nougat-mr1-release/src/com/android/messaging/sms/SmsSender.java">release/src/com/android/messaging/sms/SmsSender.java</a></p> <pre data-bbox="512 305 1671 870">288     try { 289         if (sSendMultipartSmsAsSeparateMessages) { 290             // If multipart sms is not supported, send them as separate messages 291             for (int i = 0; i &lt; messageCount; i++) { 292                 smsManager.sendTextMessage(dest, 293                     serviceCenter, 294                     messages.get(i), 295                     sentIntents.get(i), 296                     deliveryIntents.get(i)); 297             } 298         } else { 299             smsManager.sendMultipartTextMessage( 300                 dest, serviceCenter, messages, sentIntents, deliveryIntents); 301         } 302     } catch (final Exception e) { 303         throw new SmsException("SmsSender: caught exception in sending " + e); 304     } 305 }</pre> <p data-bbox="499 917 1583 987"><a href="https://android.googlesource.com/platform/packages/apps/Messaging/+nougat-mr1-release/src/com/android/messaging/sms/SmsSender.java">https://android.googlesource.com/platform/packages/apps/Messaging/+nougat-mr1-release/src/com/android/messaging/sms/SmsSender.java</a></p>

**Exhibit C for U.S. Patent No. 9,445,251 Against HTC Accused Products**

US9445251B2	HTC
	<pre>56 * Class that receives incoming SMS messages through android.provider.Telephony.SMS_RECEIVED 57 * 58 * This class serves two purposes: 59 * - Process phone verification SMS messages 60 * - Handle SMS messages when the user has enabled us to be the default SMS app (Pre-KLP) 61 */ 62 public final class SmsReceiver extends BroadcastReceiver { 63     private static final String TAG = LogUtil.BUGLE_TAG; 64 65     private static ArrayList&lt;Pattern&gt; sIgnoreSmsPatterns; 66</pre> <p><a href="https://android.gogglesource.com/platform/packages/apps/Messaging/+nougat-mr1-release/src/com/android/messaging/receiver/SmsReceiver.java">https://android.gogglesource.com/platform/packages/apps/Messaging/+nougat-mr1-release/src/com/android/messaging/receiver/SmsReceiver.java</a></p>

**Exhibit C for U.S. Patent No. 9,445,251 Against HTC Accused Products**

US9445251B2	HTC
	<pre>200     public static void deliverSmsMessages(final Context context, final int subId, 201         final int errorCode, final android.telephony.SmsMessage[] messages) { 202         final ContentValues messageValues = 203             MmsUtils.parseReceivedSmsMessage(context, messages, errorCode); 204 205         LogUtil.v(TAG, "SmsReceiver.deliverSmsMessages"); 206 207         final long nowInMillis = System.currentTimeMillis(); 208         final long receivedTimestampMs = MmsUtils.getMessageDate(messages[0], nowInMillis); 209 210         messageValues.put(Sms.Inbox.DATE, receivedTimestampMs); 211         // Default to unread and unseen for us but ReceiveSmsMessageAction will override 212         // seen for the telephony db. 213         messageValues.put(Sms.Inbox.READ, 0); 214         messageValues.put(Sms.Inbox.SEEN, 0); 215         if (OsUtil.isAtLeastLMR1()) { 216             messageValues.put(Sms.SUBSCRIPTION_ID, subId); 217         } 218 219         if (messages[0].getMessageClass() == android.telephony.SmsMessage.MessageClass.CLASS_0    220             DebugUtils.debugClassZeroSmsEnabled()) { 221             Factory.get().getUIIntents().launchClassZeroActivity(context, messageValues); 222         } else { 223             final ReceiveSmsMessageAction action = new ReceiveSmsMessageAction(messageValues); 224             action.start(); 225         } 226     }</pre> <p data-bbox="499 1063 1583 1138"><a href="https://android.googlesource.com/platform/packages/apps/Messaging/+nougat-mr1-release/src/com/android/messaging/receiver/SmsReceiver.java">https://android.googlesource.com/platform/packages/apps/Messaging/+nougat-mr1-release/src/com/android/messaging/receiver/SmsReceiver.java</a></p>



**Exhibit C for U.S. Patent No. 9,445,251 Against HTC Accused Products**

US9445251B2	HTC
	<pre>228     @Override 229     public void onReceive(final Context context, final Intent intent) { 230         LogUtil.v(TAG, "SmsReceiver.onReceive " + intent); 231         // On KLP+ we only take delivery of SMS messages in SmsDeliverReceiver. 232         if (PhoneUtils.getDefault().isSmsEnabled()) { 233             final String action = intent.getAction(); 234             if (OsUtil.isSecondaryUser() &amp;&amp; 235                 (Telephony.Sms.Intents.SMS_RECEIVED_ACTION.equals(action)    236                  // TODO: update this with the actual constant from Telephony 237                  "android.provider.Telephony.MMS_DOWNLOADED".equals(action))) { 238                 postNewMessageSecondaryUserNotification(); 239             } else if (!OsUtil.isAtLeastKLP()) { 240                 deliverSmsIntent(context, intent); 241             } 242         } 243     }</pre> <p><a href="https://android.goesource.com/platform/packages/apps/Messaging/+nougat-mr1-release/src/com/android/messaging/receiver/SmsReceiver.java">https://android.goesource.com/platform/packages/apps/Messaging/+nougat-mr1-release/src/com/android/messaging/receiver/SmsReceiver.java</a></p>

**Exhibit C for U.S. Patent No. 9,445,251 Against HTC Accused Products**

US9445251B2	HTC
	<pre>52  * Class that sends chat message via MMS. 53  * 54  * The interface emulates a blocking send similar to making an HTTP request. 55  */ 56  public class MmsSender { 57      private static final String TAG = LogUtil.BUGLE_TAG; 58 59      /** 60       * Send an MMS message. 61       * 62       * @param context Context 63       * @param messageUri The unique URI of the message for identifying it during sending 64       * @param sendReq The SendReq PDU of the message 65       * @throws MmsFailureException 66       */ 67      public static void sendMms(final Context context, final int subId, final Uri messageUri, 68                               final SendReq sendReq, final Bundle sentIntentExtras) throws MmsFailureException { 69          sendMms(context, 70                subId, 71                messageUri, 72                null /* locationUri */, 73                sendReq, 74                true /* responseImportant */, 75                sentIntentExtras); 76      }</pre> <p><a href="https://android.googlesource.com/platform/packages/apps/Messaging/+nougat-mr1-release/src/com/android/messaging/sms/MmsSender.java">https://android.googlesource.com/platform/packages/apps/Messaging/+nougat-mr1-release/src/com/android/messaging/sms/MmsSender.java</a></p>

**Exhibit C for U.S. Patent No. 9,445,251 Against HTC Accused Products**

US9445251B2	HTC
	<pre>240     * Download an MMS message. 241     * 242     * @param context Context 243     * @param contentLocation The url of the MMS message 244     * @throws MmsFailureException 245     * @throws InvalidHeaderValueException 246     */ 247     public static void downloadMms(final Context context, final int subId, 248         final String contentLocation, Bundle extras) throws MmsFailureException, 249         InvalidHeaderValueException { 250         final Uri requestUri = Uri.parse(contentLocation); 251         final Uri contentUri = MmsFileProvider.buildRawMmsUri(); 252 253         final Intent downloadedIntent = new Intent(SendStatusReceiver.MMS_DOWNLOADED_ACTION, 254             requestUri, 255             context, 256             SendStatusReceiver.class); 257         downloadedIntent.putExtra(SendMessageAction.EXTRA_CONTENT_URI, contentUri); 258         if (extras != null) { 259             downloadedIntent.putExtras(extras); 260         } 261         final PendingIntent downloadedPendingIntent = PendingIntent.getBroadcast( 262             context, 263             0 /*request code*/, 264             downloadedIntent, 265             PendingIntent.FLAG_UPDATE_CURRENT); 266 267         MmsManager.downloadMultimediaMessage(subId, context, contentLocation, contentUri, 268             downloadedPendingIntent); 269     }</pre> <p><a href="https://android.goesource.com/platform/packages/apps/Messaging/+nougat-mr1-release/src/com/android/messaging/sms/MmsSender.java">https://android.goesource.com/platform/packages/apps/Messaging/+nougat-mr1-release/src/com/android/messaging/sms/MmsSender.java</a></p>

**Exhibit C for U.S. Patent No. 9,445,251 Against HTC Accused Products**

US9445251B2	HTC
	<pre> 97      * Execute an MMS HTTP request, either a POST (sending) or a GET (downloading) 98      * 99      * @param urlString The request URL, for sending it is usually the MMSC, and for downloading 100     *           it is the message URL 101     * @param pdu For POST (sending) only, the PDU to send 102     * @param method HTTP method, POST for sending and GET for downloading 103     * @param isProxySet Is there a proxy for the MMSC 104     * @param proxyHost The proxy host 105     * @param proxyPort The proxy port 106     * @param mmsConfig The MMS config to use 107     * @param userAgent The user agent header value 108     * @param uaProfUrl The UA Prof URL header value 109     * @return The HTTP response body 110     * @throws MmsHttpException For any failures 111     */ 112     public byte[] execute(String urlString, byte[] pdu, String method, boolean isProxySet, 113                          String proxyHost, int proxyPort, Bundle mmsConfig, String userAgent, String uaProfUrl) 114         throws MmsHttpException { 115         Log.d(MmsService.TAG, "HTTP: " + method + " " + Utils.redactUrlForNonVerbose(urlString) 116             + (isProxySet ? (" proxy=" + proxyHost + ":" + proxyPort) : "") 117             + ", PDU size=" + (pdu != null ? pdu.length : 0)); 118         checkMethod(method); 119         HttpURLConnection connection = null; 120         try { 121             Proxy proxy = Proxy.NO_PROXY; 122             if (isProxySet) { 123                 proxy = new Proxy(Proxy.Type.HTTP, new InetSocketAddress(proxyHost, proxyPort)); 124             } 125             final URL url = new URL(urlString); 126             // Now get the connection 127             connection = (HttpURLConnection) url.openConnection(proxy); 128             connection.setDoInput(true); 129             connection.setConnectTimeout( 130                 mmsConfig.getInt(CarrierConfigValuesLoader.CONFIG_HTTP_SOCKET_TIMEOUT, 131                                 CarrierConfigValuesLoader.CONFIG_HTTP_SOCKET_TIMEOUT_DEFAULT)); </pre> <p data-bbox="499 1256 1583 1328"><a href="https://android.googlesource.com/platform/packages/apps/Messaging+/nougat-mr1-release/src/android/support/v7/mms/MmsHttpClient.java">https://android.googlesource.com/platform/packages/apps/Messaging+/nougat-mr1-release/src/android/support/v7/mms/MmsHttpClient.java</a></p>

**Exhibit C for U.S. Patent No. 9,445,251 Against HTC Accused Products**

US9445251B2	HTC
	<pre>132 // ----- COMMON HEADERS ----- 133 // Header: Accept 134 connection.setRequestProperty(HEADER_ACCEPT, HEADER_VALUE_ACCEPT); 135 // Header: Accept-Language 136 connection.setRequestProperty( 137     HEADER_ACCEPT_LANGUAGE, getCurrentAcceptLanguage(Locale.getDefault())); 138 // Header: User-Agent 139 Log.i(MmsService.TAG, "HTTP: User-Agent=" + userAgent); 140 connection.setRequestProperty(HEADER_USER_AGENT, userAgent); 141 // Header: x-wap-profile 142 final String uaProfUrlTagName = mmsConfig.getString( 143     CarrierConfigValuesLoader.CONFIG_UA_PROF_TAG_NAME, 144     CarrierConfigValuesLoader.CONFIG_UA_PROF_TAG_NAME_DEFAULT); 145 if (uaProfUrl != null) { 146     Log.i(MmsService.TAG, "HTTP: UaProfUrl=" + uaProfUrl); 147     connection.setRequestProperty(uaProfUrlTagName, uaProfUrl); 148 } 149 // Add extra headers specified by mms_config.xml's httpparams 150 addExtraHeaders(connection, mmsConfig); 151 // Different stuff for GET and POST 152 if (METHOD_POST.equals(method)) { 153     if (pdu == null    pdu.length &lt; 1) { 154         Log.e(MmsService.TAG, "HTTP: empty pdu"); 155         throw new MmsHttpException(0/*statusCode*/, "Sending empty PDU"); 156     } 157     connection.setDoOutput(true); 158     connection.setRequestMethod(METHOD_POST); 159     if (mmsConfig.getBoolean( 160         CarrierConfigValuesLoader.CONFIG_SUPPORT_HTTP_CHARSET_HEADER, 161         CarrierConfigValuesLoader.CONFIG_SUPPORT_HTTP_CHARSET_HEADER_DEFAULT)) { 162         connection.setRequestProperty(HEADER_CONTENT_TYPE, 163             HEADER_VALUE_CONTENT_TYPE_WITH_CHARSET); 164     } else { 165         connection.setRequestProperty(HEADER_CONTENT_TYPE, 166             HEADER_VALUE_CONTENT_TYPE_WITHOUT_CHARSET); 167     } 168 }</pre> <p><a href="https://android.goesource.com/platform/packages/apps/Messaging/+/nougat-mr1-release/src/android/support/v7/mms/MmsHttpClient.java">https://android.goesource.com/platform/packages/apps/Messaging/+/nougat-mr1-release/src/android/support/v7/mms/MmsHttpClient.java</a></p>

**Exhibit C for U.S. Patent No. 9,445,251 Against HTC Accused Products**

US9445251B2	HTC
	<pre> 167         } 168         if (Log.isLoggable(MmsService.TAG, Log.VERBOSE)) { 169             logHttpHeaders(connection.getRequestProperties()); 170         } 171         connection.setFixedLengthStreamingMode(pdu.length); 172         // Sending request body 173         final OutputStream out = 174             new BufferedOutputStream(connection.getOutputStream()); 175         out.write(pdu); 176         out.flush(); 177         out.close(); 178     } else if (METHOD_GET.equals(method)) { 179         if (Log.isLoggable(MmsService.TAG, Log.VERBOSE)) { 180             logHttpHeaders(connection.getRequestProperties()); 181         } 182         connection.setRequestMethod(METHOD_GET); 183     } 184     // Get response 185     final int responseCode = connection.getResponseCode(); 186     final String responseMessage = connection.getResponseMessage(); 187     Log.d(MmsService.TAG, "HTTP: " + responseCode + " " + responseMessage); 188     if (Log.isLoggable(MmsService.TAG, Log.VERBOSE)) { 189         logHttpHeaders(connection.getHeaderFields()); 190     } 191     if (responseCode / 100 != 2) { 192         throw new MmsHttpException(responseCode, responseMessage); 193     } 194     final InputStream in = new BufferedInputStream(connection.getInputStream()); 195     final ByteArrayOutputStream byteOut = new ByteArrayOutputStream(); 196     final byte[] buf = new byte[4096]; 197     int count = 0; 198     while ((count = in.read(buf)) &gt; 0) { 199         byteOut.write(buf, 0, count); 200     } 201     in.close(); 202     final byte[] responseBody = byteOut.toByteArray(); 203     Log.d(MmsService.TAG, "HTTP: response size=" 204         + (responseBody != null ? responseBody.length : 0)); 205     return responseBody; </pre> <p><a href="https://android.googlesource.com/platform/packages/apps/Messaging/+/nougat-mr1-">https://android.googlesource.com/platform/packages/apps/Messaging/+/nougat-mr1-</a></p>

**Exhibit C for U.S. Patent No. 9,445,251 Against HTC Accused Products**

US9445251B2	HTC
	<a href="release/src/android/support/v7/mms/MmsHttpClient.java">release/src/android/support/v7/mms/MmsHttpClient.java</a>

**Exhibit C for U.S. Patent No. 9,445,251 Against HTC Accused Products**

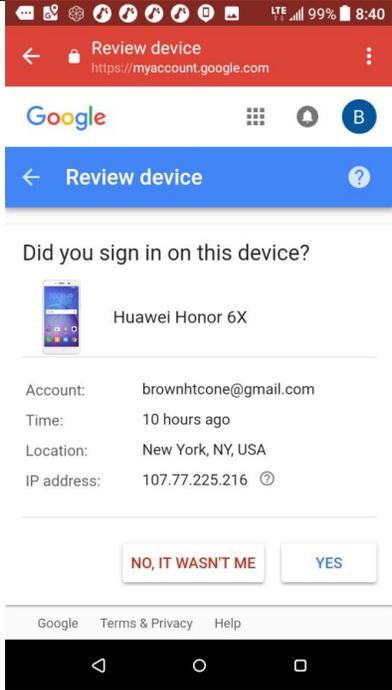
US9445251B2	HTC
	<pre> 38  * Request to send an MMS 39  */ 40  class SendRequest extends MmsRequest { 41      // Max send response PDU size in bytes (exceeding this may cause problem with 42      // system intent delivery). 43      private static final int MAX_SEND_RESPONSE_SIZE = 1000 * 1024; 44 45      private byte[] mPduData; 46 47      SendRequest(final String locationUri, final Uri pduUri, final PendingIntent sentIntent) { 48          super(locationUri, pduUri, sentIntent); 49      } 50 51      @Override 52      protected boolean loadRequest(final Context context, final Bundle mmsConfig) { 53          mPduData = readPduFromContentUri( 54              context, 55              mPduUri, 56              mmsConfig.getInt( 57                  CarrierConfigValuesLoader.CONFIG_MAX_MESSAGE_SIZE, 58                  CarrierConfigValuesLoader.CONFIG_MAX_MESSAGE_SIZE_DEFAULT)); 59          return (mPduData != null); 60      } 61 62      @Override 63      protected boolean transferResponse(final Context context, final Intent fillIn, 64          final byte[] response) { 65          // SendConf pdus are always small and can be included in the intent 66          if (response != null &amp;&amp; fillIn != null) { 67              if (response.length &gt; MAX_SEND_RESPONSE_SIZE) { 68                  // If the response PDU is too large, it won't be able to fit in 69                  // the PendingIntent to be transferred via system IPC. 70                  return false; 71              } 72              fillIn.putExtra(SmsManager.EXTRA_MMS_DATA, response); 73          } 74          return true; 75      } </pre>



**Exhibit C for U.S. Patent No. 9,445,251 Against HTC Accused Products**

<b>US9445251B2</b>	<b>HTC</b>
<p>[1B] based on receiving the message from the second device, participating in the group, wherein participating in the group includes sending first location information to a server and receiving second location information from the server, the first location information comprising a location of the first device, the second location information comprising a plurality of locations of a respective plurality of second devices included in the group;</p>	<p>HTC infringes directly and/or indirectly by performing, inducing others to perform, and/or contributing to the performance of: based on receiving the message from the second device, participating in the group, wherein participating in the group includes sending first location information to a server and receiving second location information from the server, the first location information comprising a location of the first device, the second location information comprising a plurality of locations of a respective plurality of second devices included in the group.</p> <p><b><u>Regarding Find My Device</u></b> and Android Device Manager, the Accused Products require a user to join the corresponding network by: signing-in to the device with an identifier (e.g., Google Account) or linking the device to the identifier by remote means. The Accused Products require that a signed-in user of a first device to share its location by enabling Location Access or Location Services. When a user of a first device associated with an identity (e.g. Google Account) enables the Find My Device feature and Location Services on the first device, the user shares its location and the first device's location is sent to a server. When the user makes a request to view a map of device locations associated with the identity, the device receives one or more locations corresponding to one or more second devices associated with the identity (e.g. when multiple phones are logged into the same account). Alternatively, the locations corresponding to one or more second devices associated with the identity are sent to the first device on a rolling basis.</p> <p>The first device's participation in the group is based on receiving the message from the second device, wherein the message indicates that second device has logged into the same Google account:</p> <p>e.g.,</p>

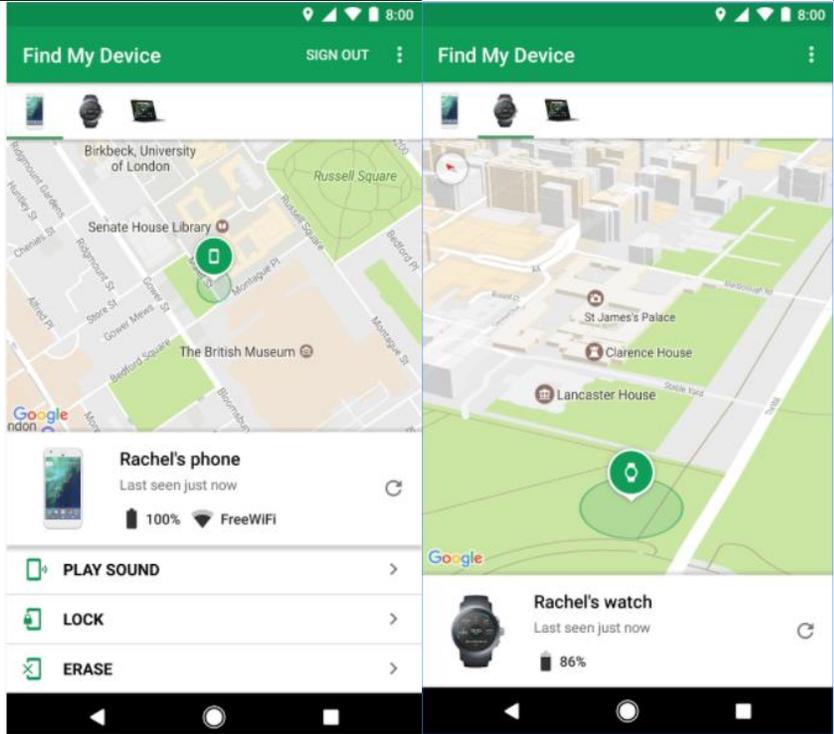
**Exhibit C for U.S. Patent No. 9,445,251 Against HTC Accused Products**

US9445251B2	HTC
	 <p>By participating in the Find My Device program, the device sends location information to a server (e.g., a network server provided by an ISP such as AT&amp;T and/or a server running Google’s services). The device also receives location information from the server indicating the location of other devices that are logged into the Google account.</p> <p><b>Regarding Google Maps,</b> Google Latitude, Google Plus, Google Hangouts, Google Messages, Google Allo, Google Duo, Google Chrome, and Android Messenger, the Accused Products require a user to join the corresponding network by signing-in to the device with an identifier (e.g., Google Account). When the signed-in user enables Google Maps Share Location and Location Services on the first device, the user shares its location and the first device’s location is sent to a server. When the user enables sharing to one or more contacts (of respective devices) and the one or more contacts enable sharing their location to the user of the</p>

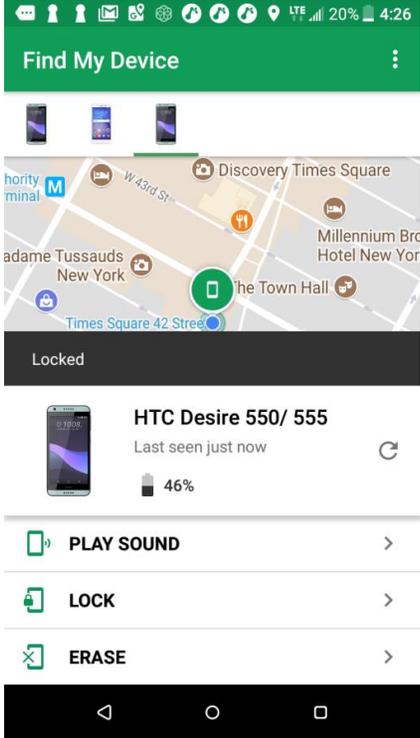
**Exhibit C for U.S. Patent No. 9,445,251 Against HTC Accused Products**

US9445251B2	HTC
	<p>first device, the user of the first device receives the locations of the one or more contacts.</p> <p>The first device's participation in the group is based on receiving the message from the second device, i.e. a message indicating that the second device is sharing its location.</p> <p>By participating in the Maps location sharing functionality, the device sends location information to a server (e.g., a network server provided by an ISP such as AT&amp;T and/or a server running Google's services). The device also receives location information from the server indicating the location of other devices that are sharing location information via Maps.</p> <p><b><u>Further regarding Google Maps</u></b>, Google Latitude, Google Plus, Google Hangouts, Google Messages, Google Allo, Google Duo, Google Chrome, and Android Messenger, the Accused Products alternatively require a user to send a request containing the identifier (e.g., Google Account). When the signed-in user enables Google Maps Share Location and Location Services on the first device, the user shares its location and the first device's location is sent to a server. When the user sends a message to another contact through Google Maps, Google Messages, and/or another means from within the Google Maps application, the message including location information are sent to a server before transmission to the intended contact. When one or more contacts enable sharing their location to the user of the first device, or alternatively send a message containing location information, or alternatively accept a request to share their location with the first user, the user of the first device receives the locations of the one or more contacts.</p> <p><b><u>Exemplary Support for Find My Device:</u></b></p>

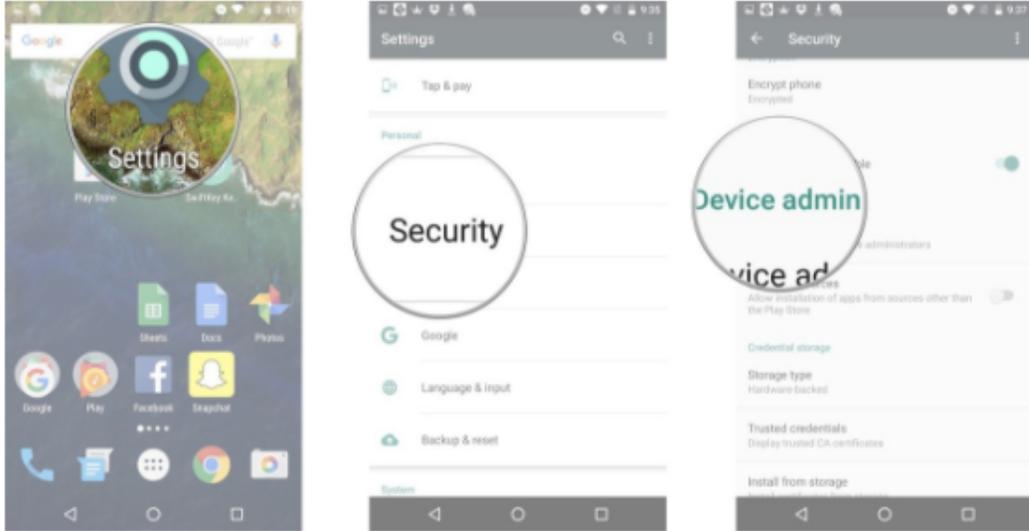
### Exhibit C for U.S. Patent No. 9,445,251 Against HTC Accused Products

US9445251B2	HTC
	 <p><a href="https://play.google.com/store/apps/details?id=com.google.android.apps.adm&amp;hl=en">https://play.google.com/store/apps/details?id=com.google.android.apps.adm&amp;hl=en</a></p>

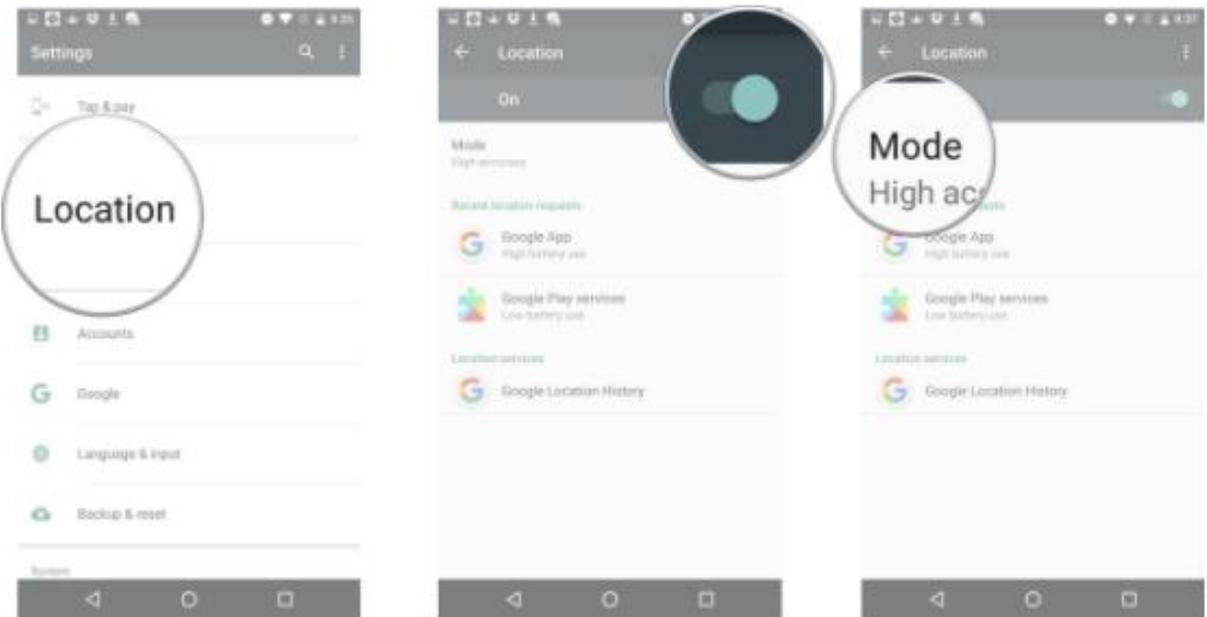
**Exhibit C for U.S. Patent No. 9,445,251 Against HTC Accused Products**

US9445251B2	HTC
	

**Exhibit C for U.S. Patent No. 9,445,251 Against HTC Accused Products**

US9445251B2	HTC
	<p data-bbox="506 248 1329 285"><b>How to enable Find My Device on your phone</b></p> <p data-bbox="506 326 1528 444">In newer Android phones, the Find My Device service is already located conveniently in your Settings app, but if you can't find it you can always <a href="#">download Find My Device from the Google Play Store</a>. This locating service has essentially amalgamated with Google to make finding your phone easier. There are just a couple of things you'll need to activate.</p> <ol data-bbox="506 493 821 623" style="list-style-type: none"><li>1. Launch <b>Settings</b>.</li><li>2. Tap <b>Security</b>.</li><li>3. Tap <b>Device Administration</b>.</li></ol> <div data-bbox="541 659 1570 1190"></div> <p data-bbox="506 1247 1226 1271">4. Tap <b>Find My Device</b> so that a checkmark appears in the checkbox.</p> <p data-bbox="506 1320 1251 1349"><a href="https://www.androidcentral.com/how-track-android-phone">https://www.androidcentral.com/how-track-android-phone</a></p>

### Exhibit C for U.S. Patent No. 9,445,251 Against HTC Accused Products

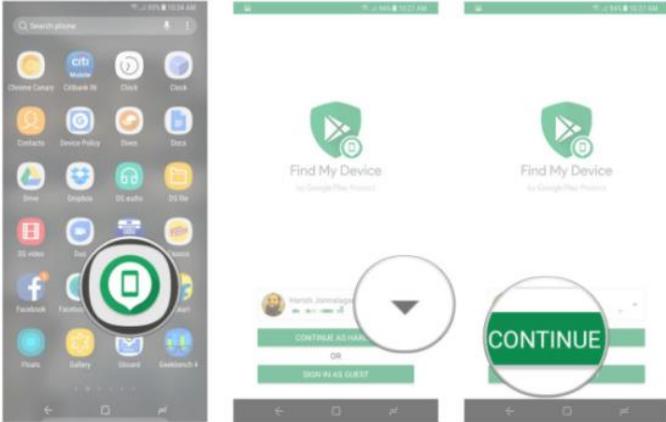
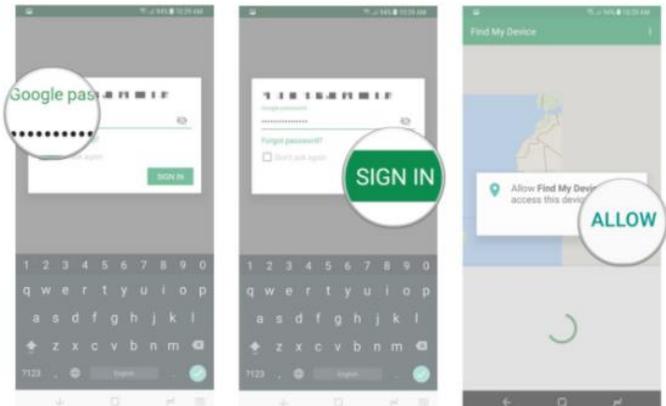
US9445251B2	HTC
	<p data-bbox="520 245 1451 402"><b>7.</b> Tap <b>Location</b> in the main Settings menu. <b>8.</b> Tap the <b>switch</b> beside <b>Location</b> at the top of the screen so that it turns on. <b>9.</b> Tap <b>Mode</b>.</p>  <p data-bbox="499 1096 1249 1128"><a href="https://www.androidcentral.com/how-track-android-phone">https://www.androidcentral.com/how-track-android-phone</a></p>

**Exhibit C for U.S. Patent No. 9,445,251 Against HTC Accused Products**

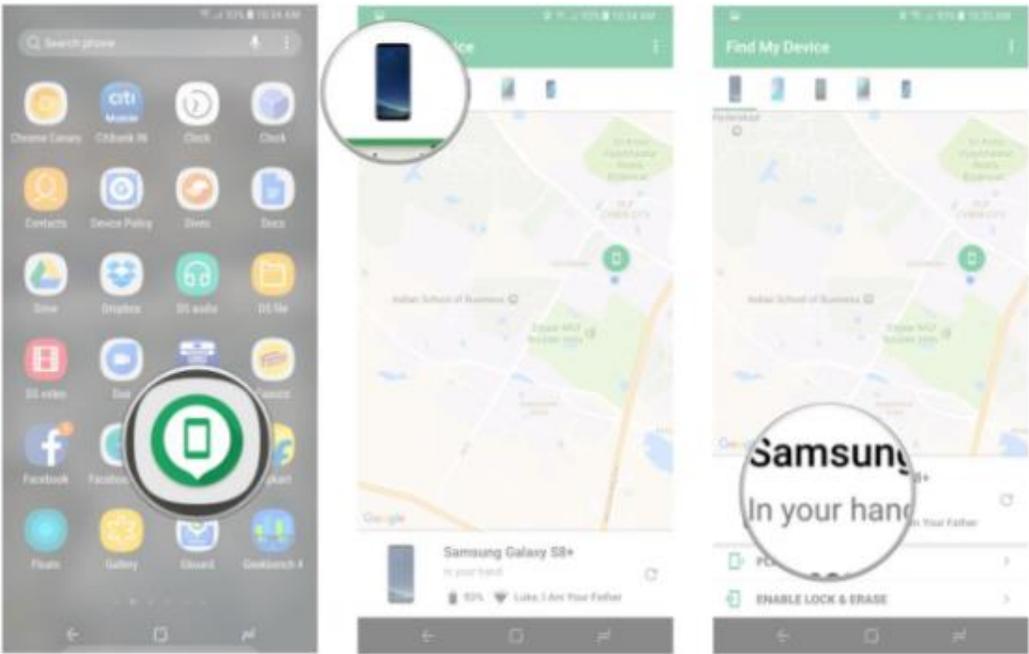
US9445251B2	HTC
	<p data-bbox="520 240 1167 277"><b>How to locate your phone with Google</b></p> <p data-bbox="520 310 1482 363">Should you happen to lose your phone, you can locate its whereabouts by logging into your Google account from any computer or even from another phone.</p> <ol data-bbox="520 407 1241 537" style="list-style-type: none"> <li data-bbox="520 407 1100 431">1. Launch a <b>web browser</b> from a phone, tablet, or computer.</li> <li data-bbox="520 456 1241 480">2. Navigate to <b>Google</b> if it is not your default search engine or home page.</li> <li data-bbox="520 505 1073 529">3. Type <b>find my phone android</b> in the Google search bar.</li> </ol> <div data-bbox="548 565 1514 1133"> <p>The image contains three sequential screenshots from an Android phone. The first screenshot shows the home screen with various app icons; the Chrome browser icon is circled in red. The second screenshot shows the Google search page with the search bar containing the word 'google' circled in red. The third screenshot shows search results for 'find my phone android', with the first result, 'find my phone android', circled in red.</p> </div> <ol data-bbox="520 1182 1514 1317" style="list-style-type: none"> <li data-bbox="520 1182 1136 1206">4. Tap on <b>Find My Device</b> (usually the first option in the search).</li> <li data-bbox="520 1230 1514 1317">5. Enter your <b>email address</b> and <b>password</b> just as though you were checking your email. If you have 2-step verification set up on your Google account (and you most certainly should), you'll need to complete that process as well.</li> </ol> <p data-bbox="499 1325 1247 1357"><a href="https://www.androidcentral.com/how-track-android-phone">https://www.androidcentral.com/how-track-android-phone</a></p>



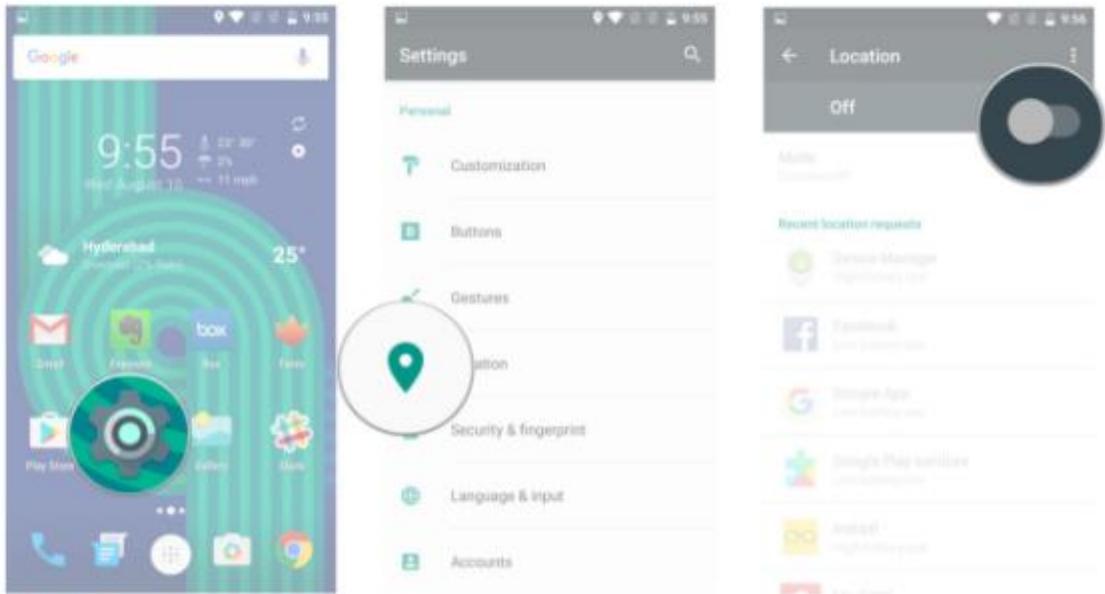
# Exhibit C for U.S. Patent No. 9,445,251 Against HTC Accused Products

US9445251B2	HTC
	<ol style="list-style-type: none"><li>1. Open Find Device from your home screen or app drawer.</li><li>2. Select the Google account you want to use the service with.</li><li>3. Hit the Continue as button.</li></ol>  <ol style="list-style-type: none"><li>4. Enter your Google account password.</li><li>5. Tap Sign in.</li><li>6. Give location access to the service.</li></ol>  <p><a href="https://www.androidcentral.com/find-my-device">https://www.androidcentral.com/find-my-device</a></p>

**Exhibit C for U.S. Patent No. 9,445,251 Against HTC Accused Products**

US9445251B2	HTC
	<p data-bbox="514 245 1533 337">Once you're signed in to Find My Device, you'll see a map with your current location as well as the make and model of your phone, and two options — Play Sound, and Enable Lock &amp; Erase. Hitting the latter option will allow you to start using the Lock and Erase functions.</p> <p data-bbox="514 375 1545 431">If you've signed into more than one phone, you can select a particular device by browsing the list at the top of the screen.</p> <ol data-bbox="514 480 1230 613" style="list-style-type: none"> <li>1. Open <b>Find My Device</b> from your home screen or app drawer.</li> <li>2. Select your phone from the <b>list of devices</b> at the top of the screen.</li> <li>3. See if your phone is <b>discoverable</b>.</li> </ol>  <p data-bbox="499 1325 1121 1357"><a href="https://www.androidcentral.com/find-my-device">https://www.androidcentral.com/find-my-device</a></p>

**Exhibit C for U.S. Patent No. 9,445,251 Against HTC Accused Products**

US9445251B2	HTC
	<p data-bbox="506 240 1604 337">If you're not able to find your phone or if it says that the device is unavailable, it is likely that the location services are disabled. Find My Device relies on GPS to track your phone, so now would be a good time to enable location services.</p> <ol data-bbox="506 386 1136 532" style="list-style-type: none"><li>1. Open <b>Settings</b> from your home screen or app drawer.</li><li>2. Tap <b>Location</b>.</li><li>3. Toggle <b>Enable location services</b>.</li></ol>  <p data-bbox="506 1214 1121 1252"><a href="https://www.androidcentral.com/find-my-device">https://www.androidcentral.com/find-my-device</a></p> <p data-bbox="506 1289 1016 1326"><b><u>Exemplary Support for Google Maps:</u></b></p>

## Exhibit C for U.S. Patent No. 9,445,251 Against HTC Accused Products

US9445251B2	HTC
	<p data-bbox="537 245 953 264">COMPUTER <b>ANDROID</b> IPHONE &amp; IPAD</p> <hr data-bbox="525 289 1558 292"/> <h3 data-bbox="525 350 1010 386">If they have a Google Account</h3> <ol data-bbox="525 407 1407 699" style="list-style-type: none"><li data-bbox="525 407 1205 427">1. If you haven't already, add their Gmail address to your <a href="#">Google Contacts</a>.</li><li data-bbox="525 444 1407 464">2. On your Android phone or tablet, open the Google Maps app  and sign in. <a href="#">Learn how to sign in.</a></li><li data-bbox="525 482 1020 501">3. Tap Menu ≡ &gt; <b>Location sharing</b> &gt; Add People .</li><li data-bbox="525 519 993 539">4. Choose how long you want to share your location.</li><li data-bbox="525 557 1127 613">5. Tap <b>Select People</b>.<ul data-bbox="556 591 1127 613" style="list-style-type: none"><li data-bbox="556 591 1127 613">• If you're asked about your contacts, give Google Maps access.</li></ul></li><li data-bbox="525 641 873 660">6. Choose who you want to share with.</li><li data-bbox="525 678 653 698">7. Tap <b>Share</b>.</li></ol> <h3 data-bbox="525 760 1098 795">If they don't have a Google Account</h3> <ol data-bbox="525 816 1549 943" style="list-style-type: none"><li data-bbox="525 816 1407 836">1. On your Android phone or tablet, open the Google Maps app  and sign in. <a href="#">Learn how to sign in.</a></li><li data-bbox="525 854 1020 873">2. Tap Menu ≡ &gt; <b>Location sharing</b> &gt; Add People .</li><li data-bbox="525 891 1549 943">3. Tap More ... &gt; <b>Copy to clipboard</b>. People with this link can see your location for as long as you choose, up to 72 hours.</li></ol> <h3 data-bbox="525 992 856 1027">Share using another app</h3> <p data-bbox="525 1045 1192 1065">You can also share through messaging apps. Tap More ... &gt; select an app.</p> <h3 data-bbox="525 1127 730 1162">Stop sharing</h3> <ol data-bbox="525 1183 1192 1281" style="list-style-type: none"><li data-bbox="525 1183 831 1203">1. Open the Google Maps app .</li><li data-bbox="525 1221 856 1240">2. Tap Menu ≡ &gt; <b>Location sharing</b>.</li><li data-bbox="525 1258 1192 1278">3. Next to the person with whom you want to stop sharing, tap Remove .</li></ol> <p data-bbox="499 1300 1688 1330"><a href="https://support.google.com/maps/answer/7326816?co=GENIE.Platform%3DAndroid&amp;oco=1">https://support.google.com/maps/answer/7326816?co=GENIE.Platform%3DAndroid&amp;oco=1</a></p>

## Exhibit C for U.S. Patent No. 9,445,251 Against HTC Accused Products

US9445251B2	HTC
	<h3 data-bbox="520 256 827 297">Share your E.T.A</h3> <p data-bbox="520 326 1650 347">After you start your drive, you can share your destination, estimated arrival time, and where you are on the route.</p> <ol data-bbox="531 380 1188 618" style="list-style-type: none"><li data-bbox="531 380 873 401">1. Open the Google Maps app .</li><li data-bbox="531 423 1146 444">2. Set a driving destination. <a href="#">Learn how to navigate to a place.</a></li><li data-bbox="531 467 1188 488">3. After you start navigation, tap More ^ &gt; <b>Share trip progress.</b></li><li data-bbox="531 511 863 532">4. Choose a person from the list.</li><li data-bbox="531 555 663 576">5. Tap <b>Share.</b></li><li data-bbox="531 599 1341 620">6. Location Sharing will stop when you reach your destination or stop navigating.</li></ol> <ul data-bbox="531 651 1188 672" style="list-style-type: none"><li data-bbox="531 651 1188 672">• To stop sharing before you arrive, tap More ^ &gt; <b>Stop sharing.</b></li></ul> <h3 data-bbox="520 743 942 784">See where someone is</h3> <p data-bbox="520 813 1251 834">If someone shares their location with you, you can see them on the map.</p> <ol data-bbox="531 867 905 976" style="list-style-type: none"><li data-bbox="531 867 873 888">1. Open the Google Maps app .</li><li data-bbox="531 911 905 932">2. Tap Menu ≡ &gt; <b>Location sharing.</b></li><li data-bbox="531 954 737 976">3. Choose someone.</li></ol> <ul data-bbox="531 1008 1289 1029" style="list-style-type: none"><li data-bbox="531 1008 1289 1029">• To see an updated location, tap on a friend's icon &gt; More ⋮ &gt; <b>Refresh.</b></li></ul> <h3 data-bbox="520 1089 1020 1130">Stop seeing someone's location</h3> <ol data-bbox="531 1149 1446 1300" style="list-style-type: none"><li data-bbox="531 1149 873 1170">1. Open the Google Maps app .</li><li data-bbox="531 1193 821 1214">2. On the map, tap their icon.</li><li data-bbox="531 1237 842 1258">3. At the bottom, tap More ^ .</li><li data-bbox="531 1281 1446 1302">4. To temporarily hide someone, tap <b>Hide from map.</b> You can stop hiding them at any time.</li></ol> <p data-bbox="520 1333 1728 1354"><b>Note:</b> You can stop someone's location from ever appearing on your map. <a href="#">Learn how to block another person's account.</a></p> <p data-bbox="499 1382 1686 1403"><a href="https://support.google.com/maps/answer/7326816?co=GENIE.Platform%3DAndroid&amp;oco=1">https://support.google.com/maps/answer/7326816?co=GENIE.Platform%3DAndroid&amp;oco=1</a></p>

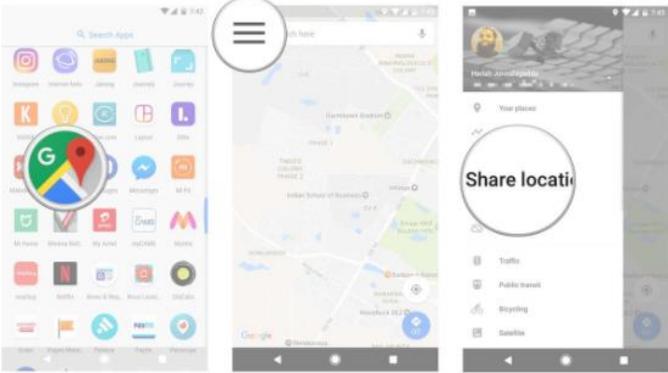
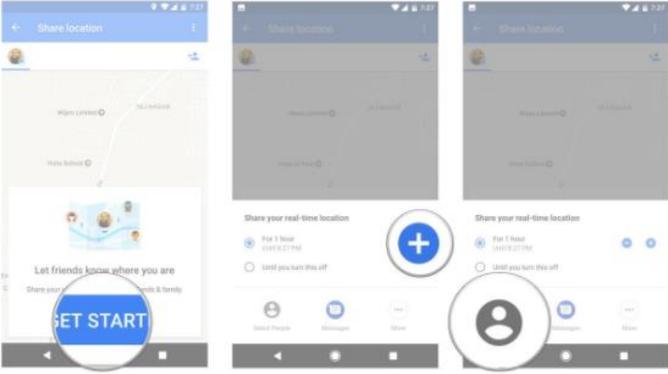
## Exhibit C for U.S. Patent No. 9,445,251 Against HTC Accused Products

US9445251B2	HTC
	<h3>Create a list of places</h3> <p>In Google Maps, you can create a list of places, like your favorite places or places you want to visit.</p> <p>COMPUTER <b>ANDROID</b> IPHONE &amp; IPAD</p> <hr/> <h3>Make a new list</h3> <ol style="list-style-type: none"><li>1. On your Android phone or tablet, open the Google Maps app .</li><li>2. Tap Menu  &gt; <b>Your places</b> &gt; <b>Saved</b>.</li><li>3. In the bottom right, tap Add .</li><li>4. Enter a name and description.</li><li>5. Tap <b>Save</b>.</li></ol> <h3>Save a place to a list</h3> <ol style="list-style-type: none"><li>1. Open the Google Maps app .</li><li>2. Search for a place or tap it on the map.</li><li>3. At the bottom, tap the place's name or address.</li><li>4. Tap <b>Save</b>.</li><li>5. Choose a list. To create a new list, tap <b>New list</b> .</li></ol> <h3>See your lists</h3> <ol style="list-style-type: none"><li>1. Open the Google Maps app .</li><li>2. Tap Menu  &gt; <b>Your places</b> &gt; <b>Saved</b>.</li></ol> <p><a href="https://support.google.com/maps/answer/7280933?hl=en&amp;ref_topic=7301134&amp;co=GENIE.Platform%3DAndroid&amp;oco=1">https://support.google.com/maps/answer/7280933?hl=en&amp;ref_topic=7301134&amp;co=GENIE.Platform%3DAndroid&amp;oco=1</a></p>

## Exhibit C for U.S. Patent No. 9,445,251 Against HTC Accused Products

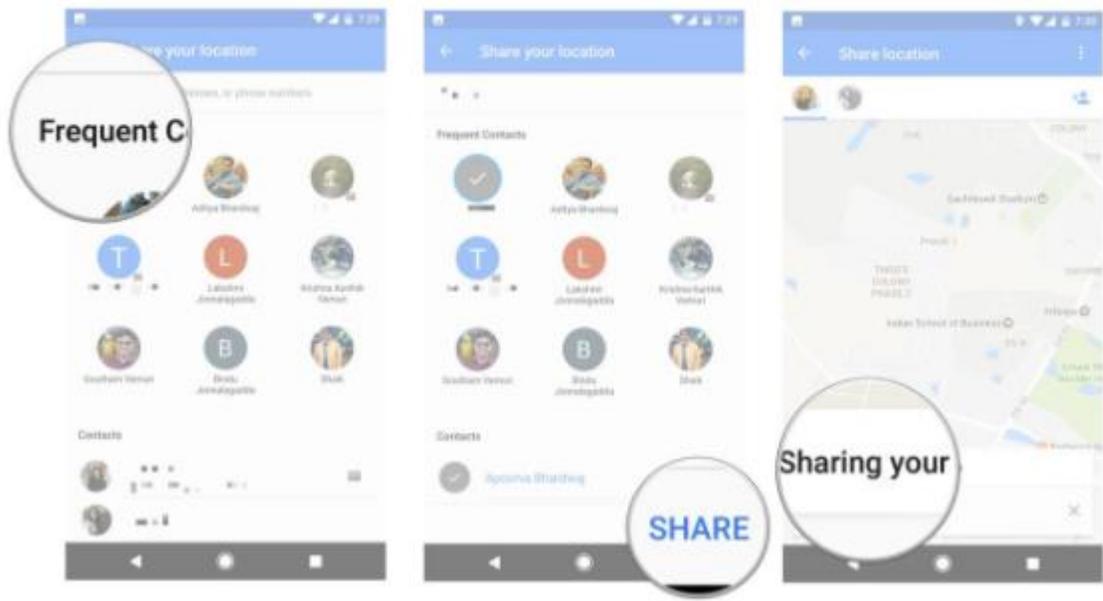
US9445251B2	HTC
	<h3 data-bbox="531 248 865 285">Hide or share lists</h3> <p data-bbox="531 315 894 337"><b>Note:</b> You can't share starred places.</p> <ol data-bbox="531 370 1241 477" style="list-style-type: none"><li>1. Open the Google Maps app .</li><li>2. Tap Menu  &gt; <b>Your places</b> &gt; <b>Saved</b>.</li><li>3. Next to the list you want to share, tap More  &gt; choose an option:</li></ol> <ul data-bbox="569 493 1671 636" style="list-style-type: none"><li>• <b>Hide/Show on your map:</b> Display or hide your saved places when looking at the map.</li><li>• <b>Share list:</b> Allow others to see your saved list.</li><li>• <b>Sharing options:</b> You can make your list public, private, or shared. To let anyone with the link see your list, tap <b>Shared</b>. To let anyone find and follow your list, tap <b>Public</b>.</li></ul> <h3 data-bbox="531 703 751 740">Follow a list</h3> <p data-bbox="531 769 1713 824">If you follow a list made by someone else, their saved places will show up in Your Places. The places will also appear as suggested locations in Google Maps.</p> <h3 data-bbox="531 881 898 919">Follow a list using a link</h3> <ol data-bbox="531 941 1339 1049" style="list-style-type: none"><li>1. Tap on the link you received to open it.</li><li>2. Tap <b>Follow</b>. This list will now be added to the group of lists you follow.</li><li>3. <b>Optional:</b> To unfollow a list someone shared with you, tap the list &gt; <b>Following</b>.</li></ol> <h3 data-bbox="531 1101 909 1138">See lists made by others</h3> <p data-bbox="531 1161 1318 1183">If a user has any Google Maps lists that were made public, you can follow them.</p> <ol data-bbox="531 1216 1121 1323" style="list-style-type: none"><li>1. Tap on the name of a user whose list you want to follow.</li><li>2. Tap <b>Lists</b>.</li><li>3. Tap on the list you want to follow &gt; More  &gt; <b>Follow</b>.</li></ol> <p data-bbox="499 1339 1902 1403"><a href="https://support.google.com/maps/answer/7280933?hl=en&amp;ref_topic=7301134&amp;co=GENIE.Platform%3DAndroid&amp;oco=1">https://support.google.com/maps/answer/7280933?hl=en&amp;ref_topic=7301134&amp;co=GENIE.Platform%3DAndroid&amp;oco=1</a></p>

**Exhibit C for U.S. Patent No. 9,445,251 Against HTC Accused Products**

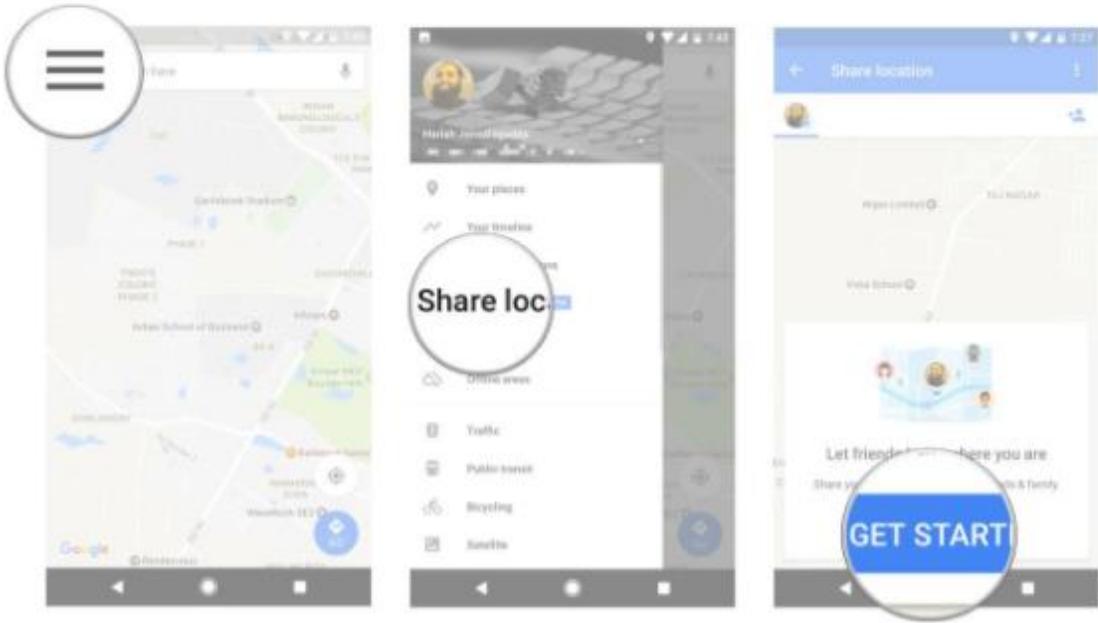
US9445251B2	HTC
	<p data-bbox="506 277 1140 310"><b>How to share your location in Google Maps</b></p> <ol data-bbox="506 337 1121 427" style="list-style-type: none"> <li>1. Open <b>Google Maps</b> from the app drawer or the home screen.</li> <li>2. Tap the <b>hamburger menu</b> (the three horizontal lines) on the top left corner of the screen.</li> <li>3. Select <b>Share location</b>.</li> </ol>  <ol data-bbox="506 865 1152 971" style="list-style-type: none"> <li>4. Tap <b>Get Started</b>.</li> <li>5. Use the <b>+</b> icon to select a time period or select the <b>Until you turn this off</b> setting to share your location indefinitely.</li> <li>6. Tap <b>Select People</b>.</li> </ol>  <p data-bbox="499 1377 1346 1409"><a href="https://www.androidcentral.com/how-share-location-google-maps">https://www.androidcentral.com/how-share-location-google-maps</a></p>



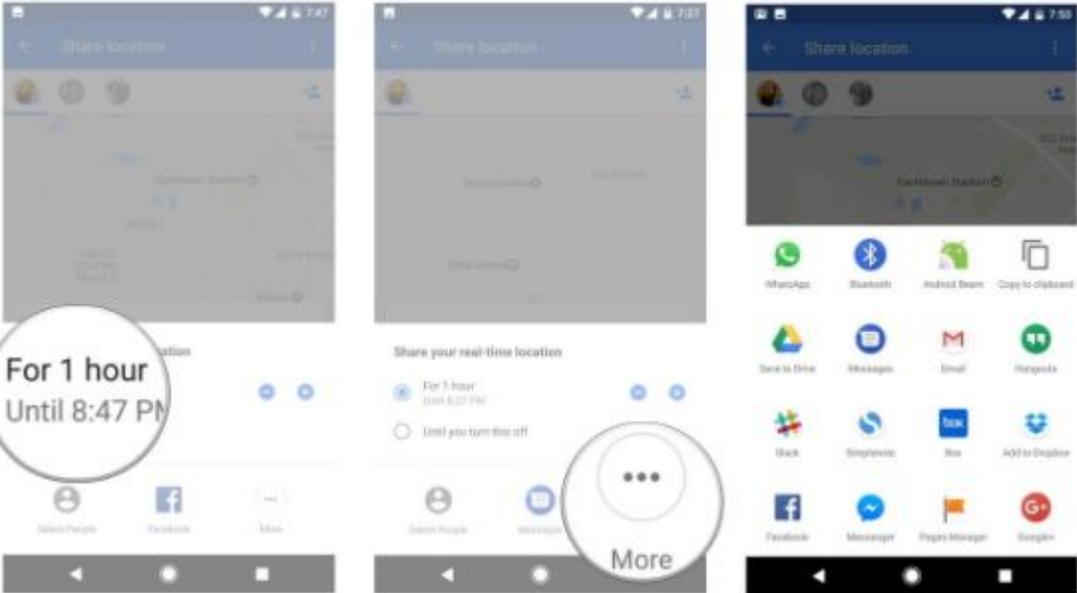
### Exhibit C for U.S. Patent No. 9,445,251 Against HTC Accused Products

US9445251B2	HTC
	<p data-bbox="514 289 1564 349">7. You'll see a list of your frequent contacts at the top, along with a full list of contacts. Pick the contacts by tapping their name.</p> <p data-bbox="514 373 1438 406">8. Once you've selected the contacts you want to share your location to, tap <b>Share</b>.</p> <p data-bbox="514 430 1407 462">9. You'll see a message saying that the selected contact can view your location.</p>  <p data-bbox="493 1136 1344 1177"><a href="https://www.androidcentral.com/how-share-location-google-maps">https://www.androidcentral.com/how-share-location-google-maps</a></p>

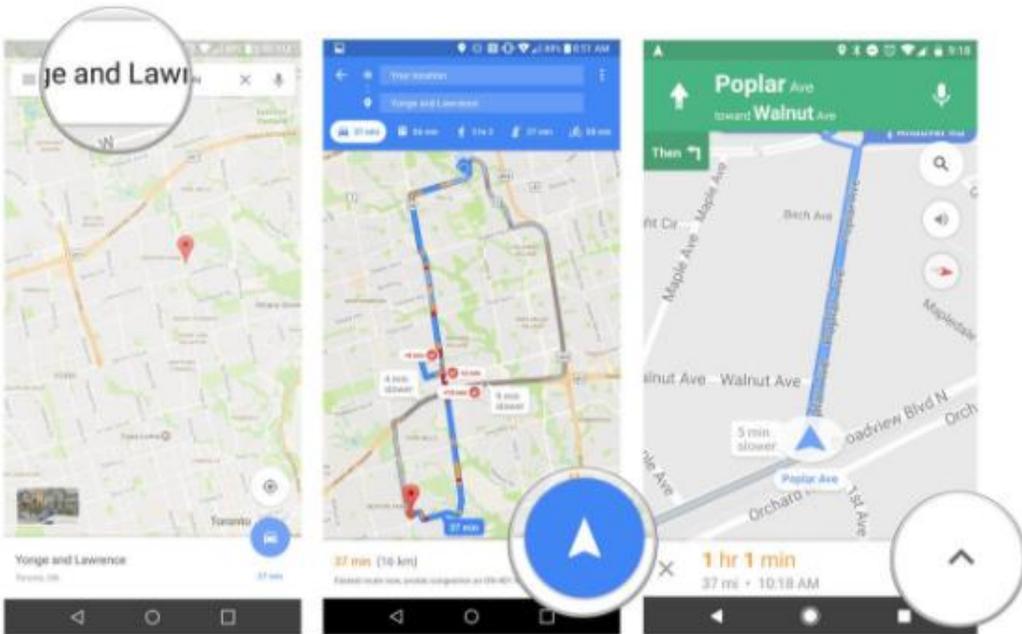
## Exhibit C for U.S. Patent No. 9,445,251 Against HTC Accused Products

US9445251B2	HTC
	<h3 data-bbox="512 245 1243 289">How to create a shareable link</h3> <p data-bbox="512 334 1451 362">You can also create a link and use it to share your location easily. Here's how to do it:</p> <ol data-bbox="506 412 1224 553" style="list-style-type: none"><li data-bbox="506 412 1224 440">1. Tap the <b>hamburger menu</b> on the top left corner of the screen.</li><li data-bbox="506 467 789 495">2. Select <b>Share location</b>.</li><li data-bbox="506 522 726 550">3. Tap <b>Get Started</b>.</li></ol>  <p data-bbox="499 1232 1346 1263"><a href="https://www.androidcentral.com/how-share-location-google-maps">https://www.androidcentral.com/how-share-location-google-maps</a></p>

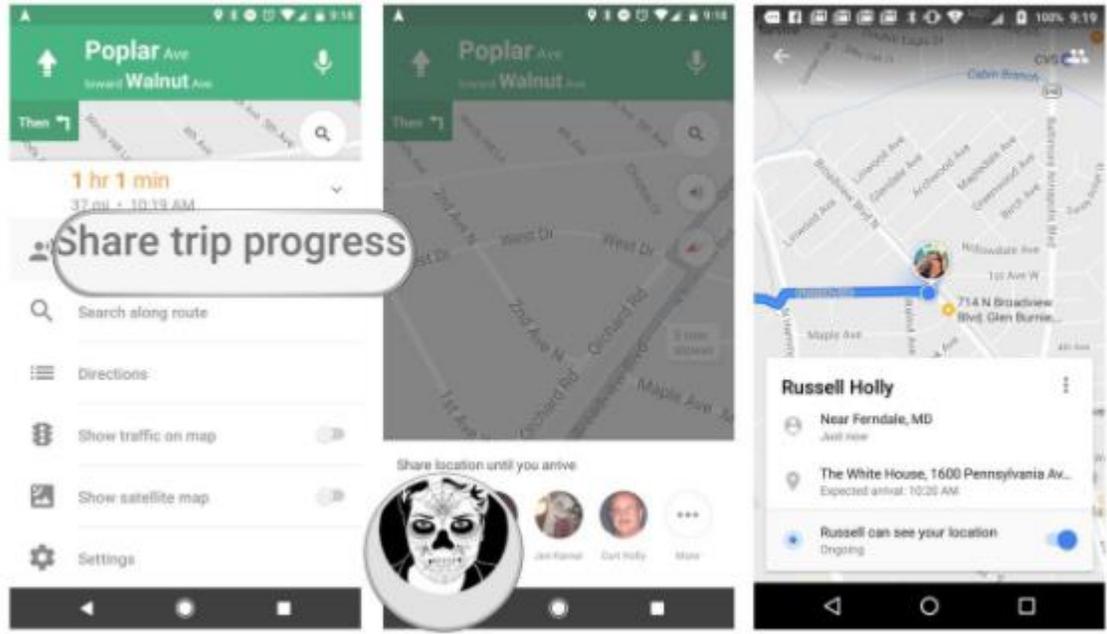
### Exhibit C for U.S. Patent No. 9,445,251 Against HTC Accused Products

US9445251B2	HTC
	<p>4. Select the amount of time you want to share your location.</p> <p>5. Tap More.</p> <p>6. Select your app of choice to create and send a unique URL that broadcasts your current location. You can email it, send the link via Messenger, or even tweet it to the intended recipient.</p>  <p><a href="https://www.androidcentral.com/how-share-location-google-maps">https://www.androidcentral.com/how-share-location-google-maps</a></p>

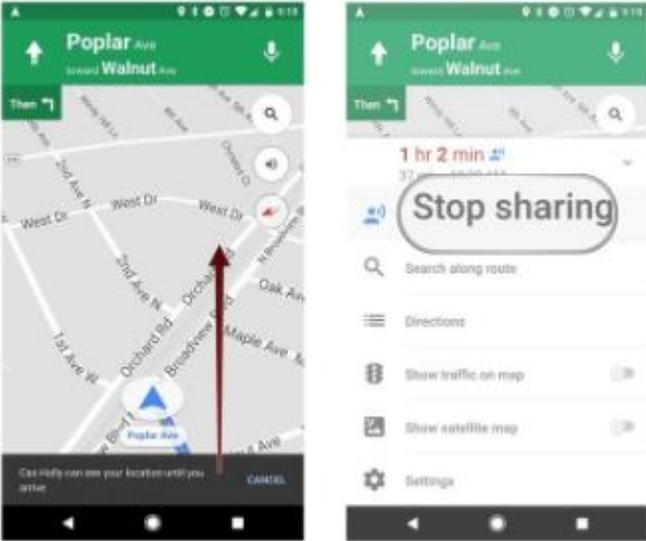
**Exhibit C for U.S. Patent No. 9,445,251 Against HTC Accused Products**

US9445251B2	HTC
	<p data-bbox="514 240 1415 337"><b>How to share your navigation directions while you walk, drive or transit</b></p> <p data-bbox="514 375 1541 464">One of the best ways to share your location with someone is during a drive, walk or transit. If you're meeting a friend or family member somewhere, or navigating towards their home, Google Maps lets you share your location with them for the duration of the trip. It's magic!</p> <ol data-bbox="514 513 1381 646" style="list-style-type: none"><li>1. In the <b>search bar</b> enter your destination.</li><li>2. Pick your navigation type (drive, transit, walk) and press the blue <b>navigate button</b>.</li><li>3. Tap the arrow next to the <b>time-to-destination</b> number at the bottom of the screen.</li></ol>  <p data-bbox="499 1328 1346 1360"><a href="https://www.androidcentral.com/how-share-location-google-maps">https://www.androidcentral.com/how-share-location-google-maps</a></p>

### Exhibit C for U.S. Patent No. 9,445,251 Against HTC Accused Products

US9445251B2	HTC
	<p data-bbox="514 277 823 305">4. Tap Share trip progress.</p> <p data-bbox="514 334 1136 362">5. Choose one or more contacts to share trip progress.</p>  <p data-bbox="520 1063 1333 1091">You can also stop sharing your location with someone before a trip ends.</p> <p data-bbox="520 1101 1346 1128"><a href="https://www.androidcentral.com/how-share-location-google-maps">https://www.androidcentral.com/how-share-location-google-maps</a></p>

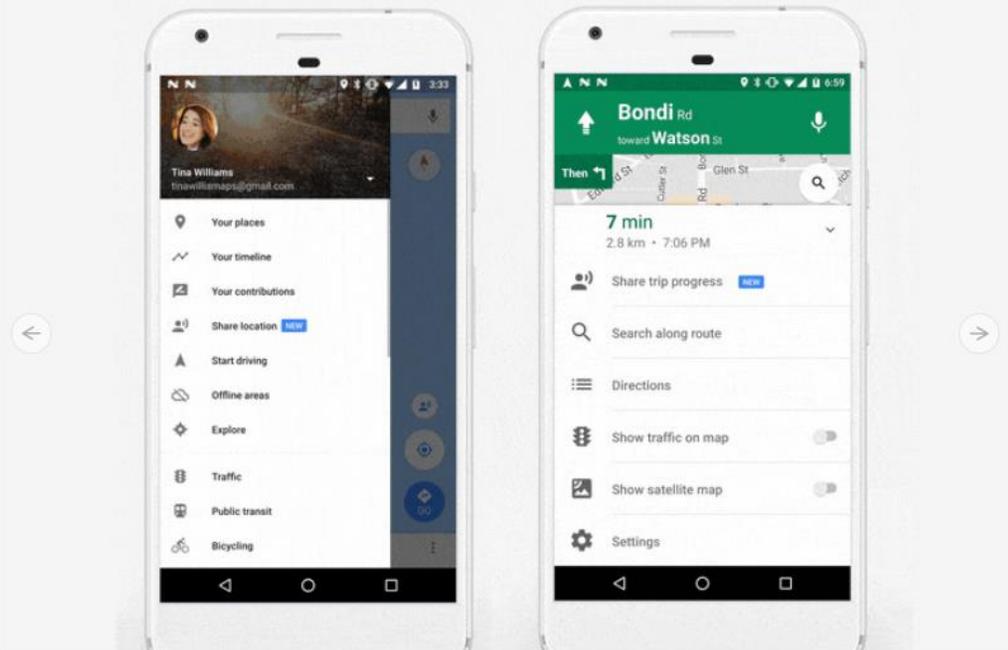
## Exhibit C for U.S. Patent No. 9,445,251 Against HTC Accused Products

US9445251B2	HTC
	<ol style="list-style-type: none"><li data-bbox="520 245 1457 272">1. Tap the arrow next to the <b>time-to-destination</b> number at the bottom of the screen.</li><li data-bbox="520 302 758 329">2. Tap <b>Stop sharing</b>.</li></ol> <div data-bbox="737 383 1383 924"></div> <p data-bbox="531 979 625 1003">That's it!</p> <p data-bbox="531 1047 1598 1071">Are you excited that location sharing is back in Google Maps? How often do you use the feature?</p> <p data-bbox="499 1089 1346 1117"><a href="https://www.androidcentral.com/how-share-location-google-maps">https://www.androidcentral.com/how-share-location-google-maps</a></p> <p data-bbox="499 1161 1409 1188">As shown below, a group may also be defined within Google Contacts.</p>

## Exhibit C for U.S. Patent No. 9,445,251 Against HTC Accused Products

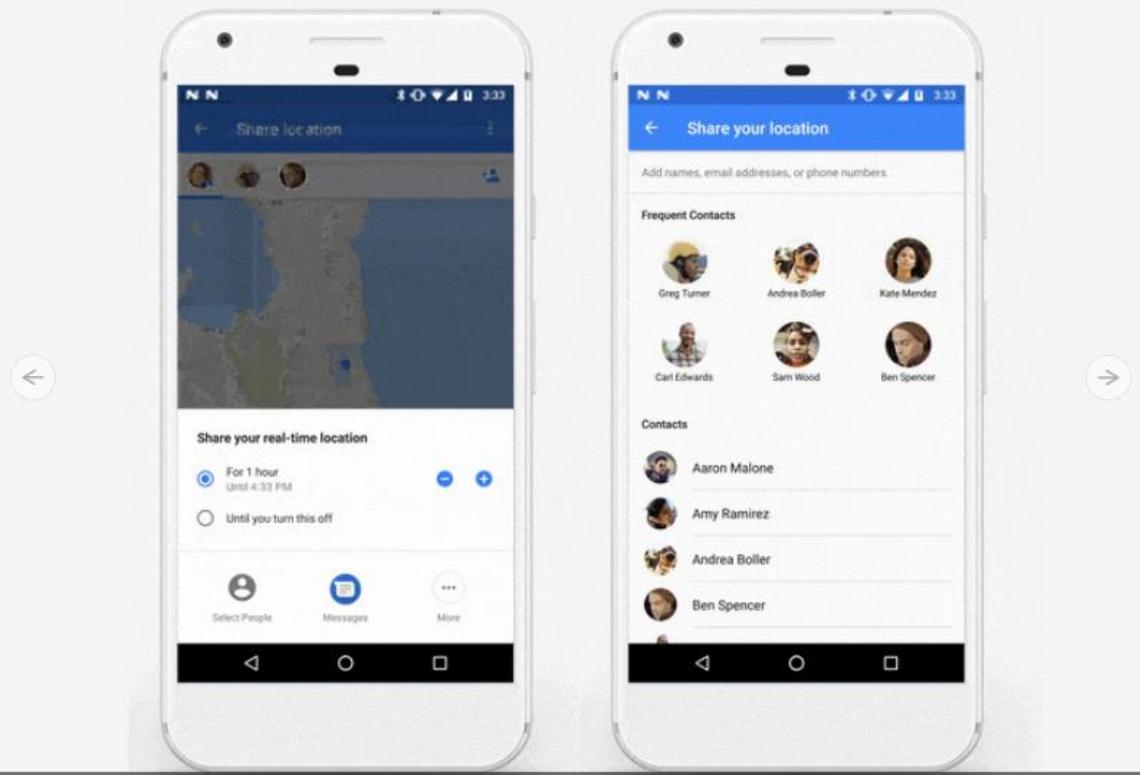
US9445251B2	HTC
	<h3 data-bbox="527 237 884 277">See your contacts</h3> <ol data-bbox="527 305 957 378" style="list-style-type: none"><li data-bbox="527 305 957 337">1. Open your device's Contacts app .</li><li data-bbox="527 350 716 378">2. Tap Menu .</li></ol> <ul data-bbox="527 407 1724 605" style="list-style-type: none"><li data-bbox="527 407 1094 435">• <b>See contacts by label:</b> Choose a label from the list.</li><li data-bbox="527 451 1346 479">• <b>See contacts for another account:</b> Tap Down arrow  &gt; pick an account.</li><li data-bbox="527 495 1199 522">• <b>See the contacts for all your accounts:</b> Choose <b>All contacts</b>.</li></ul> <p data-bbox="558 532 1724 560"><b>Tip:</b> If you have multiple contacts with the same information, the information will be grouped into one contact.</p> <ul data-bbox="527 576 1346 605" style="list-style-type: none"><li data-bbox="527 576 1346 605">• <b>See your Google Account contacts on the web:</b> Go to <a href="#">Google Contacts</a> .</li></ul> <p data-bbox="499 634 1524 667"><a href="https://support.google.com/android/answer/6118731?hl=en&amp;ref_topic=6118711">https://support.google.com/android/answer/6118731?hl=en&amp;ref_topic=6118711</a></p> <h3 data-bbox="527 711 884 751">Label your contacts</h3> <p data-bbox="527 776 972 803">You can group contacts together using labels.</p> <ol data-bbox="527 829 919 938" style="list-style-type: none"><li data-bbox="527 829 919 862">1. Open your device's Contacts app .</li><li data-bbox="527 870 852 898">2. Tap Menu  &gt; <b>Create label</b>.</li><li data-bbox="527 911 856 938">3. Enter a label name and tap <b>Ok</b>.</li></ol> <ul data-bbox="527 964 1703 1029" style="list-style-type: none"><li data-bbox="527 964 1220 992">• <b>Add one contact to a label:</b> Tap Add contact  &gt; choose a contact.</li><li data-bbox="527 1005 1703 1029">• <b>Add multiple contacts to a label:</b> Tap Add contact  &gt; touch and hold a contact &gt; tap the other contacts &gt; tap <b>Add</b>.</li></ul> <p data-bbox="499 1040 1524 1073"><a href="https://support.google.com/android/answer/6118731?hl=en&amp;ref_topic=6118711">https://support.google.com/android/answer/6118731?hl=en&amp;ref_topic=6118711</a></p> <h3 data-bbox="527 1127 930 1167">Share your contacts</h3> <ol data-bbox="527 1193 1031 1360" style="list-style-type: none"><li data-bbox="527 1193 957 1226">1. Open your device's Contacts app .</li><li data-bbox="527 1239 831 1266">2. Tap a contact in the list.</li><li data-bbox="527 1279 821 1307">3. Tap More  &gt; <b>Share</b>.</li><li data-bbox="527 1320 1031 1360">4. Choose how you want to share the contact.</li></ol> <p data-bbox="499 1372 1524 1404"><a href="https://support.google.com/android/answer/6118731?hl=en&amp;ref_topic=6118711">https://support.google.com/android/answer/6118731?hl=en&amp;ref_topic=6118711</a></p>

### Exhibit C for U.S. Patent No. 9,445,251 Against HTC Accused Products

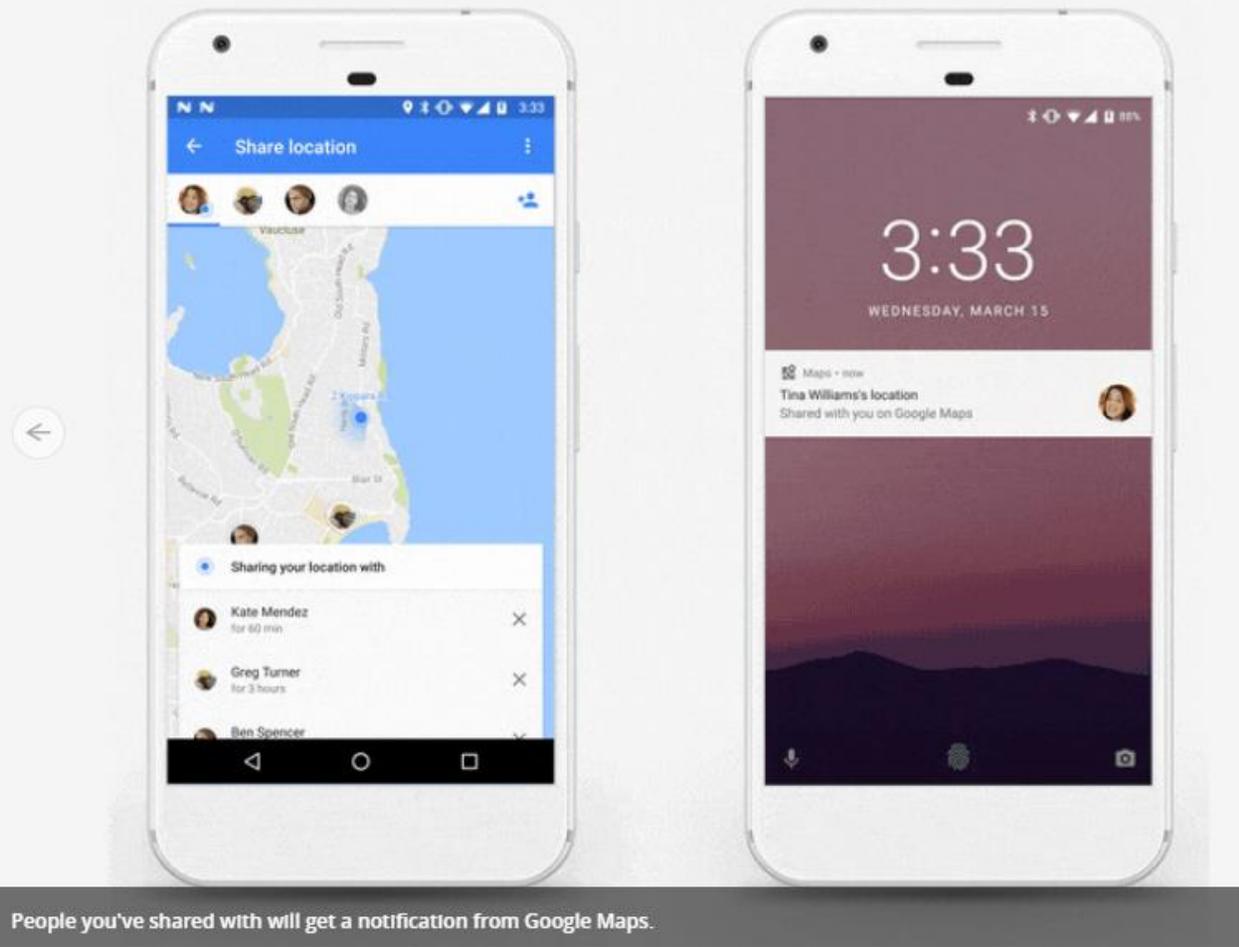
US9445251B2	HTC
	 <p data-bbox="506 938 1507 987">Ways to start location sharing: (left) from the app navigation drawer, and (right) from the menu, you can drag up while navigating somewhere.</p> <p data-bbox="499 997 1646 1029"><a href="https://arstechnica.com/gadgets/2017/03/location-sharing-finally-returns-to-google-maps/">https://arstechnica.com/gadgets/2017/03/location-sharing-finally-returns-to-google-maps/</a></p>



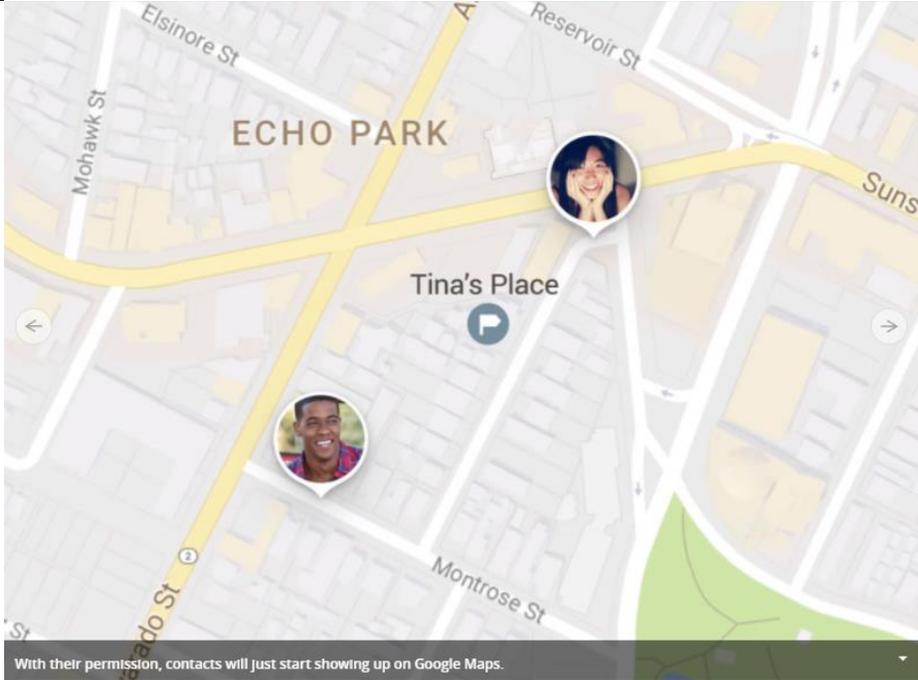
**Exhibit C for U.S. Patent No. 9,445,251 Against HTC Accused Products**

US9445251B2	HTC
	 <p data-bbox="506 1027 1646 1060">You can pick how long you want to share your location for, and with whom you want to share it.</p> <p data-bbox="506 1068 1646 1099"><a href="https://arstechnica.com/gadgets/2017/03/location-sharing-finally-returns-to-google-maps/">https://arstechnica.com/gadgets/2017/03/location-sharing-finally-returns-to-google-maps/</a></p>

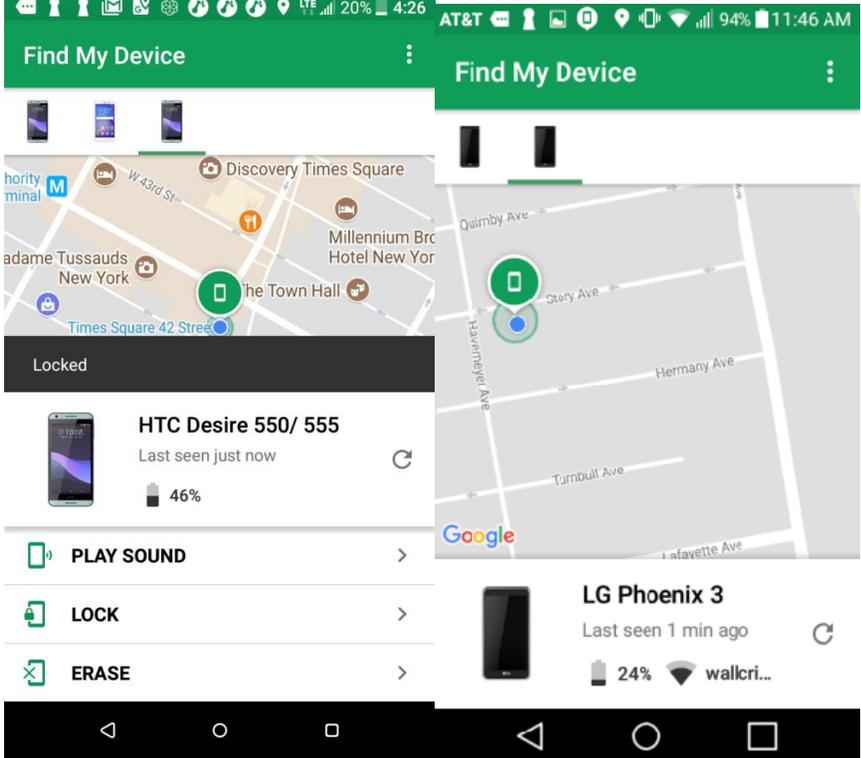
**Exhibit C for U.S. Patent No. 9,445,251 Against HTC Accused Products**

US9445251B2	HTC
	 <p data-bbox="506 1146 1163 1174">People you've shared with will get a notification from Google Maps.</p> <p data-bbox="499 1190 1646 1224"><a href="https://arstechnica.com/gadgets/2017/03/location-sharing-finally-returns-to-google-maps/">https://arstechnica.com/gadgets/2017/03/location-sharing-finally-returns-to-google-maps/</a></p>

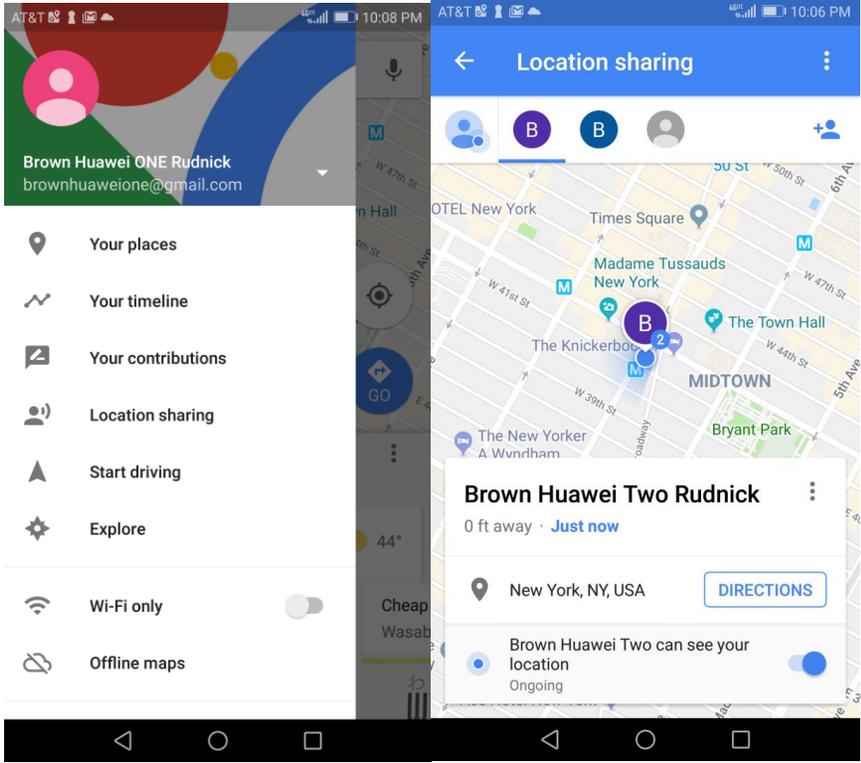
**Exhibit C for U.S. Patent No. 9,445,251 Against HTC Accused Products**

US9445251B2	HTC
	 <p data-bbox="499 885 1417 912">With their permission, contacts will just start showing up on Google Maps.</p> <p data-bbox="499 917 1648 950"><a href="https://arstechnica.com/gadgets/2017/03/location-sharing-finally-returns-to-google-maps/">https://arstechnica.com/gadgets/2017/03/location-sharing-finally-returns-to-google-maps/</a></p> <p data-bbox="499 982 1060 1023"><b><u>Exemplary Find My Device Screenshots:</u></b></p>

**Exhibit C for U.S. Patent No. 9,445,251 Against HTC Accused Products**

US9445251B2	HTC
	 <p><b>Find My Device</b></p> <p>Discovery Times Square Millennium Br Hotel New Yor The Town Hall Times Square 42 Street</p> <p>Locked</p> <p><b>HTC Desire 550/ 555</b> Last seen just now 46%</p> <p><b>PLAY SOUND</b></p> <p><b>LOCK</b></p> <p><b>ERASE</b></p> <p>Quimby Ave Story Ave Hermann Ave Turnbull Ave Lafayette Ave</p> <p><b>LG Phoenix 3</b> Last seen 1 min ago 24% wallcri...</p> <p><u><b>Exemplary Google Maps Screenshots:</b></u></p>

**Exhibit C for U.S. Patent No. 9,445,251 Against HTC Accused Products**

US9445251B2	HTC
	 <p><b>Exemplary Source Code:</b></p> <p>The above functionality is performed at least in part by the following publicly available source code and/or source code that invokes or is invoked by the following source code (or a substantially similar copy compiled and loaded onto the Accused Products by HTC). AGIS reserves the right to amend these contentions to include additional source code as discovery progresses and as additional source code is made available.</p>

**Exhibit C for U.S. Patent No. 9,445,251 Against HTC Accused Products**

US9445251B2	HTC
	<h2 data-bbox="516 240 1003 293">Contacts Provider</h2> <p data-bbox="516 329 1461 591">The Contacts Provider is a powerful and flexible Android component that manages the device's central repository of data about people. The Contacts Provider is the source of data you see in the device's contacts application, and you can also access its data in your own application and transfer data between the device and online services. The provider accommodates a wide range of data sources and tries to manage as much data as possible for each person, with the result that its organization is complex. Because of this, the provider's API includes an extensive set of contract classes and interfaces that facilitate both data retrieval and modification.</p> <p data-bbox="516 626 823 647">This guide describes the following:</p> <ul data-bbox="516 678 1360 850" style="list-style-type: none"><li data-bbox="516 678 793 699">• The basic provider structure.</li><li data-bbox="516 729 884 750">• How to retrieve data from the provider.</li><li data-bbox="516 779 852 800">• How to modify data in the provider.</li><li data-bbox="516 829 1360 850">• How to write a sync adapter for synchronizing data from your server to the Contacts Provider.</li></ul> <p data-bbox="499 865 1474 894"><a href="https://developer.android.com/guide/topics/providers/contacts-provider.html">https://developer.android.com/guide/topics/providers/contacts-provider.html</a></p>

## Exhibit C for U.S. Patent No. 9,445,251 Against HTC Accused Products

US9445251B2	HTC
	<p><b>Overview</b></p> <p>ContactsContract defines an extensible database of contact-related information. Contact information is stored in a three-tier data model:</p> <ul style="list-style-type: none"> <li>• A row in the <code>ContactsContract.Data</code> table can store any kind of personal data, such as a phone number or email addresses. The set of data kinds that can be stored in this table is open-ended. There is a predefined set of common kinds, but any application can add its own data kinds.</li> <li>• A row in the <code>ContactsContract.RawContacts</code> table represents a set of data describing a person and associated with a single account (for example, one of the user's Gmail accounts).</li> <li>• A row in the <code>ContactsContract.Contacts</code> table represents an aggregate of one or more RawContacts presumably describing the same person. When data in or associated with the RawContacts table is changed, the affected aggregate contacts are updated as necessary.</li> </ul> <p>Other tables include:</p> <ul style="list-style-type: none"> <li>• <code>ContactsContract.Groups</code>, which contains information about raw contact groups such as Gmail contact groups. The current API does not support the notion of groups spanning multiple accounts.</li> <li>• <code>ContactsContract.StatusUpdates</code>, which contains social status updates including IM availability.</li> <li>• <code>ContactsContract.AggregationExceptions</code>, which is used for manual aggregation and disaggregation of raw contacts</li> <li>• <code>ContactsContract.Settings</code>, which contains visibility and sync settings for accounts and groups.</li> <li>• <code>ContactsContract.SyncState</code>, which contains free-form data maintained on behalf of sync adapters</li> <li>• <code>ContactsContract.PhoneLookup</code>, which is used for quick caller-ID lookup</li> </ul> <p><a href="https://developer.android.com/reference/android/provider/ContactsContract.html">https://developer.android.com/reference/android/provider/ContactsContract.html</a></p> <p><b>Data</b></p> <p>As noted previously, the data for a raw contact is stored in a <code>ContactsContract.Data</code> row that is linked to the raw contact's <code>_ID</code> value. This allows a single raw contact to have multiple instances of the same type of data such as email addresses or phone numbers. For example, if "Thomas Higginson" for <code>emilyd@gmail.com</code> (the raw contact row for Thomas Higginson associated with the Google account <code>emilyd@gmail.com</code>) has a home email address of <code>thigg@gmail.com</code> and a work email address of <code>thomas.higginson@gmail.com</code>, the Contacts Provider stores the two email address rows and links them both to the raw contact.</p> <p>Notice that different types of data are stored in this single table. Display name, phone number, email, postal address, photo, and website detail rows are all found in the <code>ContactsContract.Data</code> table. To help manage this, the <code>ContactsContract.Data</code> table has some columns with descriptive names, and others with generic names. The contents of a descriptive-name column have the same meaning regardless of the type of data in the row, while the contents of a generic-name column have different meanings depending on the type of data.</p> <p><a href="https://developer.android.com/guide/topics/providers/contacts-provider.html">https://developer.android.com/guide/topics/providers/contacts-provider.html</a></p>

**Exhibit C for U.S. Patent No. 9,445,251 Against HTC Accused Products**

US9445251B2	HTC				
	Task	Action	Data	MIME type	Notes
	Pick a contact from a list	ACTION_PICK	One of: <ul style="list-style-type: none"> <li>• <code>Contacts.CONTENT_URI</code>, which displays a list of contacts.</li> <li>• <code>Phone.CONTENT_URI</code>, which displays a list of phone numbers for a raw contact.</li> <li>• <code>StructuredPostal.CONTENT_URI</code>, which displays a list of postal addresses for a raw contact.</li> <li>• <code>Email.CONTENT_URI</code>, which displays a list of email addresses for a raw contact.</li> </ul>	Not used	Displays a list of raw contacts or a list of data from a raw contact, depending on the content URI type you supply. Call <code>startActivityForResult()</code> , which returns the content URI of the selected row. The form of the URI is the table's content URI with the row's <code>LOOKUP_ID</code> appended to it. The device's contacts app delegates read and write permissions to this content URI for the life of your activity. See the <a href="#">Content Provider Basics</a> guide for more details.
<a href="https://developer.android.com/guide/topics/providers/contacts-provider.html">https://developer.android.com/guide/topics/providers/contacts-provider.html</a>					



**Exhibit C for U.S. Patent No. 9,445,251 Against HTC Accused Products**

US9445251B2	HTC
	<pre>25  * Parsed form of the intent sent to the Contacts application. 26  */ 27  public class ContactsRequest { 28 29      /** Default mode: browse contacts */ 30      public static final int ACTION_DEFAULT = 10; 31 32      /** Show all contacts */ 33      public static final int ACTION_ALL_CONTACTS = 15; 34 35      /** Show all contacts with phone numbers */ 36      public static final int ACTION_CONTACTS_WITH_PHONES = 17; 37 38      /** Show contents of a specific group */ 39      public static final int ACTION_GROUP = 20; 40 41      /** Show all starred contacts */ 42      public static final int ACTION_STARRED = 30; 43 44      /** Show frequently contacted contacts */ 45      public static final int ACTION_FREQUENT = 40; 46 47      /** Show starred and the frequent */ 48      public static final int ACTION_STREQUENT = 50; 49 50      /** Show all contacts and pick them when clicking */ 51      public static final int ACTION_PICK_CONTACT = 60; 52 53      /** Show all contacts as well as the option to create a new one */ 54      public static final int ACTION_PICK_OR_CREATE_CONTACT = 70; 55 56      /** Show all contacts and pick them for edit when clicking, and allow creating a new contact */ 57      public static final int ACTION_INSERT_OR_EDIT_CONTACT = 80;</pre> <p><a href="https://android.goesource.com/platform/packages/apps/Contacts/+/nougat-mr1-release/src/com/android/contacts/list/ContactsRequest.java">https://android.goesource.com/platform/packages/apps/Contacts/+/nougat-mr1-release/src/com/android/contacts/list/ContactsRequest.java</a></p>

**Exhibit C for U.S. Patent No. 9,445,251 Against HTC Accused Products**

US9445251B2	HTC
	<pre>59      /** Show all phone numbers and pick them when clicking */ 60      public static final int ACTION_PICK_PHONE = 90; 61 62      /** Show all postal addresses and pick them when clicking */ 63      public static final int ACTION_PICK_POSTAL = 100; 64 65      /** Show all postal addresses and pick them when clicking */ 66      public static final int ACTION_PICK_EMAIL = 105; 67 68      /** Show all contacts and create a shortcut for the picked contact */ 69      public static final int ACTION_CREATE_SHORTCUT_CONTACT = 110; 70 71      /** Show all phone numbers and create a call shortcut for the picked number */ 72      public static final int ACTION_CREATE_SHORTCUT_CALL = 120; 73 74      /** Show all phone numbers and create an SMS shortcut for the picked number */ 75      public static final int ACTION_CREATE_SHORTCUT_SMS = 130; 76 77      /** Show all contacts and activate the specified one */ 78      public static final int ACTION_VIEW_CONTACT = 140; 79 80      /** Show contacts recommended for joining with a specified target contact */ 81      public static final int ACTION_PICK_JOIN = 150;</pre> <p><a href="https://android.googlesource.com/platform/packages/apps/Contacts/+/nougat-mr1-release/src/com/android/contacts/list/ContactsRequest.java">https://android.googlesource.com/platform/packages/apps/Contacts/+/nougat-mr1-release/src/com/android/contacts/list/ContactsRequest.java</a></p>

**Exhibit C for U.S. Patent No. 9,445,251 Against HTC Accused Products**

US9445251B2	HTC
	<pre> 104  * Displays a list to browse contacts. 105  */ 106  public class PeopleActivity extends ContactsActivity implements 107      View.OnCreateContextMenuListener, 108      View.OnClickListener, 109      ActionBarAdapter.Listener, 110      DialogManager.DialogShowingViewActivity, 111      ContactListFilterController.ContactListFilterListener, 112      ProviderStatusListener, 113      MultiContactDeleteListener, 114      JoinContactsListener { </pre> <p data-bbox="499 662 1556 727"><a href="https://android.googlesource.com/platform/packages/apps/Contacts/+/nougat-mr1-release/src/com/android/contacts/activities/PeopleActivity.java">https://android.googlesource.com/platform/packages/apps/Contacts/+/nougat-mr1-release/src/com/android/contacts/activities/PeopleActivity.java</a></p> <pre> 145      * Showing a list of Contacts. Also used for showing search results in search mode. 146      */ 147      private MultiSelectContactsListFragment mAllFragment; 148      private ContactTileListFragment mFavoritesFragment; </pre> <p data-bbox="499 873 1556 938"><a href="https://android.googlesource.com/platform/packages/apps/Contacts/+/nougat-mr1-release/src/com/android/contacts/activities/PeopleActivity.java">https://android.googlesource.com/platform/packages/apps/Contacts/+/nougat-mr1-release/src/com/android/contacts/activities/PeopleActivity.java</a></p>

**Exhibit C for U.S. Patent No. 9,445,251 Against HTC Accused Products**

US9445251B2	HTC
	<pre>458     private void configureFragments(boolean fromRequest) { 459         if (fromRequest) { 460             ContactListFilter filter = null; 461             int actionCode = mRequest.getActionCode(); 462             boolean searchMode = mRequest.isSearchMode(); 463             final int tabToOpen; 464             switch (actionCode) { 465                 case ContactsRequest.ACTION_ALL_CONTACTS: 466                     filter = ContactListFilter.createFilterWithType( 467                         ContactListFilter.FILTER_TYPE_ALL_ACCOUNTS); 468                     tabToOpen = TabState.ALL; 469                     break; 470                 case ContactsRequest.ACTION_CONTACTS_WITH_PHONES: 471                     filter = ContactListFilter.createFilterWithType( 472                         ContactListFilter.FILTER_TYPE_WITH_PHONE_NUMBERS_ONLY); 473                     tabToOpen = TabState.ALL; 474                     break; 475 476                 case ContactsRequest.ACTION_FREQUENT: 477                 case ContactsRequest.ACTION_STREQUENT: 478                 case ContactsRequest.ACTION_STARRED: 479                     tabToOpen = TabState.FAVORITES; 480                     break; 481                 case ContactsRequest.ACTION_VIEW_CONTACT: 482                     tabToOpen = TabState.ALL; 483                     break; 484                 default: 485                     tabToOpen = -1; 486                     break; 487             } </pre> <p data-bbox="499 1321 1556 1393"><a href="https://android.goesource.com/platform/packages/apps/Contacts/+nougat-mr1-release/src/com/android/contacts/activities/PeopleActivity.java">https://android.goesource.com/platform/packages/apps/Contacts/+nougat-mr1-release/src/com/android/contacts/activities/PeopleActivity.java</a></p>

**Exhibit C for U.S. Patent No. 9,445,251 Against HTC Accused Products**

US9445251B2	HTC
	<pre>488         if (tabToOpen != -1) { 489             mActionBarAdapter.setCurrentTab(tabToOpen); 490         } 491 492         if (filter != null) { 493             mContactListFilterController.setContactListFilter(filter, false); 494             searchMode = false; 495         } 496 497         if (mRequest.getContactUri() != null) { 498             searchMode = false; 499         } 500 501         mActionBarAdapter.setSearchMode(searchMode); 502         configureContactListFragmentForRequest(); 503     } 504 505     configureContactListFragment(); 506 507     invalidateOptionsMenuIfNeeded(); 508 }</pre> <p data-bbox="499 1019 1556 1086"><a href="https://android.googlesource.com/platform/packages/apps/Contacts/+/nougat-mr1-release/src/com/android/contacts/activities/PeopleActivity.java">https://android.googlesource.com/platform/packages/apps/Contacts/+/nougat-mr1-release/src/com/android/contacts/activities/PeopleActivity.java</a></p>

**Exhibit C for U.S. Patent No. 9,445,251 Against HTC Accused Products**

US9445251B2	HTC
	<pre>35 public class ProfileAndContactsLoader extends CursorLoader { 36 37     private boolean mLoadProfile; 38 39     private String[] mProjection; 40 41     private Uri mExtraUri; 42     private String[] mExtraProjection; 43     private String mExtraSelection; 44     private String[] mExtraSelectionArgs; 45     private boolean mMergeExtraContactsAfterPrimary; 46 47     public ProfileAndContactsLoader(Context context) { 48         super(context); 49     }</pre> <p><a href="https://android.googlesource.com/platform/packages/apps/ContactsCommon/+/nougat-mr1-release/src/com/android/contacts/common/list/ProfileAndContactsLoader.java">https://android.googlesource.com/platform/packages/apps/ContactsCommon/+/nougat-mr1-release/src/com/android/contacts/common/list/ProfileAndContactsLoader.java</a></p>

**Exhibit C for U.S. Patent No. 9,445,251 Against HTC Accused Products**

US9445251B2	HTC
	<pre>35 public final class GroupMemberLoader extends CursorLoader { 36 37     public static class GroupEditorQuery { 38         private static final String[] PROJECTION = new String[] { 39             Data.CONTACT_ID,           // 0 40             Data.RAW_CONTACT_ID,       // 1 41             Data.DISPLAY_NAME_PRIMARY, // 2 42             Data.PHOTO_URI,            // 3 43             Data.LOOKUP_KEY,           // 4 44         }; 45 46         public static final int CONTACT_ID           = 0; 47         public static final int RAW_CONTACT_ID       = 1; 48         public static final int CONTACT_DISPLAY_NAME_PRIMARY = 2; 49         public static final int CONTACT_PHOTO_URI    = 3; 50         public static final int CONTACT_LOOKUP_KEY   = 4; 51     } 52 53     public static class GroupDetailQuery { 54         private static final String[] PROJECTION = new String[] { 55             Data.CONTACT_ID,           // 0 56             Data.PHOTO_URI,            // 1 57             Data.LOOKUP_KEY,           // 2 58             Data.DISPLAY_NAME_PRIMARY, // 3 59             Data.CONTACT_PRESENCE,     // 4 60             Data.CONTACT_STATUS,       // 5 61         }; 62 63         public static final int CONTACT_ID           = 0; 64         public static final int CONTACT_PHOTO_URI    = 1; 65         public static final int CONTACT_LOOKUP_KEY   = 2; 66         public static final int CONTACT_DISPLAY_NAME_PRIMARY = 3; 67         public static final int CONTACT_PRESENCE_STATUS = 4; 68         public static final int CONTACT_STATUS       = 5; 69     } 70 71     private final long mGroupId;</pre> <p><a href="https://android.googlesource.com/platform/packages/apps/Contacts/+/nougat-mr1-release/src/com/android/contacts/GroupMemberLoader.java">https://android.googlesource.com/platform/packages/apps/Contacts/+/nougat-mr1-release/src/com/android/contacts/GroupMemberLoader.java</a></p>

**Exhibit C for U.S. Patent No. 9,445,251 Against HTC Accused Products**

US9445251B2	HTC
	<pre>24 * Group loader for the group list that includes details such as the number of contacts per group 25 * and number of groups per account. This list is sorted by account type, account name, where the 26 * group names are in alphabetical order. Note that the list excludes default, favorite, and deleted 27 * groups. 28 */ 29 public final class GroupListLoader extends CursorLoader { 30 31     private final static String[] COLUMNS = new String[] { 32         Groups.ACCOUNT_NAME, 33         Groups.ACCOUNT_TYPE, 34         Groups.DATA_SET, 35         Groups._ID, 36         Groups.TITLE, 37         Groups.SUMMARY_COUNT, 38     }; 39 40     public final static int ACCOUNT_NAME = 0; 41     public final static int ACCOUNT_TYPE = 1; 42     public final static int DATA_SET = 2; 43     public final static int GROUP_ID = 3; 44     public final static int TITLE = 4; 45     public final static int MEMBER_COUNT = 5; 46 47     private static final Uri GROUP_LIST_URI = Groups.CONTENT_SUMMARY_URI; 48 49     public GroupListLoader(Context context) { 50         super(context, GROUP_LIST_URI, COLUMNS, Groups.ACCOUNT_TYPE + " NOT NULL AND " 51             + Groups.ACCOUNT_NAME + " NOT NULL AND " + Groups.AUTO_ADD + "=0 AND " + 52             Groups.FAVORITES + "=0 AND " + Groups.DELETED + "=0", null, 53             Groups.ACCOUNT_TYPE + ", " + Groups.ACCOUNT_NAME + ", " + Groups.DATA_SET + ", " + 54             Groups.TITLE + " COLLATE LOCALIZED ASC"); 55     } 56 }</pre> <p><a href="https://android.googlesource.com/platform/packages/apps/Contacts+/nougat-mr1-release/src/com/android/contacts/GroupListLoader.java">https://android.googlesource.com/platform/packages/apps/Contacts+/nougat-mr1-release/src/com/android/contacts/GroupListLoader.java</a></p>



**Exhibit C for U.S. Patent No. 9,445,251 Against HTC Accused Products**

US9445251B2	HTC
	<pre> 24  * Group meta-data loader. Loads all groups or just a single group from the 25  * database (if given a {@link Uri}). 26  */ 27  public final class GroupMetaDataLoader extends CursorLoader { 28 29      private final static String[] COLUMNS = new String[] { 30          Groups.ACCOUNT_NAME, 31          Groups.ACCOUNT_TYPE, 32          Groups.DATA_SET, 33          Groups._ID, 34          Groups.TITLE, 35          Groups.AUTO_ADD, 36          Groups.FAVORITES, 37          Groups.GROUP_IS_READ_ONLY, 38          Groups.DELETED, 39      }; 40 41      public final static int ACCOUNT_NAME = 0; 42      public final static int ACCOUNT_TYPE = 1; 43      public final static int DATA_SET = 2; 44      public final static int GROUP_ID = 3; 45      public final static int TITLE = 4; 46      public final static int AUTO_ADD = 5; 47      public final static int FAVORITES = 6; 48      public final static int IS_READ_ONLY = 7; 49      public final static int DELETED = 8; 50 51      public GroupMetaDataLoader(Context context, Uri groupUri) { 52          super(context, ensureIsGroupUri(groupUri), COLUMNS, Groups.ACCOUNT_TYPE + " NOT NULL AND " 53              + Groups.ACCOUNT_NAME + " NOT NULL", null, null); 54      } 55 56      /** 57       * Ensures that this is a valid group URI. If invalid, then an exception is 58       * thrown. Otherwise, the original URI is returned. 59       */ 60      private static Uri ensureIsGroupUri(final Uri groupUri) { 61          // TODO: Fix ContactsProvider2 getType method to resolve the group Uri 62          if (groupUri == null) { 63              throw new IllegalArgumentException("Uri must not be null"); 64          } 65          if (!groupUri.toString().startsWith(Groups.CONTENT_URI.toString())) { 66              throw new IllegalArgumentException("Invalid group Uri: " + groupUri); 67          } 68          return groupUri; 69      } 70  } </pre> <p><a href="https://android.googlesource.com/platform/packages/apps/Contacts+/nougat-mr1-release/src/com/android/contacts/GroupMetaDataLoader.java">https://android.googlesource.com/platform/packages/apps/Contacts+/nougat-mr1-release/src/com/android/contacts/GroupMetaDataLoader.java</a></p>

**Exhibit C for U.S. Patent No. 9,445,251 Against HTC Accused Products**

US9445251B2	HTC
	<pre>19  * Meta-data for a contact group. We load all groups associated with the contact's 20  * constituent accounts. 21  */ 22  public final class GroupMetaData { 23      private String mAccountName; 24      private String mAccountType; 25      private String mDataSet; 26      private long mGroupId; 27      private String mTitle; 28      private boolean mDefaultGroup; 29      private boolean mFavorites; 30 31      public GroupMetaData(String accountName, String accountType, String dataSet, long groupId, 32                          String title, boolean defaultGroup, boolean favorites) { 33          this.mAccountName = accountName; 34          this.mAccountType = accountType; 35          this.mDataSet = dataSet; 36          this.mGroupId = groupId; 37          this.mTitle = title; 38          this.mDefaultGroup = defaultGroup; 39          this.mFavorites = favorites; 40      } 41 42      public String getAccountName() { 43          return mAccountName; 44      } 45 46      public String getAccountType() { 47          return mAccountType; 48      } 49 50      public String getDataSet() { 51          return mDataSet; 52      } 53 54      public long getGroupId() { 55          return mGroupId; 56      } 57 58      public String getTitle() { 59          return mTitle; 60      } 61 62      public boolean isDefaultGroup() { 63          return mDefaultGroup; 64      } 65 66      public boolean isFavorites() { 67          return mFavorites; 68      } 69  }</pre> <p><a href="https://android.googlesource.com/platform/packages/apps/ContactsCommon/+nougat-mr1-release/src/com/android/contacts/common/GroupMetaData.java">https://android.googlesource.com/platform/packages/apps/ContactsCommon/+nougat-mr1-release/src/com/android/contacts/common/GroupMetaData.java</a></p>

**Exhibit C for U.S. Patent No. 9,445,251 Against HTC Accused Products**

US9445251B2	HTC
	<pre>44  * Class that sends chat message via SMS. 45  * 46  * The interface emulates a blocking sending similar to making an HTTP request. 47  * It calls the SmsManager to send a (potentially multipart) message and waits 48  * on the sent status on each part. The waiting has a timeout so it won't wait 49  * forever. Once the sent status of all parts received, the call returns. 50  * A successful sending requires success status for all parts. Otherwise, we 51  * pick the highest level of failure as the error for the whole message, which 52  * is used to determine if we need to retry the sending. 53  */ 54  public class SmsSender { 55      private static final String TAG = LogUtil.BUGLE_TAG; 56 57      public static final String EXTRA_PART_ID = "part_id"; 58 59      /* 60       * A map for pending sms messages. The key is the random request UUID. 61       */ 62      private static ConcurrentHashMap&lt;Uri, SendResult&gt; sPendingMessageMap = 63          new ConcurrentHashMap&lt;Uri, SendResult&gt;(); 64 65      private static final Random RANDOM = new Random(); 66 67      // Whether we should send multipart SMS as separate messages 68      private static Boolean sSendMultipartSmsAsSeparateMessages = null; 69  }</pre> <p><a href="https://android.googlesource.com/platform/packages/apps/Messaging/+nougat-mr1-release/src/com/android/messaging/sms/SmsSender.java">https://android.googlesource.com/platform/packages/apps/Messaging/+nougat-mr1-release/src/com/android/messaging/sms/SmsSender.java</a></p>

**Exhibit C for U.S. Patent No. 9,445,251 Against HTC Accused Products**

US9445251B2	HTC
	<pre>253 // Actually sending the message using SmsManager 254 private static void sendInternal(final Context context, final int subId, String dest, 255     final ArrayList&lt;String&gt; messages, final String serviceCenter, 256     final boolean requireDeliveryReport, final Uri messageUri) throws SmsException { 257     Assert.notNull(context); 258     final SmsManager smsManager = PhoneUtils.get(subId).getSmsManager(); 259     final int messageCount = messages.size(); 260     final ArrayList&lt;PendingIntent&gt; deliveryIntents = new ArrayList&lt;PendingIntent&gt;(messageCount); 261     final ArrayList&lt;PendingIntent&gt; sentIntents = new ArrayList&lt;PendingIntent&gt;(messageCount); 262     for (int i = 0; i &lt; messageCount; i++) { 263         // Make pending intents different for each message part 264         final int partId = (messageCount &lt;= 1 ? 0 : i + 1); 265         if (requireDeliveryReport &amp;&amp; (i == (messageCount - 1))) { 266             // TODO we only care about the delivery status of the last part 267             // Shall we have better tracking of delivery status of all parts? 268             deliveryIntents.add(PendingIntent.getBroadcast( 269                 context, 270                 partId, 271                 getSendStatusIntent(context, SendStatusReceiver.MESSAGE_DELIVERED_ACTION, 272                     messageUri, partId, subId), 273                 0/*flag*/)); 274         } else { 275             deliveryIntents.add(null); 276         } 277         sentIntents.add(PendingIntent.getBroadcast( 278             context, 279             partId, 280             getSendStatusIntent(context, SendStatusReceiver.MESSAGE_SENT_ACTION, 281                 messageUri, partId, subId), 282             0/*flag*/)); 283     } 284     if (sSendMultipartSmsAsSeparateMessages == null) { 285         sSendMultipartSmsAsSeparateMessages = MmsConfig.get(subId) 286             .getSendMultipartSmsAsSeparateMessages(); 287     }</pre> <p><a href="https://android.googlesource.com/platform/packages/apps/Messaging/+nougat-mr1-release/src/com/android/messaging/sms/SmsSender.java">https://android.googlesource.com/platform/packages/apps/Messaging/+nougat-mr1-release/src/com/android/messaging/sms/SmsSender.java</a></p>

**Exhibit C for U.S. Patent No. 9,445,251 Against HTC Accused Products**

US9445251B2	HTC
	<pre>288     try { 289         if (sSendMultipartSmsAsSeparateMessages) { 290             // If multipart sms is not supported, send them as separate messages 291             for (int i = 0; i &lt; messageCount; i++) { 292                 smsManager.sendMessage(dest, 293                     serviceCenter, 294                     messages.get(i), 295                     sentIntents.get(i), 296                     deliveryIntents.get(i)); 297             } 298         } else { 299             smsManager.sendMultipartTextMessage( 300                 dest, serviceCenter, messages, sentIntents, deliveryIntents); 301         } 302     } catch (final Exception e) { 303         throw new SmsException("SmsSender: caught exception in sending " + e); 304     } 305 }</pre> <p><a href="https://android.goesource.com/platform/packages/apps/Messaging+/nougat-mr1-release/src/com/android/messaging/sms/SmsSender.java">https://android.goesource.com/platform/packages/apps/Messaging+/nougat-mr1-release/src/com/android/messaging/sms/SmsSender.java</a></p> <pre>56  * Class that receives incoming SMS messages through android.provider.Telephony.SMS_RECEIVED 57  * 58  * This class serves two purposes: 59  * - Process phone verification SMS messages 60  * - Handle SMS messages when the user has enabled us to be the default SMS app (Pre-KLP) 61  */ 62  public final class SmsReceiver extends BroadcastReceiver { 63      private static final String TAG = LogUtil.BUGLE_TAG; 64 65      private static ArrayList&lt;Pattern&gt; sIgnoreSmsPatterns; 66  }</pre> <p><a href="https://android.goesource.com/platform/packages/apps/Messaging+/nougat-mr1-release/src/com/android/messaging/receiver/SmsReceiver.java">https://android.goesource.com/platform/packages/apps/Messaging+/nougat-mr1-release/src/com/android/messaging/receiver/SmsReceiver.java</a></p>

**Exhibit C for U.S. Patent No. 9,445,251 Against HTC Accused Products**

US9445251B2	HTC
	<pre>200     public static void deliverSmsMessages(final Context context, final int subId, 201         final int errorCode, final android.telephony.SmsMessage[] messages) { 202         final ContentValues messageValues = 203             MmsUtils.parseReceivedSmsMessage(context, messages, errorCode); 204 205         LogUtil.v(TAG, "SmsReceiver.deliverSmsMessages"); 206 207         final long nowInMillis = System.currentTimeMillis(); 208         final long receivedTimestampMs = MmsUtils.getMessageDate(messages[0], nowInMillis); 209 210         messageValues.put(Sms.Inbox.DATE, receivedTimestampMs); 211         // Default to unread and unseen for us but ReceiveSmsMessageAction will override 212         // seen for the telephony db. 213         messageValues.put(Sms.Inbox.READ, 0); 214         messageValues.put(Sms.Inbox.SEEN, 0); 215         if (OsUtil.isAtLeastL_MR1()) { 216             messageValues.put(Sms.SUBSCRIPTION_ID, subId); 217         } 218 219         if (messages[0].getMessageClass() == android.telephony.SmsMessage.MessageClass.CLASS_0    220             DebugUtils.debugClassZeroSmsEnabled()) { 221             Factory.get().getUIIntents().launchClassZeroActivity(context, messageValues); 222         } else { 223             final ReceiveSmsMessageAction action = new ReceiveSmsMessageAction(messageValues); 224             action.start(); 225         } 226     }</pre> <p><a href="https://android.googlesource.com/platform/packages/apps/Messaging/+nougat-mr1-release/src/com/android/messaging/receiver/SmsReceiver.java">https://android.googlesource.com/platform/packages/apps/Messaging/+nougat-mr1-release/src/com/android/messaging/receiver/SmsReceiver.java</a></p>

**Exhibit C for U.S. Patent No. 9,445,251 Against HTC Accused Products**

US9445251B2	HTC
	<pre>228     @Override 229     public void onReceive(final Context context, final Intent intent) { 230         LogUtil.v(TAG, "SmsReceiver.onReceive " + intent); 231         // On KLP+ we only take delivery of SMS messages in SmsDeliverReceiver. 232         if (PhoneUtils.getDefault().isSmsEnabled()) { 233             final String action = intent.getAction(); 234             if (OsUtil.isSecondaryUser() &amp;&amp; 235                 (Telephony.Sms.Intents.SMS_RECEIVED_ACTION.equals(action)    236                  // TODO: update this with the actual constant from Telephony 237                  "android.provider.Telephony.MMS_DOWNLOADED".equals(action))) { 238                 postNewMessageSecondaryUserNotification(); 239             } else if (!OsUtil.isAtLeastKLP()) { 240                 deliverSmsIntent(context, intent); 241             } 242         } 243     }</pre> <p><a href="https://android.googlesource.com/platform/packages/apps/Messaging/+nougat-mr1-release/src/com/android/messaging/receiver/SmsReceiver.java">https://android.googlesource.com/platform/packages/apps/Messaging/+nougat-mr1-release/src/com/android/messaging/receiver/SmsReceiver.java</a></p>

**Exhibit C for U.S. Patent No. 9,445,251 Against HTC Accused Products**

US9445251B2	HTC
	<pre>52  * Class that sends chat message via MMS. 53  * 54  * The interface emulates a blocking send similar to making an HTTP request. 55  */ 56  public class MmsSender { 57      private static final String TAG = LogUtil.BUGLE_TAG; 58 59      /** 60       * Send an MMS message. 61       * 62       * @param context Context 63       * @param messageUri The unique URI of the message for identifying it during sending 64       * @param sendReq The SendReq PDU of the message 65       * @throws MmsFailureException 66       */ 67      public static void sendMms(final Context context, final int subId, final Uri messageUri, 68                               final SendReq sendReq, final Bundle sentIntentExtras) throws MmsFailureException { 69          sendMms(context, 70                subId, 71                messageUri, 72                null /* locationUri */, 73                sendReq, 74                true /* responseImportant */, 75                sentIntentExtras); 76      }</pre> <p><a href="https://android.googlesource.com/platform/packages/apps/Messaging/+nougat-mr1-release/src/com/android/messaging/sms/MmsSender.java">https://android.googlesource.com/platform/packages/apps/Messaging/+nougat-mr1-release/src/com/android/messaging/sms/MmsSender.java</a></p>



**Exhibit C for U.S. Patent No. 9,445,251 Against HTC Accused Products**

US9445251B2	HTC
	<pre>240      * Download an MMS message. 241      * 242      * @param context Context 243      * @param contentLocation The url of the MMS message 244      * @throws MmsFailureException 245      * @throws InvalidHeaderValueException 246      */ 247      public static void downloadMms(final Context context, final int subId, 248          final String contentLocation, Bundle extras) throws MmsFailureException, 249          InvalidHeaderValueException { 250          final Uri requestUri = Uri.parse(contentLocation); 251          final Uri contentUri = MmsFileProvider.buildRawMmsUri(); 252 253          final Intent downloadedIntent = new Intent(SendStatusReceiver.MMS_DOWNLOADED_ACTION, 254              requestUri, 255              context, 256              SendStatusReceiver.class); 257          downloadedIntent.putExtra(SendMessageAction.EXTRA_CONTENT_URI, contentUri); 258          if (extras != null) { 259              downloadedIntent.putExtras(extras); 260          } 261          final PendingIntent downloadedPendingIntent = PendingIntent.getBroadcast( 262              context, 263              0 /*request code*/, 264              downloadedIntent, 265              PendingIntent.FLAG_UPDATE_CURRENT); 266 267          MmsManager.downloadMultimediaMessage(subId, context, contentLocation, contentUri, 268              downloadedPendingIntent); 269      }</pre> <p><a href="https://android.googlesource.com/platform/packages/apps/Messaging/+nougat-mr1-release/src/com/android/messaging/sms/MmsSender.java">https://android.googlesource.com/platform/packages/apps/Messaging/+nougat-mr1-release/src/com/android/messaging/sms/MmsSender.java</a></p>

**Exhibit C for U.S. Patent No. 9,445,251 Against HTC Accused Products**

US9445251B2	HTC
	<pre> 97      * Execute an MMS HTTP request, either a POST (sending) or a GET (downloading) 98      * 99      * @param urlString The request URL, for sending it is usually the MMSC, and for downloading 100     *           it is the message URL 101     * @param pdu For POST (sending) only, the PDU to send 102     * @param method HTTP method, POST for sending and GET for downloading 103     * @param isProxySet Is there a proxy for the MMSC 104     * @param proxyHost The proxy host 105     * @param proxyPort The proxy port 106     * @param mmsConfig The MMS config to use 107     * @param userAgent The user agent header value 108     * @param uaProfUrl The UA Prof URL header value 109     * @return The HTTP response body 110     * @throws MmsHttpException For any failures 111     */ 112     public byte[] execute(String urlString, byte[] pdu, String method, boolean isProxySet, 113                          String proxyHost, int proxyPort, Bundle mmsConfig, String userAgent, String uaProfUrl) 114         throws MmsHttpException { 115         Log.d(MmsService.TAG, "HTTP: " + method + " " + Utils.redactUrlForNonVerbose(urlString) 116             + (isProxySet ? (" proxy=" + proxyHost + ":" + proxyPort) : "") 117             + ", PDU size=" + (pdu != null ? pdu.length : 0)); 118         checkMethod(method); 119         HttpURLConnection connection = null; 120         try { 121             Proxy proxy = Proxy.NO_PROXY; 122             if (isProxySet) { 123                 proxy = new Proxy(Proxy.Type.HTTP, new InetSocketAddress(proxyHost, proxyPort)); 124             } 125             final URL url = new URL(urlString); 126             // Now get the connection 127             connection = (HttpURLConnection) url.openConnection(proxy); 128             connection.setDoInput(true); 129             connection.setConnectTimeout( 130                 mmsConfig.getInt(CarrierConfigValuesLoader.CONFIG_HTTP_SOCKET_TIMEOUT, 131                     CarrierConfigValuesLoader.CONFIG_HTTP_SOCKET_TIMEOUT_DEFAULT)); </pre> <p><a href="https://android.googlesource.com/platform/packages/apps/Messaging/+nougat-mr1-release/src/android/support/v7/mms/MmsHttpClient.java">https://android.googlesource.com/platform/packages/apps/Messaging/+nougat-mr1-release/src/android/support/v7/mms/MmsHttpClient.java</a></p>

**Exhibit C for U.S. Patent No. 9,445,251 Against HTC Accused Products**

US9445251B2	HTC
	<pre> 132 // ----- COMMON HEADERS ----- 133 // Header: Accept 134 connection.setRequestProperty(HEADER_ACCEPT, HEADER_VALUE_ACCEPT); 135 // Header: Accept-Language 136 connection.setRequestProperty( 137     HEADER_ACCEPT_LANGUAGE, getCurrentAcceptLanguage(Locale.getDefault())); 138 // Header: User-Agent 139 Log.i(MmsService.TAG, "HTTP: User-Agent=" + userAgent); 140 connection.setRequestProperty(HEADER_USER_AGENT, userAgent); 141 // Header: x-wap-profile 142 final String uaProfUrlTagName = mmsConfig.getString( 143     CarrierConfigValuesLoader.CONFIG_UA_PROF_TAG_NAME, 144     CarrierConfigValuesLoader.CONFIG_UA_PROF_TAG_NAME_DEFAULT); 145 if (uaProfUrl != null) { 146     Log.i(MmsService.TAG, "HTTP: UaProfUrl=" + uaProfUrl); 147     connection.setRequestProperty(uaProfUrlTagName, uaProfUrl); 148 } 149 // Add extra headers specified by mms_config.xml's httpparams 150 addExtraHeaders(connection, mmsConfig); 151 // Different stuff for GET and POST 152 if (METHOD_POST.equals(method)) { 153     if (pdu == null    pdu.length &lt; 1) { 154         Log.e(MmsService.TAG, "HTTP: empty pdu"); 155         throw new MmsHttpException(0/*statusCode*/, "Sending empty PDU"); 156     } 157     connection.setDoOutput(true); 158     connection.setRequestMethod(METHOD_POST); 159     if (mmsConfig.getBoolean( 160         CarrierConfigValuesLoader.CONFIG_SUPPORT_HTTP_CHARSET_HEADER, 161         CarrierConfigValuesLoader.CONFIG_SUPPORT_HTTP_CHARSET_HEADER_DEFAULT)) { 162         connection.setRequestProperty(HEADER_CONTENT_TYPE, 163             HEADER_VALUE_CONTENT_TYPE_WITH_CHARSET); 164     } else { 165         connection.setRequestProperty(HEADER_CONTENT_TYPE, 166             HEADER_VALUE_CONTENT_TYPE_WITHOUT_CHARSET); </pre> <p><a href="https://android.googlesource.com/platform/packages/apps/Messaging+/nougat-mr1-release/src/android/support/v7/mms/MmsHttpClient.java">https://android.googlesource.com/platform/packages/apps/Messaging+/nougat-mr1-release/src/android/support/v7/mms/MmsHttpClient.java</a></p>

**Exhibit C for U.S. Patent No. 9,445,251 Against HTC Accused Products**

US9445251B2	HTC
	<pre> 167         } 168         if (Log.isLoggable(MmsService.TAG, Log.VERBOSE)) { 169             logHttpHeaders(connection.getRequestProperties()); 170         } 171         connection.setFixedLengthStreamingMode(pdu.length); 172         // Sending request body 173         final OutputStream out = 174             new BufferedOutputStream(connection.getOutputStream()); 175         out.write(pdu); 176         out.flush(); 177         out.close(); 178     } else if (METHOD_GET.equals(method)) { 179         if (Log.isLoggable(MmsService.TAG, Log.VERBOSE)) { 180             logHttpHeaders(connection.getRequestProperties()); 181         } 182         connection.setRequestMethod(METHOD_GET); 183     } 184     // Get response 185     final int responseCode = connection.getResponseCode(); 186     final String responseMessage = connection.getResponseMessage(); 187     Log.d(MmsService.TAG, "HTTP: " + responseCode + " " + responseMessage); 188     if (Log.isLoggable(MmsService.TAG, Log.VERBOSE)) { 189         logHttpHeaders(connection.getHeaderFields()); 190     } 191     if (responseCode / 100 != 2) { 192         throw new MmsHttpException(responseCode, responseMessage); 193     } 194     final InputStream in = new BufferedInputStream(connection.getInputStream()); 195     final ByteArrayOutputStream byteOut = new ByteArrayOutputStream(); 196     final byte[] buf = new byte[4096]; 197     int count = 0; 198     while ((count = in.read(buf)) &gt; 0) { 199         byteOut.write(buf, 0, count); 200     } 201     in.close(); 202     final byte[] responseBody = byteOut.toByteArray(); 203     Log.d(MmsService.TAG, "HTTP: response size=" 204         + (responseBody != null ? responseBody.length : 0)); 205     return responseBody; </pre> <p><a href="https://android.googlesource.com/platform/packages/apps/Messaging+/nougat-mr1-release/src/android/support/v7/mms/MmsHttpClient.java">https://android.googlesource.com/platform/packages/apps/Messaging+/nougat-mr1-release/src/android/support/v7/mms/MmsHttpClient.java</a></p>

**Exhibit C for U.S. Patent No. 9,445,251 Against HTC Accused Products**

US9445251B2	HTC
	<pre>38  * Request to send an MMS 39  */ 40  class SendRequest extends MmsRequest { 41      // Max send response PDU size in bytes (exceeding this may cause problem with 42      // system intent delivery). 43      private static final int MAX_SEND_RESPONSE_SIZE = 1000 * 1024; 44 45      private byte[] mPduData; 46 47      SendRequest(final String locationUri, final Uri pduUri, final PendingIntent sentIntent) { 48          super(locationUri, pduUri, sentIntent); 49      } 50 51      @Override 52      protected boolean loadRequest(final Context context, final Bundle mmsConfig) { 53          mPduData = readPduFromContentUri( 54              context, 55              mPduUri, 56              mmsConfig.getInt( 57                  CarrierConfigValuesLoader.CONFIG_MAX_MESSAGE_SIZE, 58                  CarrierConfigValuesLoader.CONFIG_MAX_MESSAGE_SIZE_DEFAULT)); 59          return (mPduData != null); 60      } 61 62      @Override 63      protected boolean transferResponse(final Context context, final Intent fillIn, 64          final byte[] response) { 65          // SendConf pdus are always small and can be included in the intent 66          if (response != null &amp;&amp; fillIn != null) { 67              if (response.length &gt; MAX_SEND_RESPONSE_SIZE) { 68                  // If the response PDU is too large, it won't be able to fit in 69                  // the PendingIntent to be transferred via system IPC. 70                  return false; 71              } 72              fillIn.putExtra(SmsManager.EXTRA_MMS_DATA, response); 73          } 74          return true; 75      } }</pre>

**Exhibit C for U.S. Patent No. 9,445,251 Against HTC Accused Products**

US9445251B2	HTC
	<p data-bbox="499 237 1583 302"><a href="https://android.googlesource.com/platform/packages/apps/Messaging+/nougat-mr1-release/src/android/support/v7/mms/SendRequest.java">https://android.googlesource.com/platform/packages/apps/Messaging+/nougat-mr1-release/src/android/support/v7/mms/SendRequest.java</a></p> <pre data-bbox="520 358 968 391">public static LocationRequest create ()</pre> <p data-bbox="512 427 1016 451">Create a location request with default parameters.</p> <p data-bbox="512 483 1625 540">Default parameters are for a block accuracy, slowly updated location. It can then be adjusted as required by the applications before passing to the <a href="#">FusedLocationProviderApi</a>.</p> <p data-bbox="533 565 617 589"><b>Returns</b></p> <ul data-bbox="541 613 800 638" style="list-style-type: none"><li>• a new location request</li></ul> <p data-bbox="499 654 1787 682"><a href="https://developers.google.com/android/reference/com/google/android/gms/location/LocationRequest">https://developers.google.com/android/reference/com/google/android/gms/location/LocationRequest</a></p>

**Exhibit C for U.S. Patent No. 9,445,251 Against HTC Accused Products**

US9445251B2	HTC
	<p><b>public static final int PRIORITY_BALANCED_POWER_ACCURACY</b></p> <p>Used with <code>setPriority(int)</code> to request "block" level accuracy.</p> <p>Block level accuracy is considered to be about 100 meter accuracy. Using a coarse accuracy such as this often consumes less power.</p> <p>Constant Value: 102</p> <p><b>public static final int PRIORITY_HIGH_ACCURACY</b></p> <p>Used with <code>setPriority(int)</code> to request the most accurate locations available.</p> <p>This will return the finest location available.</p> <p>Constant Value: 100</p> <p><b>public static final int PRIORITY_LOW_POWER</b></p> <p>Used with <code>setPriority(int)</code> to request "city" level accuracy.</p> <p>City level accuracy is considered to be about 10km accuracy. Using a coarse accuracy such as this often consumes less power.</p> <p>Constant Value: 104</p> <p><a href="https://developers.google.com/android/reference/com/google/android/gms/location/LocationRequest">https://developers.google.com/android/reference/com/google/android/gms/location/LocationRequest</a></p>

**Exhibit C for U.S. Patent No. 9,445,251 Against HTC Accused Products**

US9445251B2	HTC
	<p data-bbox="520 253 1734 285"><code>public Task&lt;Location&gt; getLastLocation ()</code></p> <p data-bbox="520 318 1087 342">Returns the best most recent location currently available.</p> <p data-bbox="520 375 1682 431">If a location is not available, which should happen very rarely, null will be returned. The best accuracy available while respecting the location permissions will be returned.</p> <p data-bbox="520 464 1724 521">This method provides a simplified way to get location. It is particularly well suited for applications that do not require an accurate location and that do not want to maintain extra logic for location updates.</p> <p data-bbox="520 578 1734 610"><code>public Task&lt;LocationAvailability&gt; getLocationAvailability ()</code></p> <p data-bbox="520 643 1675 699">Returns the availability of location data. When <code>isLocationAvailable()</code> returns true, then the location returned by <code>getLastLocation()</code> will be reasonably up to date within the hints specified by the active <code>LocationRequest</code> s.</p> <p data-bbox="520 732 1457 756">If the client isn't connected to Google Play services and the request times out, null is returned.</p> <p data-bbox="520 789 1661 846">Note it's always possible for <code>getLastLocation()</code> to return null even when this method returns true (e.g. location settings were disabled between calls).</p> <p data-bbox="520 862 1902 927"><a href="https://developers.google.com/android/reference/com/google/android/gms/location/FusedLocationProviderClient">https://developers.google.com/android/reference/com/google/android/gms/location/FusedLocationProviderClient</a></p>



**Exhibit C for U.S. Patent No. 9,445,251 Against HTC Accused Products**

US9445251B2	HTC						
	<p data-bbox="514 245 1736 329"><code>public Task&lt;Void&gt; requestLocationUpdates (LocationRequest request, LocationCallback callback, Looper looper)</code></p> <p data-bbox="514 355 1257 380">Requests location updates with a callback on the specified Looper thread.</p> <p data-bbox="514 412 1671 472">This method is suited for the foreground use cases. For background use cases, the <code>PendingIntent</code> version of the method is recommended, see <code>requestLocationUpdates(LocationRequest, PendingIntent)</code>.</p> <p data-bbox="514 505 1356 529">Any previous LocationRequests registered on this LocationListener will be replaced.</p> <p data-bbox="514 561 1671 656">This call will keep the Google Play services connection active, so make sure to call <code>removeLocationUpdates(LocationCallback)</code> when you no longer need it, otherwise you lose the benefits of the automatic connection management.</p> <p data-bbox="514 688 1728 712">Callbacks for <code>LocationCallback</code> will be made on the specified thread, which must already be a prepared looper thread.</p> <p data-bbox="531 737 653 761"><b>Parameters</b></p> <table border="1" data-bbox="514 792 1736 1008"> <tbody> <tr> <td data-bbox="514 792 619 862"><b>request</b></td> <td data-bbox="619 792 1736 862">The location request for the updates.</td> </tr> <tr> <td data-bbox="514 862 619 932"><b>callback</b></td> <td data-bbox="619 862 1736 932">The callback for the location updates.</td> </tr> <tr> <td data-bbox="514 932 619 1008"><b>looper</b></td> <td data-bbox="619 932 1736 1008">The Looper object whose message queue will be used to implement the callback mechanism, or null to make callbacks on the calling thread.</td> </tr> </tbody> </table> <p data-bbox="504 1024 1906 1089"><a href="https://developers.google.com/android/reference/com/google/android/gms/location/FusedLocationProviderClient">https://developers.google.com/android/reference/com/google/android/gms/location/FusedLocationProviderClient</a></p>	<b>request</b>	The location request for the updates.	<b>callback</b>	The callback for the location updates.	<b>looper</b>	The Looper object whose message queue will be used to implement the callback mechanism, or null to make callbacks on the calling thread.
<b>request</b>	The location request for the updates.						
<b>callback</b>	The callback for the location updates.						
<b>looper</b>	The Looper object whose message queue will be used to implement the callback mechanism, or null to make callbacks on the calling thread.						

**Exhibit C for U.S. Patent No. 9,445,251 Against HTC Accused Products**

US9445251B2	HTC				
	<pre data-bbox="520 245 1730 326">public Task&lt;Void&gt; requestLocationUpdates (LocationRequest request, PendingIntent callbackIntent)</pre> <p data-bbox="520 354 1255 378">Requests location updates with a callback on the specified PendingIntent.</p> <p data-bbox="520 410 1717 540">This method is suited for the background use cases, more specifically for receiving location updates, even when the app has been killed by the system. In order to do so, use a <code>PendingIntent</code> for a started service. For foreground use cases, the <code>LocationCallback</code> version of the method is recommended, see <code>requestLocationUpdates(LocationRequest, LocationCallback, Looper)</code>.</p> <p data-bbox="520 573 1717 630">Any previously registered requests that have the same PendingIntent (as defined by <code>equals(Object)</code>) will be replaced by this request.</p> <p data-bbox="520 662 1717 760">Both <code>LocationResult</code> and <code>LocationAvailability</code> are sent to the given PendingIntent. You can extract data from an Intent using <code>hasResult(Intent)</code>, <code>extractResult(Intent)</code>, <code>hasLocationAvailability(Intent)</code>, and <code>extractLocationAvailability(Intent)</code>.</p> <p data-bbox="520 784 657 808"><b>Parameters</b></p> <table border="1" data-bbox="520 833 1730 971"> <tbody> <tr> <td data-bbox="520 833 825 906"><code>request</code></td> <td data-bbox="825 833 1730 906">The location request for the updates.</td> </tr> <tr> <td data-bbox="520 906 825 971"><code>callbackIntent</code></td> <td data-bbox="825 906 1730 971">A pending intent to be sent for each location update.</td> </tr> </tbody> </table> <p data-bbox="520 995 615 1019"><b>Returns</b></p> <ul data-bbox="520 1044 1350 1068" style="list-style-type: none"> <li>• a Task for the call, check <code>isSuccessful()</code> to determine if it was successful.</li> </ul> <p data-bbox="520 1084 1906 1141"><a href="https://developers.google.com/android/reference/com/google/android/gms/location/FusedLocationProviderClient">https://developers.google.com/android/reference/com/google/android/gms/location/FusedLocationProviderClient</a></p>	<code>request</code>	The location request for the updates.	<code>callbackIntent</code>	A pending intent to be sent for each location update.
<code>request</code>	The location request for the updates.				
<code>callbackIntent</code>	A pending intent to be sent for each location update.				

## Exhibit C for U.S. Patent No. 9,445,251 Against HTC Accused Products

US9445251B2	HTC						
	<p><code>public void onLocationAvailability (LocationAvailability locationAvailability)</code></p> <p>Called when there is a change in the availability of location data.</p> <p>When <code>isLocationAvailable()</code> returns <code>false</code> you can assume that location will not be returned in <code>onLocationResult(LocationResult)</code> until something changes in the device's settings or environment. Even when <code>isLocationAvailable()</code> returns <code>true</code> the <code>onLocationResult(LocationResult)</code> may not always be called regularly, however the device location is known and both the most recently delivered location and <code>getLastLocation(GoogleApiClient)</code> will be reasonably up to date given the hints specified by the active <code>LocationRequest</code>s.</p> <p><b>Parameters</b></p> <table border="1"> <tr> <td><code>locationAvailability</code></td> <td>The current status of location availability.</td> </tr> </table> <p><code>public void onLocationResult (LocationResult result)</code></p> <p>Called when device location information is available.</p> <p>The most recent location returned by <code>getLastLocation()</code> is not guaranteed to be immediately fresh, but will be reasonably up to date given the hints specified by the active <code>LocationRequest</code>s.</p> <p><b>Parameters</b></p> <table border="1"> <tr> <td><code>result</code></td> <td>The latest location result available.</td> </tr> </table> <p><a href="https://developers.google.com/android/reference/com/google/android/gms/location/LocationCallback">https://developers.google.com/android/reference/com/google/android/gms/location/LocationCallback</a></p> <p><code>public abstract void onLocationChanged (Location location)</code></p> <p>Called when the location has changed.</p> <p><b>Parameters</b></p> <table border="1"> <tr> <td><code>location</code></td> <td>The updated location.</td> </tr> </table>	<code>locationAvailability</code>	The current status of location availability.	<code>result</code>	The latest location result available.	<code>location</code>	The updated location.
<code>locationAvailability</code>	The current status of location availability.						
<code>result</code>	The latest location result available.						
<code>location</code>	The updated location.						

**Exhibit C for U.S. Patent No. 9,445,251 Against HTC Accused Products**

US9445251B2	HTC
	<p data-bbox="499 235 1787 264"><a href="https://developers.google.com/android/reference/com/google/android/gms/location/LocationListener">https://developers.google.com/android/reference/com/google/android/gms/location/LocationListener</a></p> <p data-bbox="512 318 800 347">Public Constructors</p> <hr data-bbox="512 358 1730 362"/> <p data-bbox="512 407 1730 453">public <b>MapView</b> (<a href="#">Context</a> context)</p> <p data-bbox="512 500 1730 545">public <b>MapView</b> (<a href="#">Context</a> context, <a href="#">AttributeSet</a> attrs)</p> <p data-bbox="512 592 1730 638">public <b>MapView</b> (<a href="#">Context</a> context, <a href="#">AttributeSet</a> attrs, int defStyle)</p> <p data-bbox="512 685 1730 730">public <b>MapView</b> (<a href="#">Context</a> context, <a href="#">GoogleMapOptions</a> options)</p> <p data-bbox="499 748 1661 777"><a href="https://developers.google.com/android/reference/com/google/android/gms/maps/MapView">https://developers.google.com/android/reference/com/google/android/gms/maps/MapView</a></p>

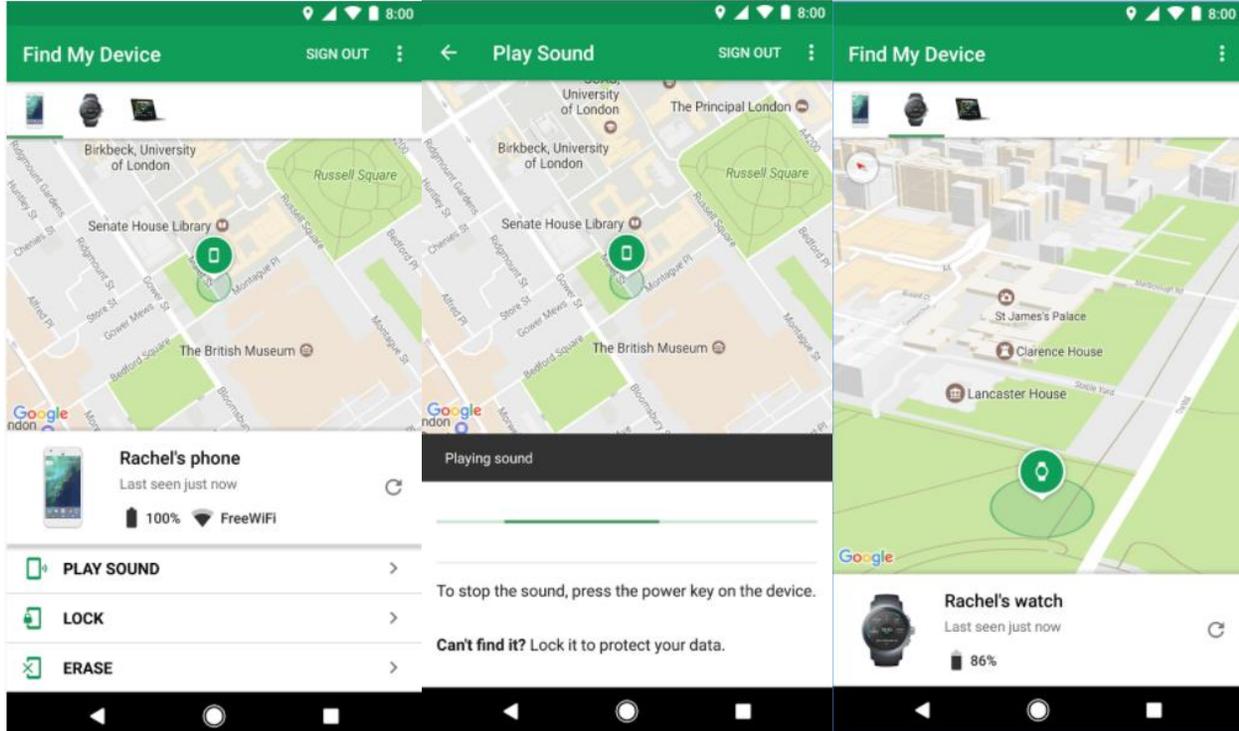
**Exhibit C for U.S. Patent No. 9,445,251 Against HTC Accused Products**

US9445251B2	HTC
	<p><code>public void <b>getMapAsync</b> (<b>OnMapReadyCallback</b> callback)</code></p> <p>Returns a non-null instance of the <code>GoogleMap</code>, ready to be used.</p> <p>Note that:</p> <ul style="list-style-type: none"> <li>• This method must be called from the main thread.</li> <li>• The callback will be executed in the main thread.</li> <li>• In the case where Google Play services is not installed on the user's device, the callback will not be triggered until the user installs it.</li> <li>• The <code>GoogleMap</code> object provided by the callback is non-null.</li> </ul> <p><b>Parameters</b></p> <p><code>callback</code> The callback object that will be triggered when the map is ready to be used.</p> <p><code>public final void <b>onCreate</b> (<b>Bundle</b> savedInstanceState)</code></p> <p>You must call this method from the parent Activity/Fragment's corresponding method.</p> <p><a href="https://developers.google.com/android/reference/com/google/android/gms/maps/MapView">https://developers.google.com/android/reference/com/google/android/gms/maps/MapView</a></p>
<p>[1C] presenting, via an interactive display of the first device, a first interactive, georeferenced map and a plurality of user-selectable symbols corresponding to the plurality of second devices, wherein the symbols are</p>	<p>HTC infringes directly and/or indirectly by performing, inducing others to perform, and/or contributing to the performance of: presenting, via an interactive display of the first device, a first interactive, georeferenced map and a plurality of user-selectable symbols corresponding to the plurality of second devices, wherein the symbols are positioned on the first georeferenced map at respective positions corresponding to the locations of the second devices, and wherein the first georeferenced map includes data relating positions on the first georeferenced map to spatial coordinates.</p> <p><b>Regarding Find My Device</b> and Android Device Manager, the Accused Products display, to the user on the display of the first device, a map with one or more symbols corresponding to one or more second devices. The map is interactive because the user may control the display of the map, e.g., pan, zoom, and/or effect change to the map in an otherwise interactive manner. The map is georeferenced for at least the reason that one or more symbols are associated with spatial locations, i.e., coordinates. The symbols are user-selectable</p>

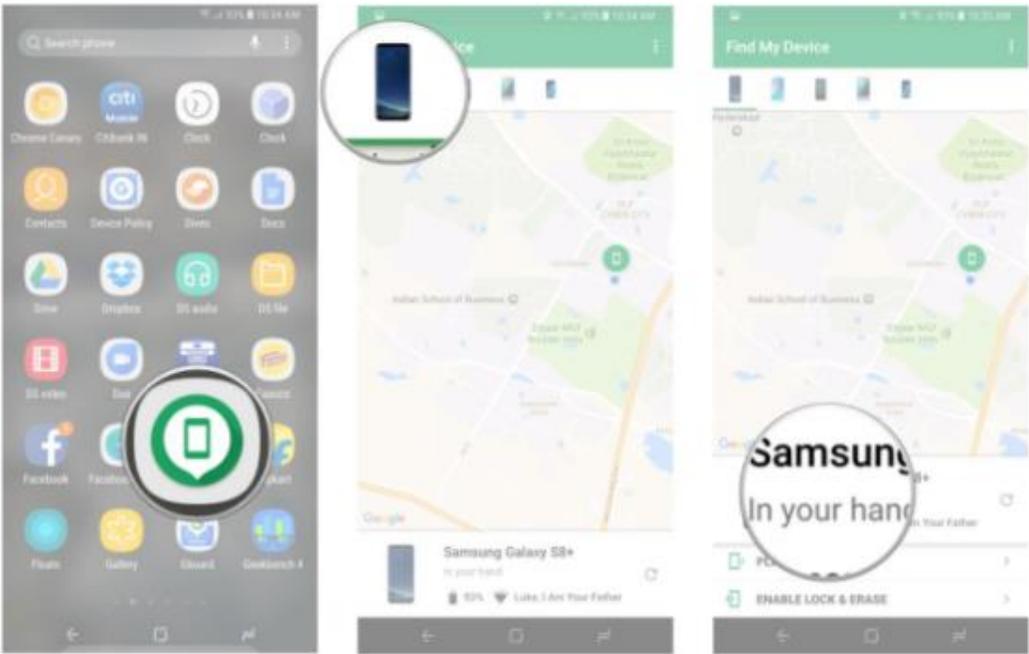
**Exhibit C for U.S. Patent No. 9,445,251 Against HTC Accused Products**

<b>US9445251B2</b>	<b>HTC</b>
<p>positioned on the first georeferenced map at respective positions corresponding to the locations of the second devices, and wherein the first georeferenced map includes data relating positions on the first georeferenced map to spatial coordinates;</p>	<p>because a user may touch the display to select the device associated with the symbol. For example, the user may input a touch selection directly on or near the portion of the display corresponding to the symbol's coordinates on the map to effect a selection of the device or to invoke sub-menus.</p> <p><b><u>Regarding Google Maps</u></b>, Google Latitude, Google Plus, Google Hangouts, Google Messages, Google Allo, Google Duo, Google Chrome, and Android Messenger, the Accused Products display, to the user on the display of the first device, a map with one or more symbols corresponding to one or more second users (or second devices corresponding to the second users). The map is interactive because the user may control the display of the map, e.g., pan, zoom, and/or effect change to the map in an otherwise interactive manner. The map is georeferenced for at least the reason that one or more symbols are associated with spatial locations, i.e., coordinates. The symbols are user-selectable because a user may touch the display to select the user or device associated with the symbol. For example, the user may input a touch selection directly on or near the portion of the display corresponding to the symbol's coordinates on the map to effect a selection of the user or device.</p> <p><b><u>Exemplary Support for Find My Device:</u></b></p>

**Exhibit C for U.S. Patent No. 9,445,251 Against HTC Accused Products**

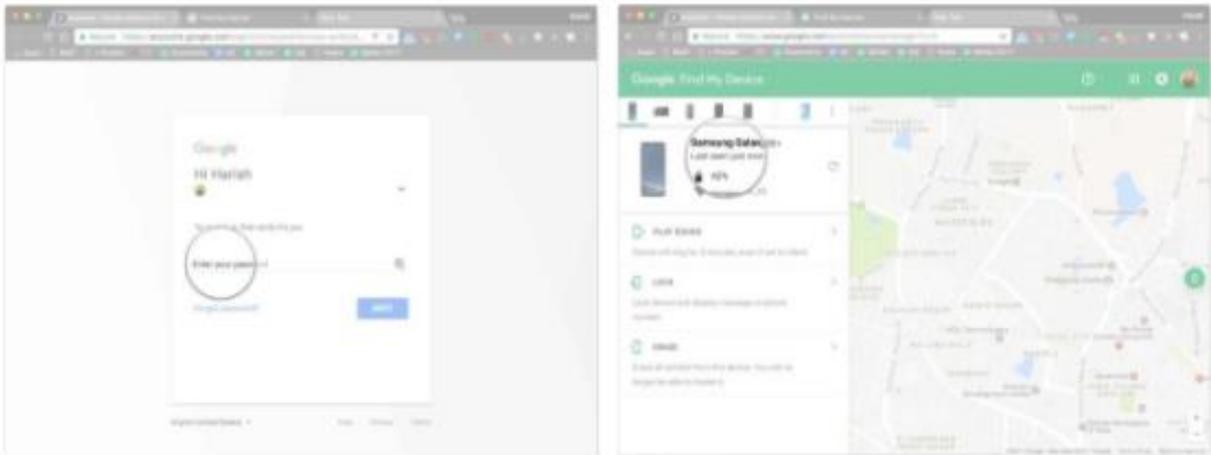
US9445251B2	HTC
	 <p><a href="https://play.google.com/store/apps/details?id=com.google.android.apps.adm&amp;hl=en">https://play.google.com/store/apps/details?id=com.google.android.apps.adm&amp;hl=en</a></p>

**Exhibit C for U.S. Patent No. 9,445,251 Against HTC Accused Products**

US9445251B2	HTC
	<p>Once you're signed in to Find My Device, you'll see a map with your current location as well as the make and model of your phone, and two options — Play Sound, and Enable Lock &amp; Erase. Hitting the latter option will allow you to start using the Lock and Erase functions.</p> <p>If you've signed into more than one phone, you can select a particular device by browsing the list at the top of the screen.</p> <ol style="list-style-type: none"> <li>1. Open <b>Find My Device</b> from your home screen or app drawer.</li> <li>2. Select your phone from the list of devices at the top of the screen.</li> <li>3. See if your phone is <b>discoverable</b>.</li> </ol>  <p><a href="https://www.androidcentral.com/find-my-device">https://www.androidcentral.com/find-my-device</a></p>



**Exhibit C for U.S. Patent No. 9,445,251 Against HTC Accused Products**

US9445251B2	HTC
	<h2 data-bbox="512 272 1650 329">How to locate your phone over the internet</h2> <p data-bbox="512 370 1659 513">If you've lost your phone, you can remotely locate it through the <a href="#">Find My Device website</a>. You'll need to sign in to the Google account that was used to set up Find My Device. It takes a few seconds, but the service should be able to track your phone. Alternatively, you can also do a Google search for "find my phone" to locate your handset.</p> <ol data-bbox="504 570 997 724" style="list-style-type: none"><li>1. Head to the <a href="#">Find My Device website</a>.</li><li>2. Sign in to your <b>Google account</b>.</li><li>3. Check if your device is visible.</li></ol> <div data-bbox="548 764 1759 1219"></div> <p data-bbox="499 1252 1121 1284"><a href="https://www.androidcentral.com/find-my-device">https://www.androidcentral.com/find-my-device</a></p>

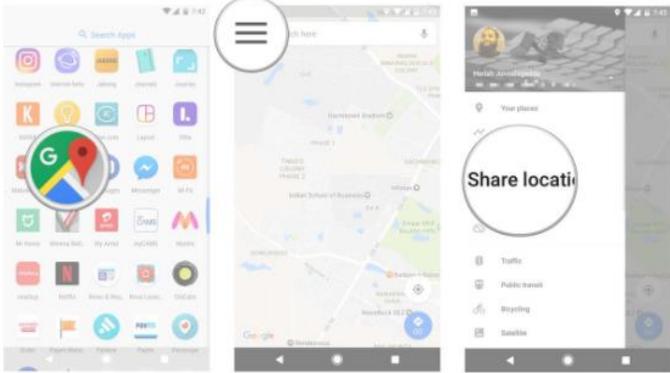
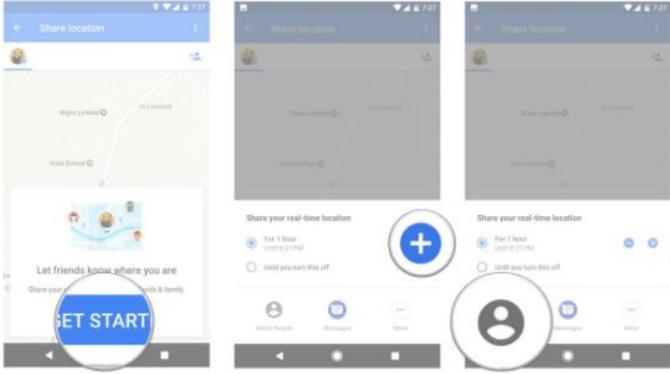
**Exhibit C for U.S. Patent No. 9,445,251 Against HTC Accused Products**

US9445251B2	HTC
	<p data-bbox="499 381 1014 414"><b><u>Exemplary Support for Google Maps:</u></b></p>

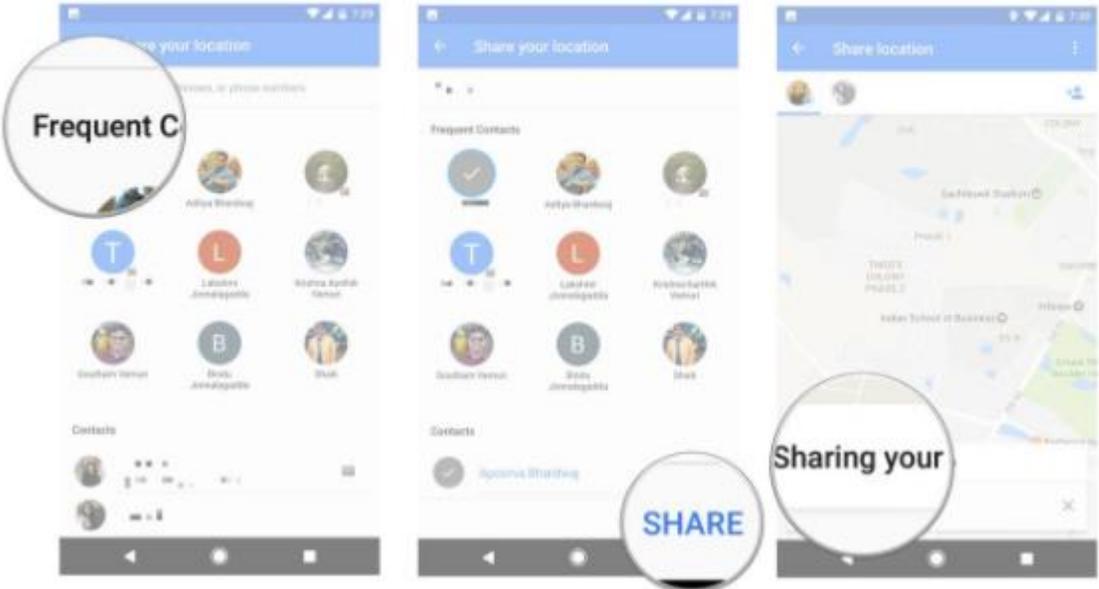
## Exhibit C for U.S. Patent No. 9,445,251 Against HTC Accused Products

US9445251B2	HTC
	<h3 data-bbox="520 256 827 298">Share your E.T.A</h3> <p data-bbox="520 326 1650 350">After you start your drive, you can share your destination, estimated arrival time, and where you are on the route.</p> <ol data-bbox="531 380 1346 618" style="list-style-type: none"><li data-bbox="531 380 873 407">1. Open the Google Maps app .</li><li data-bbox="531 423 1146 451">2. Set a driving destination. <a href="#">Learn how to navigate to a place.</a></li><li data-bbox="531 467 1188 495">3. After you start navigation, tap More ^ &gt; <b>Share trip progress.</b></li><li data-bbox="531 511 863 539">4. Choose a person from the list.</li><li data-bbox="531 555 663 583">5. Tap <b>Share.</b></li><li data-bbox="531 599 1346 626">6. Location Sharing will stop when you reach your destination or stop navigating.</li></ol> <ul data-bbox="531 651 1188 678" style="list-style-type: none"><li data-bbox="531 651 1188 678">• To stop sharing before you arrive, tap More ^ &gt; <b>Stop sharing.</b></li></ul> <h3 data-bbox="520 743 942 786">See where someone is</h3> <p data-bbox="520 813 1251 837">If someone shares their location with you, you can see them on the map.</p> <ol data-bbox="531 867 905 976" style="list-style-type: none"><li data-bbox="531 867 873 894">1. Open the Google Maps app .</li><li data-bbox="531 911 905 938">2. Tap Menu ≡ &gt; <b>Location sharing.</b></li><li data-bbox="531 954 737 982">3. Choose someone.</li></ol> <ul data-bbox="531 1008 1289 1036" style="list-style-type: none"><li data-bbox="531 1008 1289 1036">• To see an updated location, tap on a friend's icon &gt; More ⋮ &gt; <b>Refresh.</b></li></ul> <h3 data-bbox="520 1089 1020 1131">Stop seeing someone's location</h3> <ol data-bbox="531 1149 1444 1300" style="list-style-type: none"><li data-bbox="531 1149 873 1177">1. Open the Google Maps app .</li><li data-bbox="531 1193 821 1221">2. On the map, tap their icon.</li><li data-bbox="531 1237 842 1265">3. At the bottom, tap More ^ .</li><li data-bbox="531 1281 1444 1308">4. To temporarily hide someone, tap <b>Hide from map.</b> You can stop hiding them at any time.</li></ol> <p data-bbox="520 1333 1730 1357"><b>Note:</b> You can stop someone's location from ever appearing on your map. <a href="#">Learn how to block another person's account.</a></p> <p data-bbox="499 1382 1688 1409"><a href="https://support.google.com/maps/answer/7326816?co=GENIE.Platform%3DAndroid&amp;oco=1">https://support.google.com/maps/answer/7326816?co=GENIE.Platform%3DAndroid&amp;oco=1</a></p>

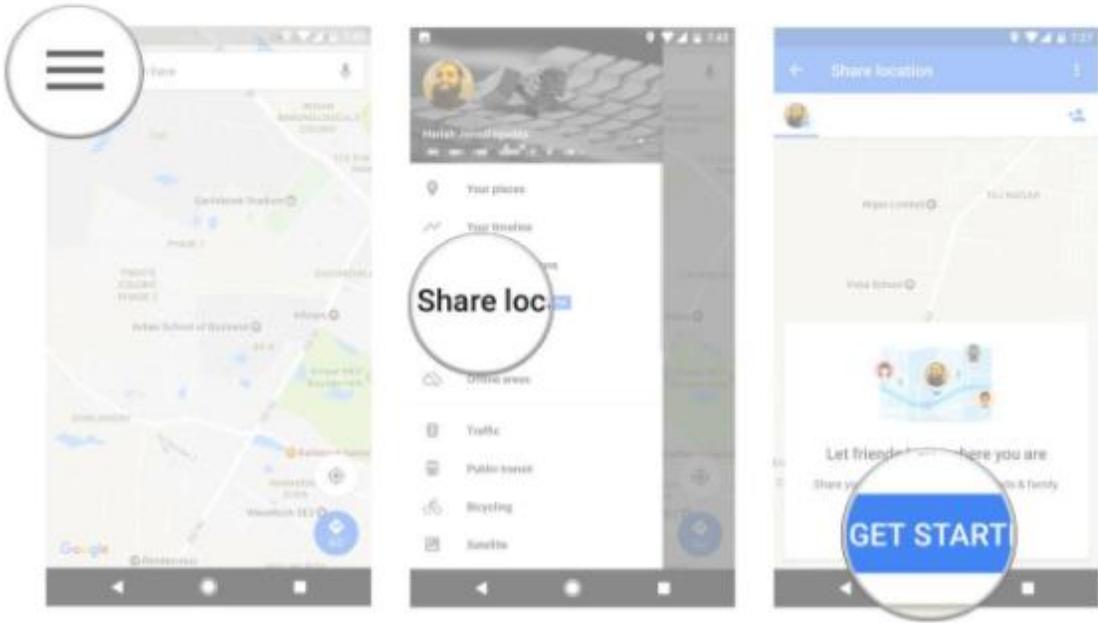
**Exhibit C for U.S. Patent No. 9,445,251 Against HTC Accused Products**

US9445251B2	HTC
	<p data-bbox="506 277 1140 310"><b>How to share your location in Google Maps</b></p> <ol data-bbox="506 337 1121 427" style="list-style-type: none"> <li>1. Open <b>Google Maps</b> from the app drawer or the home screen.</li> <li>2. Tap the <b>hamburger menu</b> (the three horizontal lines) on the top left corner of the screen.</li> <li>3. Select <b>Share location</b>.</li> </ol>  <ol data-bbox="506 865 1152 971" style="list-style-type: none"> <li>4. Tap <b>Get Started</b>.</li> <li>5. Use the <b>+</b> icon to select a time period or select the <b>Until you turn this off</b> setting to share your location indefinitely.</li> <li>6. Tap <b>Select People</b>.</li> </ol>  <p data-bbox="499 1377 1346 1409"><a href="https://www.androidcentral.com/how-share-location-google-maps">https://www.androidcentral.com/how-share-location-google-maps</a></p>

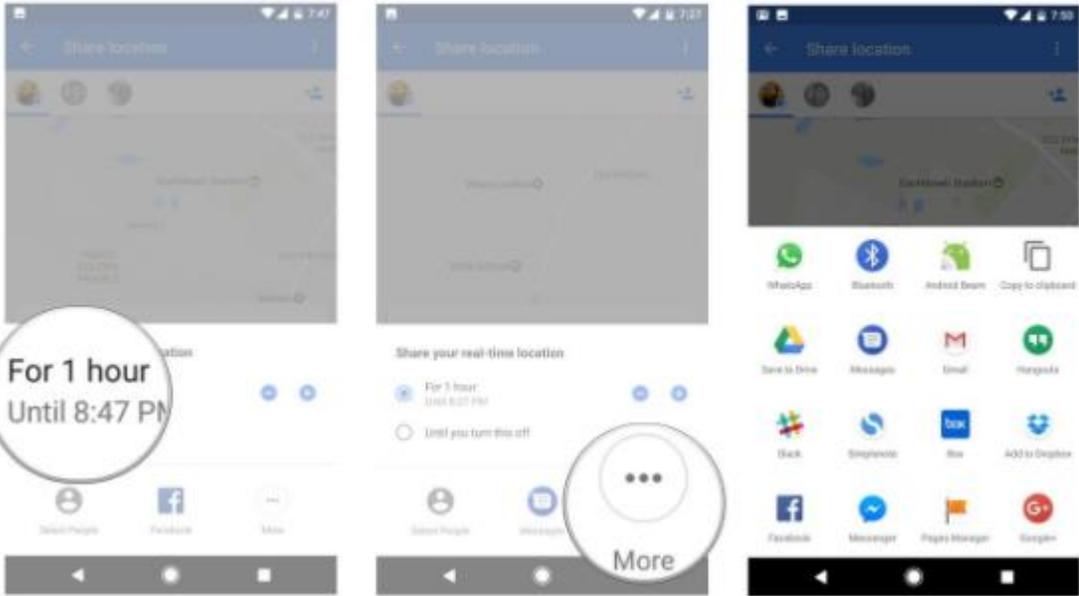
**Exhibit C for U.S. Patent No. 9,445,251 Against HTC Accused Products**

US9445251B2	HTC
	<p>7. You'll see a list of your frequent contacts at the top, along with a full list of contacts. Pick the contacts by tapping their name.</p> <p>8. Once you've selected the contacts you want to share your location to, tap <b>Share</b>.</p> <p>9. You'll see a message saying that the selected contact can view your location.</p>  <p><a href="https://www.androidcentral.com/how-share-location-google-maps">https://www.androidcentral.com/how-share-location-google-maps</a></p>

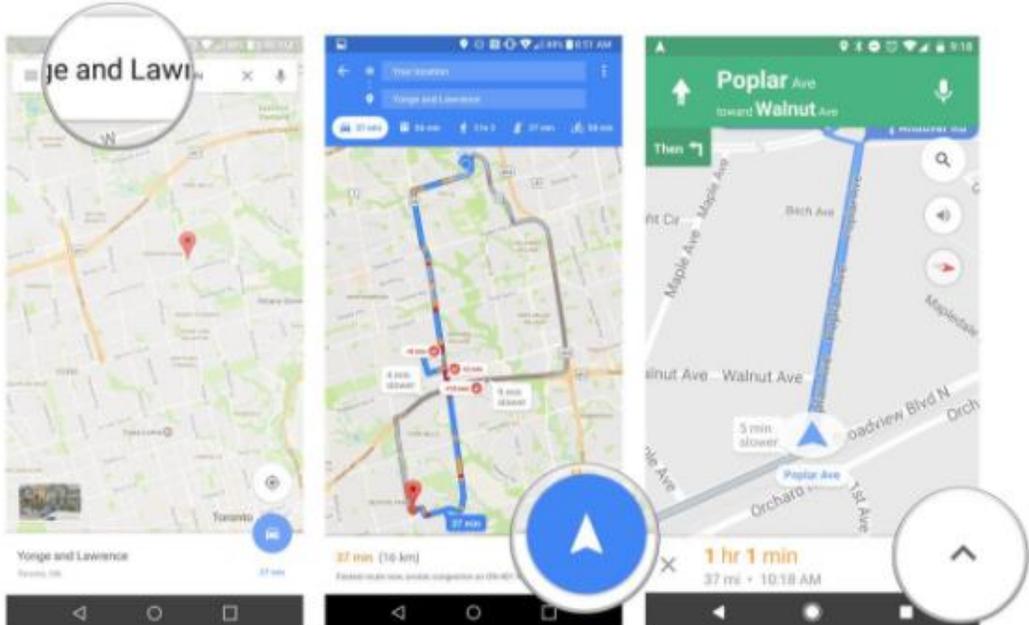
## Exhibit C for U.S. Patent No. 9,445,251 Against HTC Accused Products

US9445251B2	HTC
	<h3 data-bbox="512 245 1243 289">How to create a shareable link</h3> <p data-bbox="512 334 1451 362">You can also create a link and use it to share your location easily. Here's how to do it:</p> <ol data-bbox="506 412 1224 553" style="list-style-type: none"><li>1. Tap the <b>hamburger menu</b> on the top left corner of the screen.</li><li>2. Select <b>Share location</b>.</li><li>3. Tap <b>Get Started</b>.</li></ol>  <p data-bbox="499 1232 1346 1263"><a href="https://www.androidcentral.com/how-share-location-google-maps">https://www.androidcentral.com/how-share-location-google-maps</a></p>

### Exhibit C for U.S. Patent No. 9,445,251 Against HTC Accused Products

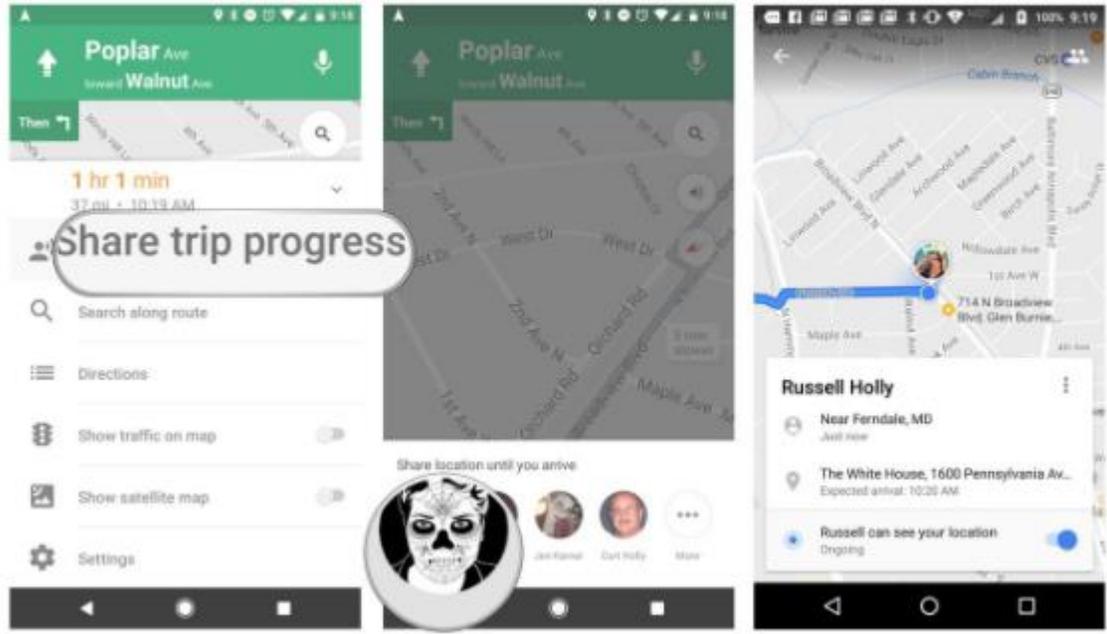
US9445251B2	HTC
	<p data-bbox="512 245 1199 272">4. Select the amount of time you want to share your location.</p> <p data-bbox="512 302 667 329">5. Tap More.</p> <p data-bbox="512 358 1619 423">6. Select your app of choice to create and send a unique URL that broadcasts your current location. You can email it, send the link via Messenger, or even tweet it to the intended recipient.</p>  <p data-bbox="499 1084 1346 1117"><a href="https://www.androidcentral.com/how-share-location-google-maps">https://www.androidcentral.com/how-share-location-google-maps</a></p>

**Exhibit C for U.S. Patent No. 9,445,251 Against HTC Accused Products**

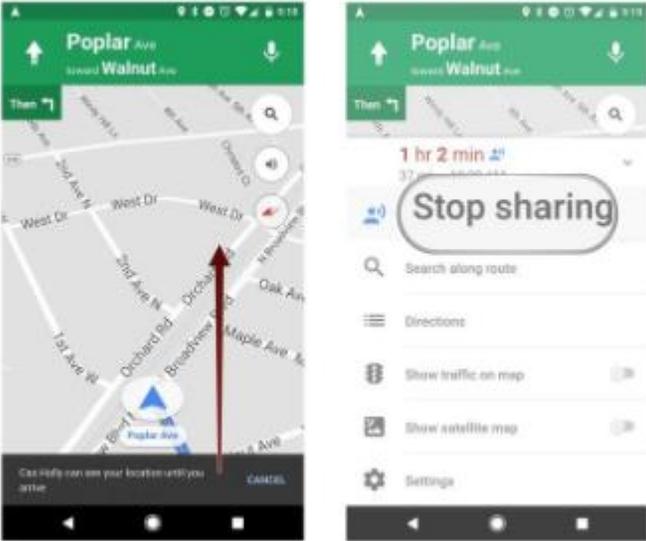
US9445251B2	HTC
	<p data-bbox="514 240 1417 337"><b>How to share your navigation directions while you walk, drive or transit</b></p> <p data-bbox="514 375 1543 467">One of the best ways to share your location with someone is during a drive, walk or transit. If you're meeting a friend or family member somewhere, or navigating towards their home, Google Maps lets you share your location with them for the duration of the trip. It's magic!</p> <ol data-bbox="514 511 1386 646" style="list-style-type: none"><li>1. In the <b>search bar</b> enter your destination.</li><li>2. Pick your navigation type (drive, transit, walk) and press the blue <b>navigate button</b>.</li><li>3. Tap the arrow next to the <b>time-to-destination</b> number at the bottom of the screen.</li></ol>  <p data-bbox="499 1328 1344 1360"><a href="https://www.androidcentral.com/how-share-location-google-maps">https://www.androidcentral.com/how-share-location-google-maps</a></p>



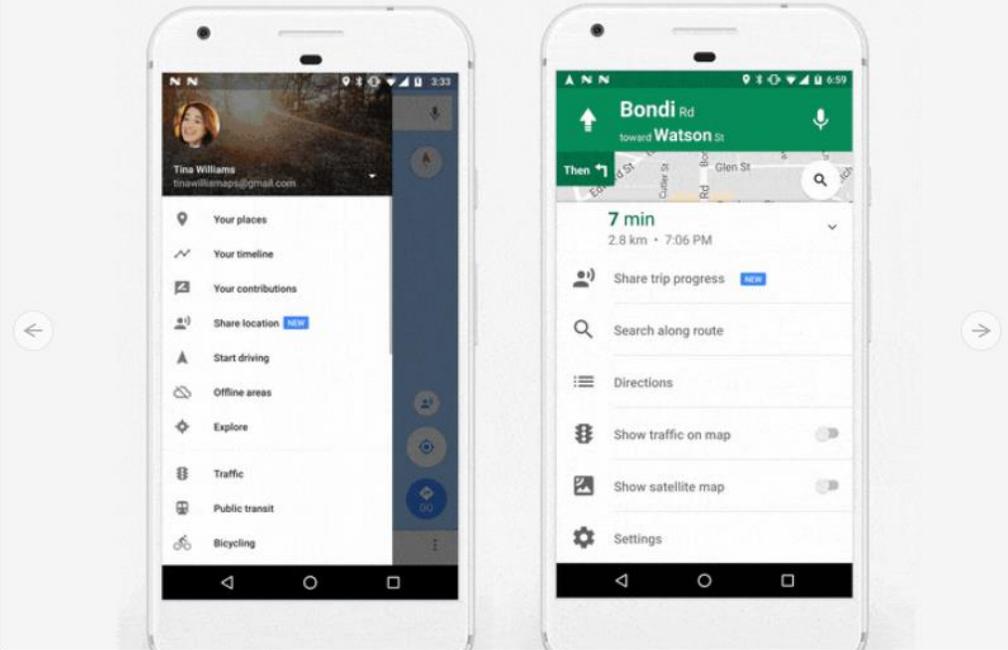
**Exhibit C for U.S. Patent No. 9,445,251 Against HTC Accused Products**

US9445251B2	HTC
	<p data-bbox="514 277 823 305">4. Tap Share trip progress.</p> <p data-bbox="514 334 1136 362">5. Choose one or more contacts to share trip progress.</p>  <p data-bbox="520 1063 1329 1091">You can also stop sharing your location with someone before a trip ends.</p> <p data-bbox="499 1101 1346 1133"><a href="https://www.androidcentral.com/how-share-location-google-maps">https://www.androidcentral.com/how-share-location-google-maps</a></p>

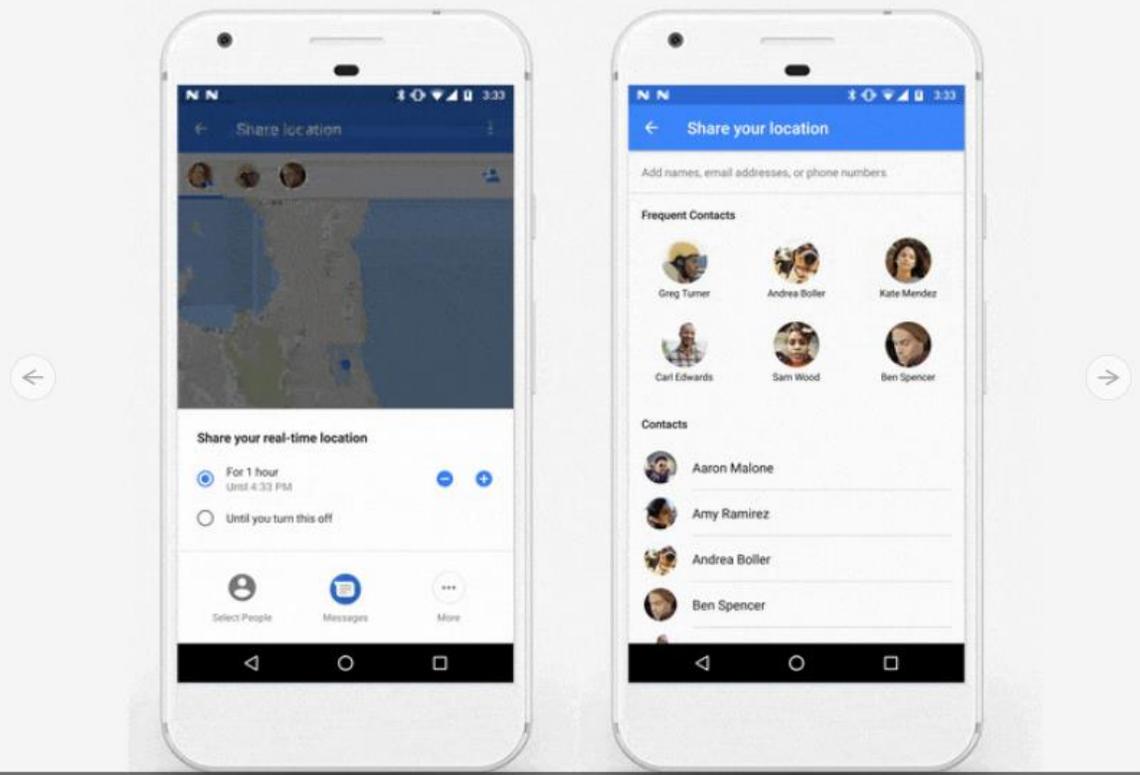
## Exhibit C for U.S. Patent No. 9,445,251 Against HTC Accused Products

US9445251B2	HTC
	<ol style="list-style-type: none"><li data-bbox="520 245 1457 272">1. Tap the arrow next to the <b>time-to-destination</b> number at the bottom of the screen.</li><li data-bbox="520 302 758 329">2. Tap <b>Stop sharing</b>.</li></ol> <div data-bbox="737 383 1383 924"></div> <p data-bbox="531 979 625 1003">That's it!</p> <p data-bbox="531 1047 1598 1071">Are you excited that location sharing is back in Google Maps? How often do you use the feature?</p> <p data-bbox="499 1089 1346 1117"><a href="https://www.androidcentral.com/how-share-location-google-maps">https://www.androidcentral.com/how-share-location-google-maps</a></p>

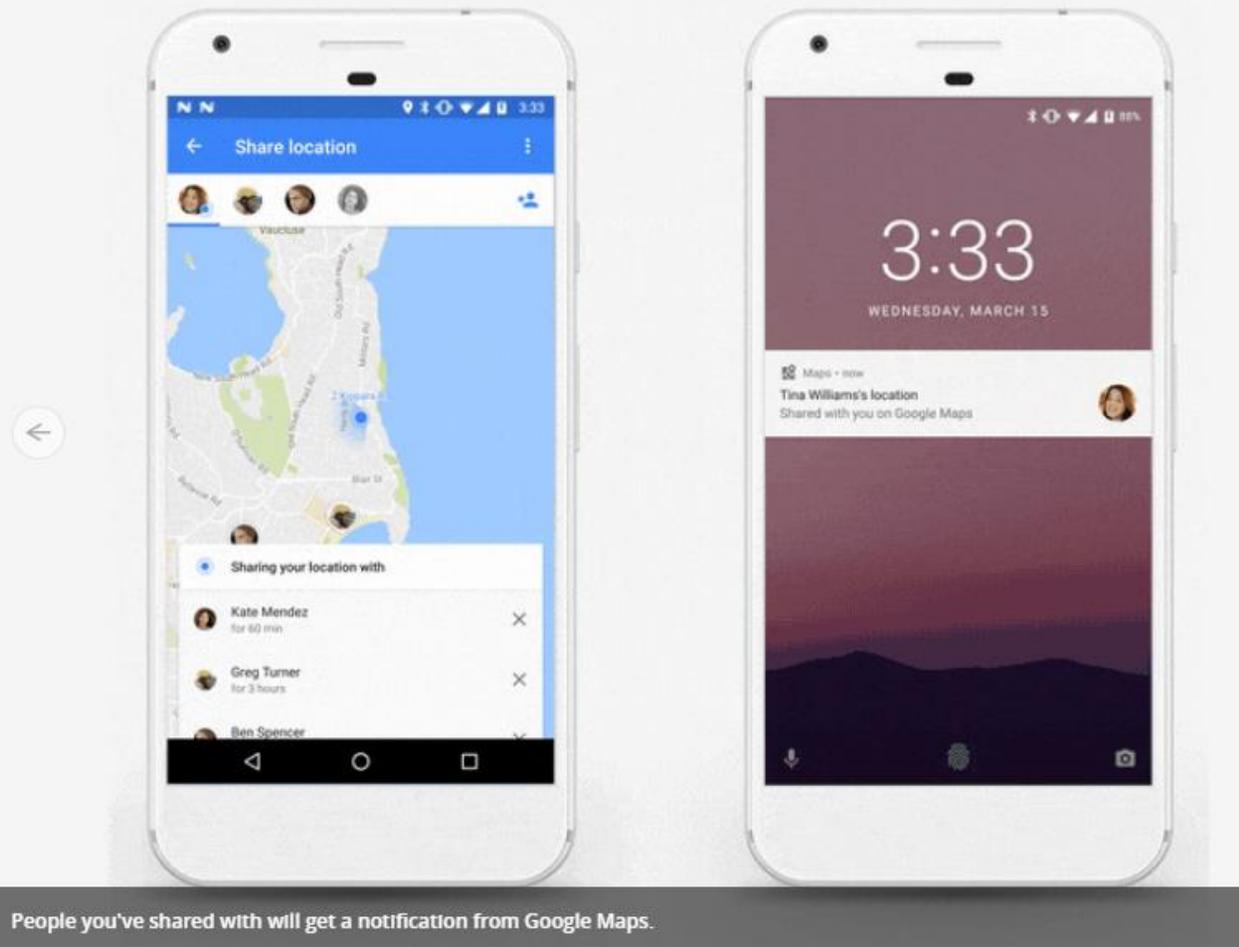
# Exhibit C for U.S. Patent No. 9,445,251 Against HTC Accused Products

US9445251B2	HTC
	 <p data-bbox="504 893 1512 950">Ways to start location sharing: (left) from the app navigation drawer, and (right) from the menu, you can drag up while navigating somewhere.</p> <p data-bbox="504 958 1648 990"><a href="https://arstechnica.com/gadgets/2017/03/location-sharing-finally-returns-to-google-maps/">https://arstechnica.com/gadgets/2017/03/location-sharing-finally-returns-to-google-maps/</a></p>

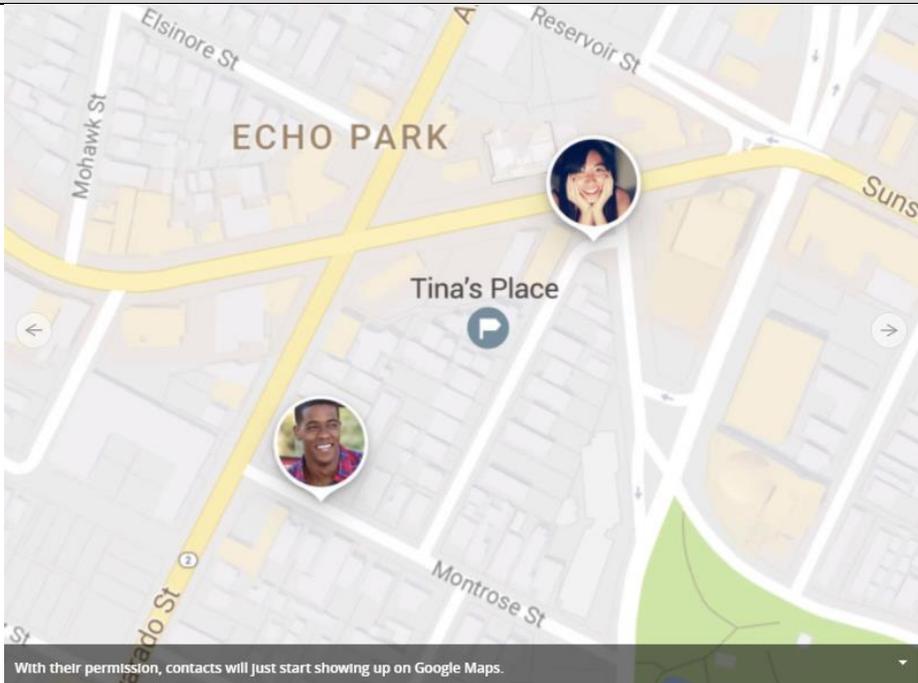
**Exhibit C for U.S. Patent No. 9,445,251 Against HTC Accused Products**

US9445251B2	HTC
	 <p data-bbox="506 1027 1646 1060">You can pick how long you want to share your location for, and with whom you want to share it.</p> <p data-bbox="506 1068 1646 1092"><a href="https://arstechnica.com/gadgets/2017/03/location-sharing-finally-returns-to-google-maps/">https://arstechnica.com/gadgets/2017/03/location-sharing-finally-returns-to-google-maps/</a></p>

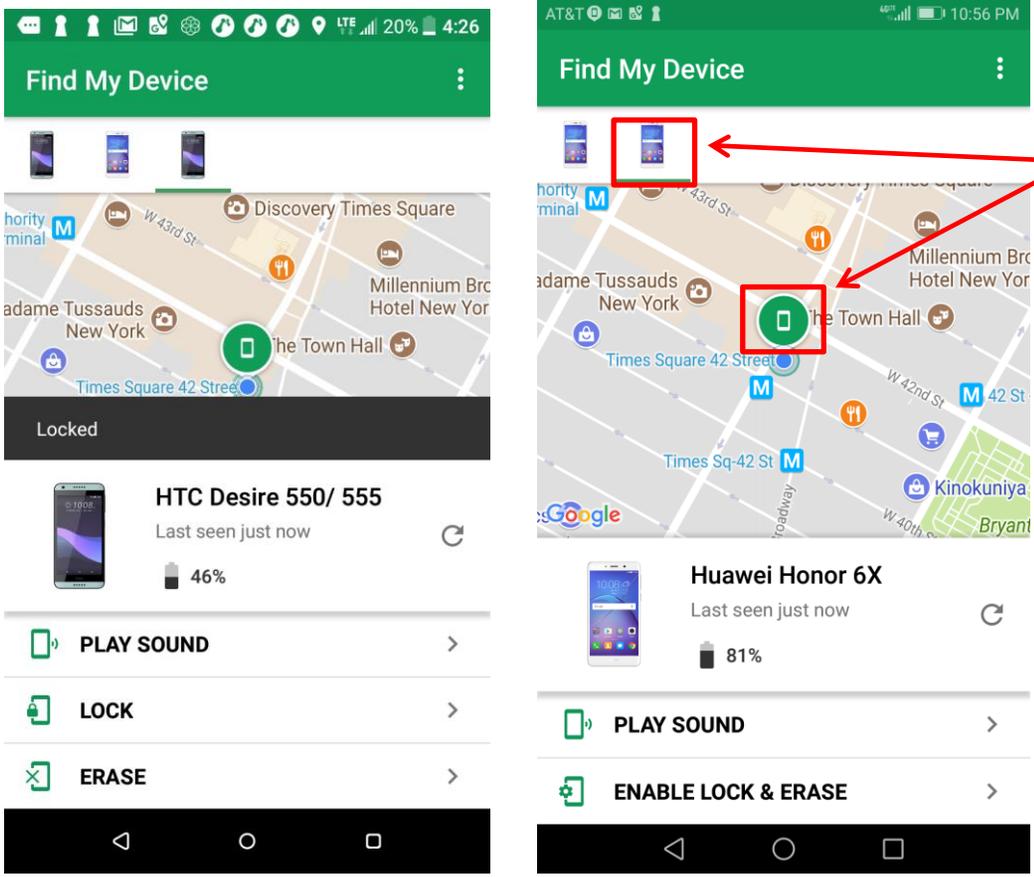
**Exhibit C for U.S. Patent No. 9,445,251 Against HTC Accused Products**

US9445251B2	HTC
	 <p data-bbox="506 1146 1161 1174">People you've shared with will get a notification from Google Maps.</p> <p data-bbox="499 1192 1646 1222"><a href="https://arstechnica.com/gadgets/2017/03/location-sharing-finally-returns-to-google-maps/">https://arstechnica.com/gadgets/2017/03/location-sharing-finally-returns-to-google-maps/</a></p>

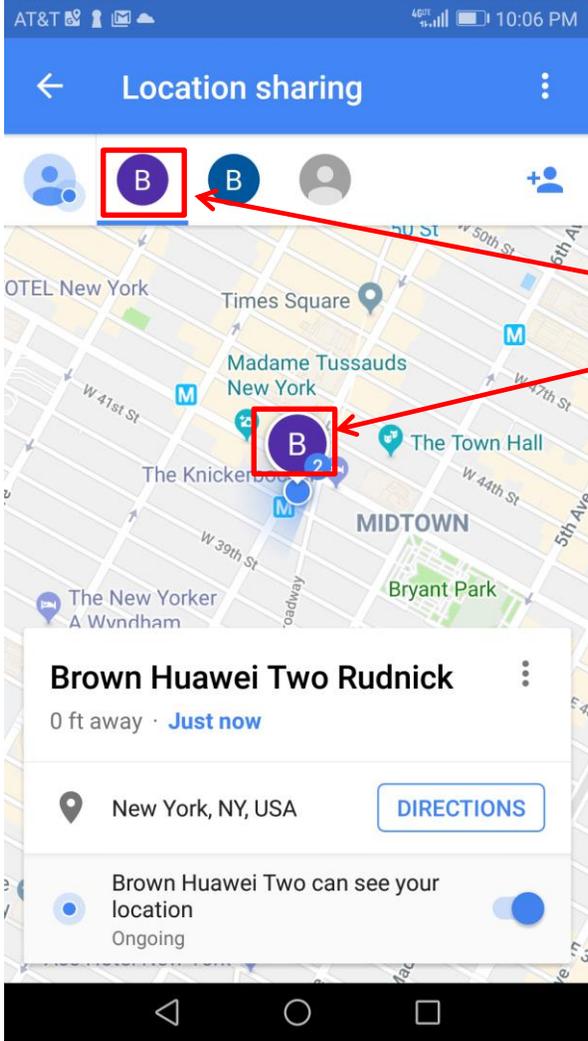
**Exhibit C for U.S. Patent No. 9,445,251 Against HTC Accused Products**

US9445251B2	HTC
	 <p data-bbox="499 885 1417 911">With their permission, contacts will just start showing up on Google Maps.</p> <p data-bbox="499 917 1648 950"><a href="https://arstechnica.com/gadgets/2017/03/location-sharing-finally-returns-to-google-maps/">https://arstechnica.com/gadgets/2017/03/location-sharing-finally-returns-to-google-maps/</a></p> <p data-bbox="499 1023 1050 1055"><b><u>Exemplary Find My Device Screenshots:</u></b></p>

### Exhibit C for U.S. Patent No. 9,445,251 Against HTC Accused Products

US9445251B2	HTC
	 <p><b>Exemplary User Selectable Symbols</b></p> <p><u>Exemplary Google Maps Screenshots:</u></p>

**Exhibit C for U.S. Patent No. 9,445,251 Against HTC Accused Products**

US9445251B2	HTC
	 <p data-bbox="1310 472 1633 641">Exemplary User Selectable Symbols</p> <p data-bbox="499 862 1066 1198"><b>Brown Huawei Two Rudnick</b> 0 ft away · <b>Just now</b> New York, NY, USA <a href="#">DIRECTIONS</a> Brown Huawei Two can see your location Ongoing</p> <p data-bbox="499 1349 842 1382"><b><u>Exemplary Source Code:</u></b></p>



**Exhibit C for U.S. Patent No. 9,445,251 Against HTC Accused Products**

US9445251B2	HTC
	<p>The above functionality is performed at least in part by the following publicly available source code and/or source code that invokes or is invoked by the following source code (or a substantially similar copy compiled and loaded onto the Accused Products by HTC). AGIS reserves the right to supplement these contentions with additional source code as discovery progresses and as additional source code is made available.</p> <pre data-bbox="520 431 1726 472">public static LocationRequest create ()</pre> <p>Create a location request with default parameters.</p> <p>Default parameters are for a block accuracy, slowly updated location. It can then be adjusted as required by the applications before passing to the <code>FusedLocationProviderApi</code>.</p> <p><b>Returns</b></p> <ul data-bbox="541 686 800 711" style="list-style-type: none"><li>• a new location request</li></ul> <p><a href="https://developers.google.com/android/reference/com/google/android/gms/location/LocationRequest">https://developers.google.com/android/reference/com/google/android/gms/location/LocationRequest</a></p>

**Exhibit C for U.S. Patent No. 9,445,251 Against HTC Accused Products**

US9445251B2	HTC
	<p><b>public static final int PRIORITY_BALANCED_POWER_ACCURACY</b></p> <p>Used with <code>setPriority(int)</code> to request "block" level accuracy.</p> <p>Block level accuracy is considered to be about 100 meter accuracy. Using a coarse accuracy such as this often consumes less power.</p> <p>Constant Value: 102</p> <hr/> <p><b>public static final int PRIORITY_HIGH_ACCURACY</b></p> <p>Used with <code>setPriority(int)</code> to request the most accurate locations available.</p> <p>This will return the finest location available.</p> <p>Constant Value: 100</p> <hr/> <p><b>public static final int PRIORITY_LOW_POWER</b></p> <p>Used with <code>setPriority(int)</code> to request "city" level accuracy.</p> <p>City level accuracy is considered to be about 10km accuracy. Using a coarse accuracy such as this often consumes less power.</p> <p>Constant Value: 104</p> <p><a href="https://developers.google.com/android/reference/com/google/android/gms/location/LocationRequest">https://developers.google.com/android/reference/com/google/android/gms/location/LocationRequest</a></p>

**Exhibit C for U.S. Patent No. 9,445,251 Against HTC Accused Products**

US9445251B2	HTC
	<p data-bbox="520 250 1738 289"><code>public Task&lt;Location&gt; getLastLocation ()</code></p> <p data-bbox="520 315 1092 341">Returns the best most recent location currently available.</p> <p data-bbox="520 371 1682 431">If a location is not available, which should happen very rarely, null will be returned. The best accuracy available while respecting the location permissions will be returned.</p> <p data-bbox="520 462 1724 522">This method provides a simplified way to get location. It is particularly well suited for applications that do not require an accurate location and that do not want to maintain extra logic for location updates.</p> <p data-bbox="520 581 1738 620"><code>public Task&lt;LocationAvailability&gt; getLocationAvailability ()</code></p> <p data-bbox="520 646 1682 706">Returns the availability of location data. When <code>isLocationAvailable()</code> returns true, then the location returned by <code>getLastLocation()</code> will be reasonably up to date within the hints specified by the active <code>LocationRequest</code> s.</p> <p data-bbox="520 737 1459 763">If the client isn't connected to Google Play services and the request times out, null is returned.</p> <p data-bbox="520 794 1661 854">Note it's always possible for <code>getLastLocation()</code> to return null even when this method returns true (e.g. location settings were disabled between calls).</p> <p data-bbox="520 868 1906 928"><a href="https://developers.google.com/android/reference/com/google/android/gms/location/FusedLocationProviderClient">https://developers.google.com/android/reference/com/google/android/gms/location/FusedLocationProviderClient</a></p>

**Exhibit C for U.S. Patent No. 9,445,251 Against HTC Accused Products**

US9445251B2	HTC						
	<p data-bbox="514 245 1734 326"> <code>public Task&lt;Void&gt; requestLocationUpdates (LocationRequest request, LocationCallback callback, Looper looper)</code> </p> <p data-bbox="514 354 1260 378">Requests location updates with a callback on the specified Looper thread.</p> <p data-bbox="514 412 1671 472">This method is suited for the foreground use cases. For background use cases, the <code>PendingIntent</code> version of the method is recommended, see <code>requestLocationUpdates(LocationRequest, PendingIntent)</code>.</p> <p data-bbox="514 505 1356 529">Any previous LocationRequests registered on this LocationListener will be replaced.</p> <p data-bbox="514 561 1671 654">This call will keep the Google Play services connection active, so make sure to call <code>removeLocationUpdates(LocationCallback)</code> when you no longer need it, otherwise you lose the benefits of the automatic connection management.</p> <p data-bbox="514 686 1728 711">Callbacks for <code>LocationCallback</code> will be made on the specified thread, which must already be a prepared looper thread.</p> <p data-bbox="531 737 653 761"><b>Parameters</b></p> <table border="1" data-bbox="514 792 1734 1008"> <tbody> <tr> <td data-bbox="514 792 617 857"><b>request</b></td> <td data-bbox="617 792 1734 857">The location request for the updates.</td> </tr> <tr> <td data-bbox="514 857 617 922"><b>callback</b></td> <td data-bbox="617 857 1734 922">The callback for the location updates.</td> </tr> <tr> <td data-bbox="514 922 617 1008"><b>looper</b></td> <td data-bbox="617 922 1734 1008">The Looper object whose message queue will be used to implement the callback mechanism, or null to make callbacks on the calling thread.</td> </tr> </tbody> </table> <p data-bbox="499 1024 1906 1089"> <a href="https://developers.google.com/android/reference/com/google/android/gms/location/FusedLocationProviderClient">https://developers.google.com/android/reference/com/google/android/gms/location/FusedLocationProviderClient</a> </p>	<b>request</b>	The location request for the updates.	<b>callback</b>	The callback for the location updates.	<b>looper</b>	The Looper object whose message queue will be used to implement the callback mechanism, or null to make callbacks on the calling thread.
<b>request</b>	The location request for the updates.						
<b>callback</b>	The callback for the location updates.						
<b>looper</b>	The Looper object whose message queue will be used to implement the callback mechanism, or null to make callbacks on the calling thread.						

**Exhibit C for U.S. Patent No. 9,445,251 Against HTC Accused Products**

US9445251B2	HTC				
	<pre data-bbox="516 245 1730 326">public Task&lt;Void&gt; requestLocationUpdates (LocationRequest request, PendingIntent callbackIntent)</pre> <p data-bbox="516 354 1255 378">Requests location updates with a callback on the specified PendingIntent.</p> <p data-bbox="516 412 1719 540">This method is suited for the background use cases, more specifically for receiving location updates, even when the app has been killed by the system. In order to do so, use a <code>PendingIntent</code> for a started service. For foreground use cases, the <code>LocationCallback</code> version of the method is recommended, see <code>requestLocationUpdates(LocationRequest, LocationCallback, Looper)</code>.</p> <p data-bbox="516 573 1713 630">Any previously registered requests that have the same PendingIntent (as defined by <code>equals(Object)</code>) will be replaced by this request.</p> <p data-bbox="516 662 1713 756">Both <code>LocationResult</code> and <code>LocationAvailability</code> are sent to the given PendingIntent. You can extract data from an Intent using <code>hasResult(Intent)</code>, <code>extractResult(Intent)</code>, <code>hasLocationAvailability(Intent)</code>, and <code>extractLocationAvailability(Intent)</code>.</p> <p data-bbox="533 781 655 805"><b>Parameters</b></p> <table border="1" data-bbox="516 834 1730 971"> <tbody> <tr> <td data-bbox="516 834 825 902"><code>request</code></td> <td data-bbox="825 834 1730 902">The location request for the updates.</td> </tr> <tr> <td data-bbox="516 902 825 971"><code>callbackIntent</code></td> <td data-bbox="825 902 1730 971">A pending intent to be sent for each location update.</td> </tr> </tbody> </table> <p data-bbox="533 997 617 1021"><b>Returns</b></p> <ul data-bbox="541 1045 1346 1070" style="list-style-type: none"> <li>• a Task for the call, check <code>isSuccessful()</code> to determine if it was successful.</li> </ul> <p data-bbox="499 1081 1906 1146"><a href="https://developers.google.com/android/reference/com/google/android/gms/location/FusedLocationProviderClient">https://developers.google.com/android/reference/com/google/android/gms/location/FusedLocationProviderClient</a></p>	<code>request</code>	The location request for the updates.	<code>callbackIntent</code>	A pending intent to be sent for each location update.
<code>request</code>	The location request for the updates.				
<code>callbackIntent</code>	A pending intent to be sent for each location update.				

**Exhibit C for U.S. Patent No. 9,445,251 Against HTC Accused Products**

US9445251B2	HTC						
	<p data-bbox="520 248 1728 280">public void <b>onLocationAvailability</b> (<a href="#">LocationAvailability</a> locationAvailability)</p> <p data-bbox="520 313 1157 337">Called when there is a change in the availability of location data.</p> <p data-bbox="520 370 1728 565">When <a href="#">isLocationAvailable()</a> returns <code>false</code> you can assume that location will not be returned in <a href="#">onLocationResult(LocationResult)</a> until something changes in the device's settings or environment. Even when <a href="#">isLocationAvailable()</a> returns <code>true</code> the <a href="#">onLocationResult(LocationResult)</a> may not always be called regularly, however the device location is known and both the most recently delivered location and <a href="#">getLastLocation(GoogleApiClient)</a> will be reasonably up to date given the hints specified by the active <a href="#">LocationRequest</a> s.</p> <p data-bbox="537 589 657 613"><b>Parameters</b></p> <table border="1" data-bbox="520 646 1728 711"> <tr> <td data-bbox="520 646 951 711"><b>locationAvailability</b></td> <td data-bbox="951 646 1728 711">The current status of location availability.</td> </tr> </table> <p data-bbox="520 760 1728 792">public void <b>onLocationResult</b> (<a href="#">LocationResult</a> result)</p> <p data-bbox="520 824 1041 849">Called when device location information is available.</p> <p data-bbox="520 881 1650 946">The most recent location returned by <a href="#">getLastLocation()</a> is not guaranteed to be immediately fresh, but will be reasonably up to date given the hints specified by the active <a href="#">LocationRequest</a> s.</p> <p data-bbox="537 971 657 995"><b>Parameters</b></p> <table border="1" data-bbox="520 1027 1728 1092"> <tr> <td data-bbox="520 1027 758 1092"><b>result</b></td> <td data-bbox="758 1027 1728 1092">The latest location result available.</td> </tr> </table> <p data-bbox="506 1109 1797 1141"><a href="https://developers.google.com/android/reference/com/google/android/gms/location/LocationCallback">https://developers.google.com/android/reference/com/google/android/gms/location/LocationCallback</a></p> <p data-bbox="520 1157 1728 1190">public abstract void <b>onLocationChanged</b> (<a href="#">Location</a> location)</p> <p data-bbox="520 1222 905 1247">Called when the location has changed.</p> <p data-bbox="537 1271 657 1295"><b>Parameters</b></p> <table border="1" data-bbox="520 1328 1728 1393"> <tr> <td data-bbox="520 1328 915 1393"><b>location</b></td> <td data-bbox="915 1328 1728 1393">The updated location.</td> </tr> </table>	<b>locationAvailability</b>	The current status of location availability.	<b>result</b>	The latest location result available.	<b>location</b>	The updated location.
<b>locationAvailability</b>	The current status of location availability.						
<b>result</b>	The latest location result available.						
<b>location</b>	The updated location.						

**Exhibit C for U.S. Patent No. 9,445,251 Against HTC Accused Products**

US9445251B2	HTC
	<p data-bbox="499 237 1787 266"><a href="https://developers.google.com/android/reference/com/google/android/gms/location/LocationListener">https://developers.google.com/android/reference/com/google/android/gms/location/LocationListener</a></p> <p data-bbox="512 321 800 350">Public Constructors</p> <hr data-bbox="512 363 1730 367"/> <p data-bbox="512 418 911 448">public <b>MapView</b> (<a href="#">Context</a> context)</p> <p data-bbox="512 509 1115 539">public <b>MapView</b> (<a href="#">Context</a> context, <a href="#">AttributeSet</a> attrs)</p> <p data-bbox="512 600 1255 630">public <b>MapView</b> (<a href="#">Context</a> context, <a href="#">AttributeSet</a> attrs, int defStyle)</p> <p data-bbox="512 691 1234 721">public <b>MapView</b> (<a href="#">Context</a> context, <a href="#">GoogleMapOptions</a> options)</p> <p data-bbox="499 750 1661 779"><a href="https://developers.google.com/android/reference/com/google/android/gms/maps/MapView">https://developers.google.com/android/reference/com/google/android/gms/maps/MapView</a></p>

**Exhibit C for U.S. Patent No. 9,445,251 Against HTC Accused Products**

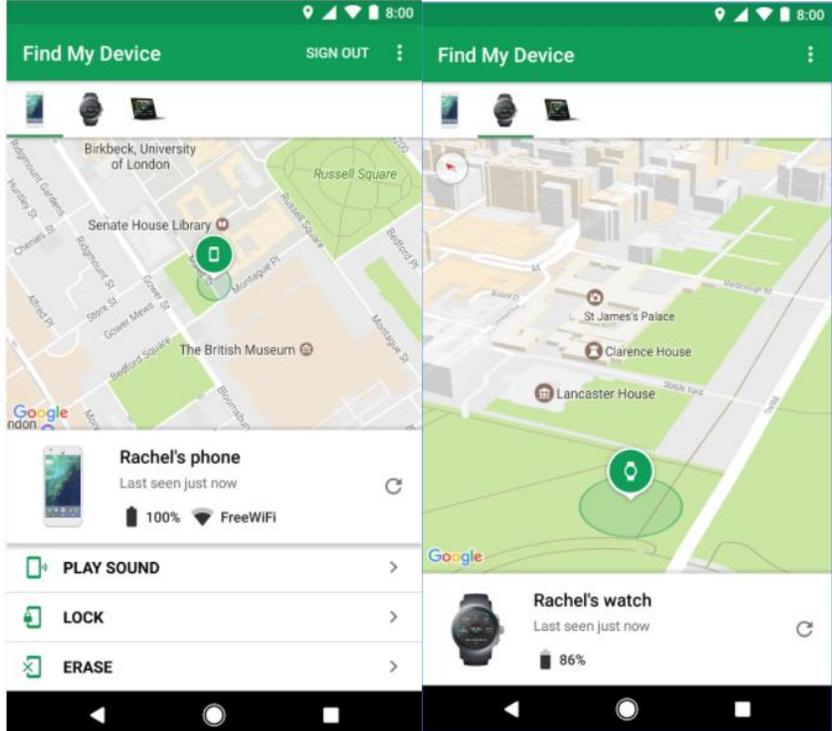
US9445251B2	HTC
	<p><code>public void <b>getMapAsync</b> (<b>OnMapReadyCallback</b> callback)</code></p> <p>Returns a non-null instance of the <code>GoogleMap</code>, ready to be used.</p> <p>Note that:</p> <ul style="list-style-type: none"> <li>• This method must be called from the main thread.</li> <li>• The callback will be executed in the main thread.</li> <li>• In the case where Google Play services is not installed on the user's device, the callback will not be triggered until the user installs it.</li> <li>• The <code>GoogleMap</code> object provided by the callback is non-null.</li> </ul> <p><b>Parameters</b></p> <p><code>callback</code> The callback object that will be triggered when the map is ready to be used.</p> <p><code>public final void <b>onCreate</b> (<b>Bundle</b> savedInstanceState)</code></p> <p>You must call this method from the parent Activity/Fragment's corresponding method.</p> <p><a href="https://developers.google.com/android/reference/com/google/android/gms/maps/MapView">https://developers.google.com/android/reference/com/google/android/gms/maps/MapView</a></p>
<p>[1D] sending, from the first device to the server, a request for a second georeferenced map different from the first georeferenced map, wherein the request specifies a map location;</p>	<p>HTC infringes directly and/or indirectly by performing, inducing others to perform, and/or contributing to the performance of: sending, from the first device to the server, a request for a second georeferenced map different from the first georeferenced map, wherein the request specifies a map location.</p> <p><b><u>Regarding Find My Device</u></b> and Android Device Manager, the Accused Products display, to the user on the display of the first device, a map with one or more symbols corresponding to one or more second devices. The user, via the first device, or the device itself requests different map data from a second server. The request occurs responsive to user input (e.g., zoom, drag, pan, change focus, change map type, symbol or device selection, another device or user selection, refresh or reload request, change in position of first device, change in position of a second device). The request must specify a map location either by map tile, by</p>



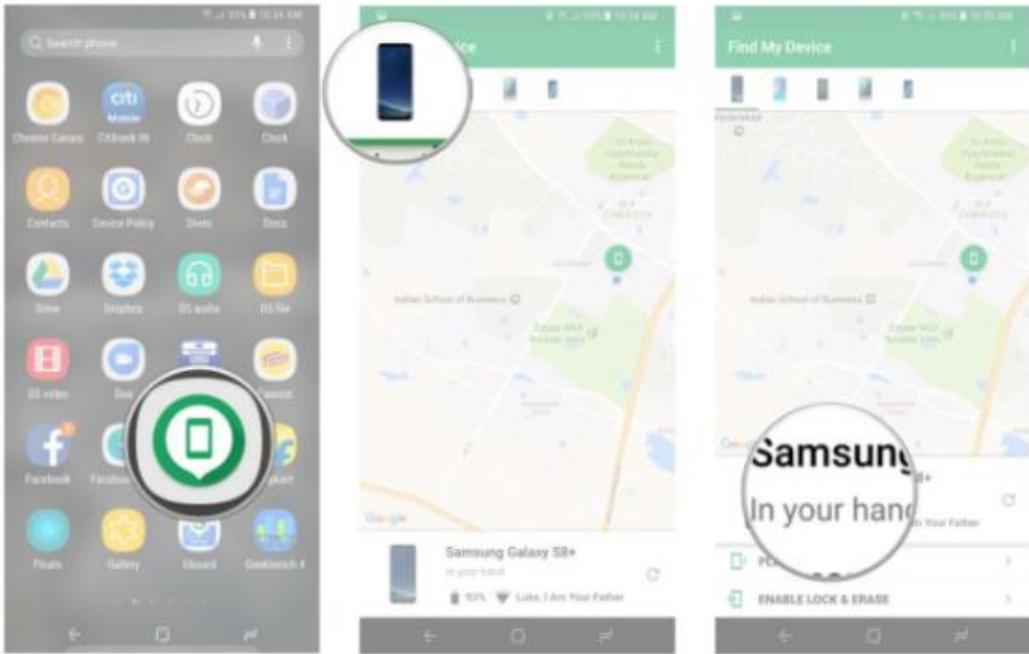
**Exhibit C for U.S. Patent No. 9,445,251 Against HTC Accused Products**

US9445251B2	HTC
	<p>coordinate, or by other identifier correlated to map location so that the proper map information can be relayed to the device. Alternatively, the request occurs responsive to an automatic and/or pre-determined control caused by an instruction from within the first device or from the one or more second devices, e.g. a refresh. The different map data includes an update to the first data or a replacement of the first data. Upon information and belief, the new map data may ultimately originate from one of many sources and servers.</p> <p><b><u>Regarding Google Maps</u></b>, Google Latitude, Google Plus, Google Hangouts, Google Messages, Google Allo, Google Duo, Google Chrome, and Android Messenger, the Accused Products display, to the user on the display of the first device, a map with one or more symbols corresponding to one or more second users (or second devices corresponding to the second users). The user, via the first device, or the device itself requests different map data from a second server. The request occurs responsive to user input (e.g., zoom, drag, pan, change focus, change map type, refresh or reload request, symbol or device selection, another device or user selection, change in position of first device, change in position of a second device). The request must specify a map location either by map tile, by coordinate, or by other identifier correlated to map location so that the proper map information can be relayed to the device. Alternatively, the request occurs responsive to an automatic and/or pre-determined control caused by an instruction from within the first device or from the one or more second devices, e.g. a refresh. The different map data includes an update to the first data or a replacement of the first data. Upon information and belief, the new map data may come from one of many sources and servers.</p> <p><b><u>Exemplary Support for Find My Device:</u></b></p>

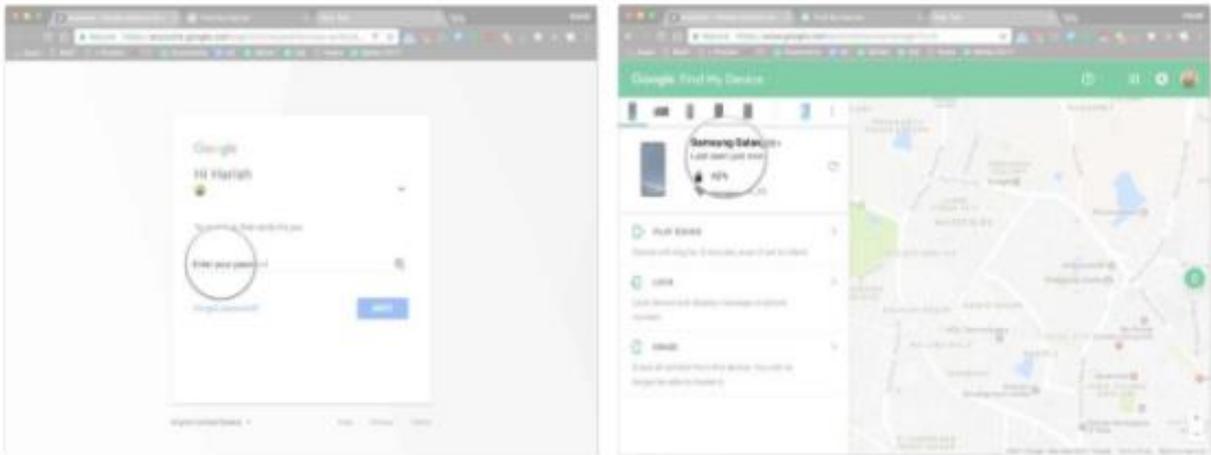
**Exhibit C for U.S. Patent No. 9,445,251 Against HTC Accused Products**

US9445251B2	HTC
	 <p><a href="https://play.google.com/store/apps/details?id=com.google.android.apps.adm&amp;hl=en">https://play.google.com/store/apps/details?id=com.google.android.apps.adm&amp;hl=en</a></p>

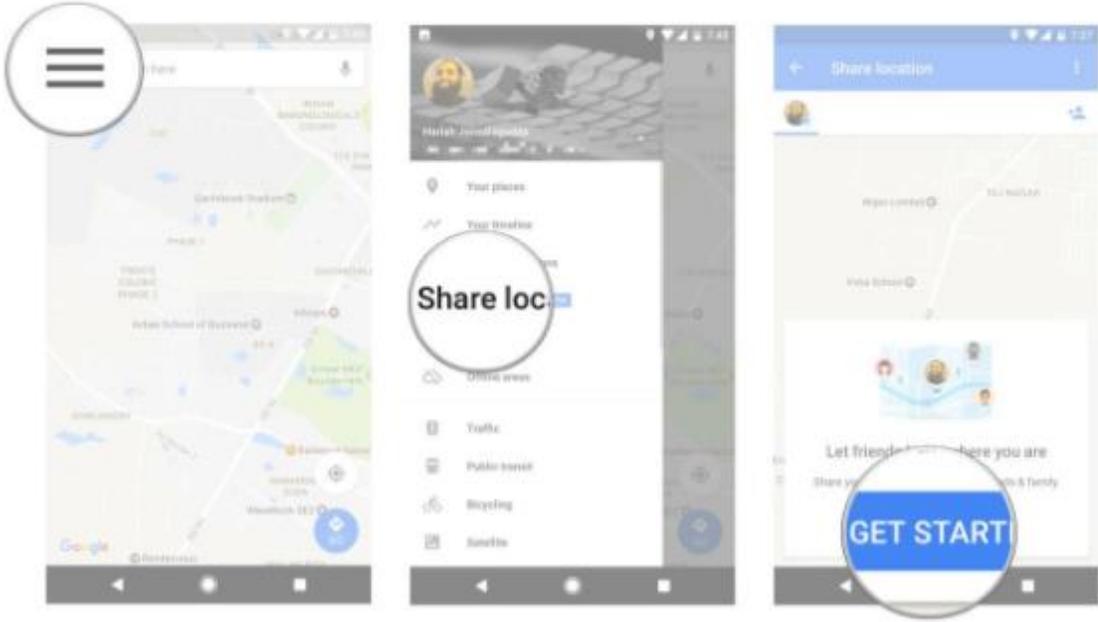
**Exhibit C for U.S. Patent No. 9,445,251 Against HTC Accused Products**

US9445251B2	HTC
	<p data-bbox="514 245 1533 337">Once you're signed in to Find My Device, you'll see a map with your current location as well as the make and model of your phone, and two options — Play Sound, and Enable Lock &amp; Erase. Hitting the latter option will allow you to start using the Lock and Erase functions.</p> <p data-bbox="514 375 1549 431">If you've signed into more than one phone, you can select a particular device by browsing the list at the top of the screen.</p> <ol data-bbox="514 480 1234 613" style="list-style-type: none"> <li>1. Open <b>Find My Device</b> from your home screen or app drawer.</li> <li>2. Select your phone from the <b>list of devices</b> at the top of the screen.</li> <li>3. See if your phone is <b>discoverable</b>.</li> </ol>  <p data-bbox="499 1325 1121 1357"><a href="https://www.androidcentral.com/find-my-device">https://www.androidcentral.com/find-my-device</a></p>

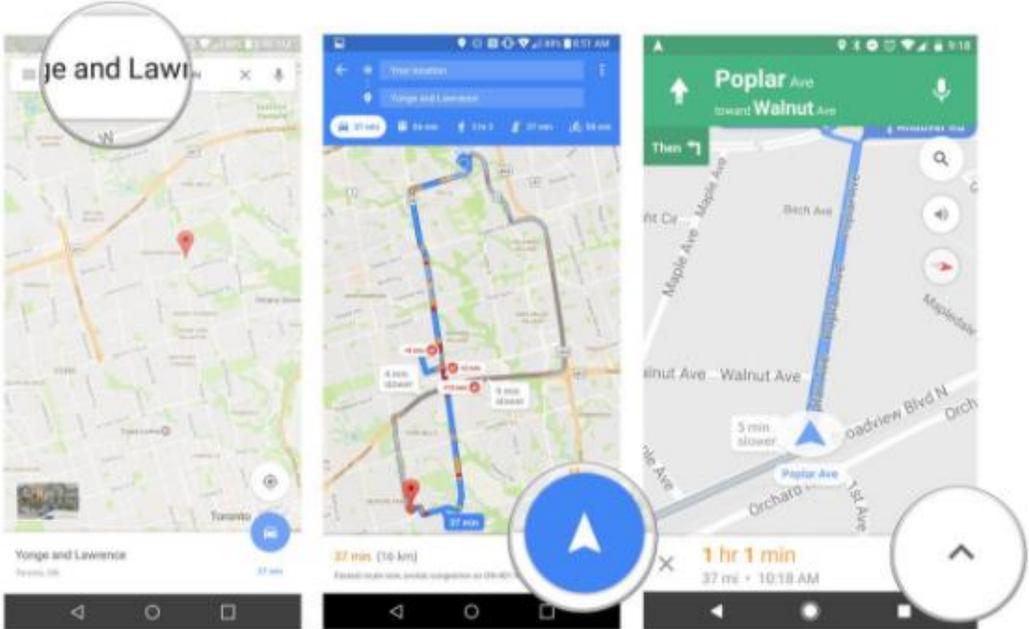
**Exhibit C for U.S. Patent No. 9,445,251 Against HTC Accused Products**

US9445251B2	HTC
	<h2 data-bbox="510 272 1650 329">How to locate your phone over the internet</h2> <p data-bbox="510 370 1659 513">If you've lost your phone, you can remotely locate it through the <a href="#">Find My Device website</a>. You'll need to sign in to the Google account that was used to set up Find My Device. It takes a few seconds, but the service should be able to track your phone. Alternatively, you can also do a Google search for "find my phone" to locate your handset.</p> <ol data-bbox="510 570 997 724" style="list-style-type: none"><li>1. Head to the <a href="#">Find My Device website</a>.</li><li>2. Sign in to your <a href="#">Google account</a>.</li><li>3. Check if your device is visible.</li></ol> <div data-bbox="548 764 1759 1219"></div> <p data-bbox="499 1252 1121 1284"><a href="https://www.androidcentral.com/find-my-device">https://www.androidcentral.com/find-my-device</a></p> <p data-bbox="499 1325 1016 1357"><b><u>Exemplary Support for Google Maps:</u></b></p>

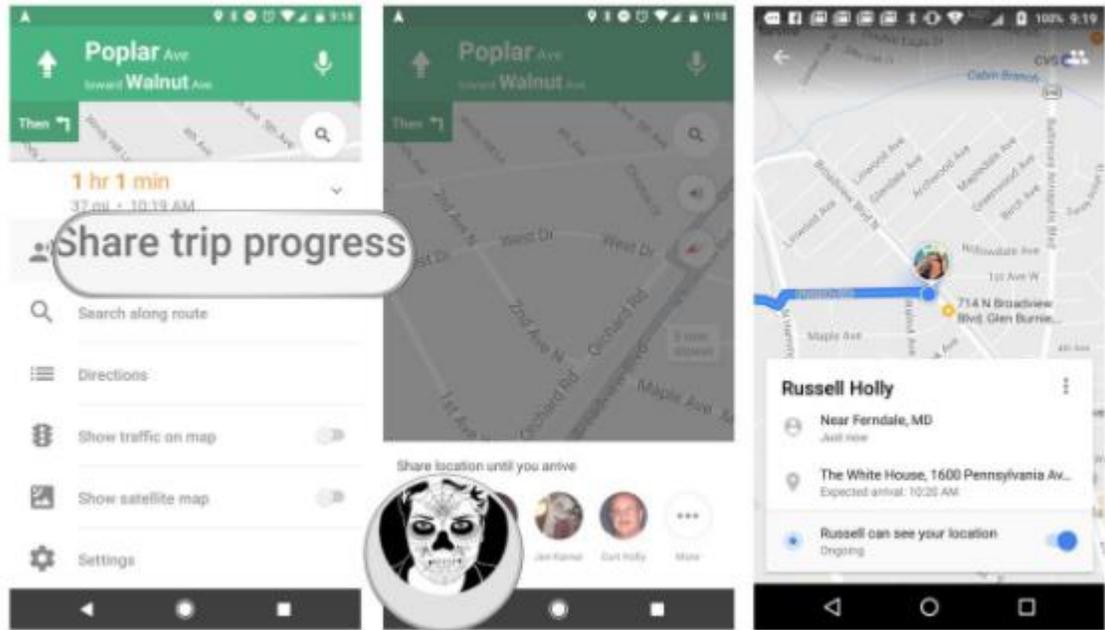
## Exhibit C for U.S. Patent No. 9,445,251 Against HTC Accused Products

US9445251B2	HTC
	<h3 data-bbox="512 321 1243 365">How to create a shareable link</h3> <p data-bbox="512 407 1451 435">You can also create a link and use it to share your location easily. Here's how to do it:</p> <ol data-bbox="506 488 1224 626" style="list-style-type: none"><li data-bbox="506 488 1224 516">1. Tap the <b>hamburger menu</b> on the top left corner of the screen.</li><li data-bbox="506 542 787 570">2. Select <b>Share location</b>.</li><li data-bbox="506 596 724 623">3. Tap <b>Get Started</b>.</li></ol>  <p data-bbox="499 1308 1346 1336"><a href="https://www.androidcentral.com/how-share-location-google-maps">https://www.androidcentral.com/how-share-location-google-maps</a></p>

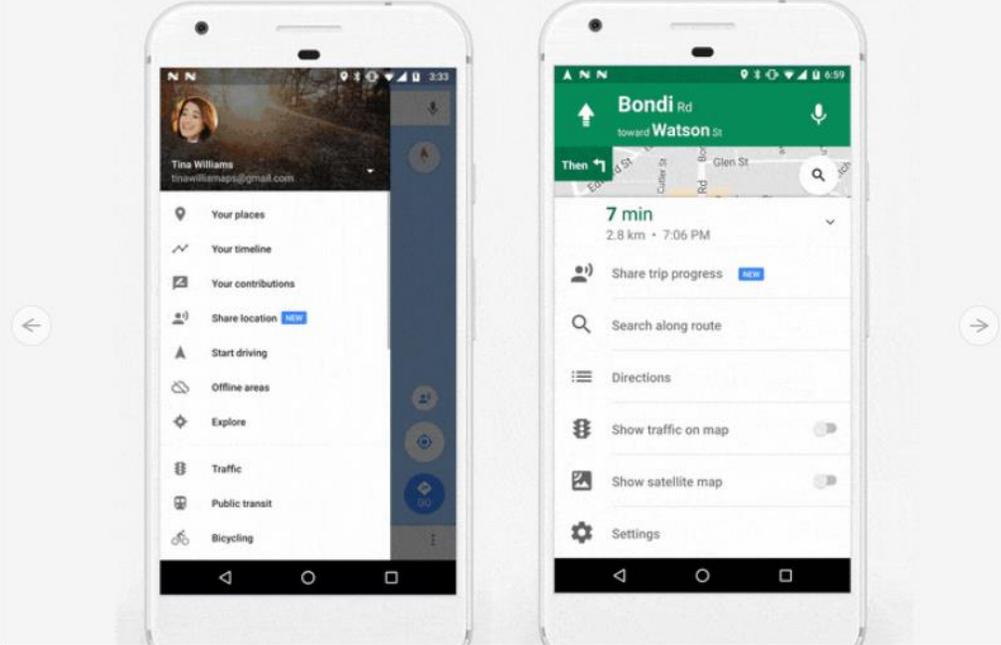
**Exhibit C for U.S. Patent No. 9,445,251 Against HTC Accused Products**

US9445251B2	HTC
	<p data-bbox="514 240 1415 342"><b>How to share your navigation directions while you walk, drive or transit</b></p> <p data-bbox="514 375 1545 467">One of the best ways to share your location with someone is during a drive, walk or transit. If you're meeting a friend or family member somewhere, or navigating towards their home, Google Maps lets you share your location with them for the duration of the trip. It's magic!</p> <ol data-bbox="514 513 1383 646" style="list-style-type: none"><li>1. In the <b>search bar</b> enter your destination.</li><li>2. Pick your navigation type (drive, transit, walk) and press the blue <b>navigate button</b>.</li><li>3. Tap the arrow next to the <b>time-to-destination</b> number at the bottom of the screen.</li></ol>  <p data-bbox="499 1328 1346 1360"><a href="https://www.androidcentral.com/how-share-location-google-maps">https://www.androidcentral.com/how-share-location-google-maps</a></p>

### Exhibit C for U.S. Patent No. 9,445,251 Against HTC Accused Products

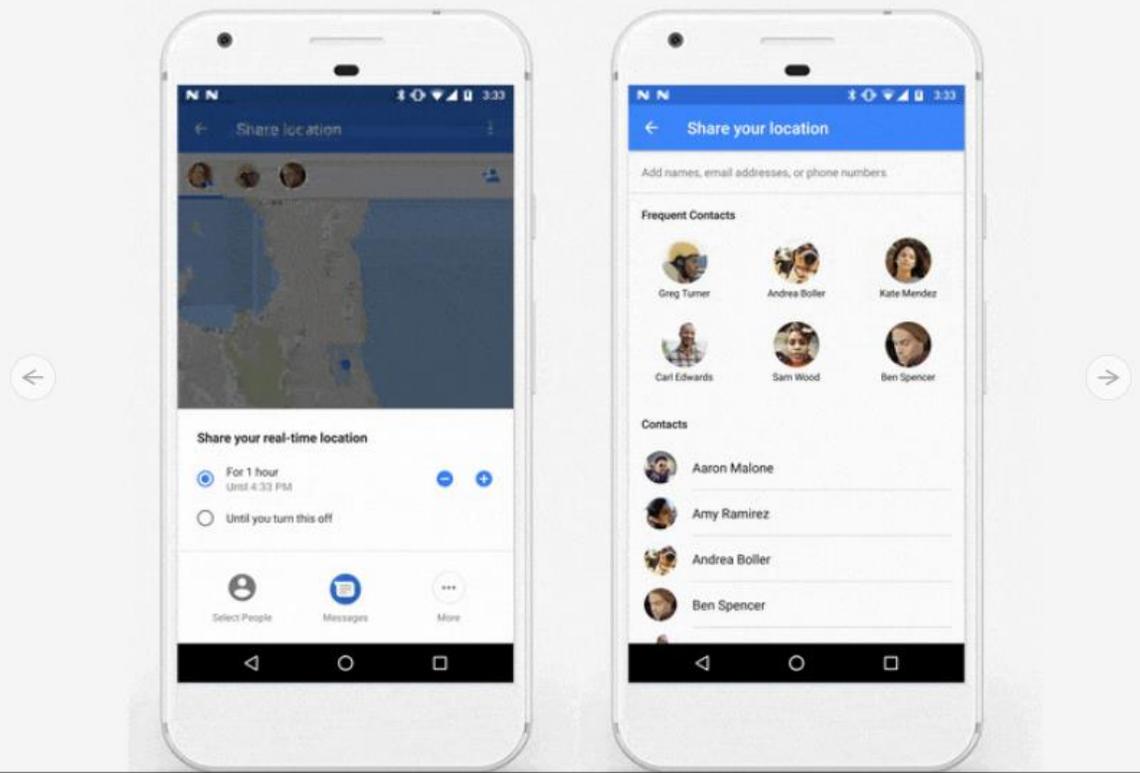
US9445251B2	HTC
	<p data-bbox="514 277 823 305">4. Tap Share trip progress.</p> <p data-bbox="514 334 1136 362">5. Choose one or more contacts to share trip progress.</p> <div data-bbox="520 396 1625 1026"></div> <p data-bbox="520 1065 1329 1092">You can also stop sharing your location with someone before a trip ends.</p> <p data-bbox="499 1102 1346 1133"><a href="https://www.androidcentral.com/how-share-location-google-maps">https://www.androidcentral.com/how-share-location-google-maps</a></p>

**Exhibit C for U.S. Patent No. 9,445,251 Against HTC Accused Products**

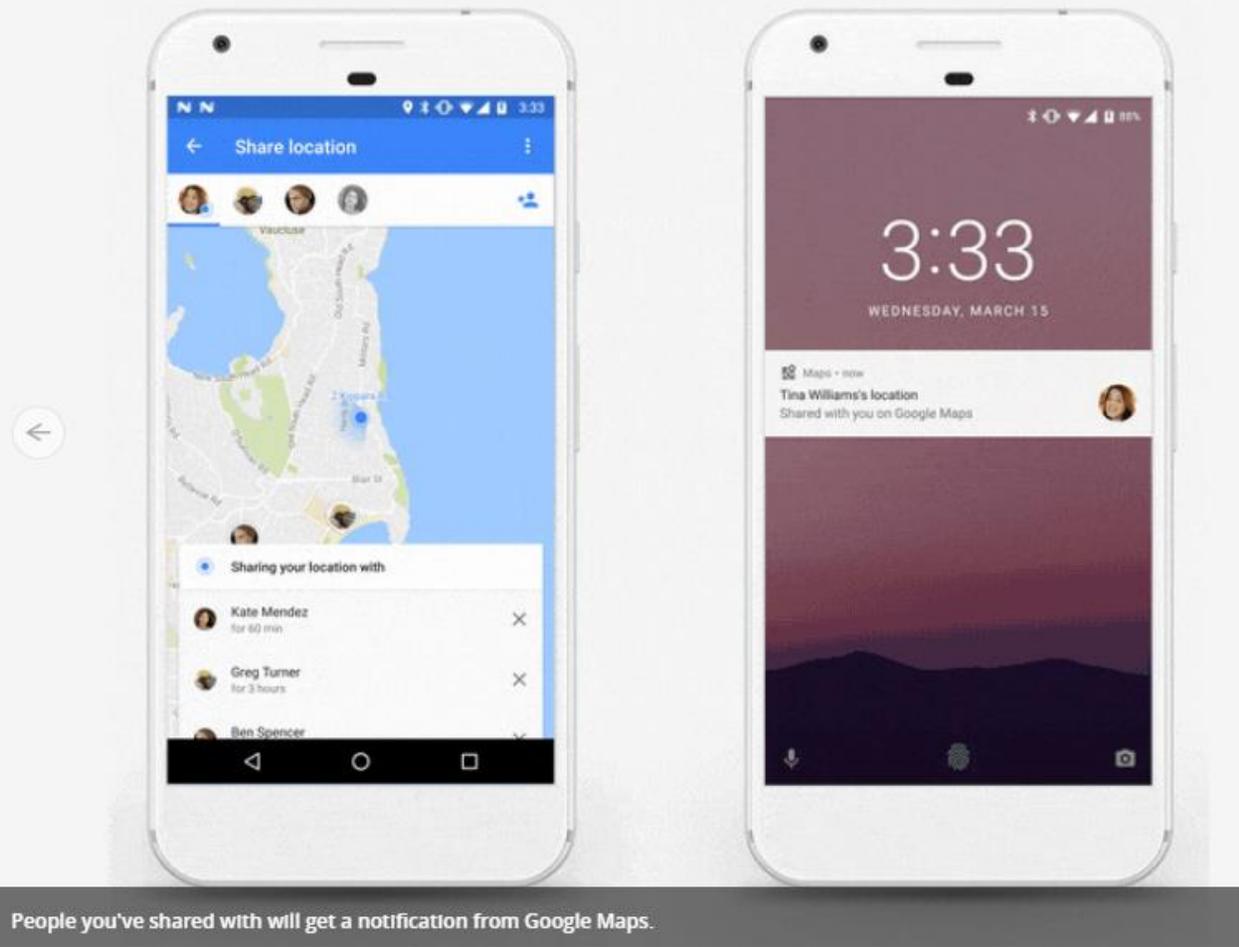
US9445251B2	HTC
	 <p data-bbox="506 901 1507 954">Ways to start location sharing: (left) from the app navigation drawer, and (right) from the menu, you can drag up while navigating somewhere.</p> <p data-bbox="506 963 1648 992"><a href="https://arstechnica.com/gadgets/2017/03/location-sharing-finally-returns-to-google-maps/">https://arstechnica.com/gadgets/2017/03/location-sharing-finally-returns-to-google-maps/</a></p>



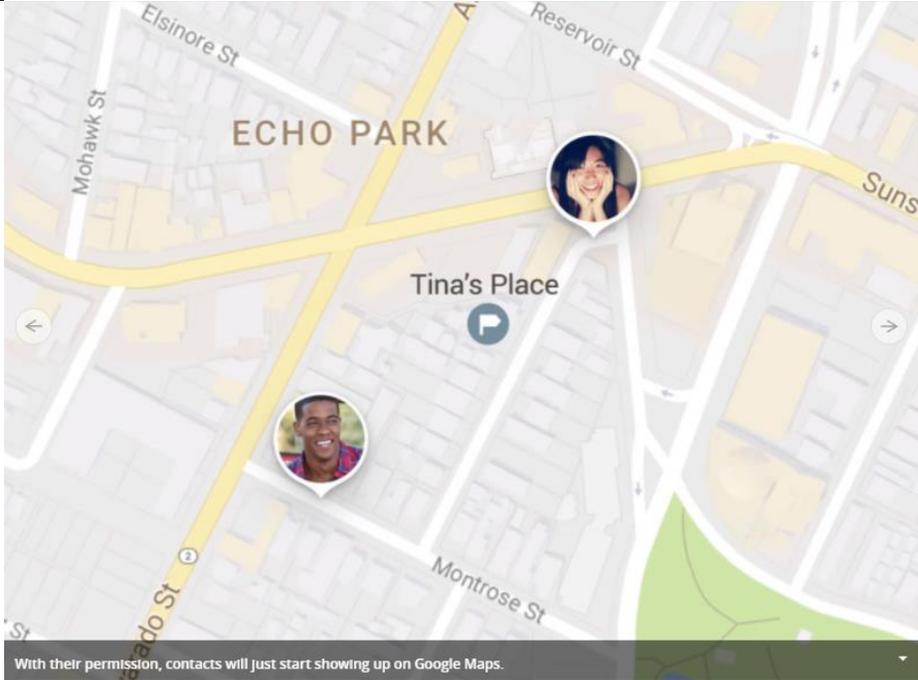
**Exhibit C for U.S. Patent No. 9,445,251 Against HTC Accused Products**

US9445251B2	HTC
	 <p data-bbox="506 1029 1646 1062">You can pick how long you want to share your location for, and with whom you want to share it.</p> <p data-bbox="506 1068 1646 1099"><a href="https://arstechnica.com/gadgets/2017/03/location-sharing-finally-returns-to-google-maps/">https://arstechnica.com/gadgets/2017/03/location-sharing-finally-returns-to-google-maps/</a></p>

**Exhibit C for U.S. Patent No. 9,445,251 Against HTC Accused Products**

US9445251B2	HTC
	 <p data-bbox="506 1146 1163 1174">People you've shared with will get a notification from Google Maps.</p> <p data-bbox="499 1192 1646 1224"><a href="https://arstechnica.com/gadgets/2017/03/location-sharing-finally-returns-to-google-maps/">https://arstechnica.com/gadgets/2017/03/location-sharing-finally-returns-to-google-maps/</a></p>

**Exhibit C for U.S. Patent No. 9,445,251 Against HTC Accused Products**

US9445251B2	HTC
	 <p data-bbox="499 917 1648 950"><a href="https://arstechnica.com/gadgets/2017/03/location-sharing-finally-returns-to-google-maps/">https://arstechnica.com/gadgets/2017/03/location-sharing-finally-returns-to-google-maps/</a></p> <p data-bbox="499 987 1919 1226">For example, Google requests and receives map data from many sources and servers, both internally and externally. In response to one of a number of user actions (e.g. zoom, drag or change focus, change map type, select another device or user), new map data is retrieved to complete the user's action and the displayed map is replaced or updated accordingly. The new map data may come from one of many sources and servers. The Accused Products request and receive the map data described above. Alternatively, any computer signed-in to the Google network services may request and receive the map data described above.</p> <p data-bbox="499 1328 1919 1403">Another example of this limitation in practice includes toggling between map types. The first-displayed map can be the standard interface map with geographical points of interest. In response to user action, a second map can be</p>

**Exhibit C for U.S. Patent No. 9,445,251 Against HTC Accused Products**

US9445251B2	HTC
	<p>called with satellite imagery, which would be retrieved from a different server source having satellite information. Another known example is when a user zooms into a standard map and a transmit map is retrieved and overlaid or otherwise included to form a second map with transit data pulled from a transit data server.</p> <p>3.36 <b>United States.</b>Google maps of the United States include data provided from the following sources:</p> <ul style="list-style-type: none"> <li>a. U.S. Fish and Wildlife Service - <a href="http://www.fws.gov/">http://www.fws.gov/</a></li> <li>b. U.S. Census Bureau - <a href="http://www.census.gov/">http://www.census.gov/</a></li> <li>c. USDA Forest Service - <a href="http://www.fs.fed.us/">http://www.fs.fed.us/</a></li> <li>d. U.S. Geological Survey, Gap Analysis Program (GAP) - <a href="http://gapanalysis.usgs.gov/padus/">http://gapanalysis.usgs.gov/padus/</a></li> <li>e. U.S. Geological Survey, U.S. Geographic Names Information System (GNIS) - <a href="http://geonames.usgs.gov/">http://geonames.usgs.gov/</a></li> <li>f. U.S. Geological Survey, National Hydrography Dataset (NHD) - <a href="http://nhd.usgs.gov/">http://nhd.usgs.gov/</a></li> <li>g. U.S. Geological Survey, Topographic Maps - <a href="http://topomaps.usgs.gov/">http://topomaps.usgs.gov/</a></li> <li>h. U.S. Geological Survey - <a href="http://www.usgs.gov/">http://www.usgs.gov/</a></li> <li>i. U.S. Coast Guard - <a href="http://www.uscg.mil/">http://www.uscg.mil/</a></li> <li>j. University of New Hampshire - <a href="http://ccom.unh.edu/">http://ccom.unh.edu/</a></li> <li>k. U.S. National Parks Service - <a href="http://www.nps.gov/">http://www.nps.gov</a></li> <li>l. U.S. Department of Transportation, Research and Innovative Technology Administration - <a href="http://www.rita.dot.gov/">http://www.rita.dot.gov/</a></li> </ul> <p><a href="https://www.google.com/intl/en_us/help/legalnotices_maps.html">https://www.google.com/intl/en_us/help/legalnotices_maps.html</a></p>

# Exhibit C for U.S. Patent No. 9,445,251 Against HTC Accused Products

US9445251B2	HTC
	<p data-bbox="506 240 695 256"><b>Data center locations</b></p> <p data-bbox="506 272 1304 305">We own and operate data centers around the world to keep our products running 24 hours a day, 7 days a week. Find out more about our data center locations, community involvement, and <a href="#">job opportunities</a> in our locations around the world.</p> <p data-bbox="506 337 569 354"><b>Americas</b></p> <ul data-bbox="506 362 674 532" style="list-style-type: none"><li>Berkeley County, South Carolina</li><li>Council Bluffs, Iowa</li><li>Douglas County, Georgia</li><li>Jackson County, Alabama</li><li>Lenoir, North Carolina</li><li>Mayes County, Oklahoma</li><li>Montgomery County, Tennessee</li><li>Quilicura, Chile</li><li>The Dalles, Oregon</li></ul> <p data-bbox="506 548 537 565"><b>Asia</b></p> <ul data-bbox="506 573 642 605" style="list-style-type: none"><li>Changhua County, Taiwan</li><li>Singapore</li></ul> <p data-bbox="506 621 558 638"><b>Europe</b></p> <ul data-bbox="506 646 642 719" style="list-style-type: none"><li>Dublin, Ireland</li><li>Eemshaven, Netherlands</li><li>Hamina, Finland</li><li>St Ghislain, Belgium</li></ul>  <p data-bbox="506 768 1377 800"><a href="https://www.google.com/about/datacenters/inside/locations/index.html">https://www.google.com/about/datacenters/inside/locations/index.html</a></p>

**Exhibit C for U.S. Patent No. 9,445,251 Against HTC Accused Products**

US9445251B2	HTC
	<h3 data-bbox="506 240 825 264">Request Location Updates</h3> <p data-bbox="506 289 1415 354">Before requesting location updates, your app must connect to location services and make a location request. The lesson on <a href="#">Changing Location Settings</a> shows you how to do this. Once a location request is in place you can start the regular updates by calling <code>requestLocationUpdates()</code>. Do this in the <code>onConnected()</code> callback provided by Google API Client, which is called when the client is ready.</p> <p data-bbox="506 370 1402 435">Depending on the form of the request, the fused location provider either invokes the <code>LocationListener.onLocationChanged()</code> callback method and passes it a <code>Location</code> object, or issues a <code>PendingIntent</code> that contains the location in its extended data. The accuracy and frequency of the updates are affected by the location permissions you've requested and the options you set in the location request object.</p> <p data-bbox="506 451 1411 516">This lesson shows you how to get the update using the <code>LocationListener</code> callback approach. Call <code>requestLocationUpdates()</code>, passing it your instance of the <code>GoogleApiClient</code>, the <code>LocationRequest</code> object, and a <code>LocationListener</code>. Define a <code>startLocationUpdates()</code> method, called from the <code>onConnected()</code> callback, as shown in the following code sample:</p> <pre data-bbox="506 532 1415 764">@Override public void onConnected(Bundle connectionHint) {     ...     if (mRequestingLocationUpdates) {         startLocationUpdates();     } }  protected void startLocationUpdates() {     LocationServices.FusedLocationApi.requestLocationUpdates(         mGoogleApiClient, mLocationRequest, this); }</pre> <p data-bbox="506 781 1402 821">Notice that the above code snippet refers to a boolean flag, <code>mRequestingLocationUpdates</code>, used to track whether the user has turned location updates on or off. For more about retaining the value of this flag across instances of the activity, see <a href="#">Save the State of the Activity</a>.</p> <p data-bbox="506 873 1451 898"><a href="https://developer.android.com/training/location/receive-location-updates.html">https://developer.android.com/training/location/receive-location-updates.html</a></p>

**Exhibit C for U.S. Patent No. 9,445,251 Against HTC Accused Products**

US9445251B2	HTC
	<p>Configure a firewall to allow access to the Google Maps APIs Services</p> <p><i>Why it's important:</i> Google Maps APIs services use a variety of domains, some of which do not belong to the *google.com domain. If you are behind a restrictive firewall, it is important to allow access to the domains used by each Maps API service. If your firewall doesn't allow access to these domains, API requests will fail, which can break your applications. You can find a complete list of domains used by the Maps APIs in the Support Portal:</p> <ol style="list-style-type: none"> <li>1. Log in to the <a href="#">Google Cloud Support Portal</a>. The Support Portal is available only to customers with the Google Maps APIs Premium Plan or a previous Google Maps APIs for Work or Google Maps for Business license.</li> <li>2. Navigate to the <b>Resources</b> tab.</li> <li>3. Select the <b>list of domains used by the Google Maps APIs family</b>. (Here's the <a href="#">direct link</a>.)</li> <li>4. Allow your applications to access the listed domains.</li> </ol> <p>We do not recommend managing firewall restrictions by IP address, as the IPs associated with these domains are not static.</p> <p><b>Note:</b> Google Maps APIs services use port 80 (http) and 443 (https) for inbound and outbound traffic. These services also require GET, POST, PUT, DELETE, and HEAD requests. Configure your firewall to allow traffic over these ports and to allow requests, depending on API and use case.</p> <p><a href="https://developers.google.com/maps/premium/prelaunch-checklist#firewall">https://developers.google.com/maps/premium/prelaunch-checklist#firewall</a></p> <p><b><u>Exemplary Screenshots:</u></b></p> <p>See, e.g., 1B and 1C above.</p> <p><b><u>Exemplary Source Code:</u></b></p> <p>The above functionality is performed at least in part by the following publicly available source code and/or source code that invokes or is invoked by the following source code (or a substantially similar copy compiled and loaded onto the Accused Products by <b>HTC</b>). AGIS reserves the right to supplement these contentions with additional source code as discovery progresses and as additional source code is made available.</p>

**Exhibit C for U.S. Patent No. 9,445,251 Against HTC Accused Products**

US9445251B2	HTC
	<pre>44  * Class that sends chat message via SMS. 45  * 46  * The interface emulates a blocking sending similar to making an HTTP request. 47  * It calls the SmsManager to send a (potentially multipart) message and waits 48  * on the sent status on each part. The waiting has a timeout so it won't wait 49  * forever. Once the sent status of all parts received, the call returns. 50  * A successful sending requires success status for all parts. Otherwise, we 51  * pick the highest level of failure as the error for the whole message, which 52  * is used to determine if we need to retry the sending. 53  */ 54  public class SmsSender { 55      private static final String TAG = LogUtil.BUGLE_TAG; 56 57      public static final String EXTRA_PART_ID = "part_id"; 58 59      /* 60       * A map for pending sms messages. The key is the random request UUID. 61       */ 62      private static ConcurrentHashMap&lt;Uri, SendResult&gt; sPendingMessageMap = 63          new ConcurrentHashMap&lt;Uri, SendResult&gt;(); 64 65      private static final Random RANDOM = new Random(); 66 67      // Whether we should send multipart SMS as separate messages 68      private static Boolean sSendMultipartSmsAsSeparateMessages = null; 69</pre> <p><a href="https://android.googlesource.com/platform/packages/apps/Messaging/+/nougat-mr1-release/src/com/android/messaging/sms/SmsSender.java">https://android.googlesource.com/platform/packages/apps/Messaging/+/nougat-mr1-release/src/com/android/messaging/sms/SmsSender.java</a></p>



**Exhibit C for U.S. Patent No. 9,445,251 Against HTC Accused Products**

US9445251B2	HTC
	<pre> 253 // Actually sending the message using SmsManager 254 private static void sendInternal(final Context context, final int subId, String dest, 255     final ArrayList&lt;String&gt; messages, final String serviceCenter, 256     final boolean requireDeliveryReport, final Uri messageUri) throws SmsException { 257     Assert.notNull(context); 258     final SmsManager smsManager = PhoneUtils.get(subId).getSmsManager(); 259     final int messageCount = messages.size(); 260     final ArrayList&lt;PendingIntent&gt; deliveryIntents = new ArrayList&lt;PendingIntent&gt;(messageCount); 261     final ArrayList&lt;PendingIntent&gt; sentIntents = new ArrayList&lt;PendingIntent&gt;(messageCount); 262     for (int i = 0; i &lt; messageCount; i++) { 263         // Make pending intents different for each message part 264         final int partId = (messageCount &lt;= 1 ? 0 : i + 1); 265         if (requireDeliveryReport &amp;&amp; (i == (messageCount - 1))) { 266             // TODO we only care about the delivery status of the last part 267             // Shall we have better tracking of delivery status of all parts? 268             deliveryIntents.add(PendingIntent.getBroadcast( 269                 context, 270                 partId, 271                 getSendStatusIntent(context, SendStatusReceiver.MESSAGE_DELIVERED_ACTION, 272                     messageUri, partId, subId), 273                 0/*flag*/)); 274         } else { 275             deliveryIntents.add(null); 276         } 277         sentIntents.add(PendingIntent.getBroadcast( 278             context, 279             partId, 280             getSendStatusIntent(context, SendStatusReceiver.MESSAGE_SENT_ACTION, 281                 messageUri, partId, subId), 282             0/*flag*/)); 283     } 284     if (sSendMultipartSmsAsSeparateMessages == null) { 285         sSendMultipartSmsAsSeparateMessages = MmsConfig.get(subId) 286             .getSendMultipartSmsAsSeparateMessages(); 287     } </pre> <p><a href="https://android.googlesource.com/platform/packages/apps/Messaging/+nougat-mr1-release/src/com/android/messaging/sms/SmsSender.java">https://android.googlesource.com/platform/packages/apps/Messaging/+nougat-mr1-release/src/com/android/messaging/sms/SmsSender.java</a></p>

**Exhibit C for U.S. Patent No. 9,445,251 Against HTC Accused Products**

US9445251B2	HTC
	<pre> 288     try { 289         if (sSendMultipartSmsAsSeparateMessages) { 290             // If multipart sms is not supported, send them as separate messages 291             for (int i = 0; i &lt; messageCount; i++) { 292                 smsManager.sendTextMessage(dest, 293                     serviceCenter, 294                     messages.get(i), 295                     sentIntents.get(i), 296                     deliveryIntents.get(i)); 297             } 298         } else { 299             smsManager.sendMultipartTextMessage( 300                 dest, serviceCenter, messages, sentIntents, deliveryIntents); 301         } 302     } catch (final Exception e) { 303         throw new SmsException("SmsSender: caught exception in sending " + e); 304     } 305 } </pre> <p><a href="https://android.googlesource.com/platform/packages/apps/Messaging+/nougat-mr1-release/src/com/android/messaging/sms/SmsSender.java">https://android.googlesource.com/platform/packages/apps/Messaging+/nougat-mr1-release/src/com/android/messaging/sms/SmsSender.java</a></p> <pre> 56  * Class that receives incoming SMS messages through android.provider.Telephony.SMS_RECEIVED 57  * 58  * This class serves two purposes: 59  * - Process phone verification SMS messages 60  * - Handle SMS messages when the user has enabled us to be the default SMS app (Pre-KLP) 61  */ 62  public final class SmsReceiver extends BroadcastReceiver { 63      private static final String TAG = LogUtil.BUGLE_TAG; 64 65      private static ArrayList&lt;Pattern&gt; sIgnoreSmsPatterns; 66 </pre> <p><a href="https://android.googlesource.com/platform/packages/apps/Messaging+/nougat-mr1-release/src/com/android/messaging/receiver/SmsReceiver.java">https://android.googlesource.com/platform/packages/apps/Messaging+/nougat-mr1-release/src/com/android/messaging/receiver/SmsReceiver.java</a></p>

**Exhibit C for U.S. Patent No. 9,445,251 Against HTC Accused Products**

US9445251B2	HTC
	<pre>200     public static void deliverSmsMessages(final Context context, final int subId, 201         final int errorCode, final android.telephony.SmsMessage[] messages) { 202         final ContentValues messageValues = 203             MmsUtils.parseReceivedSmsMessage(context, messages, errorCode); 204 205         LogUtil.v(TAG, "SmsReceiver.deliverSmsMessages"); 206 207         final long nowInMillis = System.currentTimeMillis(); 208         final long receivedTimestampMs = MmsUtils.getMessageDate(messages[0], nowInMillis); 209 210         messageValues.put(Sms.Inbox.DATE, receivedTimestampMs); 211         // Default to unread and unseen for us but ReceiveSmsMessageAction will override 212         // seen for the telephony db. 213         messageValues.put(Sms.Inbox.READ, 0); 214         messageValues.put(Sms.Inbox.SEEN, 0); 215         if (OsUtil.isAtLeastL_MR1()) { 216             messageValues.put(Sms.SUBSCRIPTION_ID, subId); 217         } 218 219         if (messages[0].getMessageClass() == android.telephony.SmsMessage.MessageClass.CLASS_0    220             DebugUtils.debugClassZeroSmsEnabled()) { 221             Factory.get().getUIIntents().launchClassZeroActivity(context, messageValues); 222         } else { 223             final ReceiveSmsMessageAction action = new ReceiveSmsMessageAction(messageValues); 224             action.start(); 225         } 226     }</pre> <p><a href="https://android.googlesource.com/platform/packages/apps/Messaging/+nougat-mr1-release/src/com/android/messaging/receiver/SmsReceiver.java">https://android.googlesource.com/platform/packages/apps/Messaging/+nougat-mr1-release/src/com/android/messaging/receiver/SmsReceiver.java</a></p>

**Exhibit C for U.S. Patent No. 9,445,251 Against HTC Accused Products**

US9445251B2	HTC
	<pre>228     @Override 229     public void onReceive(final Context context, final Intent intent) { 230         LogUtil.v(TAG, "SmsReceiver.onReceive " + intent); 231         // On KLP+ we only take delivery of SMS messages in SmsDeliverReceiver. 232         if (PhoneUtils.getDefault().isSmsEnabled()) { 233             final String action = intent.getAction(); 234             if (OsUtil.isSecondaryUser() &amp;&amp; 235                 (Telephony.Sms.Intents.SMS_RECEIVED_ACTION.equals(action)    236                  // TODO: update this with the actual constant from Telephony 237                  "android.provider.Telephony.MMS_DOWNLOADED".equals(action))) { 238                 postNewMessageSecondaryUserNotification(); 239             } else if (!OsUtil.isAtLeastKLP()) { 240                 deliverSmsIntent(context, intent); 241             } 242         } 243     }</pre> <p><a href="https://android.googlesource.com/platform/packages/apps/Messaging/+nougat-mr1-release/src/com/android/messaging/receiver/SmsReceiver.java">https://android.googlesource.com/platform/packages/apps/Messaging/+nougat-mr1-release/src/com/android/messaging/receiver/SmsReceiver.java</a></p>

**Exhibit C for U.S. Patent No. 9,445,251 Against HTC Accused Products**

US9445251B2	HTC
	<pre>52  * Class that sends chat message via MMS. 53  * 54  * The interface emulates a blocking send similar to making an HTTP request. 55  */ 56  public class MmsSender { 57      private static final String TAG = LogUtil.BUGLE_TAG; 58 59      /** 60       * Send an MMS message. 61       * 62       * @param context Context 63       * @param messageUri The unique URI of the message for identifying it during sending 64       * @param sendReq The SendReq PDU of the message 65       * @throws MmsFailureException 66       */ 67      public static void sendMms(final Context context, final int subId, final Uri messageUri, 68                               final SendReq sendReq, final Bundle sentIntentExtras) throws MmsFailureException { 69          sendMms(context, 70                 subId, 71                 messageUri, 72                 null /* locationUri */, 73                 sendReq, 74                 true /* responseImportant */, 75                 sentIntentExtras); 76      }</pre> <p><a href="https://android.googlesource.com/platform/packages/apps/Messaging/+nougat-mr1-release/src/com/android/messaging/sms/MmsSender.java">https://android.googlesource.com/platform/packages/apps/Messaging/+nougat-mr1-release/src/com/android/messaging/sms/MmsSender.java</a></p>

**Exhibit C for U.S. Patent No. 9,445,251 Against HTC Accused Products**

US9445251B2	HTC
	<pre> 240      * Download an MMS message. 241      * 242      * @param context Context 243      * @param contentLocation The url of the MMS message 244      * @throws MmsFailureException 245      * @throws InvalidHeaderValueException 246      */ 247      public static void downloadMms(final Context context, final int subId, 248          final String contentLocation, Bundle extras) throws MmsFailureException, 249          InvalidHeaderValueException { 250          final Uri requestUri = Uri.parse(contentLocation); 251          final Uri contentUri = MmsFileProvider.buildRawMmsUri(); 252 253          final Intent downloadedIntent = new Intent(SendStatusReceiver.MMS_DOWNLOADED_ACTION, 254              requestUri, 255              context, 256              SendStatusReceiver.class); 257          downloadedIntent.putExtra(SendMessageAction.EXTRA_CONTENT_URI, contentUri); 258          if (extras != null) { 259              downloadedIntent.putExtras(extras); 260          } 261          final PendingIntent downloadedPendingIntent = PendingIntent.getBroadcast( 262              context, 263              0 /*request code*/, 264              downloadedIntent, 265              PendingIntent.FLAG_UPDATE_CURRENT); 266 267          MmsManager.downloadMultimediaMessage(subId, context, contentLocation, contentUri, 268              downloadedPendingIntent); 269      } </pre> <p><a href="https://android.googlesource.com/platform/packages/apps/Messaging/+nougat-mr1-release/src/com/android/messaging/sms/MmsSender.java">https://android.googlesource.com/platform/packages/apps/Messaging/+nougat-mr1-release/src/com/android/messaging/sms/MmsSender.java</a></p>

**Exhibit C for U.S. Patent No. 9,445,251 Against HTC Accused Products**

US9445251B2	HTC
	<pre> 97      * Execute an MMS HTTP request, either a POST (sending) or a GET (downloading) 98      * 99      * @param urlString The request URL, for sending it is usually the MMSC, and for downloading 100     *           it is the message URL 101     * @param pdu For POST (sending) only, the PDU to send 102     * @param method HTTP method, POST for sending and GET for downloading 103     * @param isProxySet Is there a proxy for the MMSC 104     * @param proxyHost The proxy host 105     * @param proxyPort The proxy port 106     * @param mmsConfig The MMS config to use 107     * @param userAgent The user agent header value 108     * @param uaProfUrl The UA Prof URL header value 109     * @return The HTTP response body 110     * @throws MmsHttpException For any failures 111     */ 112     public byte[] execute(String urlString, byte[] pdu, String method, boolean isProxySet, 113                          String proxyHost, int proxyPort, Bundle mmsConfig, String userAgent, String uaProfUrl) 114         throws MmsHttpException { 115         Log.d(MmsService.TAG, "HTTP: " + method + " " + Utils.redactUrlForNonVerbose(urlString) 116             + (isProxySet ? (" proxy=" + proxyHost + ":" + proxyPort) : "") 117             + ", PDU size=" + (pdu != null ? pdu.length : 0)); 118         checkMethod(method); 119         HttpURLConnection connection = null; 120         try { 121             Proxy proxy = Proxy.NO_PROXY; 122             if (isProxySet) { 123                 proxy = new Proxy(Proxy.Type.HTTP, new InetSocketAddress(proxyHost, proxyPort)); 124             } 125             final URL url = new URL(urlString); 126             // Now get the connection 127             connection = (HttpURLConnection) url.openConnection(proxy); 128             connection.setDoInput(true); 129             connection.setConnectTimeout( 130                 mmsConfig.getInt(CarrierConfigValuesLoader.CONFIG_HTTP_SOCKET_TIMEOUT, 131                                 CarrierConfigValuesLoader.CONFIG_HTTP_SOCKET_TIMEOUT_DEFAULT)); </pre> <p data-bbox="499 1222 1583 1289"><a href="https://android.googlesource.com/platform/packages/apps/Messaging/+nougat-mr1-release/src/android/support/v7/mms/MmsHttpClient.java">https://android.googlesource.com/platform/packages/apps/Messaging/+nougat-mr1-release/src/android/support/v7/mms/MmsHttpClient.java</a></p>

**Exhibit C for U.S. Patent No. 9,445,251 Against HTC Accused Products**

US9445251B2	HTC
	<pre> 132 // ----- COMMON HEADERS ----- 133 // Header: Accept 134 connection.setRequestProperty(HEADER_ACCEPT, HEADER_VALUE_ACCEPT); 135 // Header: Accept-Language 136 connection.setRequestProperty( 137     HEADER_ACCEPT_LANGUAGE, getCurrentAcceptLanguage(Locale.getDefault())); 138 // Header: User-Agent 139 Log.i(MmsService.TAG, "HTTP: User-Agent=" + userAgent); 140 connection.setRequestProperty(HEADER_USER_AGENT, userAgent); 141 // Header: x-wap-profile 142 final String uaProfUrlTagName = mmsConfig.getString( 143     CarrierConfigValuesLoader.CONFIG_UA_PROF_TAG_NAME, 144     CarrierConfigValuesLoader.CONFIG_UA_PROF_TAG_NAME_DEFAULT); 145 if (uaProfUrl != null) { 146     Log.i(MmsService.TAG, "HTTP: UaProfUrl=" + uaProfUrl); 147     connection.setRequestProperty(uaProfUrlTagName, uaProfUrl); 148 } 149 // Add extra headers specified by mms_config.xml's httpparams 150 addExtraHeaders(connection, mmsConfig); 151 // Different stuff for GET and POST 152 if (METHOD_POST.equals(method)) { 153     if (pdu == null    pdu.length &lt; 1) { 154         Log.e(MmsService.TAG, "HTTP: empty pdu"); 155         throw new MmsHttpException(0/*statusCode*/, "Sending empty PDU"); 156     } 157     connection.setDoOutput(true); 158     connection.setRequestMethod(METHOD_POST); 159     if (mmsConfig.getBoolean( 160         CarrierConfigValuesLoader.CONFIG_SUPPORT_HTTP_CHARSET_HEADER, 161         CarrierConfigValuesLoader.CONFIG_SUPPORT_HTTP_CHARSET_HEADER_DEFAULT)) { 162         connection.setRequestProperty(HEADER_CONTENT_TYPE, 163             HEADER_VALUE_CONTENT_TYPE_WITH_CHARSET); 164     } else { 165         connection.setRequestProperty(HEADER_CONTENT_TYPE, 166             HEADER_VALUE_CONTENT_TYPE_WITHOUT_CHARSET); </pre> <p><a href="https://android.goesource.com/platform/packages/apps/Messaging+/nougat-mr1-release/src/android/support/v7/mms/MmsHttpClient.java">https://android.goesource.com/platform/packages/apps/Messaging+/nougat-mr1-release/src/android/support/v7/mms/MmsHttpClient.java</a></p>



**Exhibit C for U.S. Patent No. 9,445,251 Against HTC Accused Products**

US9445251B2	HTC
	<pre> 167         } 168         if (Log.isLoggable(MmsService.TAG, Log.VERBOSE)) { 169             logHttpHeaders(connection.getRequestProperties()); 170         } 171         connection.setFixedLengthStreamingMode(pdu.length); 172         // Sending request body 173         final OutputStream out = 174             new BufferedOutputStream(connection.getOutputStream()); 175         out.write(pdu); 176         out.flush(); 177         out.close(); 178     } else if (METHOD_GET.equals(method)) { 179         if (Log.isLoggable(MmsService.TAG, Log.VERBOSE)) { 180             logHttpHeaders(connection.getRequestProperties()); 181         } 182         connection.setRequestMethod(METHOD_GET); 183     } 184     // Get response 185     final int responseCode = connection.getResponseCode(); 186     final String responseMessage = connection.getResponseMessage(); 187     Log.d(MmsService.TAG, "HTTP: " + responseCode + " " + responseMessage); 188     if (Log.isLoggable(MmsService.TAG, Log.VERBOSE)) { 189         logHttpHeaders(connection.getHeaderFields()); 190     } 191     if (responseCode / 100 != 2) { 192         throw new MmsHttpException(responseCode, responseMessage); 193     } 194     final InputStream in = new BufferedInputStream(connection.getInputStream()); 195     final ByteArrayOutputStream byteOut = new ByteArrayOutputStream(); 196     final byte[] buf = new byte[4096]; 197     int count = 0; 198     while ((count = in.read(buf)) &gt; 0) { 199         byteOut.write(buf, 0, count); 200     } 201     in.close(); 202     final byte[] responseBody = byteOut.toByteArray(); 203     Log.d(MmsService.TAG, "HTTP: response size=" 204         + (responseBody != null ? responseBody.length : 0)); 205     return responseBody; </pre> <p><a href="https://android.googlesource.com/platform/packages/apps/Messaging+/nougat-mr1-release/src/android/support/v7/mms/MmsHttpClient.java">https://android.googlesource.com/platform/packages/apps/Messaging+/nougat-mr1-release/src/android/support/v7/mms/MmsHttpClient.java</a></p>

**Exhibit C for U.S. Patent No. 9,445,251 Against HTC Accused Products**

US9445251B2	HTC
	<pre>38  * Request to send an MMS 39  */ 40  class SendRequest extends MmsRequest { 41      // Max send response PDU size in bytes (exceeding this may cause problem with 42      // system intent delivery). 43      private static final int MAX_SEND_RESPONSE_SIZE = 1000 * 1024; 44 45      private byte[] mPduData; 46 47      SendRequest(final String locationUri, final Uri pduUri, final PendingIntent sentIntent) { 48          super(locationUri, pduUri, sentIntent); 49      } 50 51      @Override 52      protected boolean loadRequest(final Context context, final Bundle mmsConfig) { 53          mPduData = readPduFromContentUri( 54              context, 55              mPduUri, 56              mmsConfig.getInt( 57                  CarrierConfigValuesLoader.CONFIG_MAX_MESSAGE_SIZE, 58                  CarrierConfigValuesLoader.CONFIG_MAX_MESSAGE_SIZE_DEFAULT)); 59          return (mPduData != null); 60      } 61 62      @Override 63      protected boolean transferResponse(final Context context, final Intent fillIn, 64          final byte[] response) { 65          // SendConf pdus are always small and can be included in the intent 66          if (response != null &amp;&amp; fillIn != null) { 67              if (response.length &gt; MAX_SEND_RESPONSE_SIZE) { 68                  // If the response PDU is too large, it won't be able to fit in 69                  // the PendingIntent to be transferred via system IPC. 70                  return false; 71              } 72              fillIn.putExtra(SmsManager.EXTRA_MMS_DATA, response); 73          } 74          return true; 75      } }</pre>

**Exhibit C for U.S. Patent No. 9,445,251 Against HTC Accused Products**

US9445251B2	HTC
	<p data-bbox="499 235 1583 302"><a href="https://android.googlesource.com/platform/packages/apps/Messaging+/nougat-mr1-release/src/android/support/v7/mms/SendRequest.java">https://android.googlesource.com/platform/packages/apps/Messaging+/nougat-mr1-release/src/android/support/v7/mms/SendRequest.java</a></p> <pre data-bbox="520 358 968 391">public static LocationRequest create ()</pre> <p data-bbox="512 427 1016 451">Create a location request with default parameters.</p> <p data-bbox="512 483 1625 542">Default parameters are for a block accuracy, slowly updated location. It can then be adjusted as required by the applications before passing to the <a href="#">FusedLocationProviderApi</a>.</p> <p data-bbox="533 566 617 591"><b>Returns</b></p> <ul data-bbox="541 613 800 638" style="list-style-type: none"><li>• a new location request</li></ul> <p data-bbox="499 651 1785 683"><a href="https://developers.google.com/android/reference/com/google/android/gms/location/LocationRequest">https://developers.google.com/android/reference/com/google/android/gms/location/LocationRequest</a></p>

**Exhibit C for U.S. Patent No. 9,445,251 Against HTC Accused Products**

US9445251B2	HTC
	<p><b>public static final int PRIORITY_BALANCED_POWER_ACCURACY</b></p> <p>Used with <code>setPriority(int)</code> to request "block" level accuracy.</p> <p>Block level accuracy is considered to be about 100 meter accuracy. Using a coarse accuracy such as this often consumes less power.</p> <p>Constant Value: 102</p> <hr/> <p><b>public static final int PRIORITY_HIGH_ACCURACY</b></p> <p>Used with <code>setPriority(int)</code> to request the most accurate locations available.</p> <p>This will return the finest location available.</p> <p>Constant Value: 100</p> <hr/> <p><b>public static final int PRIORITY_LOW_POWER</b></p> <p>Used with <code>setPriority(int)</code> to request "city" level accuracy.</p> <p>City level accuracy is considered to be about 10km accuracy. Using a coarse accuracy such as this often consumes less power.</p> <p>Constant Value: 104</p> <p><a href="https://developers.google.com/android/reference/com/google/android/gms/location/LocationRequest">https://developers.google.com/android/reference/com/google/android/gms/location/LocationRequest</a></p>

**Exhibit C for U.S. Patent No. 9,445,251 Against HTC Accused Products**

US9445251B2	HTC
	<p data-bbox="520 250 1738 289"><code>public Task&lt;Location&gt; getLastLocation ()</code></p> <p data-bbox="520 318 1094 341">Returns the best most recent location currently available.</p> <p data-bbox="520 375 1682 431">If a location is not available, which should happen very rarely, null will be returned. The best accuracy available while respecting the location permissions will be returned.</p> <p data-bbox="520 466 1724 522">This method provides a simplified way to get location. It is particularly well suited for applications that do not require an accurate location and that do not want to maintain extra logic for location updates.</p> <p data-bbox="520 581 1738 620"><code>public Task&lt;LocationAvailability&gt; getLocationAvailability ()</code></p> <p data-bbox="520 649 1675 706">Returns the availability of location data. When <code>isLocationAvailable()</code> returns true, then the location returned by <code>getLastLocation()</code> will be reasonably up to date within the hints specified by the active <code>LocationRequest</code> s.</p> <p data-bbox="520 740 1457 763">If the client isn't connected to Google Play services and the request times out, null is returned.</p> <p data-bbox="520 797 1661 854">Note it's always possible for <code>getLastLocation()</code> to return null even when this method returns true (e.g. location settings were disabled between calls).</p> <p data-bbox="520 867 1906 925"><a href="https://developers.google.com/android/reference/com/google/android/gms/location/FusedLocationProviderClient">https://developers.google.com/android/reference/com/google/android/gms/location/FusedLocationProviderClient</a></p>

**Exhibit C for U.S. Patent No. 9,445,251 Against HTC Accused Products**

US9445251B2	HTC						
	<p data-bbox="512 245 1734 326"><code>public Task&lt;Void&gt; requestLocationUpdates (LocationRequest request, LocationCallback callback, Looper looper)</code></p> <p data-bbox="512 354 1255 378">Requests location updates with a callback on the specified Looper thread.</p> <p data-bbox="512 412 1671 472">This method is suited for the foreground use cases. For background use cases, the <code>PendingIntent</code> version of the method is recommended, see <code>requestLocationUpdates(LocationRequest, PendingIntent)</code>.</p> <p data-bbox="512 505 1356 529">Any previous LocationRequests registered on this LocationListener will be replaced.</p> <p data-bbox="512 561 1671 654">This call will keep the Google Play services connection active, so make sure to call <code>removeLocationUpdates(LocationCallback)</code> when you no longer need it, otherwise you lose the benefits of the automatic connection management.</p> <p data-bbox="512 686 1728 711">Callbacks for <code>LocationCallback</code> will be made on the specified thread, which must already be a prepared looper thread.</p> <p data-bbox="533 735 653 760"><b>Parameters</b></p> <table border="1" data-bbox="512 792 1734 1008"> <tbody> <tr> <td data-bbox="512 792 617 857"><b>request</b></td> <td data-bbox="617 792 1734 857">The location request for the updates.</td> </tr> <tr> <td data-bbox="512 857 617 922"><b>callback</b></td> <td data-bbox="617 857 1734 922">The callback for the location updates.</td> </tr> <tr> <td data-bbox="512 922 617 1008"><b>looper</b></td> <td data-bbox="617 922 1734 1008">The Looper object whose message queue will be used to implement the callback mechanism, or null to make callbacks on the calling thread.</td> </tr> </tbody> </table> <p data-bbox="512 1024 1906 1089"><a href="https://developers.google.com/android/reference/com/google/android/gms/location/FusedLocationProviderClient">https://developers.google.com/android/reference/com/google/android/gms/location/FusedLocationProviderClient</a></p>	<b>request</b>	The location request for the updates.	<b>callback</b>	The callback for the location updates.	<b>looper</b>	The Looper object whose message queue will be used to implement the callback mechanism, or null to make callbacks on the calling thread.
<b>request</b>	The location request for the updates.						
<b>callback</b>	The callback for the location updates.						
<b>looper</b>	The Looper object whose message queue will be used to implement the callback mechanism, or null to make callbacks on the calling thread.						

**Exhibit C for U.S. Patent No. 9,445,251 Against HTC Accused Products**

US9445251B2	HTC				
	<pre data-bbox="514 243 1732 324">public Task&lt;Void&gt; requestLocationUpdates (LocationRequest request, PendingIntent callbackIntent)</pre> <p data-bbox="514 349 1260 381">Requests location updates with a callback on the specified PendingIntent.</p> <p data-bbox="514 406 1722 544">This method is suited for the background use cases, more specifically for receiving location updates, even when the app has been killed by the system. In order to do so, use a <code>PendingIntent</code> for a started service. For foreground use cases, the <code>LocationCallback</code> version of the method is recommended, see <code>requestLocationUpdates(LocationRequest, LocationCallback, Looper)</code>.</p> <p data-bbox="514 568 1722 633">Any previously registered requests that have the same PendingIntent (as defined by <code>equals(Object)</code>) will be replaced by this request.</p> <p data-bbox="514 657 1722 755">Both <code>LocationResult</code> and <code>LocationAvailability</code> are sent to the given PendingIntent. You can extract data from an Intent using <code>hasResult(Intent)</code>, <code>extractResult(Intent)</code>, <code>hasLocationAvailability(Intent)</code>, and <code>extractLocationAvailability(Intent)</code>.</p> <p data-bbox="514 779 661 812"><b>Parameters</b></p> <table border="1" data-bbox="514 836 1732 966"> <tbody> <tr> <td data-bbox="514 836 829 901"><code>request</code></td> <td data-bbox="829 836 1732 901">The location request for the updates.</td> </tr> <tr> <td data-bbox="514 901 829 966"><code>callbackIntent</code></td> <td data-bbox="829 901 1732 966">A pending intent to be sent for each location update.</td> </tr> </tbody> </table> <p data-bbox="514 990 619 1023"><b>Returns</b></p> <ul data-bbox="514 1039 1354 1071" style="list-style-type: none"> <li>• a Task for the call, check <code>isSuccessful()</code> to determine if it was successful.</li> </ul> <p data-bbox="514 1079 1911 1144"><a href="https://developers.google.com/android/reference/com/google/android/gms/location/FusedLocationProviderClient">https://developers.google.com/android/reference/com/google/android/gms/location/FusedLocationProviderClient</a></p>	<code>request</code>	The location request for the updates.	<code>callbackIntent</code>	A pending intent to be sent for each location update.
<code>request</code>	The location request for the updates.				
<code>callbackIntent</code>	A pending intent to be sent for each location update.				

## Exhibit C for U.S. Patent No. 9,445,251 Against HTC Accused Products

US9445251B2	HTC						
	<p><code>public void onLocationAvailability (LocationAvailability locationAvailability)</code></p> <p>Called when there is a change in the availability of location data.</p> <p>When <code>isLocationAvailable()</code> returns <code>false</code> you can assume that location will not be returned in <code>onLocationResult(LocationResult)</code> until something changes in the device's settings or environment. Even when <code>isLocationAvailable()</code> returns <code>true</code> the <code>onLocationResult(LocationResult)</code> may not always be called regularly, however the device location is known and both the most recently delivered location and <code>getLastLocation(GoogleApiClient)</code> will be reasonably up to date given the hints specified by the active <code>LocationRequest</code>s.</p> <p><b>Parameters</b></p> <table border="1"> <tr> <td><code>locationAvailability</code></td> <td>The current status of location availability.</td> </tr> </table> <p><code>public void onLocationResult (LocationResult result)</code></p> <p>Called when device location information is available.</p> <p>The most recent location returned by <code>getLastLocation()</code> is not guaranteed to be immediately fresh, but will be reasonably up to date given the hints specified by the active <code>LocationRequest</code>s.</p> <p><b>Parameters</b></p> <table border="1"> <tr> <td><code>result</code></td> <td>The latest location result available.</td> </tr> </table> <p><a href="https://developers.google.com/android/reference/com/google/android/gms/location/LocationCallback">https://developers.google.com/android/reference/com/google/android/gms/location/LocationCallback</a></p> <p><code>public abstract void onLocationChanged (Location location)</code></p> <p>Called when the location has changed.</p> <p><b>Parameters</b></p> <table border="1"> <tr> <td><code>location</code></td> <td>The updated location.</td> </tr> </table>	<code>locationAvailability</code>	The current status of location availability.	<code>result</code>	The latest location result available.	<code>location</code>	The updated location.
<code>locationAvailability</code>	The current status of location availability.						
<code>result</code>	The latest location result available.						
<code>location</code>	The updated location.						



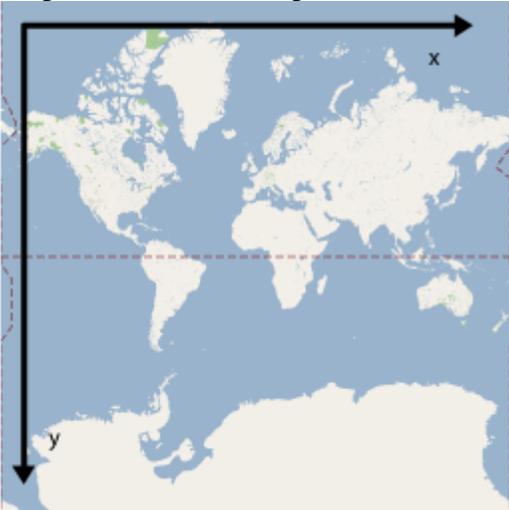
**Exhibit C for U.S. Patent No. 9,445,251 Against HTC Accused Products**

US9445251B2	HTC
	<p><a href="https://developers.google.com/android/reference/com/google/android/gms/location/LocationListener">https://developers.google.com/android/reference/com/google/android/gms/location/LocationListener</a></p> <p>Public Constructors</p> <hr/> <p>public <b>MapView</b> (<a href="#">Context</a> context)</p> <p>public <b>MapView</b> (<a href="#">Context</a> context, <a href="#">AttributeSet</a> attrs)</p> <p>public <b>MapView</b> (<a href="#">Context</a> context, <a href="#">AttributeSet</a> attrs, int defStyle)</p> <p>public <b>MapView</b> (<a href="#">Context</a> context, <a href="#">GoogleMapOptions</a> options)</p> <p><a href="https://developers.google.com/android/reference/com/google/android/gms/maps/MapView">https://developers.google.com/android/reference/com/google/android/gms/maps/MapView</a></p>

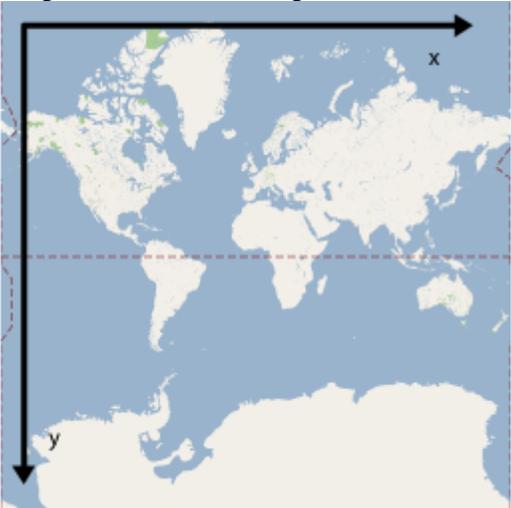
**Exhibit C for U.S. Patent No. 9,445,251 Against HTC Accused Products**

US9445251B2	HTC
	<p><code>public void <b>getMapAsync</b> (<b>OnMapReadyCallback</b> callback)</code></p> <p>Returns a non-null instance of the <code>GoogleMap</code>, ready to be used.</p> <p>Note that:</p> <ul style="list-style-type: none"> <li>• This method must be called from the main thread.</li> <li>• The callback will be executed in the main thread.</li> <li>• In the case where Google Play services is not installed on the user's device, the callback will not be triggered until the user installs it.</li> <li>• The <code>GoogleMap</code> object provided by the callback is non-null.</li> </ul> <p><b>Parameters</b></p> <p><code>callback</code> The callback object that will be triggered when the map is ready to be used.</p> <p><code>public final void <b>onCreate</b> (<b>Bundle</b> savedInstanceState)</code></p> <p>You must call this method from the parent Activity/Fragment's corresponding method.</p> <p><a href="https://developers.google.com/android/reference/com/google/android/gms/maps/MapView">https://developers.google.com/android/reference/com/google/android/gms/maps/MapView</a></p>
<p>[1E] receiving, from the server, the second georeferenced map, wherein the second georeferenced map includes the requested location and data relating positions on the second georeferenced map to</p>	<p><b>HTC</b> infringes directly and/or indirectly by performing, inducing others to perform, and/or contributing to the performance of: receiving, from the server, the second georeferenced map, wherein the second georeferenced map includes the requested location and data relating positions on the second georeferenced map to spatial coordinates.</p> <p><b>Regarding Find My Device</b> and Android Device Manager, the user, via the first device, or the device itself receives second map data from a second server. The received second map data occurs responsive to user input (e.g., zoom, drag, pan, change focus, change map type, refresh or reload request device or symbol selection, another device or user selection, change in position of first device, change in position of a second device). Alternatively, the second may data may be received responsive to an automatic and/or pre-</p>

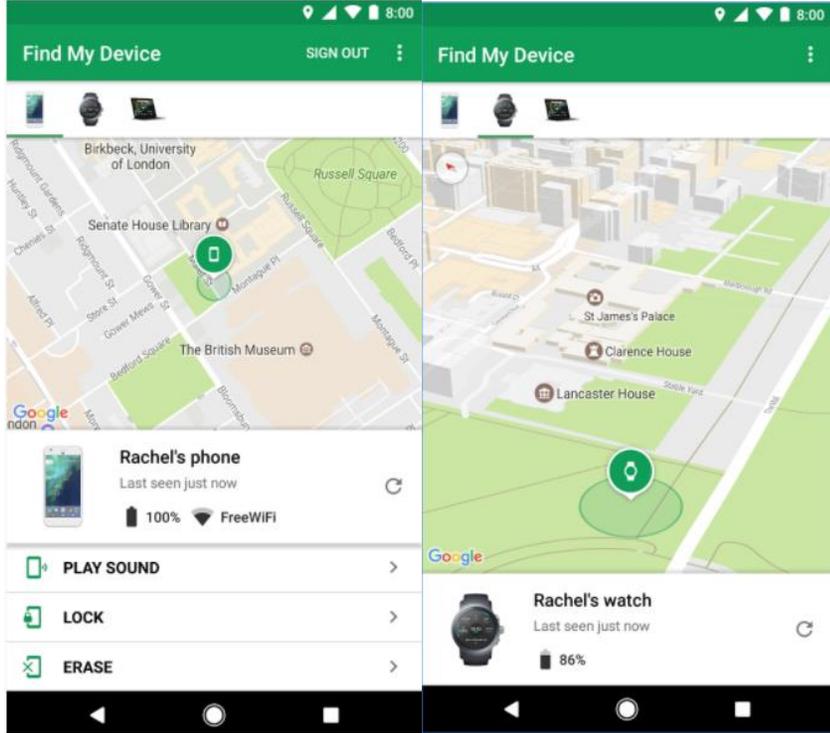
**Exhibit C for U.S. Patent No. 9,445,251 Against HTC Accused Products**

US9445251B2	HTC
spatial coordinates;	<p>determined control caused by an instruction from within the first device or from the one or more second devices, e.g. a refresh. The received second map data includes an update to the first data or a replacement of the first data. The second map data may come from one of many sources and servers. The second map data includes georeferenced data. This information includes locations and data relating the positions to spatial coordinates. See, e.g.:</p> <p>World coordinates in Google Maps are measured from the Mercator projection's origin (the northwest corner of the map at 180 degrees longitude and approximately 85 degrees latitude) and increase in the x direction towards the east (right) and increase in the y direction towards the south (down). Because the basic Mercator Google Maps tile is 256 x 256 pixels, the usable world coordinate space is {0-256}, {0-256}.</p>  <p><a href="https://developers.google.com/maps/documentation/javascript/coordinates">https://developers.google.com/maps/documentation/javascript/coordinates</a></p> <p><b>Regarding Google Maps</b>, Google Latitude, Google Plus, Google Hangouts, Google Messages, Google Allo, Google Duo, Google Chrome, and Android Messenger, the user, via the first device, or the device itself receives second map data from a second server. The received second map data occurs responsive to user input (e.g., zoom, drag, pan, change focus, refresh or reload request, change map type, device or symbol selection, another device or user selection, change in position of first device, change in position of a second</p>

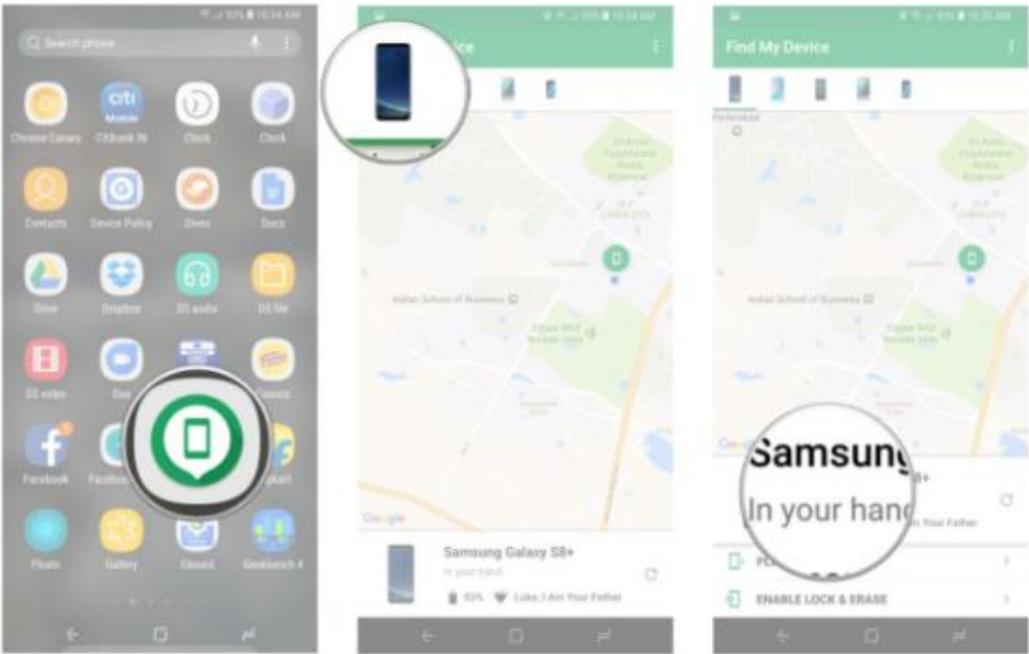
**Exhibit C for U.S. Patent No. 9,445,251 Against HTC Accused Products**

US9445251B2	HTC
	<p data-bbox="499 237 1881 412">device). Alternatively, the second map data may be received responsive to an automatic and/or pre-determined control caused by an instruction from within the first device or from the one or more second devices, e.g. a refresh. The received second map data includes an update to the first data or a replacement of the first data. The second map data may come from one of many sources and servers. This information includes locations and data relating the positions to spatial coordinates. See, e.g.:</p> <p data-bbox="499 456 1881 594">World coordinates in Google Maps are measured from the Mercator projection's origin (the northwest corner of the map at 180 degrees longitude and approximately 85 degrees latitude) and increase in the x direction towards the east (right) and increase in the y direction towards the south (down). Because the basic Mercator Google Maps tile is 256 x 256 pixels, the usable world coordinate space is {0-256}, {0-256}.</p>  <p data-bbox="499 1114 1444 1141"><a href="https://developers.google.com/maps/documentation/javascript/coordinates">https://developers.google.com/maps/documentation/javascript/coordinates</a></p> <p data-bbox="499 1222 1052 1250"><b><u>Exemplary Support for Find My Device:</u></b></p>

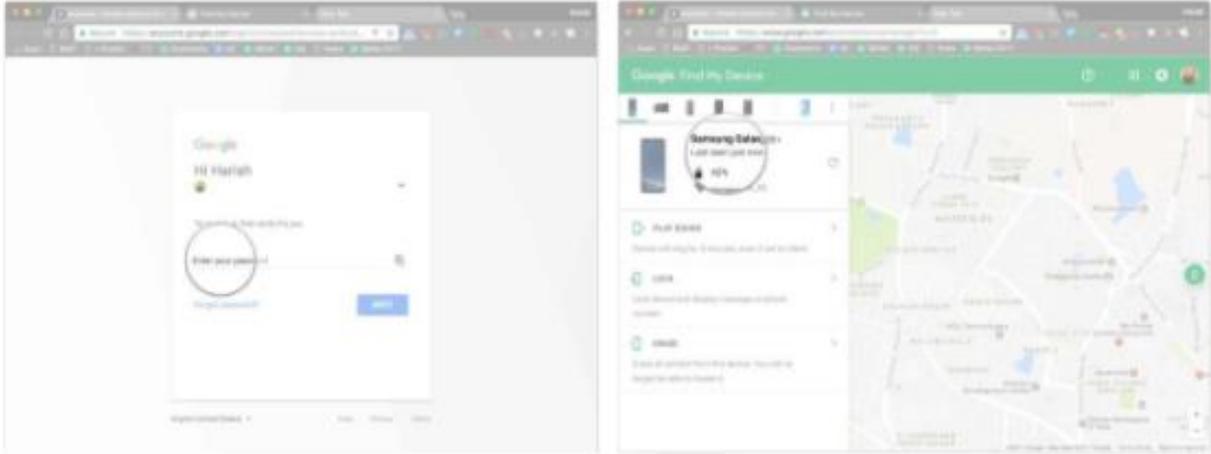
**Exhibit C for U.S. Patent No. 9,445,251 Against HTC Accused Products**

US9445251B2	HTC
	 <p><a href="https://play.google.com/store/apps/details?id=com.google.android.apps.adm&amp;hl=en">https://play.google.com/store/apps/details?id=com.google.android.apps.adm&amp;hl=en</a></p>

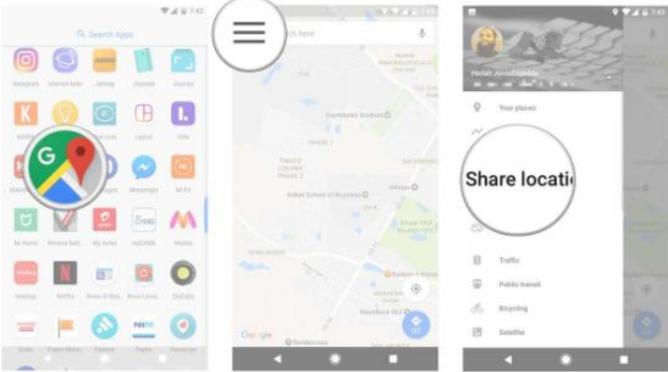
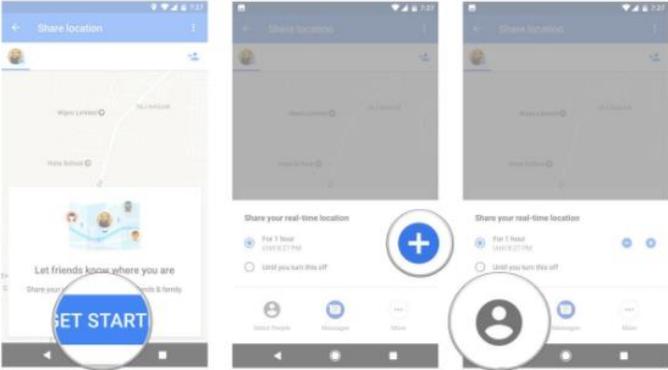
**Exhibit C for U.S. Patent No. 9,445,251 Against HTC Accused Products**

US9445251B2	HTC
	<p data-bbox="516 245 1533 337">Once you're signed in to Find My Device, you'll see a map with your current location as well as the make and model of your phone, and two options — Play Sound, and Enable Lock &amp; Erase. Hitting the latter option will allow you to start using the Lock and Erase functions.</p> <p data-bbox="516 375 1545 431">If you've signed into more than one phone, you can select a particular device by browsing the list at the top of the screen.</p> <ol data-bbox="516 480 1230 613" style="list-style-type: none"> <li>1. Open <b>Find My Device</b> from your home screen or app drawer.</li> <li>2. Select your phone from the <b>list of devices</b> at the top of the screen.</li> <li>3. See if your phone is <b>discoverable</b>.</li> </ol>  <p data-bbox="499 1325 1121 1357"><a href="https://www.androidcentral.com/find-my-device">https://www.androidcentral.com/find-my-device</a></p>

## Exhibit C for U.S. Patent No. 9,445,251 Against HTC Accused Products

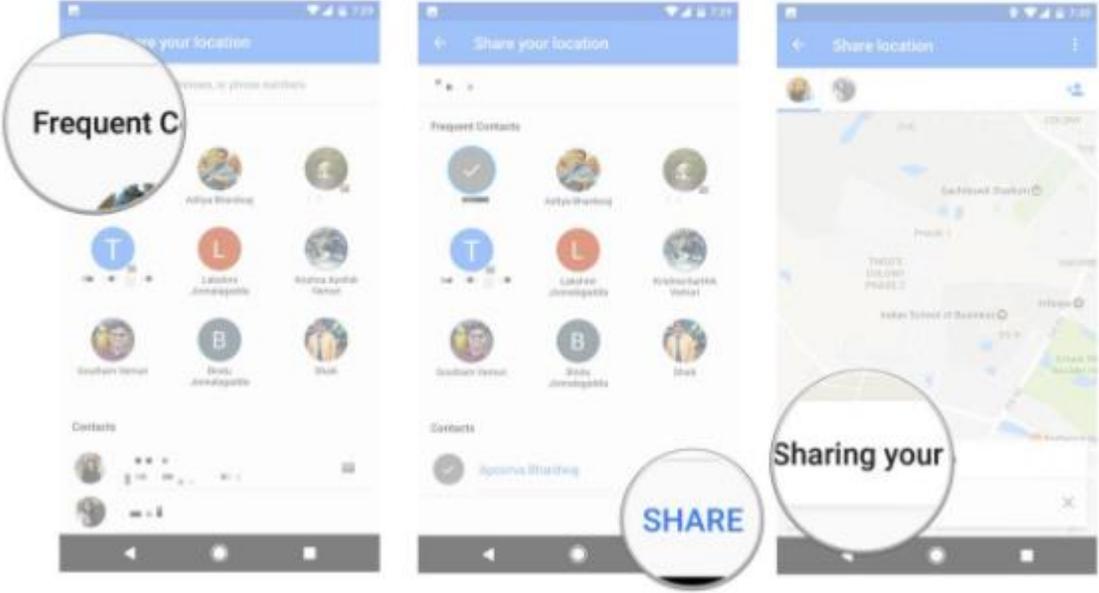
US9445251B2	HTC
	<h3 data-bbox="510 349 1650 402">How to locate your phone over the internet</h3> <p data-bbox="510 446 1661 586">If you've lost your phone, you can remotely locate it through the <a href="#">Find My Device website</a>. You'll need to sign in to the Google account that was used to set up Find My Device. It takes a few seconds, but the service should be able to track your phone. Alternatively, you can also do a Google search for "<a href="#">find my phone</a>" to locate your handset.</p> <ol data-bbox="510 646 997 800" style="list-style-type: none"><li>1. Head to the <a href="#">Find My Device website</a>.</li><li>2. Sign in to your <a href="#">Google account</a>.</li><li>3. Check if your device is visible.</li></ol> <div data-bbox="548 841 1759 1295"></div> <p data-bbox="499 1323 1121 1357"><a href="https://www.androidcentral.com/find-my-device">https://www.androidcentral.com/find-my-device</a></p>

**Exhibit C for U.S. Patent No. 9,445,251 Against HTC Accused Products**

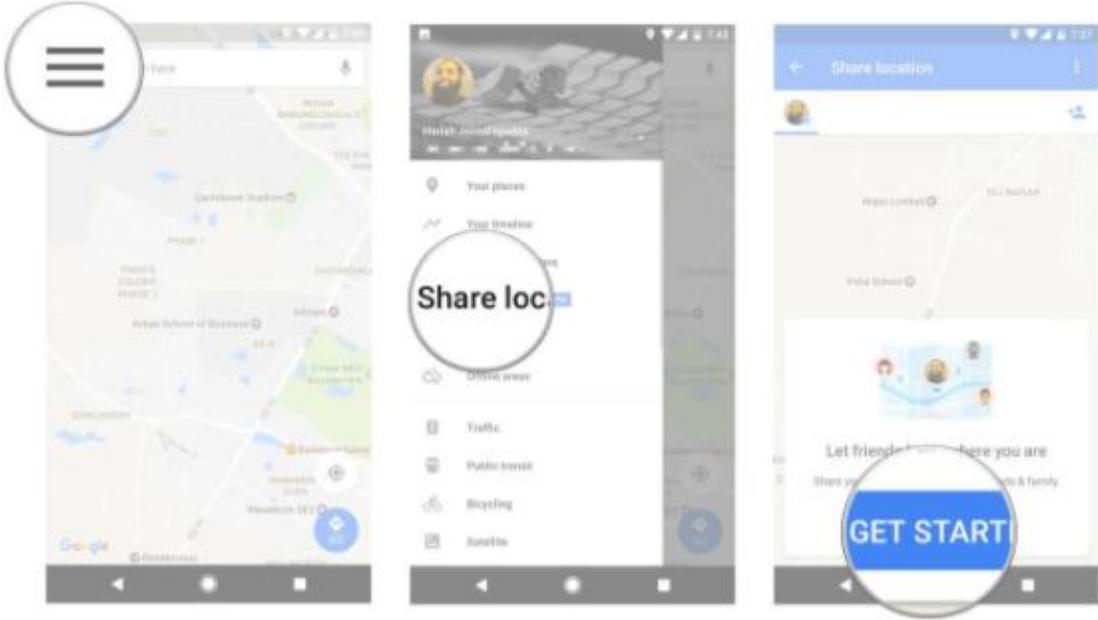
US9445251B2	HTC
	<p><b><u>Exemplary Support for Google Maps:</u></b></p> <p><b>How to share your location in Google Maps</b></p> <ol style="list-style-type: none"> <li>1. Open Google Maps from the app drawer or the home screen.</li> <li>2. Tap the hamburger menu (the three horizontal lines) on the top left corner of the screen.</li> <li>3. Select Share location.</li> </ol>  <ol style="list-style-type: none"> <li>4. Tap Get Started.</li> <li>5. Use the + icon to select a time period or select the Until you turn this off setting to share your location indefinitely.</li> <li>6. Tap Select People.</li> </ol> 



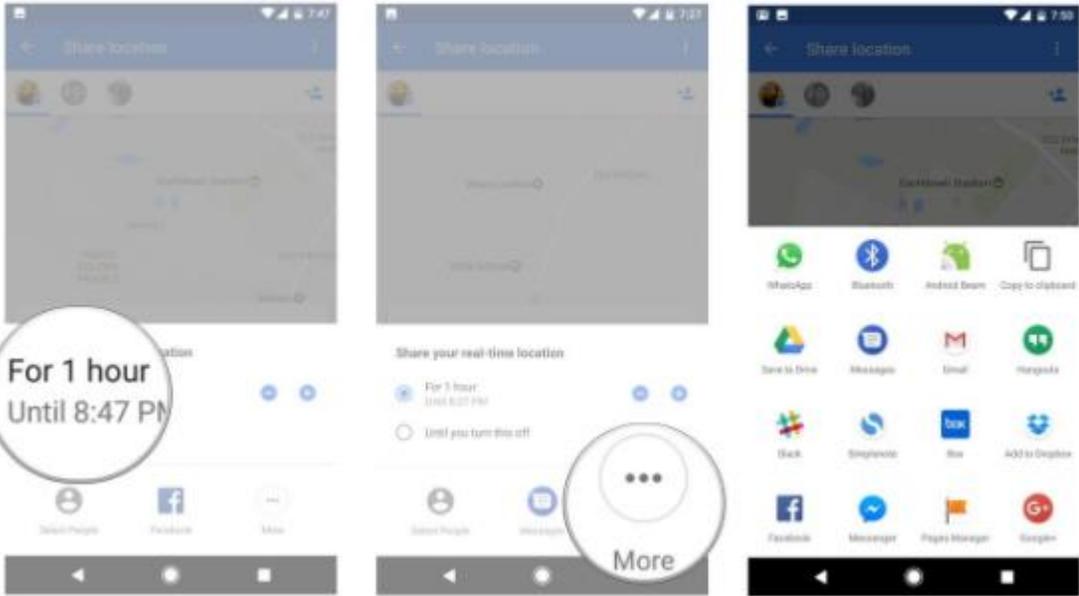
**Exhibit C for U.S. Patent No. 9,445,251 Against HTC Accused Products**

US9445251B2	HTC
	<p data-bbox="499 235 1346 267"><a href="https://www.androidcentral.com/how-share-location-google-maps">https://www.androidcentral.com/how-share-location-google-maps</a></p> <ol data-bbox="514 329 1566 505" style="list-style-type: none"><li data-bbox="514 329 1566 386">7. You'll see a list of your frequent contacts at the top, along with a full list of contacts. <b>Pick the contacts</b> by tapping their name.</li><li data-bbox="514 415 1444 443">8. Once you've selected the contacts you want to share your location to, tap <b>Share</b>.</li><li data-bbox="514 472 1409 500">9. You'll see a message saying that the selected contact can view your location.</li></ol>  <p data-bbox="499 1177 1346 1209"><a href="https://www.androidcentral.com/how-share-location-google-maps">https://www.androidcentral.com/how-share-location-google-maps</a></p>

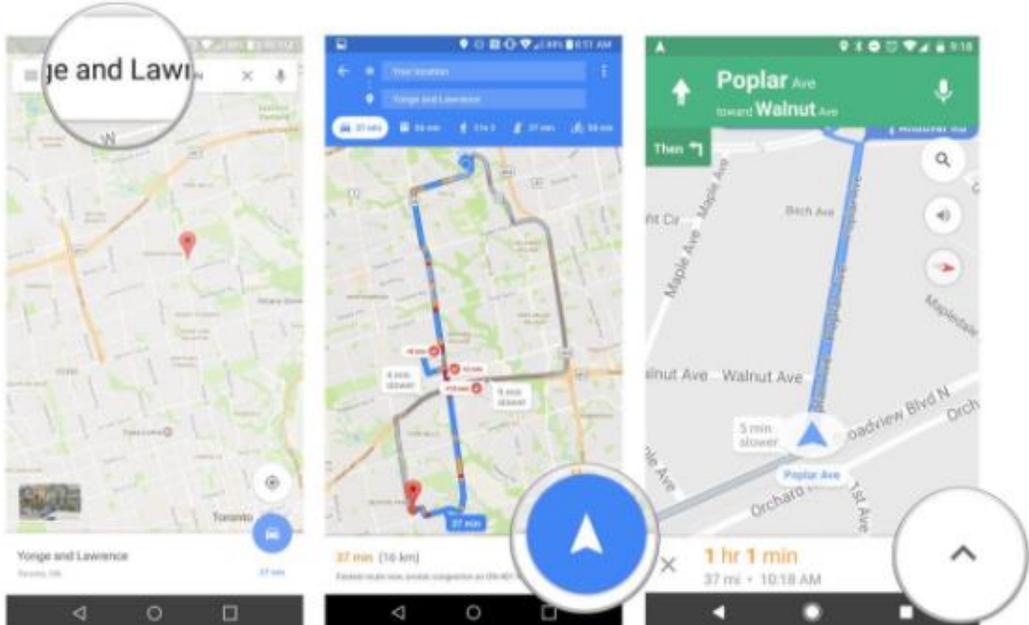
**Exhibit C for U.S. Patent No. 9,445,251 Against HTC Accused Products**

US9445251B2	HTC
	<h2 data-bbox="512 245 1243 293">How to create a shareable link</h2> <p data-bbox="512 334 1451 363">You can also create a link and use it to share your location easily. Here's how to do it:</p> <ol data-bbox="506 412 1224 553" style="list-style-type: none"><li data-bbox="506 412 1224 441">1. Tap the <b>hamburger menu</b> on the top left corner of the screen.</li><li data-bbox="506 469 787 498">2. Select <b>Share location</b>.</li><li data-bbox="506 526 726 555">3. Tap <b>Get Started</b>.</li></ol>  <p data-bbox="499 1235 1346 1265"><a href="https://www.androidcentral.com/how-share-location-google-maps">https://www.androidcentral.com/how-share-location-google-maps</a></p>

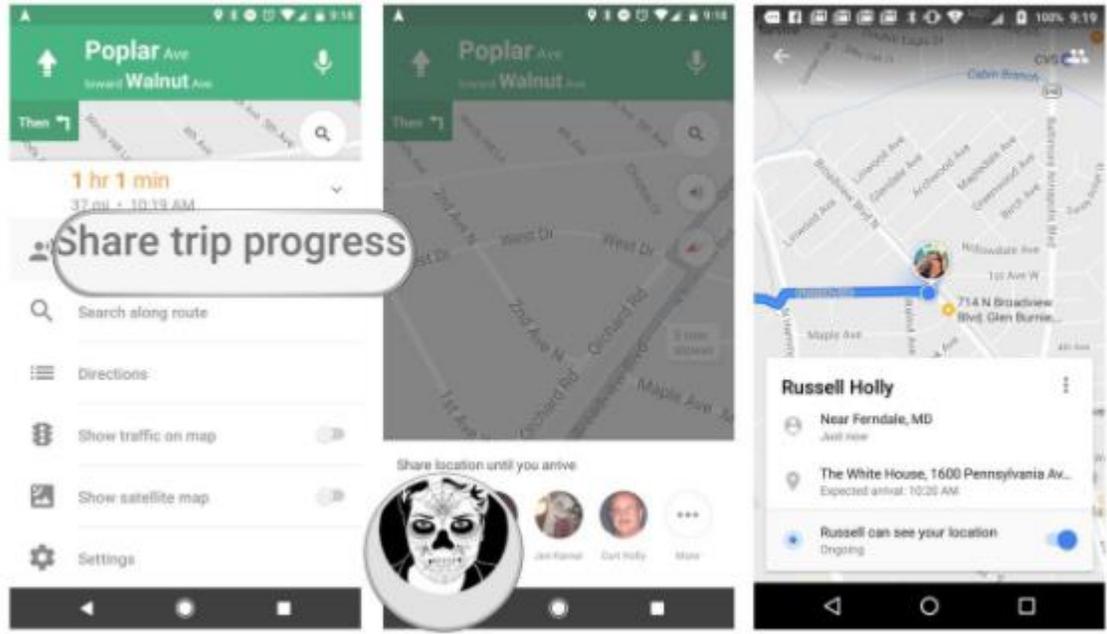
### Exhibit C for U.S. Patent No. 9,445,251 Against HTC Accused Products

US9445251B2	HTC
	<p data-bbox="512 245 1199 272">4. Select the amount of time you want to share your location.</p> <p data-bbox="512 302 667 329">5. Tap More.</p> <p data-bbox="512 358 1619 423">6. Select your app of choice to create and send a unique URL that broadcasts your current location. You can email it, send the link via Messenger, or even tweet it to the intended recipient.</p>  <p data-bbox="499 1084 1346 1117"><a href="https://www.androidcentral.com/how-share-location-google-maps">https://www.androidcentral.com/how-share-location-google-maps</a></p>

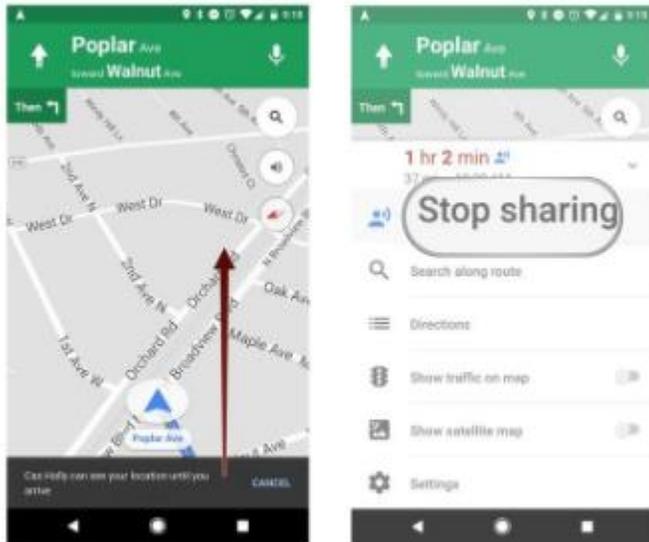
**Exhibit C for U.S. Patent No. 9,445,251 Against HTC Accused Products**

US9445251B2	HTC
	<p data-bbox="514 240 1417 337"><b>How to share your navigation directions while you walk, drive or transit</b></p> <p data-bbox="514 375 1543 467">One of the best ways to share your location with someone is during a drive, walk or transit. If you're meeting a friend or family member somewhere, or navigating towards their home, Google Maps lets you share your location with them for the duration of the trip. It's magic!</p> <ol data-bbox="514 511 1386 646" style="list-style-type: none"><li>1. In the <b>search bar</b> enter your destination.</li><li>2. Pick your navigation type (drive, transit, walk) and press the blue <b>navigate button</b>.</li><li>3. Tap the arrow next to the <b>time-to-destination</b> number at the bottom of the screen.</li></ol>  <p data-bbox="499 1328 1344 1360"><a href="https://www.androidcentral.com/how-share-location-google-maps">https://www.androidcentral.com/how-share-location-google-maps</a></p>

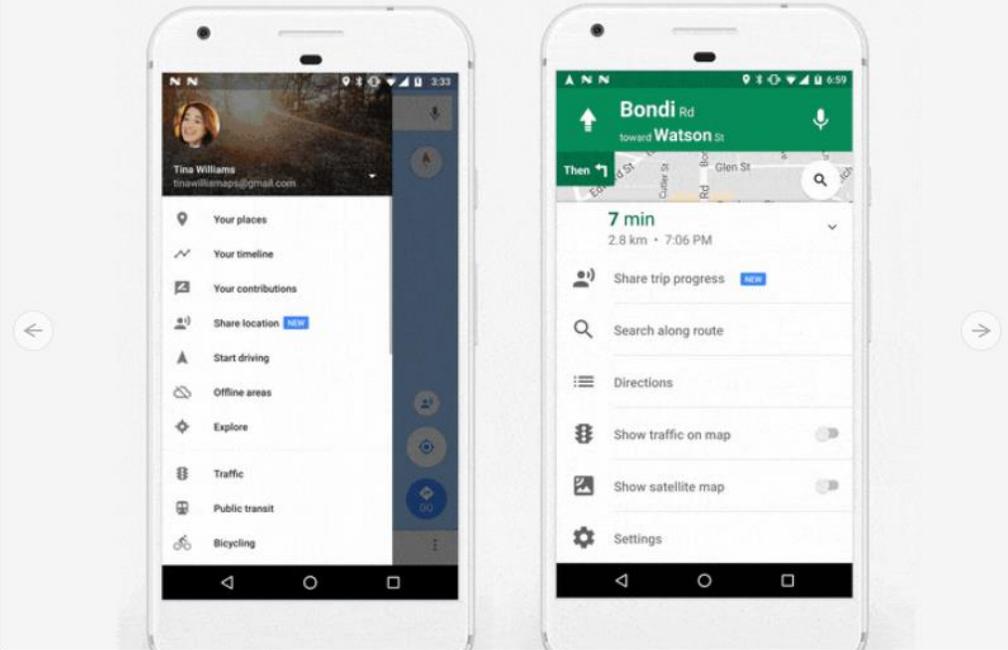
### Exhibit C for U.S. Patent No. 9,445,251 Against HTC Accused Products

US9445251B2	HTC
	<p data-bbox="514 277 823 305">4. Tap Share trip progress.</p> <p data-bbox="514 334 1136 362">5. Choose one or more contacts to share trip progress.</p>  <p data-bbox="520 1063 1333 1091">You can also stop sharing your location with someone before a trip ends.</p> <p data-bbox="520 1101 1346 1133"><a href="https://www.androidcentral.com/how-share-location-google-maps">https://www.androidcentral.com/how-share-location-google-maps</a></p>

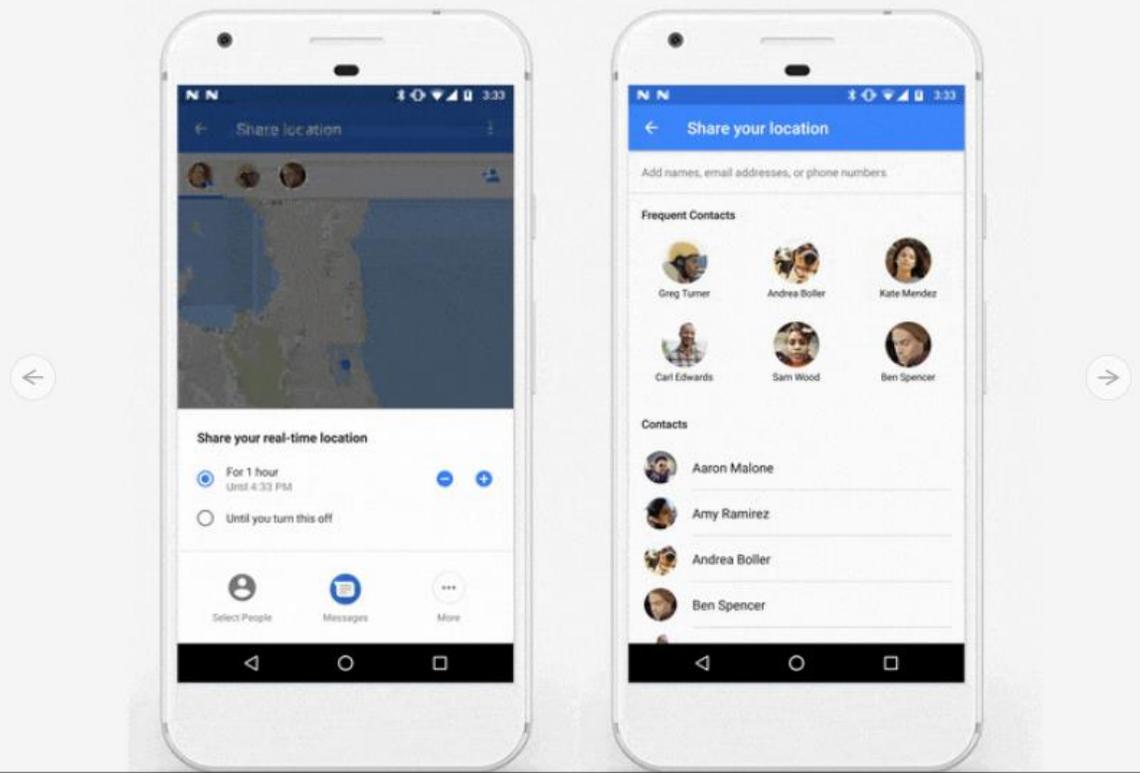
### Exhibit C for U.S. Patent No. 9,445,251 Against HTC Accused Products

US9445251B2	HTC
	<ol style="list-style-type: none"><li data-bbox="520 245 1457 272">1. Tap the arrow next to the <b>time-to-destination</b> number at the bottom of the screen.</li><li data-bbox="520 302 758 329">2. Tap <b>Stop sharing</b>.</li></ol> <div data-bbox="732 380 1381 922"></div> <p data-bbox="527 976 625 1003">That's it!</p> <p data-bbox="527 1045 1598 1073">Are you excited that location sharing is back in Google Maps? How often do you use the feature?</p> <p data-bbox="499 1084 1346 1117"><a href="https://www.androidcentral.com/how-share-location-google-maps">https://www.androidcentral.com/how-share-location-google-maps</a></p>

# Exhibit C for U.S. Patent No. 9,445,251 Against HTC Accused Products

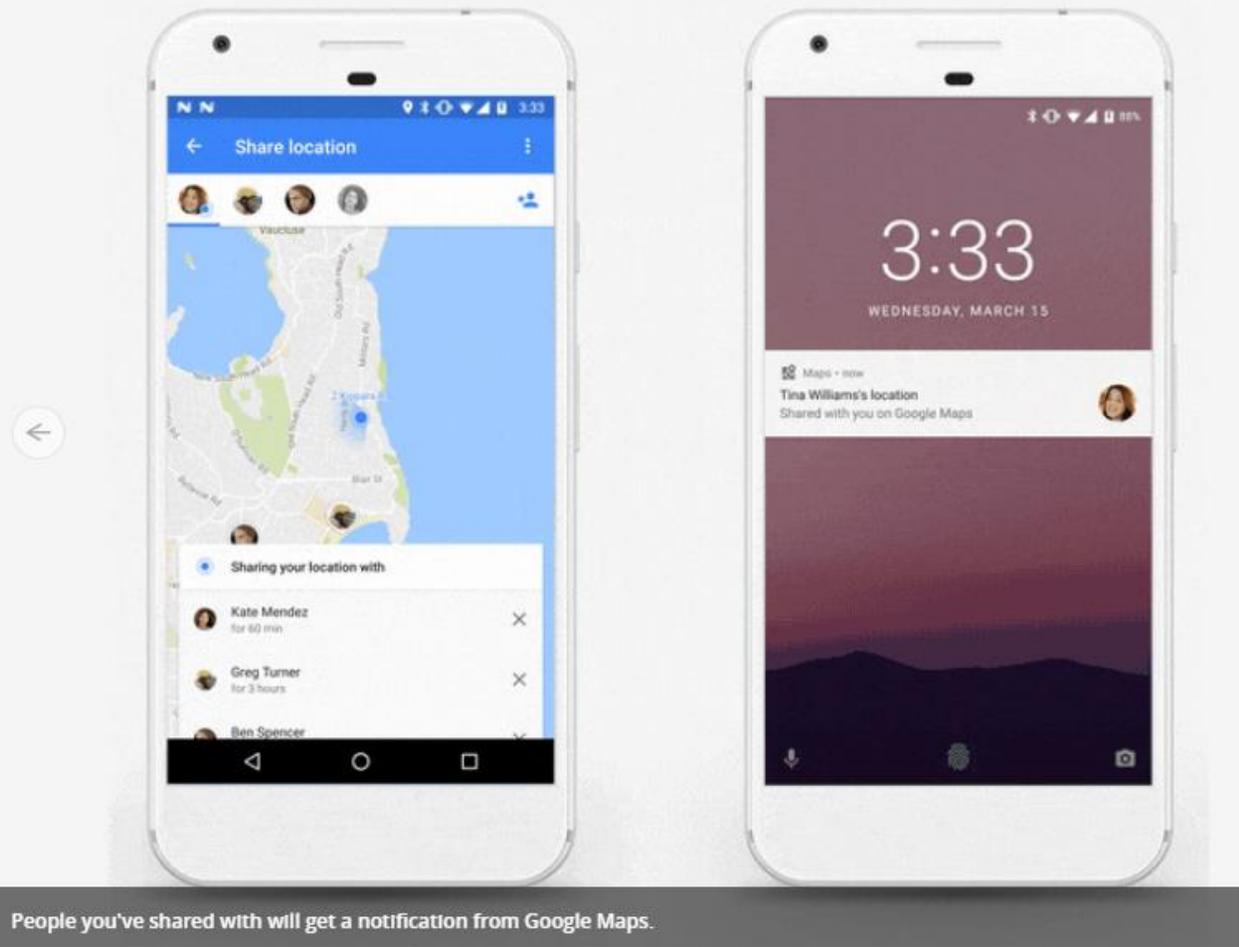
US9445251B2	HTC
	 <p data-bbox="504 893 1512 950">Ways to start location sharing: (left) from the app navigation drawer, and (right) from the menu, you can drag up while navigating somewhere.</p> <p data-bbox="504 958 1648 990"><a href="https://arstechnica.com/gadgets/2017/03/location-sharing-finally-returns-to-google-maps/">https://arstechnica.com/gadgets/2017/03/location-sharing-finally-returns-to-google-maps/</a></p>

**Exhibit C for U.S. Patent No. 9,445,251 Against HTC Accused Products**

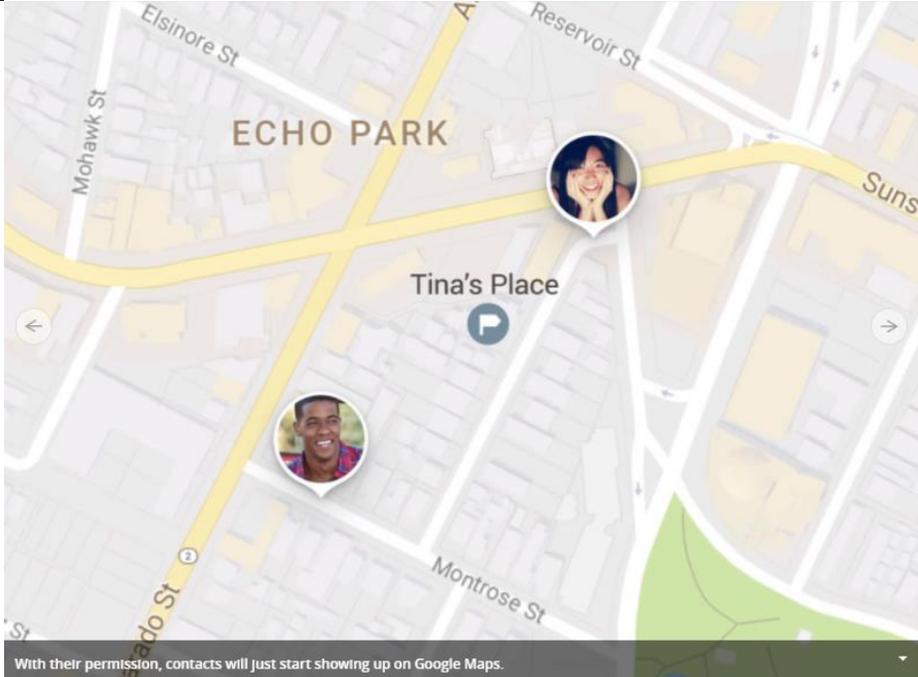
US9445251B2	HTC
	 <p data-bbox="506 1029 1646 1062">You can pick how long you want to share your location for, and with whom you want to share it.</p> <p data-bbox="506 1068 1646 1092"><a href="https://arstechnica.com/gadgets/2017/03/location-sharing-finally-returns-to-google-maps/">https://arstechnica.com/gadgets/2017/03/location-sharing-finally-returns-to-google-maps/</a></p>



**Exhibit C for U.S. Patent No. 9,445,251 Against HTC Accused Products**

US9445251B2	HTC
	 <p data-bbox="506 1146 1163 1174">People you've shared with will get a notification from Google Maps.</p> <p data-bbox="499 1190 1646 1224"><a href="https://arstechnica.com/gadgets/2017/03/location-sharing-finally-returns-to-google-maps/">https://arstechnica.com/gadgets/2017/03/location-sharing-finally-returns-to-google-maps/</a></p>

**Exhibit C for U.S. Patent No. 9,445,251 Against HTC Accused Products**

US9445251B2	HTC
	 <p data-bbox="499 885 1003 906">With their permission, contacts will just start showing up on Google Maps.</p> <p data-bbox="499 917 1648 950"><a href="https://arstechnica.com/gadgets/2017/03/location-sharing-finally-returns-to-google-maps/">https://arstechnica.com/gadgets/2017/03/location-sharing-finally-returns-to-google-maps/</a></p> <p data-bbox="499 982 829 1015"><b><u>Exemplary Screenshots:</u></b></p> <p data-bbox="499 1055 850 1088">See, e.g., 1C and 1D above.</p> <p data-bbox="499 1128 840 1161"><b><u>Exemplary Source Code:</u></b></p> <p data-bbox="499 1201 1900 1347">The above functionality is performed at least in part by the following publicly available source code and/or source code that invokes or is invoked by the following source code (or a substantially similar copy compiled and loaded onto the Accused Products by <b>HTC</b>). AGIS reserves the right to supplement these contentions with additional source code as discovery progresses and as additional source code is made available.</p>

**Exhibit C for U.S. Patent No. 9,445,251 Against HTC Accused Products**

US9445251B2	HTC
	<pre>44 * Class that sends chat message via SMS. 45 * 46 * The interface emulates a blocking sending similar to making an HTTP request. 47 * It calls the SmsManager to send a (potentially multipart) message and waits 48 * on the sent status on each part. The waiting has a timeout so it won't wait 49 * forever. Once the sent status of all parts received, the call returns. 50 * A successful sending requires success status for all parts. Otherwise, we 51 * pick the highest level of failure as the error for the whole message, which 52 * is used to determine if we need to retry the sending. 53 */ 54 public class SmsSender { 55     private static final String TAG = LogUtil.BUGLE_TAG; 56 57     public static final String EXTRA_PART_ID = "part_id"; 58 59     /* 60      * A map for pending sms messages. The key is the random request UUID. 61      */ 62     private static ConcurrentHashMap&lt;Uri, SendResult&gt; sPendingMessageMap = 63         new ConcurrentHashMap&lt;Uri, SendResult&gt;(); 64 65     private static final Random RANDOM = new Random(); 66 67     // Whether we should send multipart SMS as separate messages 68     private static Boolean sSendMultipartSmsAsSeparateMessages = null; 69</pre> <p><a href="https://android.googlesource.com/platform/packages/apps/Messaging/+nougat-mr1-release/src/com/android/messaging/sms/SmsSender.java">https://android.googlesource.com/platform/packages/apps/Messaging/+nougat-mr1-release/src/com/android/messaging/sms/SmsSender.java</a></p>

**Exhibit C for U.S. Patent No. 9,445,251 Against HTC Accused Products**

US9445251B2	HTC
	<pre>253 // Actually sending the message using SmsManager 254 private static void sendInternal(final Context context, final int subId, String dest, 255     final ArrayList&lt;String&gt; messages, final String serviceCenter, 256     final boolean requireDeliveryReport, final Uri messageUri) throws SmsException { 257     Assert.notNull(context); 258     final SmsManager smsManager = PhoneUtils.get(subId).getSmsManager(); 259     final int messageCount = messages.size(); 260     final ArrayList&lt;PendingIntent&gt; deliveryIntents = new ArrayList&lt;PendingIntent&gt;(messageCount); 261     final ArrayList&lt;PendingIntent&gt; sentIntents = new ArrayList&lt;PendingIntent&gt;(messageCount); 262     for (int i = 0; i &lt; messageCount; i++) { 263         // Make pending intents different for each message part 264         final int partId = (messageCount &lt;= 1 ? 0 : i + 1); 265         if (requireDeliveryReport &amp;&amp; (i == (messageCount - 1))) { 266             // TODO we only care about the delivery status of the last part 267             // Shall we have better tracking of delivery status of all parts? 268             deliveryIntents.add(PendingIntent.getBroadcast( 269                 context, 270                 partId, 271                 getSendStatusIntent(context, SendStatusReceiver.MESSAGE_DELIVERED_ACTION, 272                     messageUri, partId, subId), 273                 0/*flag*/)); 274         } else { 275             deliveryIntents.add(null); 276         } 277         sentIntents.add(PendingIntent.getBroadcast( 278             context, 279             partId, 280             getSendStatusIntent(context, SendStatusReceiver.MESSAGE_SENT_ACTION, 281                 messageUri, partId, subId), 282             0/*flag*/)); 283     } 284     if (sSendMultipartSmsAsSeparateMessages == null) { 285         sSendMultipartSmsAsSeparateMessages = MmsConfig.get(subId) 286             .getSendMultipartSmsAsSeparateMessages(); 287     }</pre> <p><a href="https://android.googlesource.com/platform/packages/apps/Messaging/+nougat-mr1-release/src/com/android/messaging/sms/SmsSender.java">https://android.googlesource.com/platform/packages/apps/Messaging/+nougat-mr1-release/src/com/android/messaging/sms/SmsSender.java</a></p>

**Exhibit C for U.S. Patent No. 9,445,251 Against HTC Accused Products**

US9445251B2	HTC
	<pre>288     try { 289         if (sSendMultipartSmsAsSeparateMessages) { 290             // If multipart sms is not supported, send them as separate messages 291             for (int i = 0; i &lt; messageCount; i++) { 292                 smsManager.sendTextMessage(dest, 293                     serviceCenter, 294                     messages.get(i), 295                     sentIntents.get(i), 296                     deliveryIntents.get(i)); 297             } 298         } else { 299             smsManager.sendMultipartTextMessage( 300                 dest, serviceCenter, messages, sentIntents, deliveryIntents); 301         } 302     } catch (final Exception e) { 303         throw new SmsException("SmsSender: caught exception in sending " + e); 304     } 305 }</pre> <p><a href="https://android.googlesource.com/platform/packages/apps/Messaging+/nougat-mr1-release/src/com/android/messaging/sms/SmsSender.java">https://android.googlesource.com/platform/packages/apps/Messaging+/nougat-mr1-release/src/com/android/messaging/sms/SmsSender.java</a></p> <pre>56  * Class that receives incoming SMS messages through android.provider.Telephony.SMS_RECEIVED 57  * 58  * This class serves two purposes: 59  * - Process phone verification SMS messages 60  * - Handle SMS messages when the user has enabled us to be the default SMS app (Pre-KLP) 61  */ 62  public final class SmsReceiver extends BroadcastReceiver { 63      private static final String TAG = LogUtil.BUGLE_TAG; 64 65      private static ArrayList&lt;Pattern&gt; sIgnoreSmsPatterns; 66</pre> <p><a href="https://android.googlesource.com/platform/packages/apps/Messaging+/nougat-mr1-release/src/com/android/messaging/receiver/SmsReceiver.java">https://android.googlesource.com/platform/packages/apps/Messaging+/nougat-mr1-release/src/com/android/messaging/receiver/SmsReceiver.java</a></p>

**Exhibit C for U.S. Patent No. 9,445,251 Against HTC Accused Products**

US9445251B2	HTC
	<pre>200     public static void deliverSmsMessages(final Context context, final int subId, 201         final int errorCode, final android.telephony.SmsMessage[] messages) { 202         final ContentValues messageValues = 203             MmsUtils.parseReceivedSmsMessage(context, messages, errorCode); 204 205         LogUtil.v(TAG, "SmsReceiver.deliverSmsMessages"); 206 207         final long nowInMillis = System.currentTimeMillis(); 208         final long receivedTimestampMs = MmsUtils.getMessageDate(messages[0], nowInMillis); 209 210         messageValues.put(Sms.Inbox.DATE, receivedTimestampMs); 211         // Default to unread and unseen for us but ReceiveSmsMessageAction will override 212         // seen for the telephony db. 213         messageValues.put(Sms.Inbox.READ, 0); 214         messageValues.put(Sms.Inbox.SEEN, 0); 215         if (OsUtil.isAtLeastL_MR1()) { 216             messageValues.put(Sms.SUBSCRIPTION_ID, subId); 217         } 218 219         if (messages[0].getMessageClass() == android.telephony.SmsMessage.MessageClass.CLASS_0    220             DebugUtils.debugClassZeroSmsEnabled()) { 221             Factory.get().getUIIntents().launchClassZeroActivity(context, messageValues); 222         } else { 223             final ReceiveSmsMessageAction action = new ReceiveSmsMessageAction(messageValues); 224             action.start(); 225         } 226     }</pre> <p><a href="https://android.googlesource.com/platform/packages/apps/Messaging/+nougat-mr1-release/src/com/android/messaging/receiver/SmsReceiver.java">https://android.googlesource.com/platform/packages/apps/Messaging/+nougat-mr1-release/src/com/android/messaging/receiver/SmsReceiver.java</a></p>

**Exhibit C for U.S. Patent No. 9,445,251 Against HTC Accused Products**

US9445251B2	HTC
	<pre>228     @Override 229     public void onReceive(final Context context, final Intent intent) { 230         LogUtil.v(TAG, "SmsReceiver.onReceive " + intent); 231         // On KLP+ we only take delivery of SMS messages in SmsDeliverReceiver. 232         if (PhoneUtils.getDefault().isSmsEnabled()) { 233             final String action = intent.getAction(); 234             if (OsUtil.isSecondaryUser() &amp;&amp; 235                 (Telephony.Sms.Intents.SMS_RECEIVED_ACTION.equals(action)    236                  // TODO: update this with the actual constant from Telephony 237                  "android.provider.Telephony.MMS_DOWNLOADED".equals(action))) { 238                 postNewMessageSecondaryUserNotification(); 239             } else if (!OsUtil.isAtLeastKLP()) { 240                 deliverSmsIntent(context, intent); 241             } 242         } 243     }</pre> <p><a href="https://android.googlesource.com/platform/packages/apps/Messaging/+nougat-mr1-release/src/com/android/messaging/receiver/SmsReceiver.java">https://android.googlesource.com/platform/packages/apps/Messaging/+nougat-mr1-release/src/com/android/messaging/receiver/SmsReceiver.java</a></p>

**Exhibit C for U.S. Patent No. 9,445,251 Against HTC Accused Products**

US9445251B2	HTC
	<pre>52 * Class that sends chat message via MMS. 53 * 54 * The interface emulates a blocking send similar to making an HTTP request. 55 */ 56 public class MmsSender { 57     private static final String TAG = LogUtil.BUGLE_TAG; 58 59     /** 60      * Send an MMS message. 61      * 62      * @param context Context 63      * @param messageUri The unique URI of the message for identifying it during sending 64      * @param sendReq The SendReq PDU of the message 65      * @throws MmsFailureException 66      */ 67     public static void sendMms(final Context context, final int subId, final Uri messageUri, 68         final SendReq sendReq, final Bundle sentIntentExtras) throws MmsFailureException { 69         sendMms(context, 70             subId, 71             messageUri, 72             null /* locationUri */, 73             sendReq, 74             true /* responseImportant */, 75             sentIntentExtras); 76     }</pre> <p><a href="https://android.googlesource.com/platform/packages/apps/Messaging/+nougat-mr1-release/src/com/android/messaging/sms/MmsSender.java">https://android.googlesource.com/platform/packages/apps/Messaging/+nougat-mr1-release/src/com/android/messaging/sms/MmsSender.java</a></p>



**Exhibit C for U.S. Patent No. 9,445,251 Against HTC Accused Products**

US9445251B2	HTC
	<pre>240      * Download an MMS message. 241      * 242      * @param context Context 243      * @param contentLocation The url of the MMS message 244      * @throws MmsFailureException 245      * @throws InvalidHeaderValueException 246      */ 247      public static void downloadMms(final Context context, final int subId, 248          final String contentLocation, Bundle extras) throws MmsFailureException, 249          InvalidHeaderValueException { 250          final Uri requestUri = Uri.parse(contentLocation); 251          final Uri contentUri = MmsFileProvider.buildRawMmsUri(); 252 253          final Intent downloadedIntent = new Intent(SendStatusReceiver.MMS_DOWNLOADED_ACTION, 254              requestUri, 255              context, 256              SendStatusReceiver.class); 257          downloadedIntent.putExtra(SendMessageAction.EXTRA_CONTENT_URI, contentUri); 258          if (extras != null) { 259              downloadedIntent.putExtras(extras); 260          } 261          final PendingIntent downloadedPendingIntent = PendingIntent.getBroadcast( 262              context, 263              0 /*request code*/, 264              downloadedIntent, 265              PendingIntent.FLAG_UPDATE_CURRENT); 266 267          MmsManager.downloadMultimediaMessage(subId, context, contentLocation, contentUri, 268              downloadedPendingIntent); 269      }</pre> <p><a href="https://android.googlesource.com/platform/packages/apps/Messaging/+nougat-mr1-release/src/com/android/messaging/sms/MmsSender.java">https://android.googlesource.com/platform/packages/apps/Messaging/+nougat-mr1-release/src/com/android/messaging/sms/MmsSender.java</a></p>

**Exhibit C for U.S. Patent No. 9,445,251 Against HTC Accused Products**

US9445251B2	HTC
	<pre> 97      * Execute an MMS HTTP request, either a POST (sending) or a GET (downloading) 98      * 99      * @param urlString The request URL, for sending it is usually the MMSC, and for downloading 100     *           it is the message URL 101     * @param pdu For POST (sending) only, the PDU to send 102     * @param method HTTP method, POST for sending and GET for downloading 103     * @param isProxySet Is there a proxy for the MMSC 104     * @param proxyHost The proxy host 105     * @param proxyPort The proxy port 106     * @param mmsConfig The MMS config to use 107     * @param userAgent The user agent header value 108     * @param uaProfUrl The UA Prof URL header value 109     * @return The HTTP response body 110     * @throws MmsHttpException For any failures 111     */ 112     public byte[] execute(String urlString, byte[] pdu, String method, boolean isProxySet, 113                          String proxyHost, int proxyPort, Bundle mmsConfig, String userAgent, String uaProfUrl) 114         throws MmsHttpException { 115         Log.d(MmsService.TAG, "HTTP: " + method + " " + Utils.redactUrlForNonVerbose(urlString) 116             + (isProxySet ? (" proxy=" + proxyHost + ":" + proxyPort) : "") 117             + ", PDU size=" + (pdu != null ? pdu.length : 0)); 118         checkMethod(method); 119         HttpURLConnection connection = null; 120         try { 121             Proxy proxy = Proxy.NO_PROXY; 122             if (isProxySet) { 123                 proxy = new Proxy(Proxy.Type.HTTP, new InetSocketAddress(proxyHost, proxyPort)); 124             } 125             final URL url = new URL(urlString); 126             // Now get the connection 127             connection = (HttpURLConnection) url.openConnection(proxy); 128             connection.setDoInput(true); 129             connection.setConnectTimeout( 130                 mmsConfig.getInt(CarrierConfigValuesLoader.CONFIG_HTTP_SOCKET_TIMEOUT, 131                                 CarrierConfigValuesLoader.CONFIG_HTTP_SOCKET_TIMEOUT_DEFAULT)); </pre> <p data-bbox="499 1222 1583 1289"><a href="https://android.googlesource.com/platform/packages/apps/Messaging+/nougat-mr1-release/src/android/support/v7/mms/MmsHttpClient.java">https://android.googlesource.com/platform/packages/apps/Messaging+/nougat-mr1-release/src/android/support/v7/mms/MmsHttpClient.java</a></p>

**Exhibit C for U.S. Patent No. 9,445,251 Against HTC Accused Products**

US9445251B2	HTC
	<pre> 132 // ----- COMMON HEADERS ----- 133 // Header: Accept 134 connection.setRequestProperty(HEADER_ACCEPT, HEADER_VALUE_ACCEPT); 135 // Header: Accept-Language 136 connection.setRequestProperty( 137     HEADER_ACCEPT_LANGUAGE, getCurrentAcceptLanguage(Locale.getDefault())); 138 // Header: User-Agent 139 Log.i(MmsService.TAG, "HTTP: User-Agent=" + userAgent); 140 connection.setRequestProperty(HEADER_USER_AGENT, userAgent); 141 // Header: x-wap-profile 142 final String uaProfUrlTagName = mmsConfig.getString( 143     CarrierConfigValuesLoader.CONFIG_UA_PROF_TAG_NAME, 144     CarrierConfigValuesLoader.CONFIG_UA_PROF_TAG_NAME_DEFAULT); 145 if (uaProfUrl != null) { 146     Log.i(MmsService.TAG, "HTTP: UaProfUrl=" + uaProfUrl); 147     connection.setRequestProperty(uaProfUrlTagName, uaProfUrl); 148 } 149 // Add extra headers specified by mms_config.xml's httpparams 150 addExtraHeaders(connection, mmsConfig); 151 // Different stuff for GET and POST 152 if (METHOD_POST.equals(method)) { 153     if (pdu == null    pdu.length &lt; 1) { 154         Log.e(MmsService.TAG, "HTTP: empty pdu"); 155         throw new MmsHttpException(0/*statusCode*/, "Sending empty PDU"); 156     } 157     connection.setDoOutput(true); 158     connection.setRequestMethod(METHOD_POST); 159     if (mmsConfig.getBoolean( 160         CarrierConfigValuesLoader.CONFIG_SUPPORT_HTTP_CHARSET_HEADER, 161         CarrierConfigValuesLoader.CONFIG_SUPPORT_HTTP_CHARSET_HEADER_DEFAULT)) { 162         connection.setRequestProperty(HEADER_CONTENT_TYPE, 163             HEADER_VALUE_CONTENT_TYPE_WITH_CHARSET); 164     } else { 165         connection.setRequestProperty(HEADER_CONTENT_TYPE, 166             HEADER_VALUE_CONTENT_TYPE_WITHOUT_CHARSET); </pre> <p><a href="https://android.globalsources.com/platform/packages/apps/Messaging+/nougat-mr1-release/src/android/support/v7/mms/MmsHttpClient.java">https://android.globalsources.com/platform/packages/apps/Messaging+/nougat-mr1-release/src/android/support/v7/mms/MmsHttpClient.java</a></p>

**Exhibit C for U.S. Patent No. 9,445,251 Against HTC Accused Products**

US9445251B2	HTC
	<pre> 167         } 168         if (Log.isLoggable(MmsService.TAG, Log.VERBOSE)) { 169             logHttpHeaders(connection.getRequestProperties()); 170         } 171         connection.setFixedLengthStreamingMode(pdu.length); 172         // Sending request body 173         final OutputStream out = 174             new BufferedOutputStream(connection.getOutputStream()); 175         out.write(pdu); 176         out.flush(); 177         out.close(); 178     } else if (METHOD_GET.equals(method)) { 179         if (Log.isLoggable(MmsService.TAG, Log.VERBOSE)) { 180             logHttpHeaders(connection.getRequestProperties()); 181         } 182         connection.setRequestMethod(METHOD_GET); 183     } 184     // Get response 185     final int responseCode = connection.getResponseCode(); 186     final String responseMessage = connection.getResponseMessage(); 187     Log.d(MmsService.TAG, "HTTP: " + responseCode + " " + responseMessage); 188     if (Log.isLoggable(MmsService.TAG, Log.VERBOSE)) { 189         logHttpHeaders(connection.getHeaderFields()); 190     } 191     if (responseCode / 100 != 2) { 192         throw new MmsHttpException(responseCode, responseMessage); 193     } 194     final InputStream in = new BufferedInputStream(connection.getInputStream()); 195     final ByteArrayOutputStream byteOut = new ByteArrayOutputStream(); 196     final byte[] buf = new byte[4096]; 197     int count = 0; 198     while ((count = in.read(buf)) &gt; 0) { 199         byteOut.write(buf, 0, count); 200     } 201     in.close(); 202     final byte[] responseBody = byteOut.toByteArray(); 203     Log.d(MmsService.TAG, "HTTP: response size=" 204         + (responseBody != null ? responseBody.length : 0)); 205     return responseBody; </pre> <p><a href="https://android.googlesource.com/platform/packages/apps/Messaging+/nougat-mr1-release/src/android/support/v7/mms/MmsHttpClient.java">https://android.googlesource.com/platform/packages/apps/Messaging+/nougat-mr1-release/src/android/support/v7/mms/MmsHttpClient.java</a></p>

**Exhibit C for U.S. Patent No. 9,445,251 Against HTC Accused Products**

US9445251B2	HTC
	<pre>38  * Request to send an MMS 39  */ 40  class SendRequest extends MmsRequest { 41      // Max send response PDU size in bytes (exceeding this may cause problem with 42      // system intent delivery). 43      private static final int MAX_SEND_RESPONSE_SIZE = 1000 * 1024; 44 45      private byte[] mPduData; 46 47      SendRequest(final String locationUri, final Uri pduUri, final PendingIntent sentIntent) { 48          super(locationUri, pduUri, sentIntent); 49      } 50 51      @Override 52      protected boolean loadRequest(final Context context, final Bundle mmsConfig) { 53          mPduData = readPduFromContentUri( 54              context, 55              mPduUri, 56              mmsConfig.getInt( 57                  CarrierConfigValuesLoader.CONFIG_MAX_MESSAGE_SIZE, 58                  CarrierConfigValuesLoader.CONFIG_MAX_MESSAGE_SIZE_DEFAULT)); 59          return (mPduData != null); 60      } 61 62      @Override 63      protected boolean transferResponse(final Context context, final Intent fillIn, 64          final byte[] response) { 65          // SendConf pdus are always small and can be included in the intent 66          if (response != null &amp;&amp; fillIn != null) { 67              if (response.length &gt; MAX_SEND_RESPONSE_SIZE) { 68                  // If the response PDU is too large, it won't be able to fit in 69                  // the PendingIntent to be transferred via system IPC. 70                  return false; 71              } 72              fillIn.putExtra(SmsManager.EXTRA_MMS_DATA, response); 73          } 74          return true; 75      } }</pre>

**Exhibit C for U.S. Patent No. 9,445,251 Against HTC Accused Products**

US9445251B2	HTC
	<p><a href="https://android.googlesource.com/platform/packages/apps/Messaging+/nougat-mr1-release/src/android/support/v7/mms/SendRequest.java">https://android.googlesource.com/platform/packages/apps/Messaging+/nougat-mr1-release/src/android/support/v7/mms/SendRequest.java</a></p> <pre data-bbox="520 358 1724 399">public static LocationRequest create ()</pre> <p data-bbox="512 423 1016 451">Create a location request with default parameters.</p> <p data-bbox="512 480 1625 542">Default parameters are for a block accuracy, slowly updated location. It can then be adjusted as required by the applications before passing to the <a href="#">FusedLocationProviderApi</a>.</p> <p data-bbox="533 565 617 589"><b>Returns</b></p> <ul data-bbox="541 610 800 634" style="list-style-type: none"> <li>• a new location request</li> </ul> <p><a href="https://developers.google.com/android/reference/com/google/android/gms/location/LocationRequest">https://developers.google.com/android/reference/com/google/android/gms/location/LocationRequest</a></p>

**Exhibit C for U.S. Patent No. 9,445,251 Against HTC Accused Products**

US9445251B2	HTC
	<p><b>public static final int PRIORITY_BALANCED_POWER_ACCURACY</b></p> <p>Used with <code>setPriority(int)</code> to request "block" level accuracy.</p> <p>Block level accuracy is considered to be about 100 meter accuracy. Using a coarse accuracy such as this often consumes less power.</p> <p>Constant Value: 102</p> <p><b>public static final int PRIORITY_HIGH_ACCURACY</b></p> <p>Used with <code>setPriority(int)</code> to request the most accurate locations available.</p> <p>This will return the finest location available.</p> <p>Constant Value: 100</p> <p><b>public static final int PRIORITY_LOW_POWER</b></p> <p>Used with <code>setPriority(int)</code> to request "city" level accuracy.</p> <p>City level accuracy is considered to be about 10km accuracy. Using a coarse accuracy such as this often consumes less power.</p> <p>Constant Value: 104</p> <p><a href="https://developers.google.com/android/reference/com/google/android/gms/location/LocationRequest">https://developers.google.com/android/reference/com/google/android/gms/location/LocationRequest</a></p>

**Exhibit C for U.S. Patent No. 9,445,251 Against HTC Accused Products**

US9445251B2	HTC
	<p data-bbox="520 250 1738 289"><code>public Task&lt;Location&gt; getLastLocation ()</code></p> <p data-bbox="520 318 1094 341">Returns the best most recent location currently available.</p> <p data-bbox="520 375 1682 431">If a location is not available, which should happen very rarely, null will be returned. The best accuracy available while respecting the location permissions will be returned.</p> <p data-bbox="520 466 1724 522">This method provides a simplified way to get location. It is particularly well suited for applications that do not require an accurate location and that do not want to maintain extra logic for location updates.</p> <p data-bbox="520 581 1738 620"><code>public Task&lt;LocationAvailability&gt; getLocationAvailability ()</code></p> <p data-bbox="520 649 1677 706">Returns the availability of location data. When <code>isLocationAvailable()</code> returns true, then the location returned by <code>getLastLocation()</code> will be reasonably up to date within the hints specified by the active <code>LocationRequest</code> s.</p> <p data-bbox="520 740 1457 763">If the client isn't connected to Google Play services and the request times out, null is returned.</p> <p data-bbox="520 797 1661 854">Note it's always possible for <code>getLastLocation()</code> to return null even when this method returns true (e.g. location settings were disabled between calls).</p> <p data-bbox="520 867 1906 933"><a href="https://developers.google.com/android/reference/com/google/android/gms/location/FusedLocationProviderClient">https://developers.google.com/android/reference/com/google/android/gms/location/FusedLocationProviderClient</a></p>



**Exhibit C for U.S. Patent No. 9,445,251 Against HTC Accused Products**

US9445251B2	HTC						
	<p data-bbox="516 245 1734 326"> <code>public Task&lt;Void&gt; requestLocationUpdates (LocationRequest request, LocationCallback callback, Looper looper)</code> </p> <p data-bbox="516 354 1255 378">Requests location updates with a callback on the specified Looper thread.</p> <p data-bbox="516 412 1671 472">This method is suited for the foreground use cases. For background use cases, the <code>PendingIntent</code> version of the method is recommended, see <code>requestLocationUpdates(LocationRequest, PendingIntent)</code>.</p> <p data-bbox="516 506 1356 531">Any previous LocationRequests registered on this LocationListener will be replaced.</p> <p data-bbox="516 565 1671 656">This call will keep the Google Play services connection active, so make sure to call <code>removeLocationUpdates(LocationCallback)</code> when you no longer need it, otherwise you lose the benefits of the automatic connection management.</p> <p data-bbox="516 690 1728 714">Callbacks for <code>LocationCallback</code> will be made on the specified thread, which must already be a prepared looper thread.</p> <p data-bbox="531 738 653 763"><b>Parameters</b></p> <table border="1" data-bbox="516 792 1734 1008"> <tbody> <tr> <td data-bbox="516 792 617 857"><b>request</b></td> <td data-bbox="617 792 1734 857">The location request for the updates.</td> </tr> <tr> <td data-bbox="516 857 617 922"><b>callback</b></td> <td data-bbox="617 857 1734 922">The callback for the location updates.</td> </tr> <tr> <td data-bbox="516 922 617 1008"><b>looper</b></td> <td data-bbox="617 922 1734 1008">The Looper object whose message queue will be used to implement the callback mechanism, or null to make callbacks on the calling thread.</td> </tr> </tbody> </table> <p data-bbox="504 1024 1902 1089"> <a href="https://developers.google.com/android/reference/com/google/android/gms/location/FusedLocationProviderClient">https://developers.google.com/android/reference/com/google/android/gms/location/FusedLocationProviderClient</a> </p>	<b>request</b>	The location request for the updates.	<b>callback</b>	The callback for the location updates.	<b>looper</b>	The Looper object whose message queue will be used to implement the callback mechanism, or null to make callbacks on the calling thread.
<b>request</b>	The location request for the updates.						
<b>callback</b>	The callback for the location updates.						
<b>looper</b>	The Looper object whose message queue will be used to implement the callback mechanism, or null to make callbacks on the calling thread.						

## Exhibit C for U.S. Patent No. 9,445,251 Against HTC Accused Products

US9445251B2	HTC				
	<pre>public Task&lt;Void&gt; requestLocationUpdates (LocationRequest request, PendingIntent callbackIntent)</pre> <p>Requests location updates with a callback on the specified PendingIntent.</p> <p>This method is suited for the background use cases, more specifically for receiving location updates, even when the app has been killed by the system. In order to do so, use a <code>PendingIntent</code> for a started service. For foreground use cases, the <code>LocationCallback</code> version of the method is recommended, see <code>requestLocationUpdates(LocationRequest, LocationCallback, Looper)</code>.</p> <p>Any previously registered requests that have the same PendingIntent (as defined by <code>equals(Object)</code>) will be replaced by this request.</p> <p>Both <code>LocationResult</code> and <code>LocationAvailability</code> are sent to the given PendingIntent. You can extract data from an Intent using <code>hasResult(Intent)</code>, <code>extractResult(Intent)</code>, <code>hasLocationAvailability(Intent)</code>, and <code>extractLocationAvailability(Intent)</code>.</p> <p><b>Parameters</b></p> <table border="1"> <tr> <td><code>request</code></td> <td>The location request for the updates.</td> </tr> <tr> <td><code>callbackIntent</code></td> <td>A pending intent to be sent for each location update.</td> </tr> </table> <p><b>Returns</b></p> <ul style="list-style-type: none"> <li>a Task for the call, check <code>isSuccessful()</code> to determine if it was successful.</li> </ul> <p><a href="https://developers.google.com/android/reference/com/google/android/gms/location/FusedLocationProviderClient">https://developers.google.com/android/reference/com/google/android/gms/location/FusedLocationProviderClient</a></p>	<code>request</code>	The location request for the updates.	<code>callbackIntent</code>	A pending intent to be sent for each location update.
<code>request</code>	The location request for the updates.				
<code>callbackIntent</code>	A pending intent to be sent for each location update.				

## Exhibit C for U.S. Patent No. 9,445,251 Against HTC Accused Products

US9445251B2	HTC						
	<p><code>public void onLocationAvailability (LocationAvailability locationAvailability)</code></p> <p>Called when there is a change in the availability of location data.</p> <p>When <code>isLocationAvailable()</code> returns <code>false</code> you can assume that location will not be returned in <code>onLocationResult(LocationResult)</code> until something changes in the device's settings or environment. Even when <code>isLocationAvailable()</code> returns <code>true</code> the <code>onLocationResult(LocationResult)</code> may not always be called regularly, however the device location is known and both the most recently delivered location and <code>getLastLocation(GoogleApiClient)</code> will be reasonably up to date given the hints specified by the active <code>LocationRequest</code>s.</p> <p><b>Parameters</b></p> <table border="1"> <tr> <td><code>locationAvailability</code></td> <td>The current status of location availability.</td> </tr> </table> <p><code>public void onLocationResult (LocationResult result)</code></p> <p>Called when device location information is available.</p> <p>The most recent location returned by <code>getLastLocation()</code> is not guaranteed to be immediately fresh, but will be reasonably up to date given the hints specified by the active <code>LocationRequest</code>s.</p> <p><b>Parameters</b></p> <table border="1"> <tr> <td><code>result</code></td> <td>The latest location result available.</td> </tr> </table> <p><a href="https://developers.google.com/android/reference/com/google/android/gms/location/LocationCallback">https://developers.google.com/android/reference/com/google/android/gms/location/LocationCallback</a></p> <p><code>public abstract void onLocationChanged (Location location)</code></p> <p>Called when the location has changed.</p> <p><b>Parameters</b></p> <table border="1"> <tr> <td><code>location</code></td> <td>The updated location.</td> </tr> </table>	<code>locationAvailability</code>	The current status of location availability.	<code>result</code>	The latest location result available.	<code>location</code>	The updated location.
<code>locationAvailability</code>	The current status of location availability.						
<code>result</code>	The latest location result available.						
<code>location</code>	The updated location.						

**Exhibit C for U.S. Patent No. 9,445,251 Against HTC Accused Products**

US9445251B2	HTC
	<p><a href="https://developers.google.com/android/reference/com/google/android/gms/location/LocationListener">https://developers.google.com/android/reference/com/google/android/gms/location/LocationListener</a></p> <p>Public Constructors</p> <hr/> <p>public <b>MapView</b> (<a href="#">Context</a> context)</p> <p>public <b>MapView</b> (<a href="#">Context</a> context, <a href="#">AttributeSet</a> attrs)</p> <p>public <b>MapView</b> (<a href="#">Context</a> context, <a href="#">AttributeSet</a> attrs, int defStyle)</p> <p>public <b>MapView</b> (<a href="#">Context</a> context, <a href="#">GoogleMapOptions</a> options)</p> <p><a href="https://developers.google.com/android/reference/com/google/android/gms/maps/MapView">https://developers.google.com/android/reference/com/google/android/gms/maps/MapView</a></p>

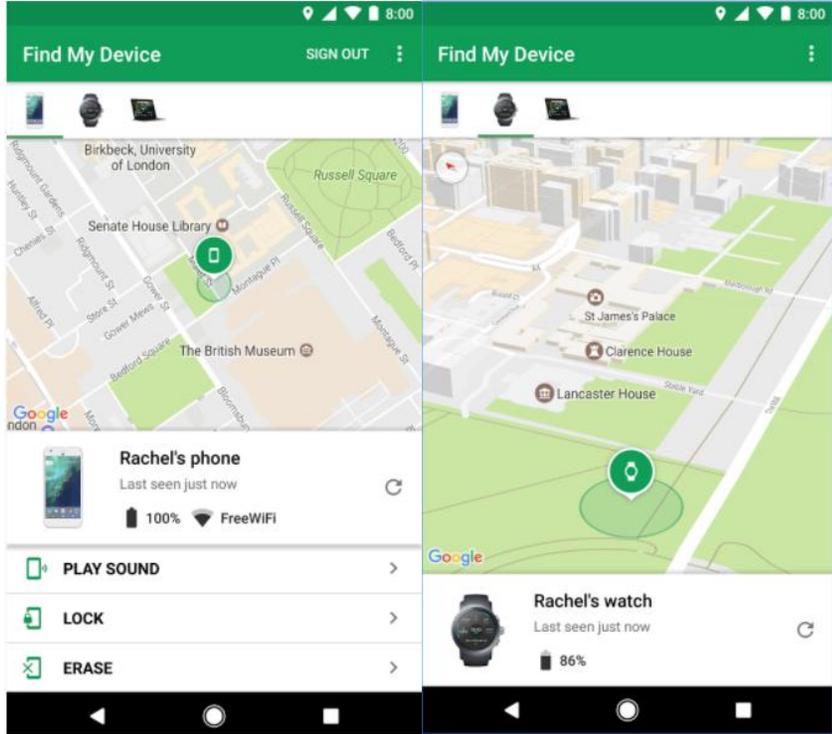
**Exhibit C for U.S. Patent No. 9,445,251 Against HTC Accused Products**

<p>US9445251B2</p>	<p><b>HTC</b></p> <p><code>public void <b>getMapAsync</b> (<b>OnMapReadyCallback</b> callback)</code></p> <p>Returns a non-null instance of the <code>GoogleMap</code>, ready to be used.</p> <p>Note that:</p> <ul style="list-style-type: none"> <li>• This method must be called from the main thread.</li> <li>• The callback will be executed in the main thread.</li> <li>• In the case where Google Play services is not installed on the user's device, the callback will not be triggered until the user installs it.</li> <li>• The <code>GoogleMap</code> object provided by the callback is non-null.</li> </ul> <p><b>Parameters</b></p> <p><code>callback</code> The callback object that will be triggered when the map is ready to be used.</p> <p><code>public final void <b>onCreate</b> (<b>Bundle</b> savedInstanceState)</code></p> <p>You must call this method from the parent Activity/Fragment's corresponding method.</p> <p><a href="https://developers.google.com/android/reference/com/google/android/gms/maps/MapView">https://developers.google.com/android/reference/com/google/android/gms/maps/MapView</a></p>
<p>[1F] presenting, via the interactive display of the first device, the second georeferenced map and the plurality of user-selectable symbols corresponding to the plurality of second devices, wherein the symbols are</p>	<p><b>HTC</b> infringes directly and/or indirectly by performing, inducing others to perform, and/or contributing to the performance of: presenting, via the interactive display of the first device, the second georeferenced map and the plurality of user-selectable symbols corresponding to the plurality of second devices, wherein the symbols are positioned on the second georeferenced map at respective positions corresponding to the locations of the second devices.</p> <p><b>Regarding Find My Device</b> and Android Device Manager, the Accused Products present the user with a second and/or updated map display on the display. The second and/or updated map comprises a first symbol positioned on the map and corresponding to the first device. The second and/or updated map comprises one or more second symbols positioned on the map and corresponding to one or more second devices. The map is interactive because the user may control the display of the map, e.g., pan, zoom, and/or effect change to the</p>

**Exhibit C for U.S. Patent No. 9,445,251 Against HTC Accused Products**

<b>US9445251B2</b>	<b>HTC</b>
<p>positioned on the second georeferenced map at respective positions corresponding to the locations of the second devices;</p>	<p>map in an otherwise interactive manner. The map is georeferenced for at least the reason that one or more symbols are associated with spatial locations, i.e., coordinates. The symbols are user-selectable because a user may touch the display to select the device associated with the symbol. For example, the user may input a touch selection directly on or near the portion of the display corresponding to the symbol's coordinates on the map to effect a selection of the device. The symbols are positions on the map at locations corresponding to the spatial positions of the symbols.</p> <p><b><u>Regarding Google Maps</u></b>, Google Latitude, Google Plus, Google Hangouts, Google Messages, Google Allo, Google Duo, Google Chrome, and Android Messenger, the Accused Products present the user of a first device with a second and/or updated map display on the display. The second and/or updated map comprises a first symbol positioned on the map and corresponding to the first user or corresponding first device. The second and/or updated map comprises one or more second symbols positioned on the map and corresponding to one or more second users, contacts and/or corresponding second devices. The map is interactive because the user may control the display of the map, e.g., pan, zoom, and/or effect change to the map in an otherwise interactive manner. The map is georeferenced for at least the reason that one or more symbols are associated with spatial locations, i.e., coordinates. The symbols are user-selectable because a user may touch the display to select the user or device associated with the symbol. For example, the user may input a touch selection directly on or near the portion of the display corresponding to the symbol's coordinates on the map to effect a selection of the user or device. The symbols are positions on the map at locations corresponding to the spatial positions of the symbols</p> <p><b><u>Exemplary Support for Find My Device:</u></b></p>

**Exhibit C for U.S. Patent No. 9,445,251 Against HTC Accused Products**

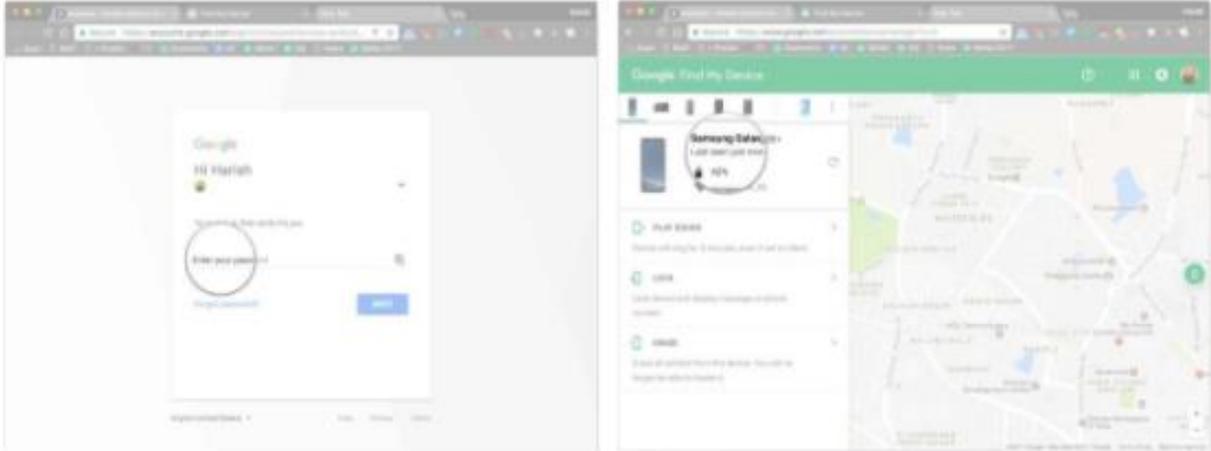
US9445251B2	HTC
	 <p><a href="https://play.google.com/store/apps/details?id=com.google.android.apps.adm&amp;hl=en">https://play.google.com/store/apps/details?id=com.google.android.apps.adm&amp;hl=en</a></p>

**Exhibit C for U.S. Patent No. 9,445,251 Against HTC Accused Products**

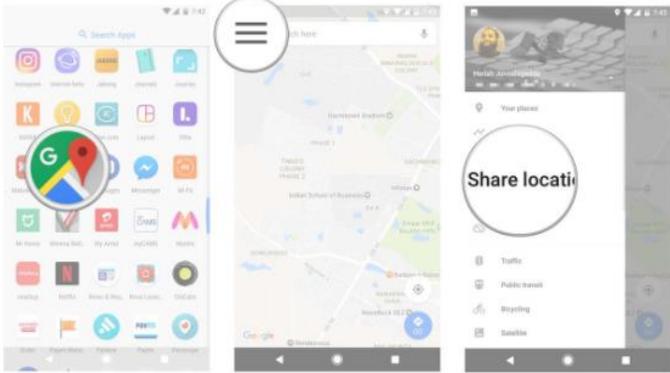
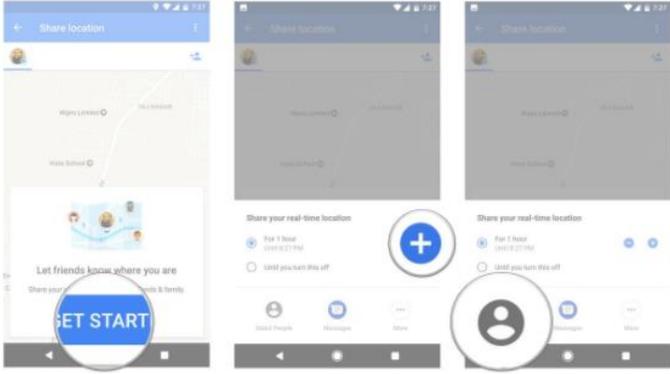
US9445251B2	HTC
	<p data-bbox="514 245 1533 337">Once you're signed in to Find My Device, you'll see a map with your current location as well as the make and model of your phone, and two options — Play Sound, and Enable Lock &amp; Erase. Hitting the latter option will allow you to start using the Lock and Erase functions.</p> <p data-bbox="514 375 1549 431">If you've signed into more than one phone, you can select a particular device by browsing the list at the top of the screen.</p> <ol data-bbox="514 480 1234 613" style="list-style-type: none"> <li>1. Open <b>Find My Device</b> from your home screen or app drawer.</li> <li>2. Select your phone from the <b>list of devices</b> at the top of the screen.</li> <li>3. See if your phone is <b>discoverable</b>.</li> </ol> <div data-bbox="550 646 1579 1300"> </div> <p data-bbox="499 1325 1121 1357"><a href="https://www.androidcentral.com/find-my-device">https://www.androidcentral.com/find-my-device</a></p>



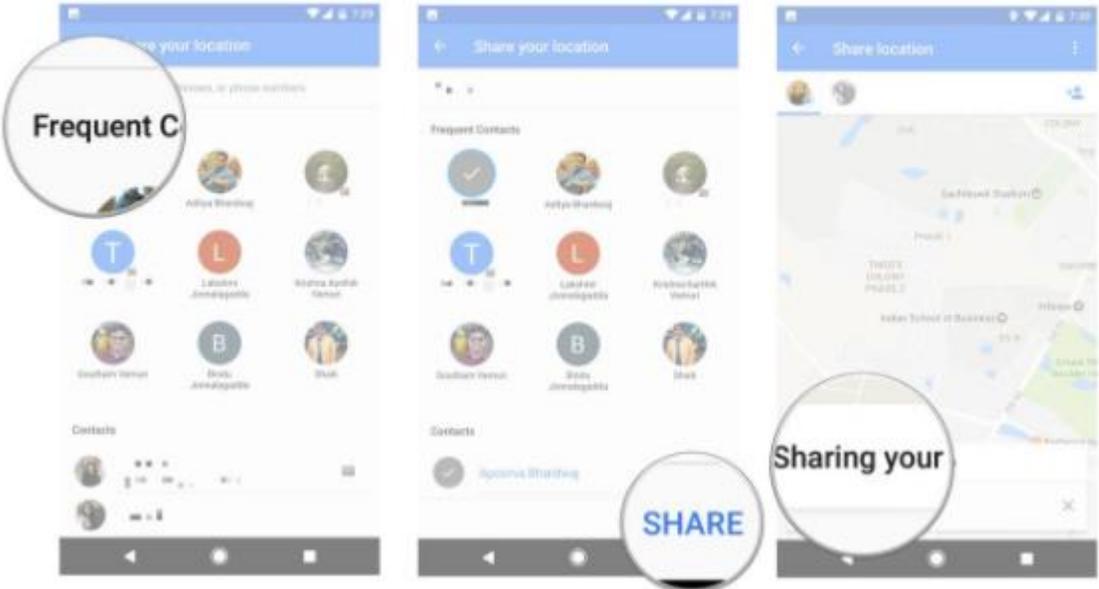
**Exhibit C for U.S. Patent No. 9,445,251 Against HTC Accused Products**

US9445251B2	HTC
	<h2 data-bbox="512 313 1650 367">How to locate your phone over the internet</h2> <p data-bbox="512 410 1656 548">If you've lost your phone, you can remotely locate it through the <a href="#">Find My Device website</a>. You'll need to sign in to the Google account that was used to set up Find My Device. It takes a few seconds, but the service should be able to track your phone. Alternatively, you can also do a Google search for "find my phone" to locate your handset.</p> <ol data-bbox="506 607 995 760" style="list-style-type: none"><li>1. Head to the <a href="#">Find My Device website</a>.</li><li>2. Sign in to your Google account.</li><li>3. Check if your device is visible.</li></ol> <div data-bbox="548 805 1759 1256"></div> <p data-bbox="499 1289 1121 1317"><a href="https://www.androidcentral.com/find-my-device">https://www.androidcentral.com/find-my-device</a></p> <p data-bbox="499 1360 1016 1390"><b><u>Exemplary Support for Google Maps:</u></b></p>

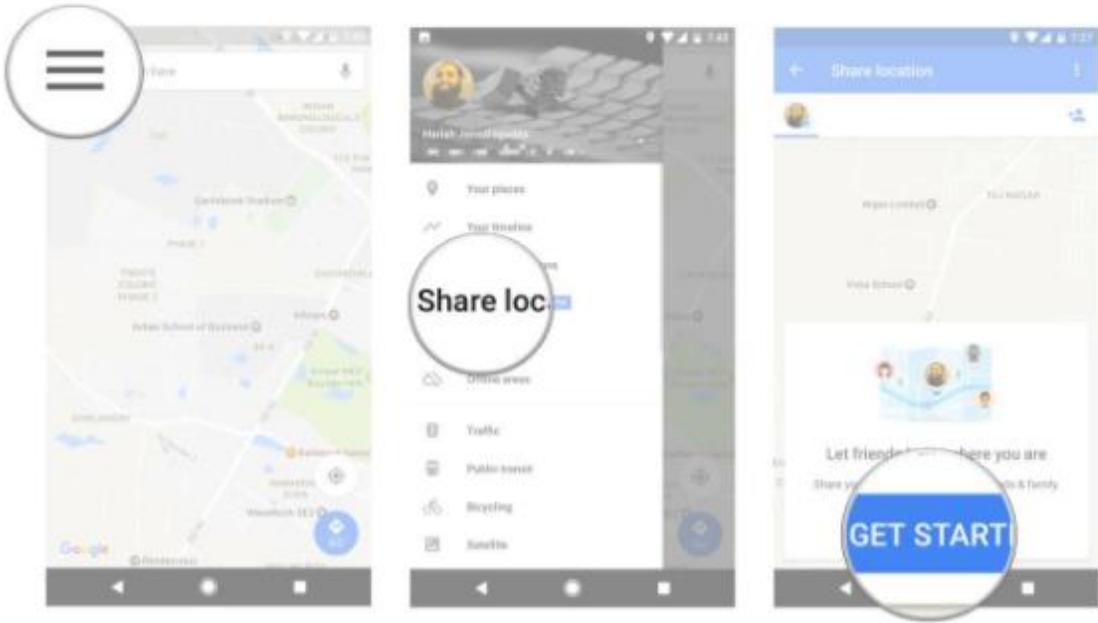
# Exhibit C for U.S. Patent No. 9,445,251 Against HTC Accused Products

US9445251B2	HTC
	<h3 data-bbox="506 277 1140 310">How to share your location in Google Maps</h3> <ol data-bbox="506 337 1119 427" style="list-style-type: none"><li>1. Open <b>Google Maps</b> from the app drawer or the home screen.</li><li>2. Tap the <b>hamburger menu</b> (the three horizontal lines) on the top left corner of the screen.</li><li>3. Select <b>Share location</b>.</li></ol>  <ol data-bbox="506 865 1150 971" style="list-style-type: none"><li>4. Tap <b>Get Started</b>.</li><li>5. Use the <b>+</b> icon to select a time period or select the <b>Until you turn this off</b> setting to share your location indefinitely.</li><li>6. Tap <b>Select People</b>.</li></ol>  <p data-bbox="499 1377 1346 1409"><a href="https://www.androidcentral.com/how-share-location-google-maps">https://www.androidcentral.com/how-share-location-google-maps</a></p>

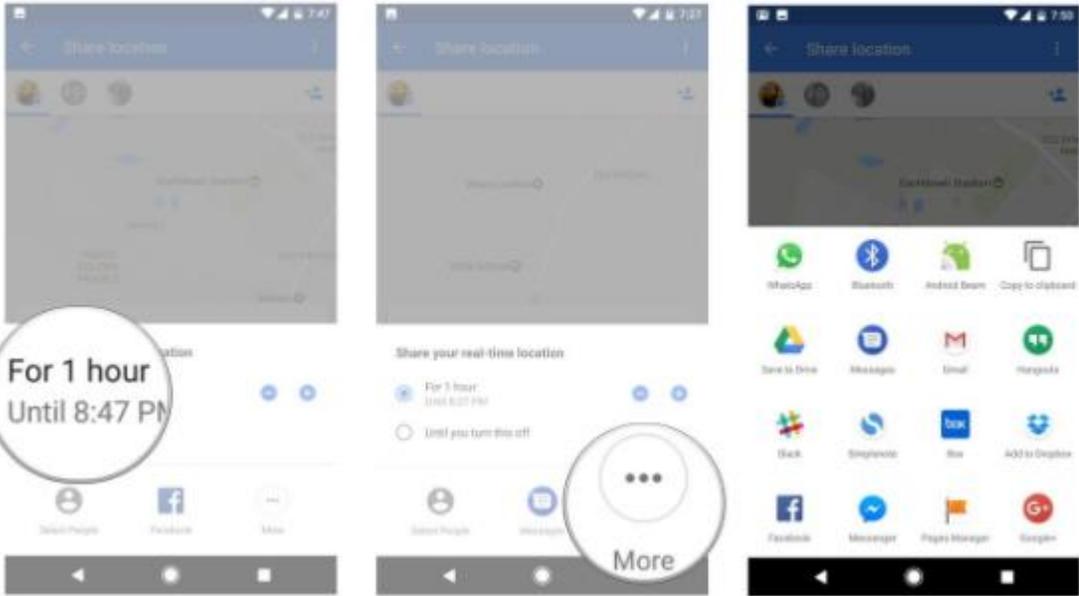
### Exhibit C for U.S. Patent No. 9,445,251 Against HTC Accused Products

US9445251B2	HTC
	<p data-bbox="514 289 1566 347">7. You'll see a list of your frequent contacts at the top, along with a full list of contacts. Pick the contacts by tapping their name.</p> <p data-bbox="514 375 1444 402">8. Once you've selected the contacts you want to share your location to, tap <b>Share</b>.</p> <p data-bbox="514 435 1409 462">9. You'll see a message saying that the selected contact can view your location.</p>  <p data-bbox="499 1138 1346 1170"><a href="https://www.androidcentral.com/how-share-location-google-maps">https://www.androidcentral.com/how-share-location-google-maps</a></p>

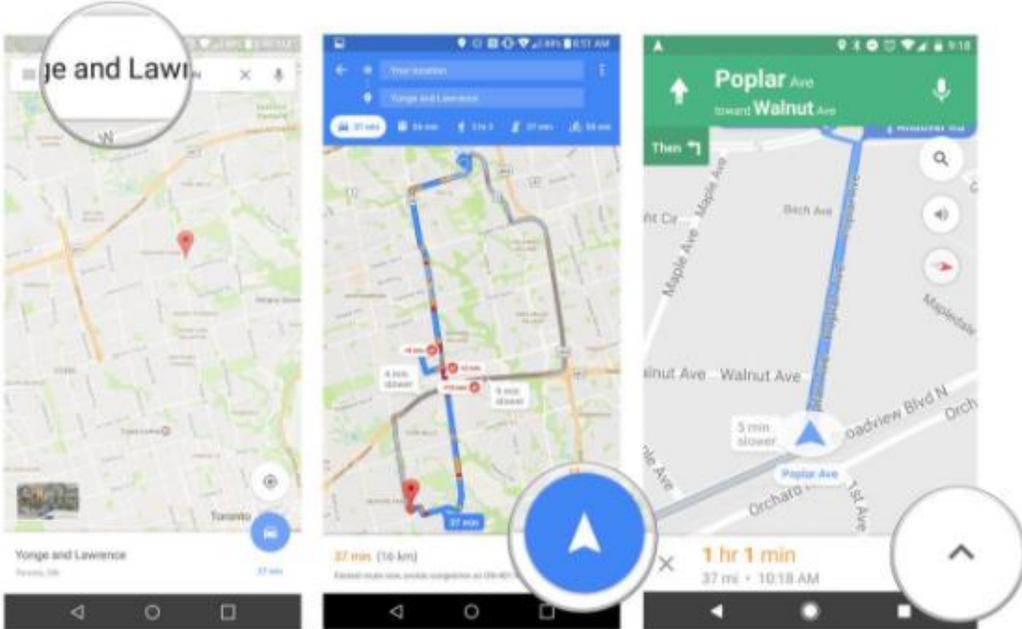
## Exhibit C for U.S. Patent No. 9,445,251 Against HTC Accused Products

US9445251B2	HTC
	<h3 data-bbox="512 245 1243 289">How to create a shareable link</h3> <p data-bbox="512 334 1451 362">You can also create a link and use it to share your location easily. Here's how to do it:</p> <ol data-bbox="506 412 1224 553" style="list-style-type: none"><li>1. Tap the <b>hamburger menu</b> on the top left corner of the screen.</li><li>2. Select <b>Share location</b>.</li><li>3. Tap <b>Get Started</b>.</li></ol>  <p data-bbox="499 1232 1346 1263"><a href="https://www.androidcentral.com/how-share-location-google-maps">https://www.androidcentral.com/how-share-location-google-maps</a></p>

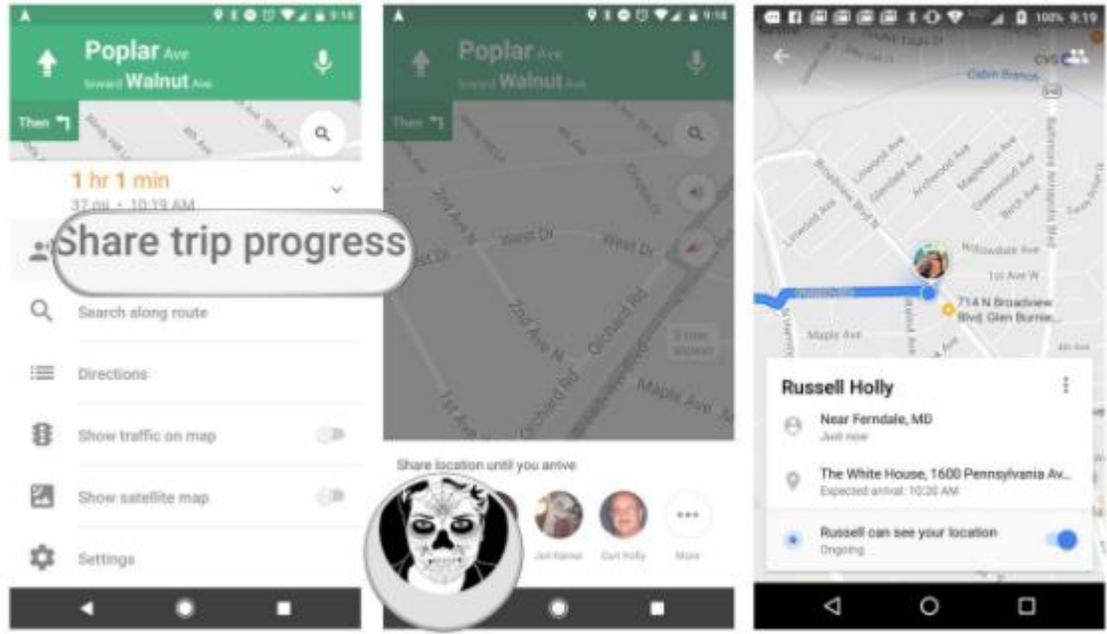
### Exhibit C for U.S. Patent No. 9,445,251 Against HTC Accused Products

US9445251B2	HTC
	<p data-bbox="510 245 1199 272">4. Select the amount of time you want to share your location.</p> <p data-bbox="510 302 667 329">5. Tap More.</p> <p data-bbox="510 358 1619 423">6. Select your app of choice to create and send a unique URL that broadcasts your current location. You can email it, send the link via Messenger, or even tweet it to the intended recipient.</p>  <p data-bbox="499 1084 1346 1117"><a href="https://www.androidcentral.com/how-share-location-google-maps">https://www.androidcentral.com/how-share-location-google-maps</a></p>

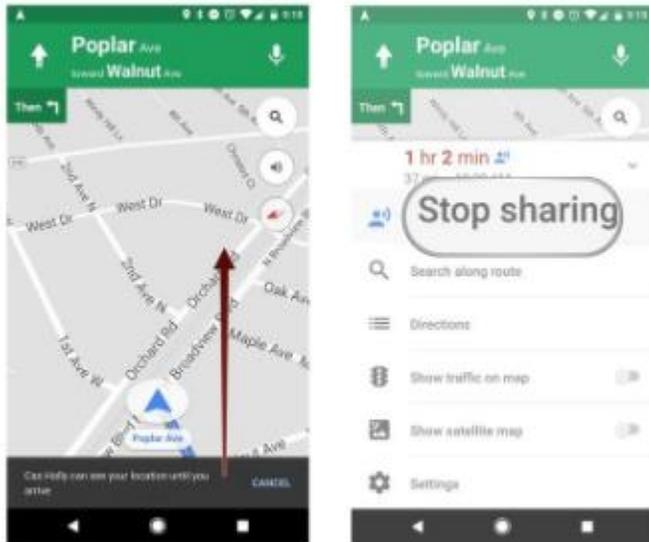
**Exhibit C for U.S. Patent No. 9,445,251 Against HTC Accused Products**

US9445251B2	HTC
	<p data-bbox="514 240 1417 337"><b>How to share your navigation directions while you walk, drive or transit</b></p> <p data-bbox="514 375 1543 464">One of the best ways to share your location with someone is during a drive, walk or transit. If you're meeting a friend or family member somewhere, or navigating towards their home, Google Maps lets you share your location with them for the duration of the trip. It's magic!</p> <ol data-bbox="514 513 1381 646" style="list-style-type: none"><li>1. In the <b>search bar</b> enter your destination.</li><li>2. Pick your navigation type (drive, transit, walk) and press the blue <b>navigate button</b>.</li><li>3. Tap the arrow next to the <b>time-to-destination</b> number at the bottom of the screen.</li></ol>  <p data-bbox="499 1328 1346 1360"><a href="https://www.androidcentral.com/how-share-location-google-maps">https://www.androidcentral.com/how-share-location-google-maps</a></p>

### Exhibit C for U.S. Patent No. 9,445,251 Against HTC Accused Products

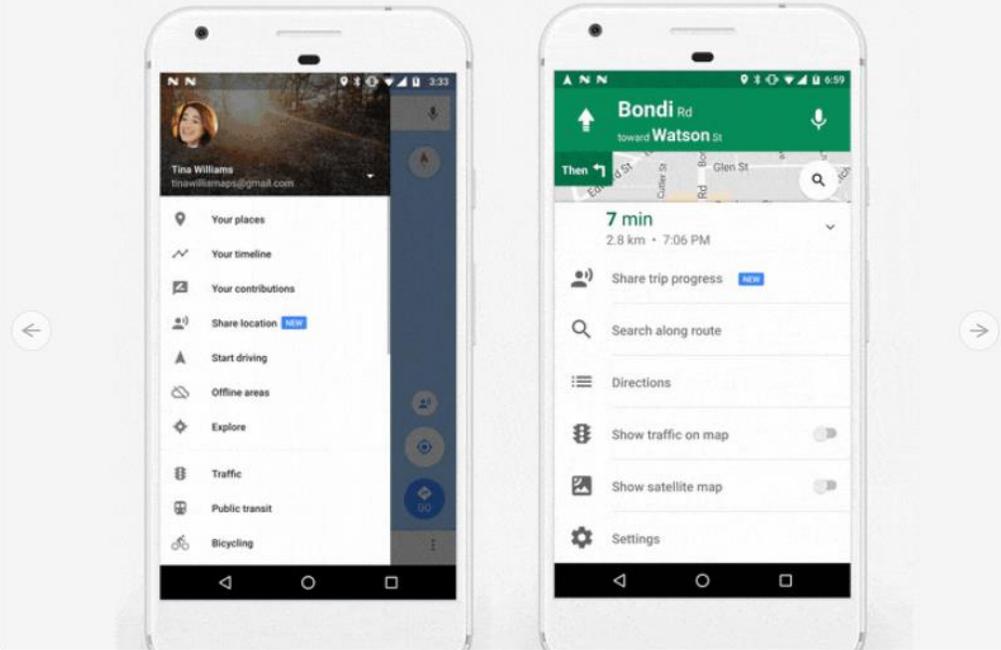
US9445251B2	HTC
	<p data-bbox="514 277 825 305">4. Tap Share trip progress.</p> <p data-bbox="514 334 1136 362">5. Choose one or more contacts to share trip progress.</p>  <p data-bbox="520 1063 1333 1091">You can also stop sharing your location with someone before a trip ends.</p> <p data-bbox="520 1101 1346 1128"><a href="https://www.androidcentral.com/how-share-location-google-maps">https://www.androidcentral.com/how-share-location-google-maps</a></p>

### Exhibit C for U.S. Patent No. 9,445,251 Against HTC Accused Products

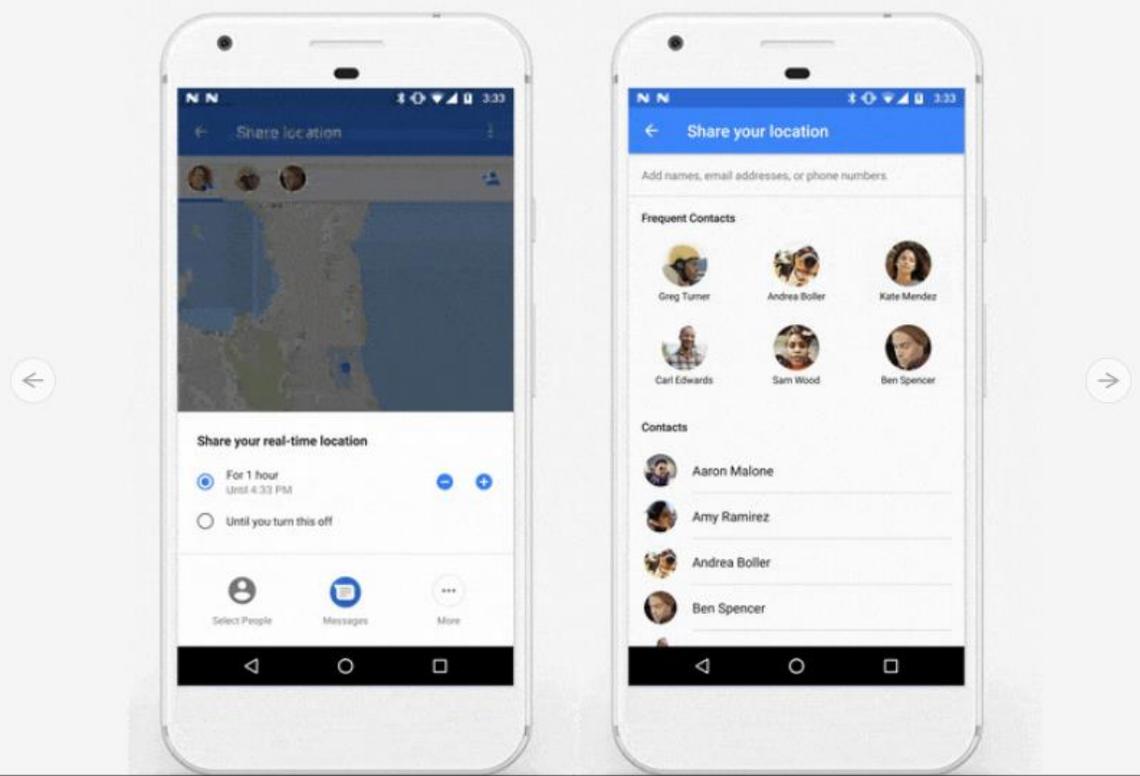
US9445251B2	HTC
	<ol style="list-style-type: none"><li data-bbox="520 245 1457 272">1. Tap the arrow next to the <b>time-to-destination</b> number at the bottom of the screen.</li><li data-bbox="520 302 758 329">2. Tap <b>Stop sharing</b>.</li></ol> <div data-bbox="732 380 1381 922"></div> <p data-bbox="531 976 625 1003">That's it!</p> <p data-bbox="531 1045 1598 1073">Are you excited that location sharing is back in Google Maps? How often do you use the feature?</p> <p data-bbox="499 1086 1346 1114"><a href="https://www.androidcentral.com/how-share-location-google-maps">https://www.androidcentral.com/how-share-location-google-maps</a></p>



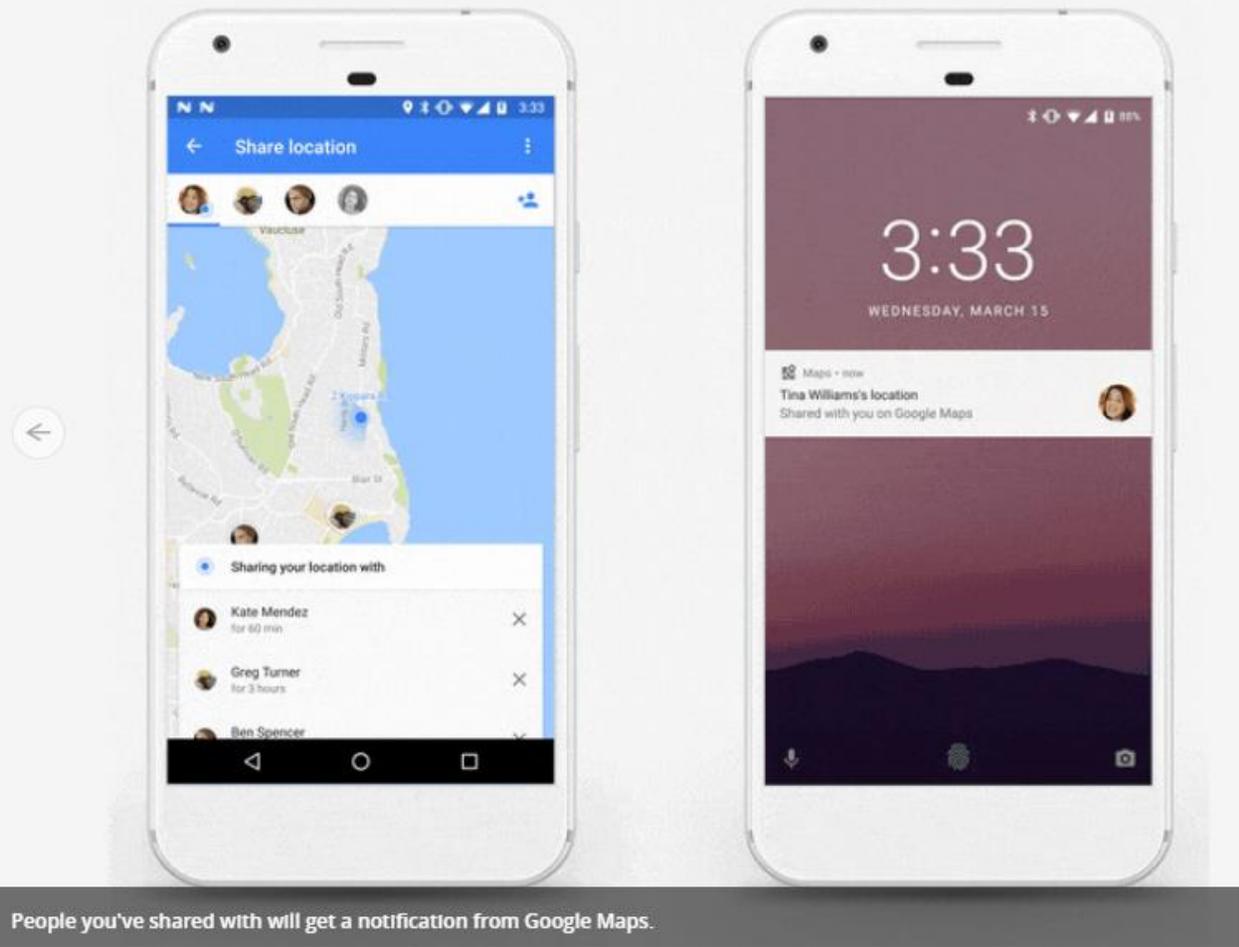
**Exhibit C for U.S. Patent No. 9,445,251 Against HTC Accused Products**

US9445251B2	HTC
	 <p data-bbox="506 901 1507 954">Ways to start location sharing: (left) from the app navigation drawer, and (right) from the menu, you can drag up while navigating somewhere.</p> <p data-bbox="506 961 1648 990"><a href="https://arstechnica.com/gadgets/2017/03/location-sharing-finally-returns-to-google-maps/">https://arstechnica.com/gadgets/2017/03/location-sharing-finally-returns-to-google-maps/</a></p>

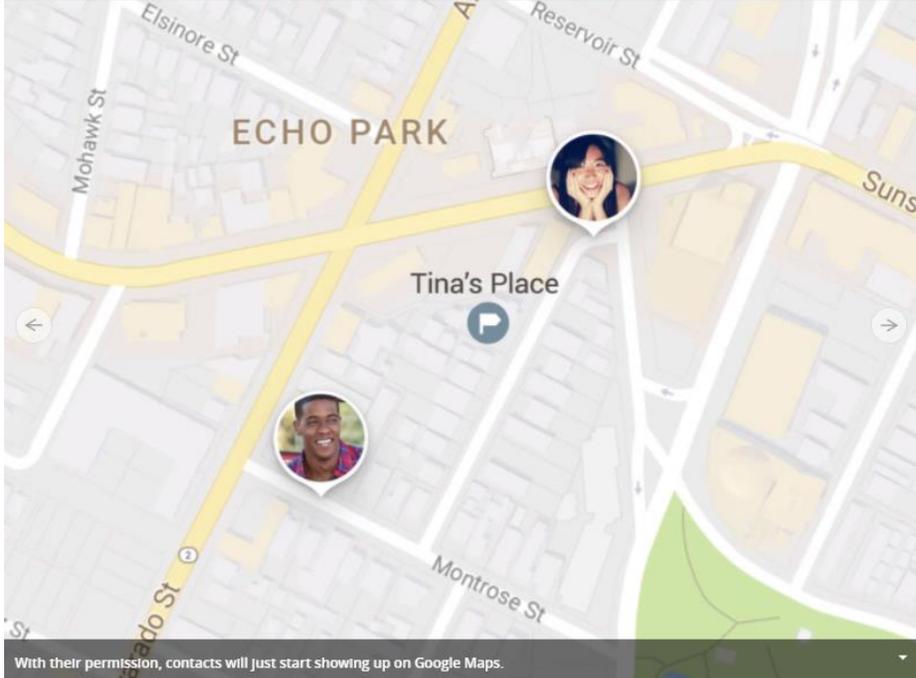
**Exhibit C for U.S. Patent No. 9,445,251 Against HTC Accused Products**

US9445251B2	HTC
	 <p data-bbox="506 1029 1646 1062">You can pick how long you want to share your location for, and with whom you want to share it.</p> <p data-bbox="506 1068 1646 1092"><a href="https://arstechnica.com/gadgets/2017/03/location-sharing-finally-returns-to-google-maps/">https://arstechnica.com/gadgets/2017/03/location-sharing-finally-returns-to-google-maps/</a></p>

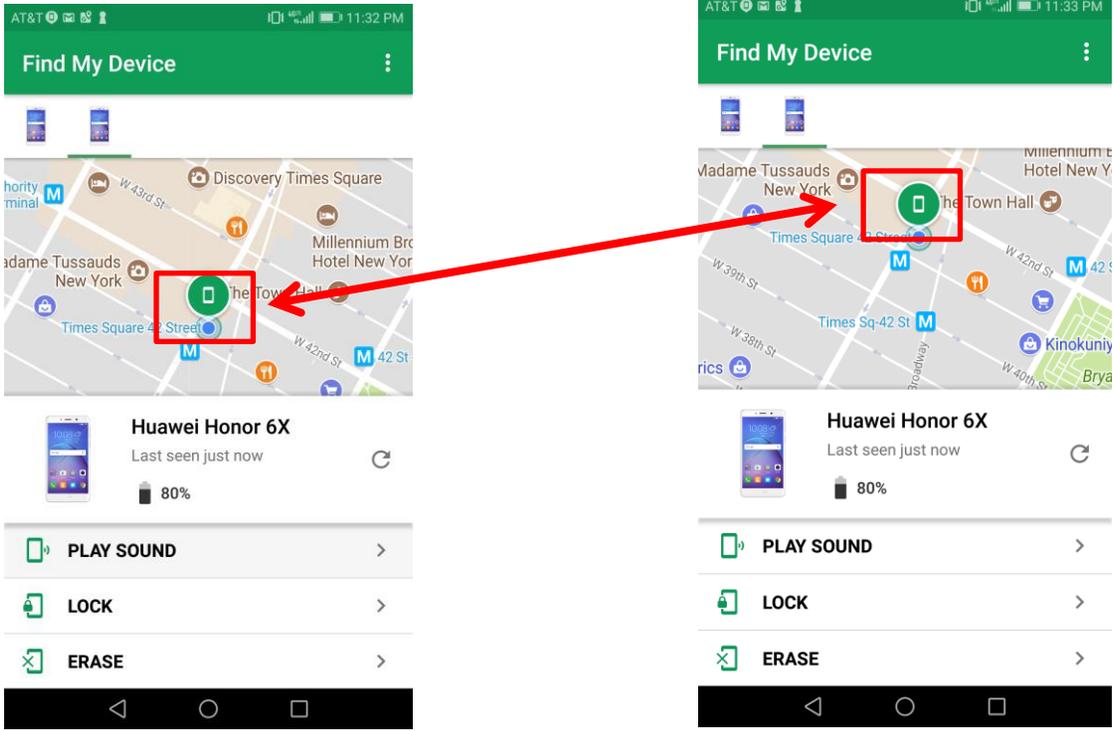
**Exhibit C for U.S. Patent No. 9,445,251 Against HTC Accused Products**

US9445251B2	HTC
	 <p data-bbox="506 1146 1161 1174">People you've shared with will get a notification from Google Maps.</p> <p data-bbox="499 1190 1646 1224"><a href="https://arstechnica.com/gadgets/2017/03/location-sharing-finally-returns-to-google-maps/">https://arstechnica.com/gadgets/2017/03/location-sharing-finally-returns-to-google-maps/</a></p>

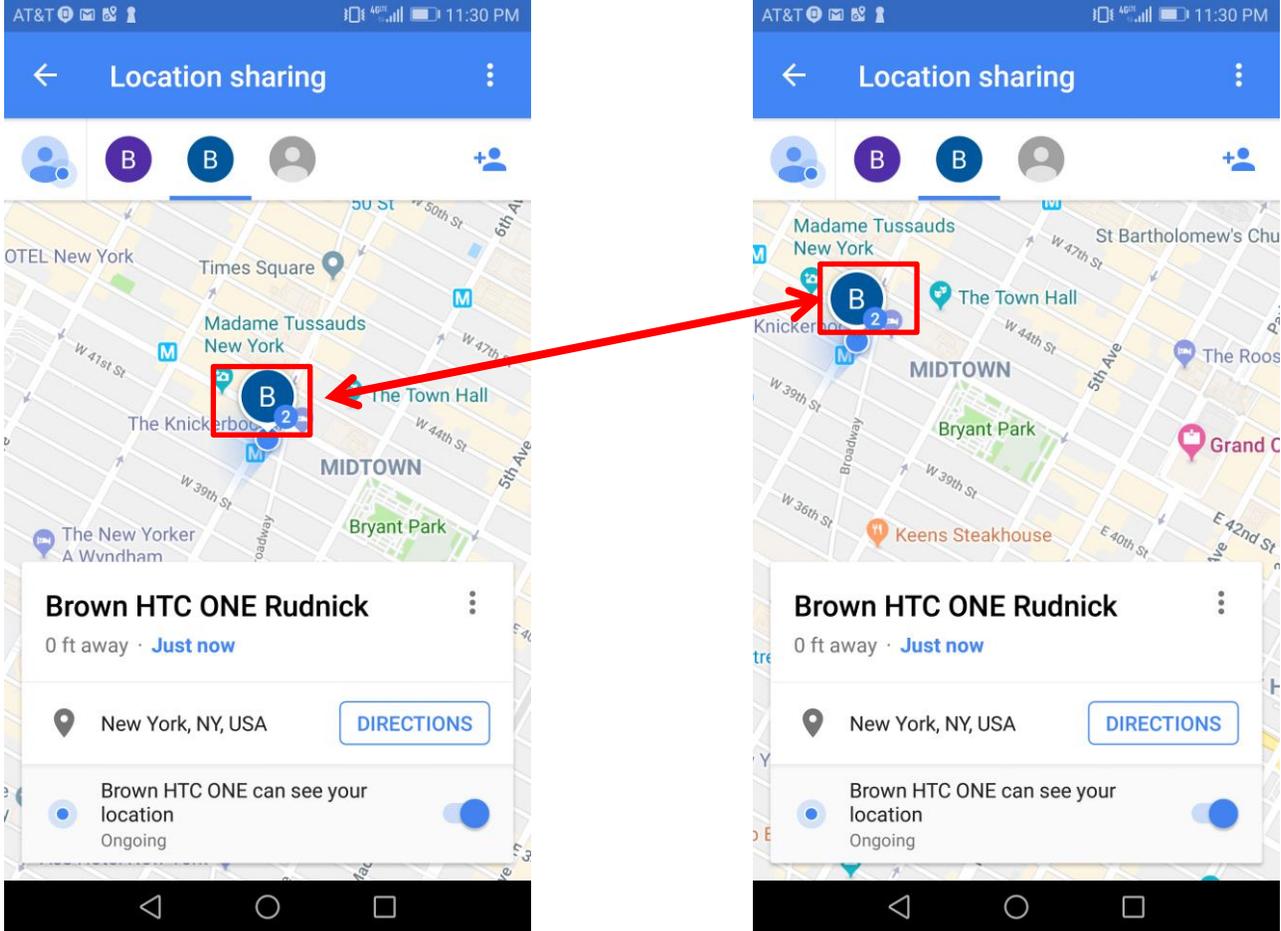
**Exhibit C for U.S. Patent No. 9,445,251 Against HTC Accused Products**

US9445251B2	HTC
	 <p data-bbox="506 889 1003 906">With their permission, contacts will just start showing up on Google Maps.</p> <p data-bbox="499 915 1646 948"><a href="https://arstechnica.com/gadgets/2017/03/location-sharing-finally-returns-to-google-maps/">https://arstechnica.com/gadgets/2017/03/location-sharing-finally-returns-to-google-maps/</a></p> <p data-bbox="499 987 1052 1019"><b><u>Exemplary Find My Device Screenshots:</u></b></p>

**Exhibit C for U.S. Patent No. 9,445,251 Against HTC Accused Products**

US9445251B2	HTC
	 <p><b>Exemplary Google Maps Screenshots:</b></p>

### Exhibit C for U.S. Patent No. 9,445,251 Against HTC Accused Products

US9445251B2	HTC
	 <p><b>Exemplary Source Code:</b></p> <p>The above functionality is performed at least in part by the following publicly available source code and/or</p>

**Exhibit C for U.S. Patent No. 9,445,251 Against HTC Accused Products**

US9445251B2	HTC
	<p>source code that invokes or is invoked by the following source code (or a substantially similar copy compiled and loaded onto the Accused Products by <b>HTC</b>):</p> <pre data-bbox="520 321 1726 365">public static LocationRequest create ()</pre> <p>Create a location request with default parameters.</p> <p>Default parameters are for a block accuracy, slowly updated location. It can then be adjusted as required by the applications before passing to the <code>FusedLocationProviderApi</code>.</p> <p><b>Returns</b></p> <ul data-bbox="541 574 802 602" style="list-style-type: none"><li>• a new location request</li></ul> <p><a href="https://developers.google.com/android/reference/com/google/android/gms/location/LocationRequest">https://developers.google.com/android/reference/com/google/android/gms/location/LocationRequest</a></p>

**Exhibit C for U.S. Patent No. 9,445,251 Against HTC Accused Products**

US9445251B2	HTC
	<p><b>public static final int PRIORITY_BALANCED_POWER_ACCURACY</b></p> <p>Used with <code>setPriority(int)</code> to request "block" level accuracy.</p> <p>Block level accuracy is considered to be about 100 meter accuracy. Using a coarse accuracy such as this often consumes less power.</p> <p>Constant Value: 102</p> <hr/> <p><b>public static final int PRIORITY_HIGH_ACCURACY</b></p> <p>Used with <code>setPriority(int)</code> to request the most accurate locations available.</p> <p>This will return the finest location available.</p> <p>Constant Value: 100</p> <hr/> <p><b>public static final int PRIORITY_LOW_POWER</b></p> <p>Used with <code>setPriority(int)</code> to request "city" level accuracy.</p> <p>City level accuracy is considered to be about 10km accuracy. Using a coarse accuracy such as this often consumes less power.</p> <p>Constant Value: 104</p> <p><a href="https://developers.google.com/android/reference/com/google/android/gms/location/LocationRequest">https://developers.google.com/android/reference/com/google/android/gms/location/LocationRequest</a></p>



**Exhibit C for U.S. Patent No. 9,445,251 Against HTC Accused Products**

US9445251B2	HTC
	<p data-bbox="520 250 1738 289"><code>public Task&lt;Location&gt; getLastLocation ()</code></p> <p data-bbox="520 315 1094 342">Returns the best most recent location currently available.</p> <p data-bbox="520 371 1686 435">If a location is not available, which should happen very rarely, null will be returned. The best accuracy available while respecting the location permissions will be returned.</p> <p data-bbox="520 464 1724 527">This method provides a simplified way to get location. It is particularly well suited for applications that do not require an accurate location and that do not want to maintain extra logic for location updates.</p> <p data-bbox="520 578 1738 617"><code>public Task&lt;LocationAvailability&gt; getLocationAvailability ()</code></p> <p data-bbox="520 646 1680 709">Returns the availability of location data. When <code>isLocationAvailable()</code> returns true, then the location returned by <code>getLastLocation()</code> will be reasonably up to date within the hints specified by the active <code>LocationRequest</code> s.</p> <p data-bbox="520 738 1461 766">If the client isn't connected to Google Play services and the request times out, null is returned.</p> <p data-bbox="520 795 1661 859">Note it's always possible for <code>getLastLocation()</code> to return null even when this method returns true (e.g. location settings were disabled between calls).</p> <p data-bbox="520 865 1906 928"><a href="https://developers.google.com/android/reference/com/google/android/gms/location/FusedLocationProviderClient">https://developers.google.com/android/reference/com/google/android/gms/location/FusedLocationProviderClient</a></p>

**Exhibit C for U.S. Patent No. 9,445,251 Against HTC Accused Products**

US9445251B2	HTC						
	<p data-bbox="506 240 1734 326"><code>public Task&lt;Void&gt; requestLocationUpdates (LocationRequest request, LocationCallback callback, Looper looper)</code></p> <p data-bbox="506 354 1257 380">Requests location updates with a callback on the specified Looper thread.</p> <p data-bbox="506 410 1671 472">This method is suited for the foreground use cases. For background use cases, the <code>PendingIntent</code> version of the method is recommended, see <code>requestLocationUpdates(LocationRequest, PendingIntent)</code>.</p> <p data-bbox="506 503 1356 529">Any previous LocationRequests registered on this LocationListener will be replaced.</p> <p data-bbox="506 560 1671 654">This call will keep the Google Play services connection active, so make sure to call <code>removeLocationUpdates(LocationCallback)</code> when you no longer need it, otherwise you lose the benefits of the automatic connection management.</p> <p data-bbox="506 685 1728 711">Callbacks for <code>LocationCallback</code> will be made on the specified thread, which must already be a prepared looper thread.</p> <p data-bbox="506 735 653 761"><b>Parameters</b></p> <table border="1" data-bbox="506 792 1734 1008"> <tbody> <tr> <td data-bbox="506 792 617 857"><b>request</b></td> <td data-bbox="617 792 1734 857">The location request for the updates.</td> </tr> <tr> <td data-bbox="506 857 617 922"><b>callback</b></td> <td data-bbox="617 857 1734 922">The callback for the location updates.</td> </tr> <tr> <td data-bbox="506 922 617 1008"><b>looper</b></td> <td data-bbox="617 922 1734 1008">The Looper object whose message queue will be used to implement the callback mechanism, or null to make callbacks on the calling thread.</td> </tr> </tbody> </table> <p data-bbox="506 1023 1906 1089"><a href="https://developers.google.com/android/reference/com/google/android/gms/location/FusedLocationProviderClient">https://developers.google.com/android/reference/com/google/android/gms/location/FusedLocationProviderClient</a></p>	<b>request</b>	The location request for the updates.	<b>callback</b>	The callback for the location updates.	<b>looper</b>	The Looper object whose message queue will be used to implement the callback mechanism, or null to make callbacks on the calling thread.
<b>request</b>	The location request for the updates.						
<b>callback</b>	The callback for the location updates.						
<b>looper</b>	The Looper object whose message queue will be used to implement the callback mechanism, or null to make callbacks on the calling thread.						

**Exhibit C for U.S. Patent No. 9,445,251 Against HTC Accused Products**

US9445251B2	HTC				
	<pre data-bbox="520 245 1730 326">public Task&lt;Void&gt; requestLocationUpdates (LocationRequest request, PendingIntent callbackIntent)</pre> <p data-bbox="520 354 1255 378">Requests location updates with a callback on the specified PendingIntent.</p> <p data-bbox="520 412 1717 540">This method is suited for the background use cases, more specifically for receiving location updates, even when the app has been killed by the system. In order to do so, use a <code>PendingIntent</code> for a started service. For foreground use cases, the <code>LocationCallback</code> version of the method is recommended, see <code>requestLocationUpdates(LocationRequest, LocationCallback, Looper)</code>.</p> <p data-bbox="520 573 1709 630">Any previously registered requests that have the same PendingIntent (as defined by <code>equals(Object)</code>) will be replaced by this request.</p> <p data-bbox="520 662 1713 756">Both <code>LocationResult</code> and <code>LocationAvailability</code> are sent to the given PendingIntent. You can extract data from an Intent using <code>hasResult(Intent)</code>, <code>extractResult(Intent)</code>, <code>hasLocationAvailability(Intent)</code>, and <code>extractLocationAvailability(Intent)</code>.</p> <p data-bbox="533 781 655 805"><b>Parameters</b></p> <table border="1" data-bbox="520 834 1730 971"> <tbody> <tr> <td data-bbox="520 834 825 902"><code>request</code></td> <td data-bbox="835 834 1730 902">The location request for the updates.</td> </tr> <tr> <td data-bbox="520 906 825 971"><code>callbackIntent</code></td> <td data-bbox="835 906 1730 971">A pending intent to be sent for each location update.</td> </tr> </tbody> </table> <p data-bbox="533 997 617 1021"><b>Returns</b></p> <ul data-bbox="541 1045 1346 1070" style="list-style-type: none"> <li>• a Task for the call, check <code>isSuccessful()</code> to determine if it was successful.</li> </ul> <p data-bbox="499 1081 1902 1146"><a href="https://developers.google.com/android/reference/com/google/android/gms/location/FusedLocationProviderClient">https://developers.google.com/android/reference/com/google/android/gms/location/FusedLocationProviderClient</a></p>	<code>request</code>	The location request for the updates.	<code>callbackIntent</code>	A pending intent to be sent for each location update.
<code>request</code>	The location request for the updates.				
<code>callbackIntent</code>	A pending intent to be sent for each location update.				

## Exhibit C for U.S. Patent No. 9,445,251 Against HTC Accused Products

US9445251B2	HTC						
	<p><code>public void onLocationAvailability (LocationAvailability locationAvailability)</code></p> <p>Called when there is a change in the availability of location data.</p> <p>When <code>isLocationAvailable()</code> returns <code>false</code> you can assume that location will not be returned in <code>onLocationResult(LocationResult)</code> until something changes in the device's settings or environment. Even when <code>isLocationAvailable()</code> returns <code>true</code> the <code>onLocationResult(LocationResult)</code> may not always be called regularly, however the device location is known and both the most recently delivered location and <code>getLastLocation(GoogleApiClient)</code> will be reasonably up to date given the hints specified by the active <code>LocationRequest</code>s.</p> <p><b>Parameters</b></p> <table border="1"> <tr> <td><code>locationAvailability</code></td> <td>The current status of location availability.</td> </tr> </table> <p><code>public void onLocationResult (LocationResult result)</code></p> <p>Called when device location information is available.</p> <p>The most recent location returned by <code>getLastLocation()</code> is not guaranteed to be immediately fresh, but will be reasonably up to date given the hints specified by the active <code>LocationRequest</code>s.</p> <p><b>Parameters</b></p> <table border="1"> <tr> <td><code>result</code></td> <td>The latest location result available.</td> </tr> </table> <p><a href="https://developers.google.com/android/reference/com/google/android/gms/location/LocationCallback">https://developers.google.com/android/reference/com/google/android/gms/location/LocationCallback</a></p> <p><code>public abstract void onLocationChanged (Location location)</code></p> <p>Called when the location has changed.</p> <p><b>Parameters</b></p> <table border="1"> <tr> <td><code>location</code></td> <td>The updated location.</td> </tr> </table>	<code>locationAvailability</code>	The current status of location availability.	<code>result</code>	The latest location result available.	<code>location</code>	The updated location.
<code>locationAvailability</code>	The current status of location availability.						
<code>result</code>	The latest location result available.						
<code>location</code>	The updated location.						

**Exhibit C for U.S. Patent No. 9,445,251 Against HTC Accused Products**

US9445251B2	HTC
	<p><a href="https://developers.google.com/android/reference/com/google/android/gms/location/LocationListener">https://developers.google.com/android/reference/com/google/android/gms/location/LocationListener</a></p> <p>Public Constructors</p> <hr/> <p>public <b>MapView</b> (<a href="#">Context</a> context)</p> <p>public <b>MapView</b> (<a href="#">Context</a> context, <a href="#">AttributeSet</a> attrs)</p> <p>public <b>MapView</b> (<a href="#">Context</a> context, <a href="#">AttributeSet</a> attrs, int defStyle)</p> <p>public <b>MapView</b> (<a href="#">Context</a> context, <a href="#">GoogleMapOptions</a> options)</p> <p><a href="https://developers.google.com/android/reference/com/google/android/gms/maps/MapView">https://developers.google.com/android/reference/com/google/android/gms/maps/MapView</a></p>

**Exhibit C for U.S. Patent No. 9,445,251 Against HTC Accused Products**

<p><b>US9445251B2</b></p>	<p><b>HTC</b></p>
	<p><code>public void <b>getMapAsync</b> (<b>OnMapReadyCallback</b> callback)</code></p> <p>Returns a non-null instance of the <code>GoogleMap</code>, ready to be used.</p> <p>Note that:</p> <ul style="list-style-type: none"> <li>• This method must be called from the main thread.</li> <li>• The callback will be executed in the main thread.</li> <li>• In the case where Google Play services is not installed on the user's device, the callback will not be triggered until the user installs it.</li> <li>• The <code>GoogleMap</code> object provided by the callback is non-null.</li> </ul> <p><b>Parameters</b></p> <p><code>callback</code> The callback object that will be triggered when the map is ready to be used.</p> <p><code>public final void <b>onCreate</b> (<b>Bundle</b> savedInstanceState)</code></p> <p>You must call this method from the parent Activity/Fragment's corresponding method.</p> <p><a href="https://developers.google.com/android/reference/com/google/android/gms/maps/MapView">https://developers.google.com/android/reference/com/google/android/gms/maps/MapView</a></p>
<p>[1G] and identifying user interaction with the interactive display selecting one or more of the user-selectable symbols corresponding to one or more of the second devices and positioned on the second georeferenced</p>	<p><b>HTC</b> infringes directly and/or indirectly by performing, inducing others to perform, and/or contributing to the performance of: identifying user interaction with the interactive display selecting one or more of the user-selectable symbols corresponding to one or more of the second devices and positioned on the second georeferenced map and user interaction with the display specifying an action and, based thereon, using an Internet Protocol to send data to the one or more second devices via the server, wherein the first device does not have access to respective Internet Protocol addresses of the second devices.</p> <p><b>Regarding Find My Device</b> and Android Device Manager, the Accused Products are configured to allow a user of a first device to interact with the display, to select a device corresponding to a symbol, and to select an action to be performed, such as: play a sound on the second device, put the second device into a lost mode, and erase the second device. Selection of one of the aforementioned actions results in sending data from the</p>

**Exhibit C for U.S. Patent No. 9,445,251 Against HTC Accused Products**

US9445251B2	HTC
<p>map and user interaction with the display specifying an action and, based thereon, using an Internet Protocol to send data to the one or more second devices via the server, wherein the first device does not have access to respective Internet Protocol addresses of the second devices.</p>	<p>first device to a server and then sending data from the server to the second device.</p> <p>On information and belief, the communication of device-location information does not reveal a device’s IP address. On information and belief, such communications are encrypted and do not reveal IP addresses to individual devices.</p> <p><b><u>Regarding Google Maps</u></b>, Google Latitude, Google Plus, Google Hangouts, Google Messages, and Android Messenger, the Accused Products are configured to allow a user of the first device to interact with the display, to select a user, contact, or device, and to select an action to be performed, such as: sending a message, initiating a call, initiating a data conference, sharing a location, stop sharing a location, block a user from location sharing, sending a location, requesting a location, or sending other data. When an action is specified, data is sent from the first device to the second device via a server.</p> <p>On information and belief, the communication of device-location information does not reveal a device’s IP address. On information and belief, such communications are encrypted and do not reveal IP addresses to individual devices.</p> <p><b><u>Exemplary Support for Find My Device:</u></b></p> <p>Introducing Find My Device - the new and improved Android Device Manager. Find My Device helps you easily locate a lost Android device, and keeps your information safe and sound while you look.</p> <p><b>Locate your phone, tablet or watch.</b> Misplaced your Android Wear device? No problem.</p> <p><b>Play a sound.</b> Find My Device helps you track down your device when it's close by.</p> <p><b>Lock, erase or show a message.</b> With Find My Device you can secure your device remotely and help someone get in touch.</p> <p>Permissions Notice</p> <ul style="list-style-type: none"> <li>- Location: Needed to show your device's current location on the map.</li> <li>- Contacts: Needed to access the email address associated with your Google account.</li> </ul>

**Exhibit C for U.S. Patent No. 9,445,251 Against HTC Accused Products**

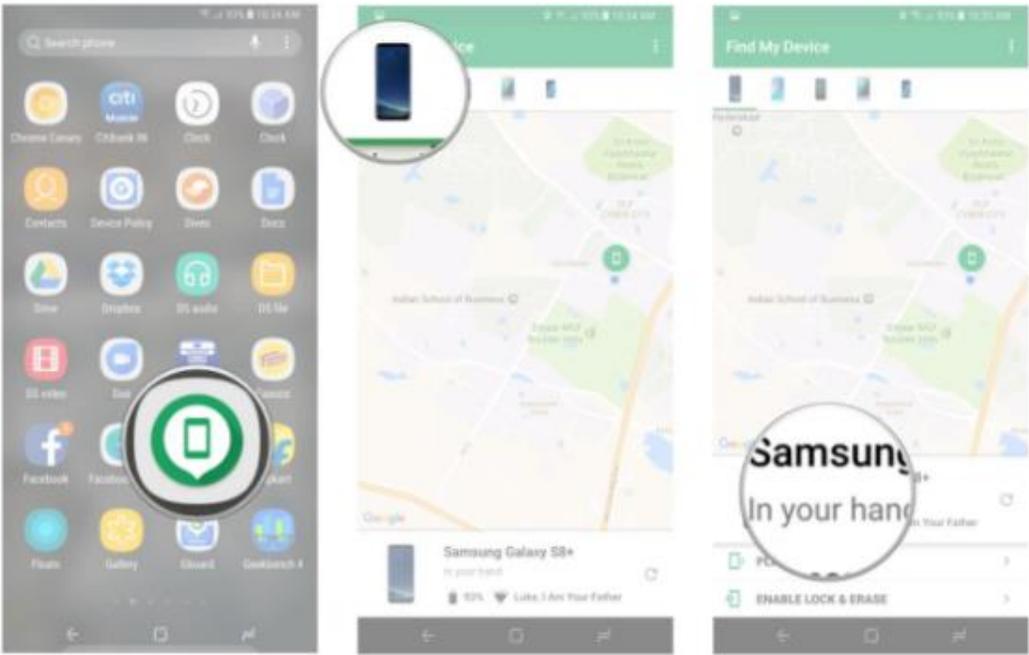
US9445251B2	HTC
	<p data-bbox="499 235 1570 267"><a href="https://play.google.com/store/apps/details?id=com.google.android.apps.adm&amp;hl=en">https://play.google.com/store/apps/details?id=com.google.android.apps.adm&amp;hl=en</a></p> <div data-bbox="499 305 1743 1039"> <p>The image contains two side-by-side screenshots of the Google Play Store app interface for the 'Find My Device' application. The left screenshot shows the 'Find My Device' screen with a map of London, a card for 'Rachel's phone' (last seen just now, 100% battery, FreeWiFi), and three action buttons: 'PLAY SOUND', 'LOCK', and 'ERASE'. The right screenshot shows the 'Lock device' screen with a 'Set password' section, an 'Add a message or phone number' section, and a green 'LOCK' button at the bottom.</p> </div> <p data-bbox="499 1047 1570 1079"><a href="https://play.google.com/store/apps/details?id=com.google.android.apps.adm&amp;hl=en">https://play.google.com/store/apps/details?id=com.google.android.apps.adm&amp;hl=en</a></p>



**Exhibit C for U.S. Patent No. 9,445,251 Against HTC Accused Products**

US9445251B2	HTC
	<p data-bbox="520 240 1167 277"><b>How to locate your phone with Google</b></p> <p data-bbox="520 310 1482 363">Should you happen to lose your phone, you can locate its whereabouts by logging into your Google account from any computer or even from another phone.</p> <ol data-bbox="520 407 1241 532" style="list-style-type: none"> <li data-bbox="520 407 1100 431">1. Launch a <b>web browser</b> from a phone, tablet, or computer.</li> <li data-bbox="520 456 1241 480">2. Navigate to <b>Google</b> if it is not your default search engine or home page.</li> <li data-bbox="520 505 1073 529">3. Type <b>find my phone android</b> in the Google search bar.</li> </ol> <div data-bbox="550 565 1514 1133"> <p>The image contains three sequential screenshots from an Android phone. The first screenshot shows the home screen with various app icons; the Chrome browser icon is circled in red. The second screenshot shows the Google search page with the search bar containing the text 'google' circled in red. The third screenshot shows the search results for 'find my phone android', with the first result, 'find my phone android', circled in red.</p> </div> <ol data-bbox="520 1182 1514 1317" style="list-style-type: none"> <li data-bbox="520 1182 1136 1206">4. Tap on <b>Find My Device</b> (usually the first option in the search).</li> <li data-bbox="520 1230 1514 1317">5. Enter your <b>email address</b> and <b>password</b> just as though you were checking your email. If you have 2-step verification set up on your Google account (and you most certainly should), you'll need to complete that process as well.</li> </ol> <p data-bbox="499 1325 1247 1357"><a href="https://www.androidcentral.com/how-track-android-phone">https://www.androidcentral.com/how-track-android-phone</a></p>

**Exhibit C for U.S. Patent No. 9,445,251 Against HTC Accused Products**

US9445251B2	HTC
	<p data-bbox="516 282 1535 375">Once you're signed in to Find My Device, you'll see a map with your current location as well as the make and model of your phone, and two options — Play Sound, and Enable Lock &amp; Erase. Hitting the latter option will allow you to start using the Lock and Erase functions.</p> <p data-bbox="516 410 1549 467">If you've signed into more than one phone, you can select a particular device by browsing the list at the top of the screen.</p> <ol data-bbox="516 516 1234 651" style="list-style-type: none"><li>1. Open <b>Find My Device</b> from your home screen or app drawer.</li><li>2. Select your phone from the list of devices at the top of the screen.</li><li>3. See if your phone is <b>discoverable</b>.</li></ol>  <p data-bbox="499 1360 1121 1391"><a href="https://www.androidcentral.com/find-my-device">https://www.androidcentral.com/find-my-device</a></p>

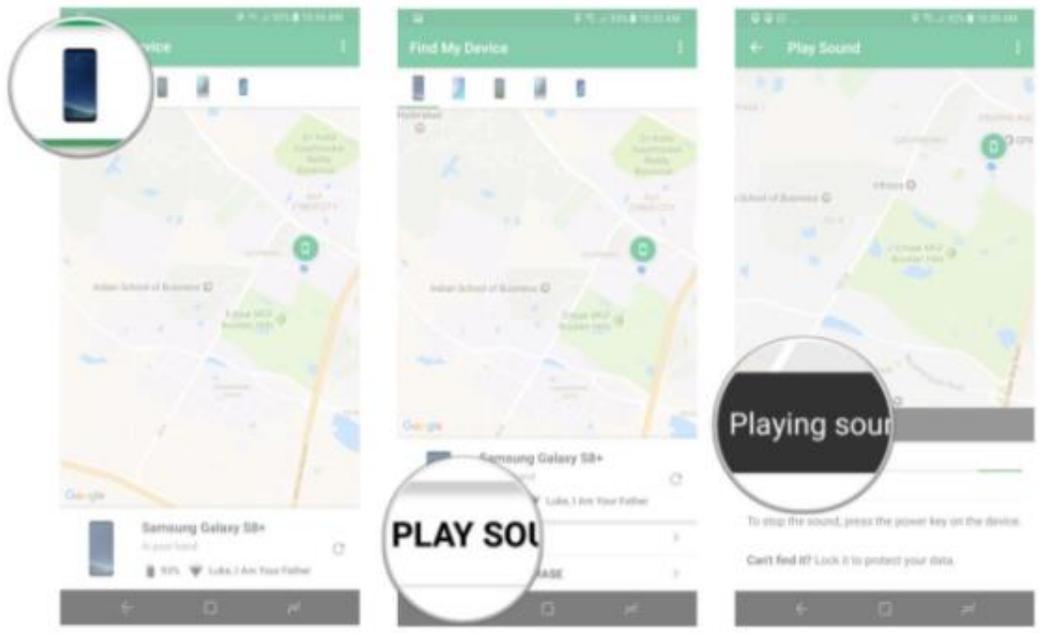
**Exhibit C for U.S. Patent No. 9,445,251 Against HTC Accused Products**

US9445251B2	HTC
	<div data-bbox="653 272 1199 695" data-label="Image"> </div> <p data-bbox="514 743 1066 764">When your phone is located, you have three options to choose from:</p> <ul data-bbox="514 800 1339 971" style="list-style-type: none"> <li>• You can <b>Ring</b> your phone so that it makes noise (even if you had it on silent). This feature is helpful if the map indicates that the phone is within earshot and you simply can't see it.</li> <li>• You can <b>Lock</b> your phone so that the finder can't access your home screen. This feature is most helpful if your phone wasn't previously secured with a passcode or a fingerprint sensor.</li> <li>• You can <b>Erase</b> your phone. This is the best option if you know for certain that you aren't likely to retrieve your phone.</li> </ul> <div data-bbox="541 995 1339 1398" data-label="Image"> </div>

**Exhibit C for U.S. Patent No. 9,445,251 Against HTC Accused Products**

US9445251B2	HTC
	<p data-bbox="499 235 1251 264"><a href="https://www.androidcentral.com/how-track-android-phone">https://www.androidcentral.com/how-track-android-phone</a></p> <h2 data-bbox="512 313 1650 367">How to locate your phone over the internet</h2> <p data-bbox="512 410 1656 548">If you've lost your phone, you can remotely locate it through the <a href="#">Find My Device website</a>. You'll need to sign in to the Google account that was used to set up Find My Device. It takes a few seconds, but the service should be able to track your phone. Alternatively, you can also do a Google search for "<a href="#">find my phone</a>" to locate your handset.</p> <ol data-bbox="504 607 995 760" style="list-style-type: none"><li>1. Head to the <a href="#">Find My Device website</a>.</li><li>2. Sign in to your Google account.</li><li>3. Check if your device is visible.</li></ol> <div data-bbox="550 803 1759 1256"></div> <p data-bbox="499 1287 1119 1317"><a href="https://www.androidcentral.com/find-my-device">https://www.androidcentral.com/find-my-device</a></p>

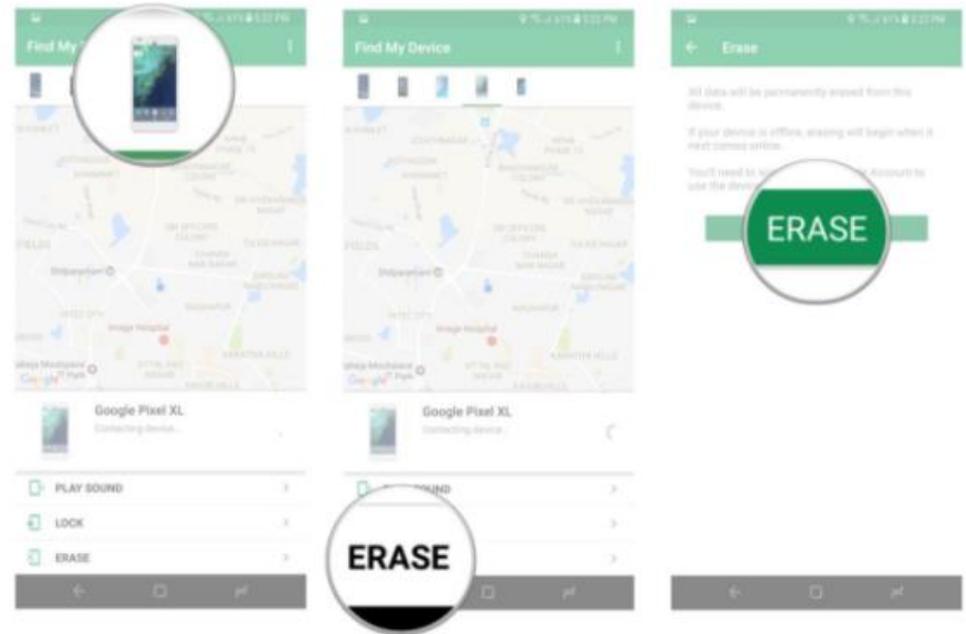
**Exhibit C for U.S. Patent No. 9,445,251 Against HTC Accused Products**

US9445251B2	HTC
	<h2 data-bbox="514 240 1522 289">How to ring your phone with Find My Device</h2> <p data-bbox="514 326 1543 479">The best part about Find My Device is that it is easily accessible. If you need to locate your phone, just head to the website or log in to the service from another phone. Once you sign in to Find My Device and locate your device, you can use the <b>Play Sound</b> option, which plays a loud tone on your phone continuously at full volume for five minutes even if you turned the ringer off. Once you find your phone, you can hit the power button to stop the ringing.</p> <ol data-bbox="514 527 1344 657" style="list-style-type: none"><li>1. Locate your phone on Find My Device.</li><li>2. Tap <b>Play Sound</b>.</li><li>3. Your device will start ringing. You can hit the power button to stop the sound.</li></ol>  <p data-bbox="493 1347 1123 1380"><a href="https://www.androidcentral.com/find-my-device">https://www.androidcentral.com/find-my-device</a></p>

**Exhibit C for U.S. Patent No. 9,445,251 Against HTC Accused Products**

US9445251B2	HTC
	<div data-bbox="506 272 1566 326" data-label="Section-Header"> <h2>How to lock your phone with Find My Device</h2> </div> <div data-bbox="506 358 1575 456" data-label="Text"> <p>There's also a <b>Lock</b> option that lets you set a new password to unlock the phone. You can also display a message over the lock screen and add a button to call back your number so that anyone that comes across your phone can easily get in touch with you.</p> </div> <div data-bbox="499 505 1409 643" data-label="List-Group"> <ol style="list-style-type: none"> <li>1. Locate your phone on <b>Find My Device</b>.</li> <li>2. Tap <b>Lock</b>.</li> <li>3. Enter a message and phone number to display on the lock screen and tap <b>Lock</b>.</li> </ol> </div> <div data-bbox="543 678 1617 1354" data-label="Image"> <p>The image consists of three sequential screenshots from the Find My Device app. The first screenshot shows the 'Find My Device' main screen with a map and a phone icon circled in a white circle. The second screenshot shows the 'Lock device' screen with a 'LOCK' button circled in a white circle. The third screenshot shows the 'Lock device' screen with a 'LOCK' button circled in a white circle.</p> </div> <div data-bbox="491 1367 1123 1404" data-label="Text"> <p><a href="https://www.androidcentral.com/find-my-device">https://www.androidcentral.com/find-my-device</a></p> </div>

**Exhibit C for U.S. Patent No. 9,445,251 Against HTC Accused Products**

US9445251B2	HTC
	<p data-bbox="514 240 1459 284"><b>How to erase your lost phone's data remotely</b></p> <p data-bbox="514 316 1470 462">If you're certain that you're not going to see your phone again, there is the nuclear option of erasing the data remotely. Selecting the <b>Erase</b> option deletes all the data on your phone. The service also deletes data from a connected SD card, but there is a chance that it may not be able to, based on the manufacturer and Android platform version. Even if your phone is switched off when you send the <b>Erase</b> command, the factory reset process will be initiated as soon as it goes online.</p> <ol data-bbox="514 495 1039 625" style="list-style-type: none"> <li>1. Locate your phone on <b>Find My Device</b>.</li> <li>2. Tap <b>Erase</b>.</li> <li>3. Confirm deletion of data by hitting the <b>Erase</b> button.</li> </ol>  <p data-bbox="493 1315 1123 1356"><a href="https://www.androidcentral.com/find-my-device">https://www.androidcentral.com/find-my-device</a></p> <p data-bbox="493 1388 1018 1421"><b>Exemplary Support for Google Maps:</b></p>

## Exhibit C for U.S. Patent No. 9,445,251 Against HTC Accused Products

US9445251B2	HTC
	<p data-bbox="537 280 953 302">COMPUTER <b>ANDROID</b> IPHONE &amp; IPAD</p> <hr data-bbox="525 326 1556 331"/> <h3 data-bbox="525 386 1010 423">If they have a Google Account</h3> <ol data-bbox="525 444 1409 737" style="list-style-type: none"><li data-bbox="525 444 1205 466">1. If you haven't already, add their Gmail address to your <a href="#">Google Contacts</a> .</li><li data-bbox="525 482 1409 503">2. On your Android phone or tablet, open the Google Maps app  and sign in. <a href="#">Learn how to sign in.</a></li><li data-bbox="525 519 1020 540">3. Tap Menu ≡ &gt; <b>Location sharing</b> &gt; Add People +.</li><li data-bbox="525 557 993 578">4. Choose how long you want to share your location.</li><li data-bbox="525 594 1125 651">5. Tap <b>Select People</b>.<ul data-bbox="556 630 1125 651" style="list-style-type: none"><li data-bbox="556 630 1125 651">• If you're asked about your contacts, give Google Maps access.</li></ul></li><li data-bbox="525 678 873 699">6. Choose who you want to share with.</li><li data-bbox="525 716 646 737">7. Tap <b>Share</b>.</li></ol> <h3 data-bbox="525 797 1094 834">If they don't have a Google Account</h3> <ol data-bbox="525 855 1545 980" style="list-style-type: none"><li data-bbox="525 855 1409 876">1. On your Android phone or tablet, open the Google Maps app  and sign in. <a href="#">Learn how to sign in.</a></li><li data-bbox="525 893 1020 914">2. Tap Menu ≡ &gt; <b>Location sharing</b> &gt; Add People +.</li><li data-bbox="525 930 1545 980">3. Tap More ... &gt; <b>Copy to clipboard</b>. People with this link can see your location for as long as you choose, up to 72 hours.</li></ol> <h3 data-bbox="525 1029 856 1066">Share using another app</h3> <p data-bbox="525 1083 1192 1104">You can also share through messaging apps. Tap More ... &gt; select an app.</p> <h3 data-bbox="525 1164 730 1201">Stop sharing</h3> <ol data-bbox="525 1222 1192 1320" style="list-style-type: none"><li data-bbox="525 1222 831 1243">1. Open the Google Maps app .</li><li data-bbox="525 1260 856 1281">2. Tap Menu ≡ &gt; <b>Location sharing</b>.</li><li data-bbox="525 1297 1192 1320">3. Next to the person with whom you want to stop sharing, tap Remove ✕ .</li></ol> <p data-bbox="499 1336 1688 1369"><a href="https://support.google.com/maps/answer/7326816?co=GENIE.Platform%3DAndroid&amp;oco=1">https://support.google.com/maps/answer/7326816?co=GENIE.Platform%3DAndroid&amp;oco=1</a></p>



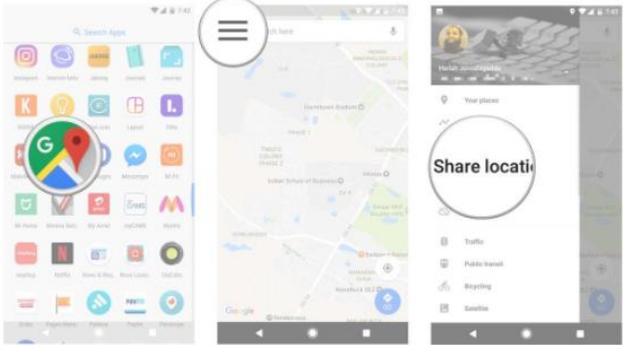
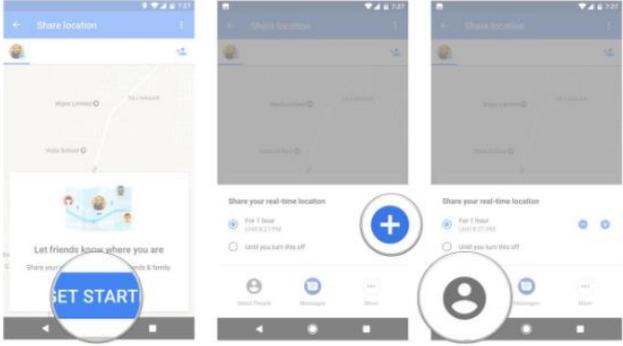
## Exhibit C for U.S. Patent No. 9,445,251 Against HTC Accused Products

US9445251B2	HTC
	<h3 data-bbox="520 256 827 298">Share your E.T.A</h3> <p data-bbox="520 326 1650 350">After you start your drive, you can share your destination, estimated arrival time, and where you are on the route.</p> <ol data-bbox="531 380 1346 618" style="list-style-type: none"><li data-bbox="531 380 873 407">1. Open the Google Maps app .</li><li data-bbox="531 423 1146 451">2. Set a driving destination. <a href="#">Learn how to navigate to a place.</a></li><li data-bbox="531 467 1188 495">3. After you start navigation, tap More ^ &gt; <b>Share trip progress.</b></li><li data-bbox="531 511 863 539">4. Choose a person from the list.</li><li data-bbox="531 555 667 583">5. Tap <b>Share.</b></li><li data-bbox="531 599 1346 626">6. Location Sharing will stop when you reach your destination or stop navigating.</li></ol> <ul data-bbox="531 651 1188 678" style="list-style-type: none"><li data-bbox="531 651 1188 678">• To stop sharing before you arrive, tap More ^ &gt; <b>Stop sharing.</b></li></ul> <h3 data-bbox="520 743 942 786">See where someone is</h3> <p data-bbox="520 813 1251 837">If someone shares their location with you, you can see them on the map.</p> <ol data-bbox="531 867 905 976" style="list-style-type: none"><li data-bbox="531 867 873 894">1. Open the Google Maps app .</li><li data-bbox="531 911 905 938">2. Tap Menu ≡ &gt; <b>Location sharing.</b></li><li data-bbox="531 954 737 982">3. Choose someone.</li></ol> <ul data-bbox="531 1008 1289 1036" style="list-style-type: none"><li data-bbox="531 1008 1289 1036">• To see an updated location, tap on a friend's icon &gt; More ⋮ &gt; <b>Refresh.</b></li></ul> <h3 data-bbox="520 1089 1020 1131">Stop seeing someone's location</h3> <ol data-bbox="531 1149 1444 1300" style="list-style-type: none"><li data-bbox="531 1149 873 1177">1. Open the Google Maps app .</li><li data-bbox="531 1193 821 1221">2. On the map, tap their icon.</li><li data-bbox="531 1237 842 1265">3. At the bottom, tap More ^ .</li><li data-bbox="531 1281 1444 1308">4. To temporarily hide someone, tap <b>Hide from map.</b> You can stop hiding them at any time.</li></ol> <p data-bbox="520 1333 1730 1357"><b>Note:</b> You can stop someone's location from ever appearing on your map. <a href="#">Learn how to block another person's account.</a></p> <p data-bbox="499 1382 1688 1409"><a href="https://support.google.com/maps/answer/7326816?co=GENIE.Platform%3DAndroid&amp;oco=1">https://support.google.com/maps/answer/7326816?co=GENIE.Platform%3DAndroid&amp;oco=1</a></p>

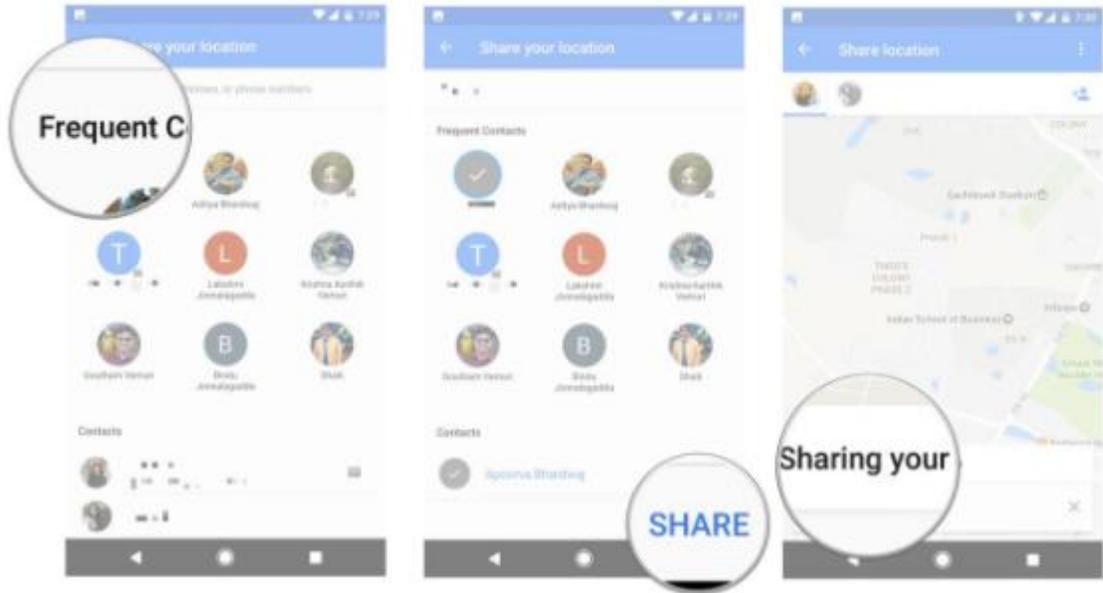
## Exhibit C for U.S. Patent No. 9,445,251 Against HTC Accused Products

US9445251B2	HTC
	<h3>Hide or share lists</h3> <p><b>Note:</b> You can't share starred places.</p> <ol style="list-style-type: none"><li>1. Open the Google Maps app .</li><li>2. Tap Menu  &gt; <b>Your places</b> &gt; <b>Saved</b>.</li><li>3. Next to the list you want to share, tap More  &gt; choose an option:<ul style="list-style-type: none"><li>• <b>Hide/Show on your map:</b> Display or hide your saved places when looking at the map.</li><li>• <b>Share list:</b> Allow others to see your saved list.</li><li>• <b>Sharing options:</b> You can make your list public, private, or shared. To let anyone with the link see your list, tap <b>Shared</b>. To let anyone find and follow your list, tap <b>Public</b>.</li></ul></li></ol> <h3>Follow a list</h3> <p>If you follow a list made by someone else, their saved places will show up in Your Places. The places will also appear as suggested locations in Google Maps.</p> <h3>Follow a list using a link</h3> <ol style="list-style-type: none"><li>1. Tap on the link you received to open it.</li><li>2. Tap <b>Follow</b>. This list will now be added to the group of lists you follow.</li><li>3. <b>Optional:</b> To unfollow a list someone shared with you, tap the list &gt; <b>Following</b>.</li></ol> <h3>See lists made by others</h3> <p>If a user has any Google Maps lists that were made public, you can follow them.</p> <ol style="list-style-type: none"><li>1. Tap on the name of a user whose list you want to follow.</li><li>2. Tap <b>Lists</b>.</li><li>3. Tap on the list you want to follow &gt; More  &gt; <b>Follow</b>.</li></ol> <p><a href="https://support.google.com/maps/answer/7280933?hl=en&amp;ref_topic=7301134&amp;co=GENIE.Platform%3DAnd">https://support.google.com/maps/answer/7280933?hl=en&amp;ref_topic=7301134&amp;co=GENIE.Platform%3DAnd</a></p>

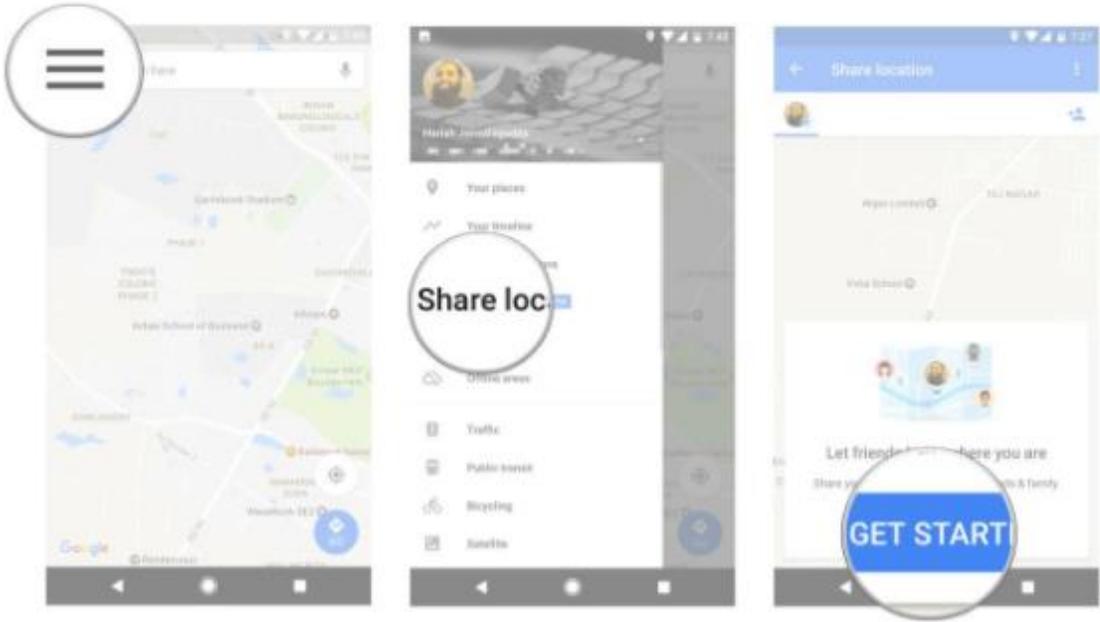
# Exhibit C for U.S. Patent No. 9,445,251 Against HTC Accused Products

US9445251B2	HTC
	<p data-bbox="499 235 655 261">roid&amp;oco=1</p> <h3 data-bbox="506 313 1094 342">How to share your location in Google Maps</h3> <ol data-bbox="506 370 1075 451" style="list-style-type: none"><li>1. Open Google Maps from the app drawer or the home screen.</li><li>2. Tap the <b>hamburger menu</b> (the three horizontal lines) on the top left corner of the screen.</li><li>3. Select <b>Share location</b>.</li></ol>  <ol data-bbox="506 857 1104 959" style="list-style-type: none"><li>4. Tap <b>Get Started</b>.</li><li>5. Use the + icon to select a time period or select the <b>Until you turn this off</b> setting to share your location indefinitely.</li><li>6. Tap <b>Select People</b>.</li></ol>  <p data-bbox="499 1333 1346 1365"><a href="https://www.androidcentral.com/how-share-location-google-maps">https://www.androidcentral.com/how-share-location-google-maps</a></p>

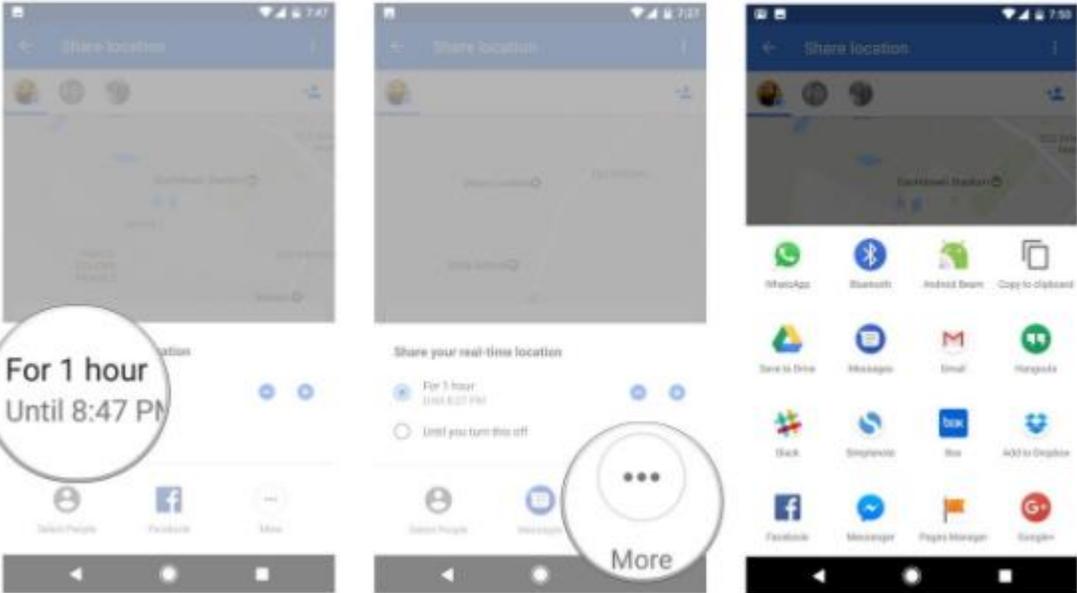
**Exhibit C for U.S. Patent No. 9,445,251 Against HTC Accused Products**

US9445251B2	HTC
	<p data-bbox="514 251 1564 430"><b>7.</b> You'll see a list of your frequent contacts at the top, along with a full list of contacts. <b>Pick the contacts</b> by tapping their name.</p> <p data-bbox="514 341 1438 365"><b>8.</b> Once you've selected the contacts you want to share your location to, tap <b>Share</b>.</p> <p data-bbox="514 397 1396 422"><b>9.</b> You'll see a message saying that the selected contact can view your location.</p>  <p data-bbox="493 1096 1344 1136"><a href="https://www.androidcentral.com/how-share-location-google-maps">https://www.androidcentral.com/how-share-location-google-maps</a></p>

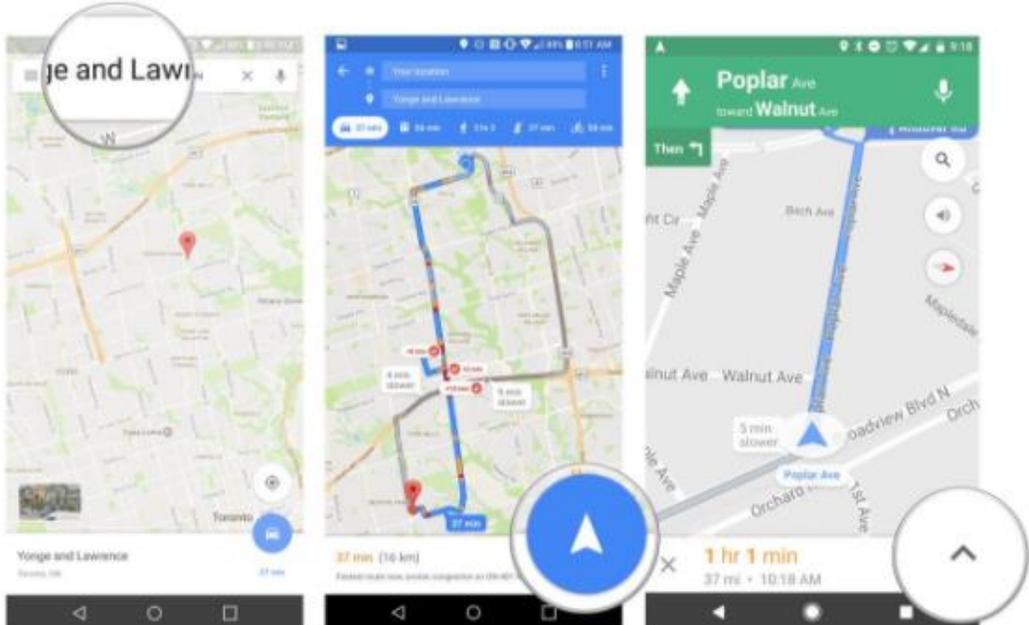
## Exhibit C for U.S. Patent No. 9,445,251 Against HTC Accused Products

US9445251B2	HTC
	<h3 data-bbox="512 245 1243 293">How to create a shareable link</h3> <p data-bbox="512 334 1451 362">You can also create a link and use it to share your location easily. Here's how to do it:</p> <ol data-bbox="506 412 1224 553" style="list-style-type: none"><li>1. Tap the <b>hamburger menu</b> on the top left corner of the screen.</li><li>2. Select <b>Share location</b>.</li><li>3. Tap <b>Get Started</b>.</li></ol>  <p data-bbox="499 1235 1346 1268"><a href="https://www.androidcentral.com/how-share-location-google-maps">https://www.androidcentral.com/how-share-location-google-maps</a></p>

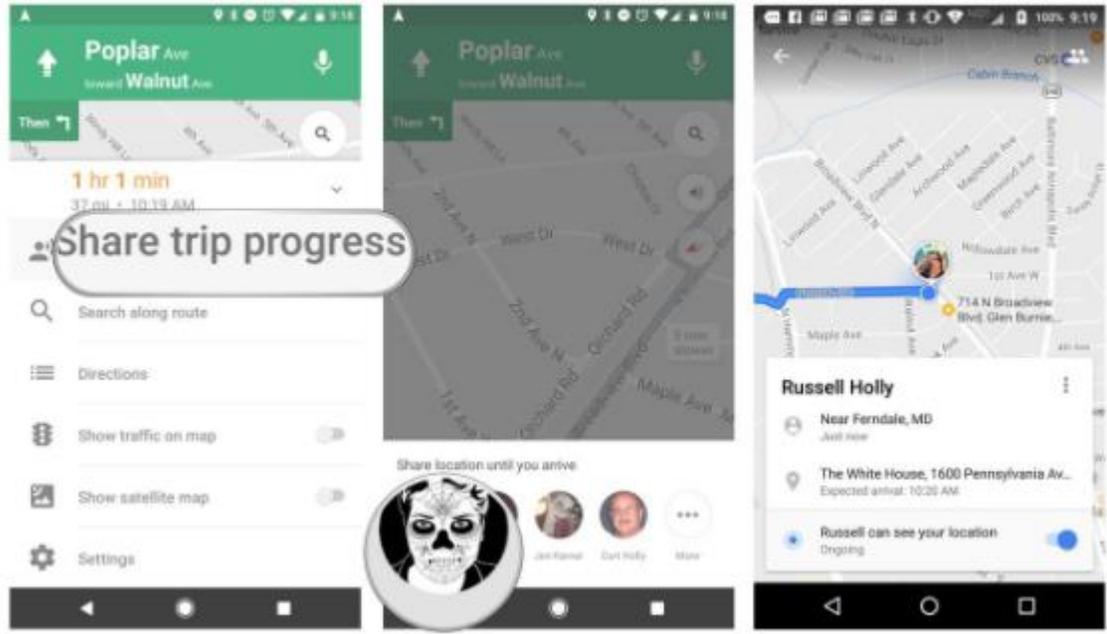
**Exhibit C for U.S. Patent No. 9,445,251 Against HTC Accused Products**

US9445251B2	HTC
	<p>4. Select the <b>amount of time</b> you want to share your location.</p> <p>5. Tap <b>More</b>.</p> <p>6. Select <b>your app of choice</b> to create and send a <b>unique URL</b> that broadcasts your current location. You can email it, send the link via Messenger, or even tweet it to the <b>intended recipient</b>.</p>  <p><a href="https://www.androidcentral.com/how-share-location-google-maps">https://www.androidcentral.com/how-share-location-google-maps</a></p>

**Exhibit C for U.S. Patent No. 9,445,251 Against HTC Accused Products**

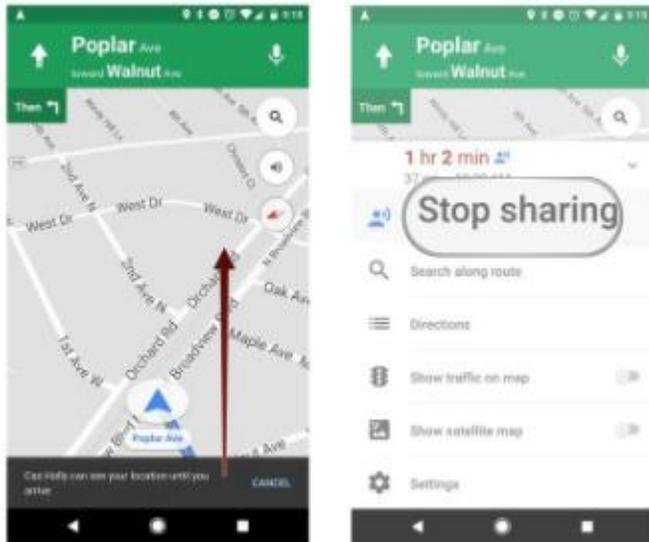
US9445251B2	HTC
	<h2 data-bbox="514 240 1417 337">How to share your navigation directions while you walk, drive or transit</h2> <p data-bbox="514 375 1543 467">One of the best ways to share your location with someone is during a drive, walk or transit. If you're meeting a friend or family member somewhere, or navigating towards their home, Google Maps lets you share your location with them for the duration of the trip. It's magic!</p> <ol data-bbox="514 511 1386 646" style="list-style-type: none"><li>1. In the <b>search bar</b> enter your destination.</li><li>2. Pick your navigation type (drive, transit, walk) and press the blue <b>navigate button</b>.</li><li>3. Tap the arrow next to the <b>time-to-destination</b> number at the bottom of the screen.</li></ol>  <p data-bbox="499 1328 1344 1360"><a href="https://www.androidcentral.com/how-share-location-google-maps">https://www.androidcentral.com/how-share-location-google-maps</a></p>

**Exhibit C for U.S. Patent No. 9,445,251 Against HTC Accused Products**

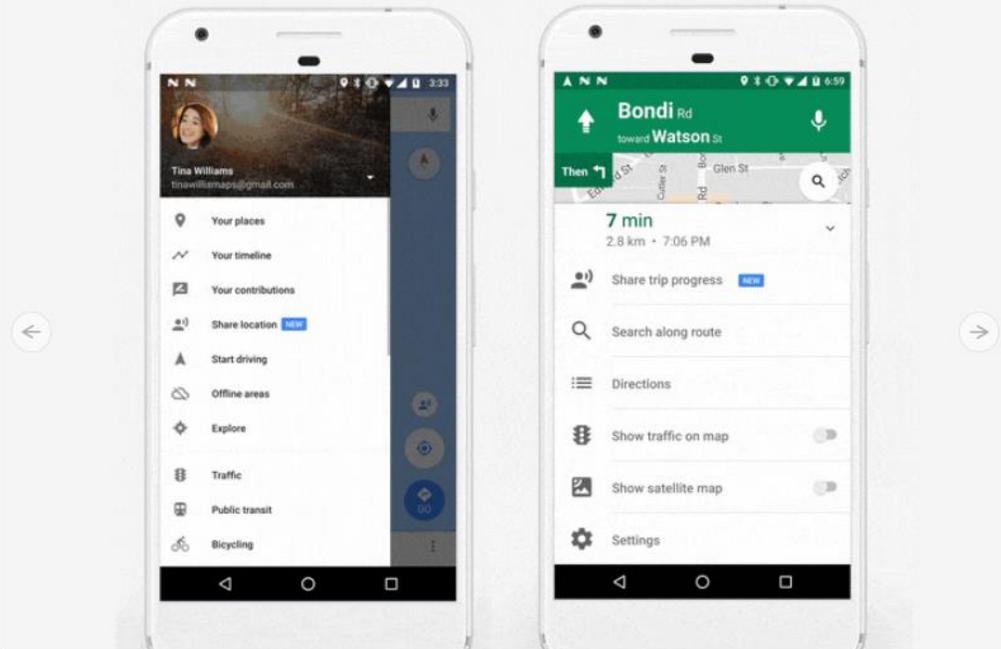
US9445251B2	HTC
	<p data-bbox="514 277 823 305">4. Tap Share trip progress.</p> <p data-bbox="514 334 1136 362">5. Choose one or more contacts to share trip progress.</p>  <p data-bbox="520 1063 1333 1091">You can also stop sharing your location with someone before a trip ends.</p> <p data-bbox="499 1101 1346 1133"><a href="https://www.androidcentral.com/how-share-location-google-maps">https://www.androidcentral.com/how-share-location-google-maps</a></p>



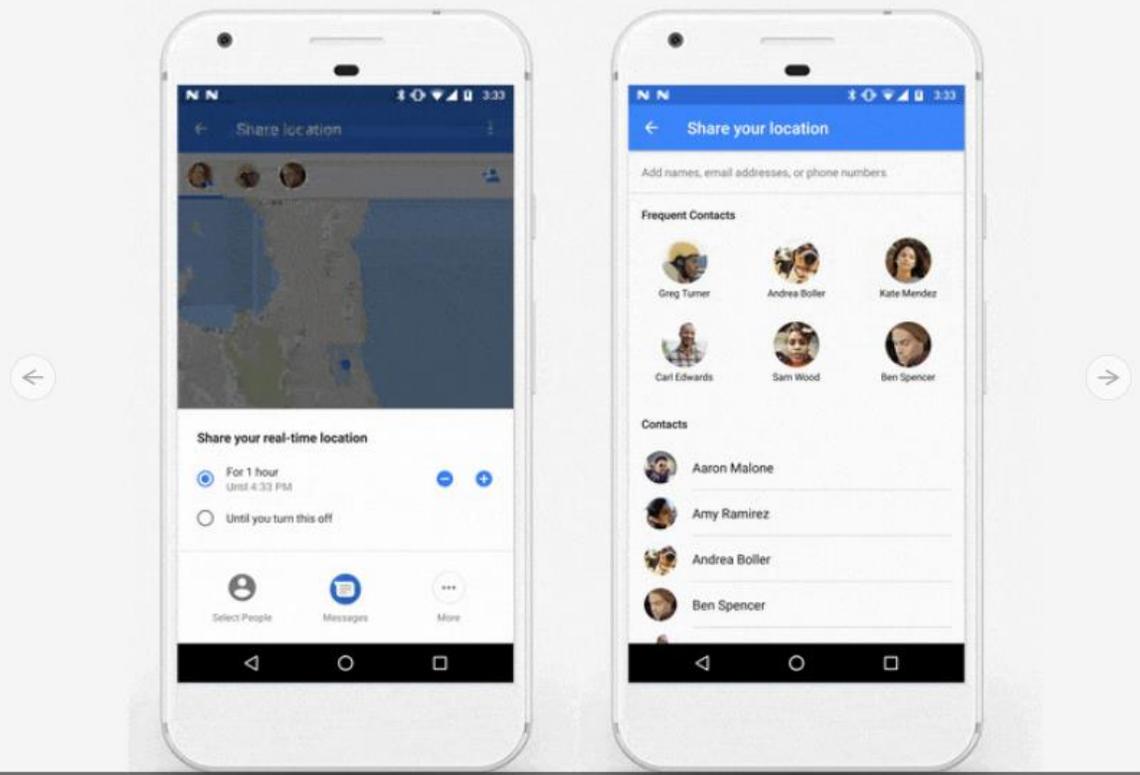
## Exhibit C for U.S. Patent No. 9,445,251 Against HTC Accused Products

US9445251B2	HTC
	<ol style="list-style-type: none"><li data-bbox="520 245 1457 272">1. Tap the arrow next to the <b>time-to-destination</b> number at the bottom of the screen.</li><li data-bbox="520 302 758 329">2. Tap <b>Stop sharing</b>.</li></ol> <div data-bbox="732 380 1381 922"></div> <p data-bbox="531 976 625 1003">That's it!</p> <p data-bbox="531 1045 1598 1073">Are you excited that location sharing is back in Google Maps? How often do you use the feature?</p> <p data-bbox="499 1086 1346 1114"><a href="https://www.androidcentral.com/how-share-location-google-maps">https://www.androidcentral.com/how-share-location-google-maps</a></p> <p data-bbox="499 1195 1409 1222">As shown below, a group may also be defined within Google Contacts.</p>

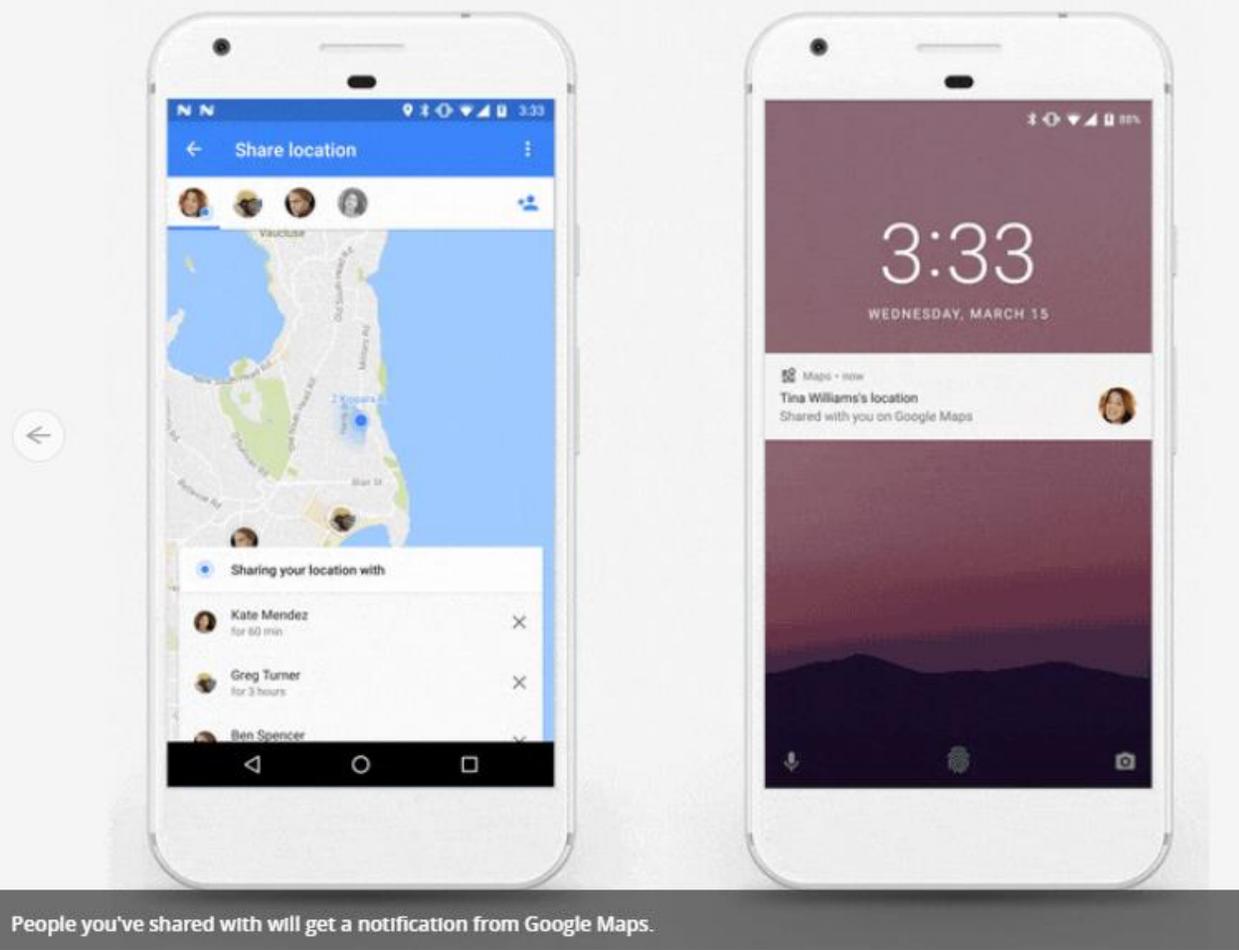
# Exhibit C for U.S. Patent No. 9,445,251 Against HTC Accused Products

US9445251B2	HTC
	<h2 data-bbox="535 245 930 293">Share your contacts</h2> <ol data-bbox="548 318 1031 480" style="list-style-type: none"><li>1. Open your device's Contacts app .</li><li>2. Tap a contact in the list.</li><li>3. Tap More  &gt; <b>Share</b>.</li><li>4. Choose how you want to share the contact.</li></ol> <p data-bbox="499 496 1524 529"><a href="https://support.google.com/android/answer/6118731?hl=en&amp;ref_topic=6118711">https://support.google.com/android/answer/6118731?hl=en&amp;ref_topic=6118711</a></p>  <p data-bbox="512 1235 1507 1284">Ways to start location sharing: (left) from the app navigation drawer, and (right) from the menu, you can drag up while navigating somewhere.</p> <p data-bbox="499 1300 1650 1328"><a href="https://arstechnica.com/gadgets/2017/03/location-sharing-finally-returns-to-google-maps/">https://arstechnica.com/gadgets/2017/03/location-sharing-finally-returns-to-google-maps/</a></p>

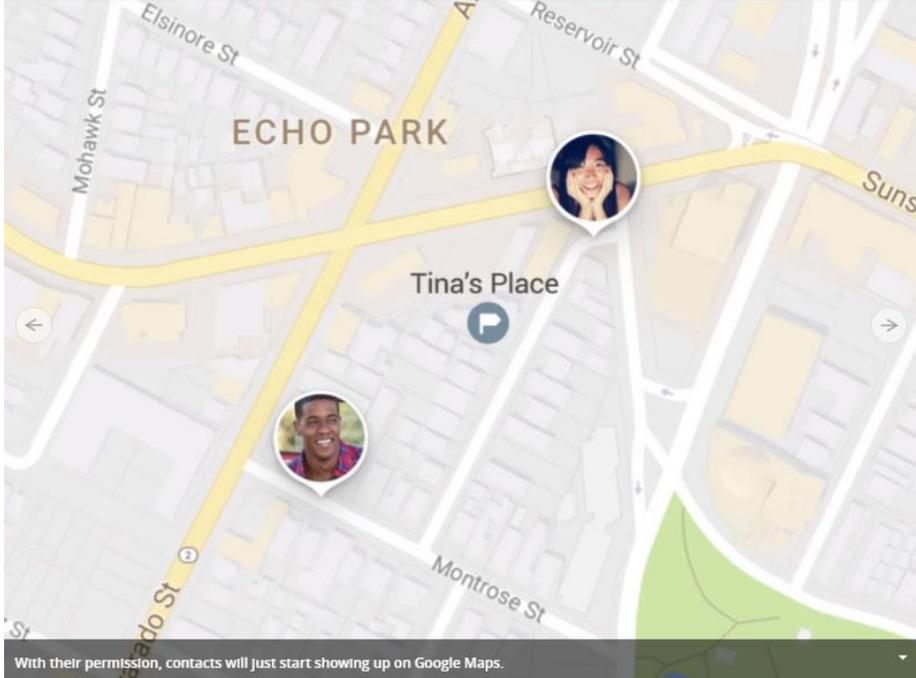
**Exhibit C for U.S. Patent No. 9,445,251 Against HTC Accused Products**

US9445251B2	HTC
	 <p data-bbox="506 1027 1646 1060">You can pick how long you want to share your location for, and with whom you want to share it.</p> <p data-bbox="506 1068 1646 1092"><a href="https://arstechnica.com/gadgets/2017/03/location-sharing-finally-returns-to-google-maps/">https://arstechnica.com/gadgets/2017/03/location-sharing-finally-returns-to-google-maps/</a></p>

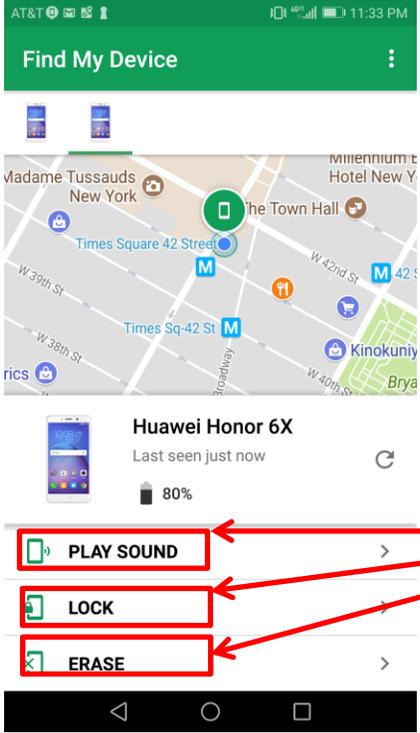
**Exhibit C for U.S. Patent No. 9,445,251 Against HTC Accused Products**

US9445251B2	HTC
	 <p data-bbox="506 1146 1163 1174">People you've shared with will get a notification from Google Maps.</p> <p data-bbox="499 1192 1646 1224"><a href="https://arstechnica.com/gadgets/2017/03/location-sharing-finally-returns-to-google-maps/">https://arstechnica.com/gadgets/2017/03/location-sharing-finally-returns-to-google-maps/</a></p>

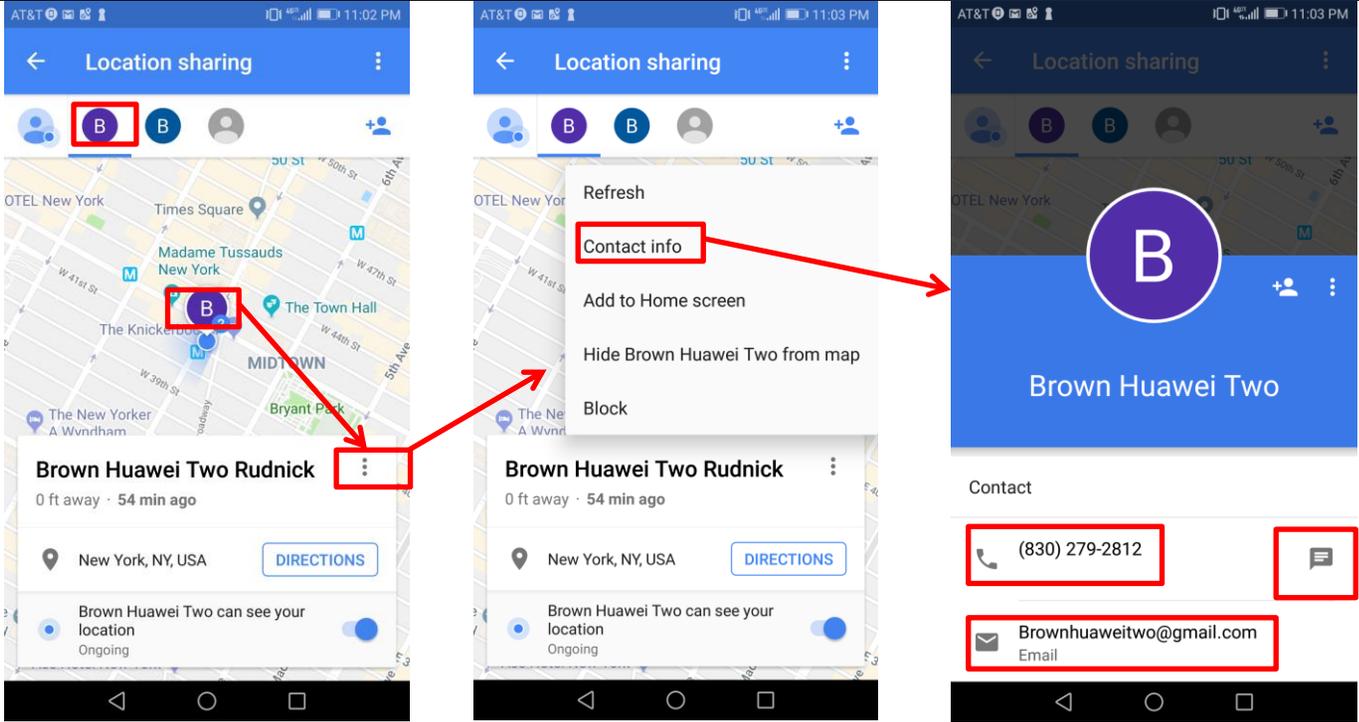
**Exhibit C for U.S. Patent No. 9,445,251 Against HTC Accused Products**

US9445251B2	HTC
	 <p data-bbox="506 889 1003 906">With their permission, contacts will just start showing up on Google Maps.</p> <p data-bbox="499 915 1646 948"><a href="https://arstechnica.com/gadgets/2017/03/location-sharing-finally-returns-to-google-maps/">https://arstechnica.com/gadgets/2017/03/location-sharing-finally-returns-to-google-maps/</a></p> <p data-bbox="499 987 1052 1019"><b><u>Exemplary Find My Device Screenshots:</u></b></p>

**Exhibit C for U.S. Patent No. 9,445,251 Against HTC Accused Products**

US9445251B2	HTC
	 <p>The screenshot shows the 'Find My Device' app interface. At the top, there's a green header with the text 'Find My Device'. Below it is a map of Times Square in New York City. A green location pin is placed on the map. Below the map, there's a card for a 'Huawei Honor 6X' with the text 'Last seen just now' and a battery icon showing '80%'. Below the card are three action buttons: 'PLAY SOUND', 'LOCK', and 'ERASE'. Each button has a red box around it, and a red arrow points from a box on the right labeled 'Exemplary communications' to each of these buttons. At the bottom of the screenshot is an Android navigation bar. Below the screenshot, the text '<u>Exemplary Maps Screenshots:</u>' is written.</p> <p><b>Exemplary Maps Screenshots:</b></p>

**Exhibit C for U.S. Patent No. 9,445,251 Against HTC Accused Products**

US9445251B2	HTC
	 <p><b>Exemplary Source Code:</b></p> <p>The above functionality is performed at least in part by the following publicly available source code and/or source code that invokes or is invoked by the following source code (or a substantially similar copy compiled and loaded onto the Accused Products by <b>HTC</b>). AGIS reserves the right to supplement these contentions with additional source code as discovery progresses and as additional source code is made available. AGIS reserves the right to supplement these contentions with additional source code as discovery progresses and as additional source code is made available.</p>

**Exhibit C for U.S. Patent No. 9,445,251 Against HTC Accused Products**

US9445251B2	HTC
	<pre>44 * Class that sends chat message via SMS. 45 * 46 * The interface emulates a blocking sending similar to making an HTTP request. 47 * It calls the SmsManager to send a (potentially multipart) message and waits 48 * on the sent status on each part. The waiting has a timeout so it won't wait 49 * forever. Once the sent status of all parts received, the call returns. 50 * A successful sending requires success status for all parts. Otherwise, we 51 * pick the highest level of failure as the error for the whole message, which 52 * is used to determine if we need to retry the sending. 53 */ 54 public class SmsSender { 55     private static final String TAG = LogUtil.BUGLE_TAG; 56 57     public static final String EXTRA_PART_ID = "part_id"; 58 59     /* 60      * A map for pending sms messages. The key is the random request UUID. 61      */ 62     private static ConcurrentHashMap&lt;Uri, SendResult&gt; sPendingMessageMap = 63         new ConcurrentHashMap&lt;Uri, SendResult&gt;(); 64 65     private static final Random RANDOM = new Random(); 66 67     // Whether we should send multipart SMS as separate messages 68     private static Boolean sSendMultipartSmsAsSeparateMessages = null; 69</pre> <p><a href="https://android.googlesource.com/platform/packages/apps/Messaging/+/nougat-mr1-release/src/com/android/messaging/sms/SmsSender.java">https://android.googlesource.com/platform/packages/apps/Messaging/+/nougat-mr1-release/src/com/android/messaging/sms/SmsSender.java</a></p>



**Exhibit C for U.S. Patent No. 9,445,251 Against HTC Accused Products**

US9445251B2	HTC
	<pre>253 // Actually sending the message using SmsManager 254 private static void sendInternal(final Context context, final int subId, String dest, 255     final ArrayList&lt;String&gt; messages, final String serviceCenter, 256     final boolean requireDeliveryReport, final Uri messageUri) throws SmsException { 257     Assert.notNull(context); 258     final SmsManager smsManager = PhoneUtils.get(subId).getSmsManager(); 259     final int messageCount = messages.size(); 260     final ArrayList&lt;PendingIntent&gt; deliveryIntents = new ArrayList&lt;PendingIntent&gt;(messageCount); 261     final ArrayList&lt;PendingIntent&gt; sentIntents = new ArrayList&lt;PendingIntent&gt;(messageCount); 262     for (int i = 0; i &lt; messageCount; i++) { 263         // Make pending intents different for each message part 264         final int partId = (messageCount &lt;= 1 ? 0 : i + 1); 265         if (requireDeliveryReport &amp;&amp; (i == (messageCount - 1))) { 266             // TODO we only care about the delivery status of the last part 267             // Shall we have better tracking of delivery status of all parts? 268             deliveryIntents.add(PendingIntent.getBroadcast( 269                 context, 270                 partId, 271                 getSendStatusIntent(context, SendStatusReceiver.MESSAGE_DELIVERED_ACTION, 272                     messageUri, partId, subId), 273                 0/*flag*/)); 274         } else { 275             deliveryIntents.add(null); 276         } 277         sentIntents.add(PendingIntent.getBroadcast( 278             context, 279             partId, 280             getSendStatusIntent(context, SendStatusReceiver.MESSAGE_SENT_ACTION, 281                 messageUri, partId, subId), 282             0/*flag*/)); 283     } 284     if (sSendMultipartSmsAsSeparateMessages == null) { 285         sSendMultipartSmsAsSeparateMessages = MmsConfig.get(subId) 286             .getSendMultipartSmsAsSeparateMessages(); 287     }</pre> <p><a href="https://android.googlesource.com/platform/packages/apps/Messaging/+nougat-mr1-release/src/com/android/messaging/sms/SmsSender.java">https://android.googlesource.com/platform/packages/apps/Messaging/+nougat-mr1-release/src/com/android/messaging/sms/SmsSender.java</a></p>

**Exhibit C for U.S. Patent No. 9,445,251 Against HTC Accused Products**

US9445251B2	HTC
	<pre>288     try { 289         if (sSendMultipartSmsAsSeparateMessages) { 290             // If multipart sms is not supported, send them as separate messages 291             for (int i = 0; i &lt; messageCount; i++) { 292                 smsManager.sendMessage(dest, 293                     serviceCenter, 294                     messages.get(i), 295                     sentIntents.get(i), 296                     deliveryIntents.get(i)); 297             } 298         } else { 299             smsManager.sendMultipartTextMessage( 300                 dest, serviceCenter, messages, sentIntents, deliveryIntents); 301         } 302     } catch (final Exception e) { 303         throw new SmsException("SmsSender: caught exception in sending " + e); 304     } 305 }</pre> <p><a href="https://android.googlesource.com/platform/packages/apps/Messaging+/nougat-mr1-release/src/com/android/messaging/sms/SmsSender.java">https://android.googlesource.com/platform/packages/apps/Messaging+/nougat-mr1-release/src/com/android/messaging/sms/SmsSender.java</a></p> <pre>56  * Class that receives incoming SMS messages through android.provider.Telephony.SMS_RECEIVED 57  * 58  * This class serves two purposes: 59  * - Process phone verification SMS messages 60  * - Handle SMS messages when the user has enabled us to be the default SMS app (Pre-KLP) 61  */ 62  public final class SmsReceiver extends BroadcastReceiver { 63      private static final String TAG = LogUtil.BUGLE_TAG; 64 65      private static ArrayList&lt;Pattern&gt; sIgnoreSmsPatterns; 66  }</pre> <p><a href="https://android.googlesource.com/platform/packages/apps/Messaging+/nougat-mr1-release/src/com/android/messaging/receiver/SmsReceiver.java">https://android.googlesource.com/platform/packages/apps/Messaging+/nougat-mr1-release/src/com/android/messaging/receiver/SmsReceiver.java</a></p>

**Exhibit C for U.S. Patent No. 9,445,251 Against HTC Accused Products**

US9445251B2	HTC
	<pre>200     public static void deliverSmsMessages(final Context context, final int subId, 201         final int errorCode, final android.telephony.SmsMessage[] messages) { 202         final ContentValues messageValues = 203             MmsUtils.parseReceivedSmsMessage(context, messages, errorCode); 204 205         LogUtil.v(TAG, "SmsReceiver.deliverSmsMessages"); 206 207         final long nowInMillis = System.currentTimeMillis(); 208         final long receivedTimestampMs = MmsUtils.getMessageDate(messages[0], nowInMillis); 209 210         messageValues.put(Sms.Inbox.DATE, receivedTimestampMs); 211         // Default to unread and unseen for us but ReceiveSmsMessageAction will override 212         // seen for the telephony db. 213         messageValues.put(Sms.Inbox.READ, 0); 214         messageValues.put(Sms.Inbox.SEEN, 0); 215         if (OsUtil.isAtLeastL_MR1()) { 216             messageValues.put(Sms.SUBSCRIPTION_ID, subId); 217         } 218 219         if (messages[0].getMessageClass() == android.telephony.SmsMessage.MessageClass.CLASS_0    220             DebugUtils.debugClassZeroSmsEnabled()) { 221             Factory.get().getUIIntents().launchClassZeroActivity(context, messageValues); 222         } else { 223             final ReceiveSmsMessageAction action = new ReceiveSmsMessageAction(messageValues); 224             action.start(); 225         } 226     }</pre> <p><a href="https://android.goesource.com/platform/packages/apps/Messaging+/nougat-mr1-release/src/com/android/messaging/receiver/SmsReceiver.java">https://android.goesource.com/platform/packages/apps/Messaging+/nougat-mr1-release/src/com/android/messaging/receiver/SmsReceiver.java</a></p>

**Exhibit C for U.S. Patent No. 9,445,251 Against HTC Accused Products**

US9445251B2	HTC
	<pre>228     @Override 229     public void onReceive(final Context context, final Intent intent) { 230         LogUtil.v(TAG, "SmsReceiver.onReceive " + intent); 231         // On KLP+ we only take delivery of SMS messages in SmsDeliverReceiver. 232         if (PhoneUtils.getDefault().isSmsEnabled()) { 233             final String action = intent.getAction(); 234             if (OsUtil.isSecondaryUser() &amp;&amp; 235                 (Telephony.Sms.Intents.SMS_RECEIVED_ACTION.equals(action)    236                  // TODO: update this with the actual constant from Telephony 237                  "android.provider.Telephony.MMS_DOWNLOADED".equals(action))) { 238                 postNewMessageSecondaryUserNotification(); 239             } else if (!OsUtil.isAtLeastKLP()) { 240                 deliverSmsIntent(context, intent); 241             } 242         } 243     }</pre> <p><a href="https://android.googlesource.com/platform/packages/apps/Messaging/+nougat-mr1-release/src/com/android/messaging/receiver/SmsReceiver.java">https://android.googlesource.com/platform/packages/apps/Messaging/+nougat-mr1-release/src/com/android/messaging/receiver/SmsReceiver.java</a></p>

**Exhibit C for U.S. Patent No. 9,445,251 Against HTC Accused Products**

US9445251B2	HTC
	<pre>52  * Class that sends chat message via MMS. 53  * 54  * The interface emulates a blocking send similar to making an HTTP request. 55  */ 56  public class MmsSender { 57      private static final String TAG = LogUtil.BUGLE_TAG; 58 59      /** 60       * Send an MMS message. 61       * 62       * @param context Context 63       * @param messageUri The unique URI of the message for identifying it during sending 64       * @param sendReq The SendReq PDU of the message 65       * @throws MmsFailureException 66       */ 67      public static void sendMms(final Context context, final int subId, final Uri messageUri, 68                               final SendReq sendReq, final Bundle sentIntentExtras) throws MmsFailureException { 69          sendMms(context, 70                 subId, 71                 messageUri, 72                 null /* locationUri */, 73                 sendReq, 74                 true /* responseImportant */, 75                 sentIntentExtras); 76      }</pre> <p><a href="https://android.googlesource.com/platform/packages/apps/Messaging/+nougat-mr1-release/src/com/android/messaging/sms/MmsSender.java">https://android.googlesource.com/platform/packages/apps/Messaging/+nougat-mr1-release/src/com/android/messaging/sms/MmsSender.java</a></p>

**Exhibit C for U.S. Patent No. 9,445,251 Against HTC Accused Products**

US9445251B2	HTC
	<pre>240      * Download an MMS message. 241      * 242      * @param context Context 243      * @param contentLocation The url of the MMS message 244      * @throws MmsFailureException 245      * @throws InvalidHeaderValueException 246      */ 247      public static void downloadMms(final Context context, final int subId, 248          final String contentLocation, Bundle extras) throws MmsFailureException, 249          InvalidHeaderValueException { 250          final Uri requestUri = Uri.parse(contentLocation); 251          final Uri contentUri = MmsFileProvider.buildRawMmsUri(); 252 253          final Intent downloadedIntent = new Intent(SendStatusReceiver.MMS_DOWNLOADED_ACTION, 254              requestUri, 255              context, 256              SendStatusReceiver.class); 257          downloadedIntent.putExtra(SendMessageAction.EXTRA_CONTENT_URI, contentUri); 258          if (extras != null) { 259              downloadedIntent.putExtras(extras); 260          } 261          final PendingIntent downloadedPendingIntent = PendingIntent.getBroadcast( 262              context, 263              0 /*request code*/, 264              downloadedIntent, 265              PendingIntent.FLAG_UPDATE_CURRENT); 266 267          MmsManager.downloadMultimediaMessage(subId, context, contentLocation, contentUri, 268              downloadedPendingIntent); 269      }</pre> <p><a href="https://android.googlesource.com/platform/packages/apps/Messaging/+nougat-mr1-release/src/com/android/messaging/sms/MmsSender.java">https://android.googlesource.com/platform/packages/apps/Messaging/+nougat-mr1-release/src/com/android/messaging/sms/MmsSender.java</a></p>

**Exhibit C for U.S. Patent No. 9,445,251 Against HTC Accused Products**

US9445251B2	HTC
	<pre> 97      * Execute an MMS HTTP request, either a POST (sending) or a GET (downloading) 98      * 99      * @param urlString The request URL, for sending it is usually the MMSC, and for downloading 100     *           it is the message URL 101     * @param pdu For POST (sending) only, the PDU to send 102     * @param method HTTP method, POST for sending and GET for downloading 103     * @param isProxySet Is there a proxy for the MMSC 104     * @param proxyHost The proxy host 105     * @param proxyPort The proxy port 106     * @param mmsConfig The MMS config to use 107     * @param userAgent The user agent header value 108     * @param uaProfUrl The UA Prof URL header value 109     * @return The HTTP response body 110     * @throws MmsHttpException For any failures 111     */ 112     public byte[] execute(String urlString, byte[] pdu, String method, boolean isProxySet, 113                          String proxyHost, int proxyPort, Bundle mmsConfig, String userAgent, String uaProfUrl) 114         throws MmsHttpException { 115         Log.d(MmsService.TAG, "HTTP: " + method + " " + Utils.redactUrlForNonVerbose(urlString) 116             + (isProxySet ? (" proxy=" + proxyHost + ":" + proxyPort) : "") 117             + ", PDU size=" + (pdu != null ? pdu.length : 0)); 118         checkMethod(method); 119         HttpURLConnection connection = null; 120         try { 121             Proxy proxy = Proxy.NO_PROXY; 122             if (isProxySet) { 123                 proxy = new Proxy(Proxy.Type.HTTP, new InetSocketAddress(proxyHost, proxyPort)); 124             } 125             final URL url = new URL(urlString); 126             // Now get the connection 127             connection = (HttpURLConnection) url.openConnection(proxy); 128             connection.setDoInput(true); 129             connection.setConnectTimeout( 130                 mmsConfig.getInt(CarrierConfigValuesLoader.CONFIG_HTTP_SOCKET_TIMEOUT, 131                                 CarrierConfigValuesLoader.CONFIG_HTTP_SOCKET_TIMEOUT_DEFAULT)); </pre> <p><a href="https://android.googlesource.com/platform/packages/apps/Messaging/+nougat-mr1-release/src/android/support/v7/mms/MmsHttpClient.java">https://android.googlesource.com/platform/packages/apps/Messaging/+nougat-mr1-release/src/android/support/v7/mms/MmsHttpClient.java</a></p>

**Exhibit C for U.S. Patent No. 9,445,251 Against HTC Accused Products**

US9445251B2	HTC
	<pre> 132 // ----- COMMON HEADERS ----- 133 // Header: Accept 134 connection.setRequestProperty(HEADER_ACCEPT, HEADER_VALUE_ACCEPT); 135 // Header: Accept-Language 136 connection.setRequestProperty( 137     HEADER_ACCEPT_LANGUAGE, getCurrentAcceptLanguage(Locale.getDefault())); 138 // Header: User-Agent 139 Log.i(MmsService.TAG, "HTTP: User-Agent=" + userAgent); 140 connection.setRequestProperty(HEADER_USER_AGENT, userAgent); 141 // Header: x-wap-profile 142 final String uaProfUrlTagName = mmsConfig.getString( 143     CarrierConfigValuesLoader.CONFIG_UA_PROF_TAG_NAME, 144     CarrierConfigValuesLoader.CONFIG_UA_PROF_TAG_NAME_DEFAULT); 145 if (uaProfUrl != null) { 146     Log.i(MmsService.TAG, "HTTP: UaProfUrl=" + uaProfUrl); 147     connection.setRequestProperty(uaProfUrlTagName, uaProfUrl); 148 } 149 // Add extra headers specified by mms_config.xml's httpparams 150 addExtraHeaders(connection, mmsConfig); 151 // Different stuff for GET and POST 152 if (METHOD_POST.equals(method)) { 153     if (pdu == null    pdu.length &lt; 1) { 154         Log.e(MmsService.TAG, "HTTP: empty pdu"); 155         throw new MmsHttpException(0/*statusCode*/, "Sending empty PDU"); 156     } 157     connection.setDoOutput(true); 158     connection.setRequestMethod(METHOD_POST); 159     if (mmsConfig.getBoolean( 160         CarrierConfigValuesLoader.CONFIG_SUPPORT_HTTP_CHARSET_HEADER, 161         CarrierConfigValuesLoader.CONFIG_SUPPORT_HTTP_CHARSET_HEADER_DEFAULT)) { 162         connection.setRequestProperty(HEADER_CONTENT_TYPE, 163             HEADER_VALUE_CONTENT_TYPE_WITH_CHARSET); 164     } else { 165         connection.setRequestProperty(HEADER_CONTENT_TYPE, 166             HEADER_VALUE_CONTENT_TYPE_WITHOUT_CHARSET); </pre> <p><a href="https://android.gogglesource.com/platform/packages/apps/Messaging+/nougat-mr1-release/src/android/support/v7/mms/MmsHttpClient.java">https://android.gogglesource.com/platform/packages/apps/Messaging+/nougat-mr1-release/src/android/support/v7/mms/MmsHttpClient.java</a></p>



**Exhibit C for U.S. Patent No. 9,445,251 Against HTC Accused Products**

US9445251B2	HTC
	<pre> 167         } 168         if (Log.isLoggable(MmsService.TAG, Log.VERBOSE)) { 169             logHttpHeaders(connection.getRequestProperties()); 170         } 171         connection.setFixedLengthStreamingMode(pdu.length); 172         // Sending request body 173         final OutputStream out = 174             new BufferedOutputStream(connection.getOutputStream()); 175         out.write(pdu); 176         out.flush(); 177         out.close(); 178     } else if (METHOD_GET.equals(method)) { 179         if (Log.isLoggable(MmsService.TAG, Log.VERBOSE)) { 180             logHttpHeaders(connection.getRequestProperties()); 181         } 182         connection.setRequestMethod(METHOD_GET); 183     } 184     // Get response 185     final int responseCode = connection.getResponseCode(); 186     final String responseMessage = connection.getResponseMessage(); 187     Log.d(MmsService.TAG, "HTTP: " + responseCode + " " + responseMessage); 188     if (Log.isLoggable(MmsService.TAG, Log.VERBOSE)) { 189         logHttpHeaders(connection.getHeaderFields()); 190     } 191     if (responseCode / 100 != 2) { 192         throw new MmsHttpException(responseCode, responseMessage); 193     } 194     final InputStream in = new BufferedInputStream(connection.getInputStream()); 195     final ByteArrayOutputStream byteOut = new ByteArrayOutputStream(); 196     final byte[] buf = new byte[4096]; 197     int count = 0; 198     while ((count = in.read(buf)) &gt; 0) { 199         byteOut.write(buf, 0, count); 200     } 201     in.close(); 202     final byte[] responseBody = byteOut.toByteArray(); 203     Log.d(MmsService.TAG, "HTTP: response size=" 204         + (responseBody != null ? responseBody.length : 0)); 205     return responseBody; </pre> <p><a href="https://android.googlesource.com/platform/packages/apps/Messaging+/nougat-mr1-release/src/android/support/v7/mms/MmsHttpClient.java">https://android.googlesource.com/platform/packages/apps/Messaging+/nougat-mr1-release/src/android/support/v7/mms/MmsHttpClient.java</a></p>

**Exhibit C for U.S. Patent No. 9,445,251 Against HTC Accused Products**

US9445251B2	HTC
	<pre>38  * Request to send an MMS 39  */ 40  class SendRequest extends MmsRequest { 41      // Max send response PDU size in bytes (exceeding this may cause problem with 42      // system intent delivery). 43      private static final int MAX_SEND_RESPONSE_SIZE = 1000 * 1024; 44 45      private byte[] mPduData; 46 47      SendRequest(final String locationUri, final Uri pduUri, final PendingIntent sentIntent) { 48          super(locationUri, pduUri, sentIntent); 49      } 50 51      @Override 52      protected boolean loadRequest(final Context context, final Bundle mmsConfig) { 53          mPduData = readPduFromContentUri( 54              context, 55              mPduUri, 56              mmsConfig.getInt( 57                  CarrierConfigValuesLoader.CONFIG_MAX_MESSAGE_SIZE, 58                  CarrierConfigValuesLoader.CONFIG_MAX_MESSAGE_SIZE_DEFAULT)); 59          return (mPduData != null); 60      } 61 62      @Override 63      protected boolean transferResponse(final Context context, final Intent fillIn, 64          final byte[] response) { 65          // SendConf pdus are always small and can be included in the intent 66          if (response != null &amp;&amp; fillIn != null) { 67              if (response.length &gt; MAX_SEND_RESPONSE_SIZE) { 68                  // If the response PDU is too large, it won't be able to fit in 69                  // the PendingIntent to be transferred via system IPC. 70                  return false; 71              } 72              fillIn.putExtra(SmsManager.EXTRA_MMS_DATA, response); 73          } 74          return true; 75      } }</pre>

**Exhibit C for U.S. Patent No. 9,445,251 Against HTC Accused Products**

US9445251B2	HTC
	<p><a href="https://android.googlesource.com/platform/packages/apps/Messaging+/nougat-mr1-release/src/android/support/v7/mms/SendRequest.java">https://android.googlesource.com/platform/packages/apps/Messaging+/nougat-mr1-release/src/android/support/v7/mms/SendRequest.java</a></p> <pre data-bbox="520 358 1724 399">public static LocationRequest create ()</pre> <p data-bbox="512 423 1016 451">Create a location request with default parameters.</p> <p data-bbox="512 480 1625 542">Default parameters are for a block accuracy, slowly updated location. It can then be adjusted as required by the applications before passing to the <a href="#">FusedLocationProviderApi</a>.</p> <p data-bbox="533 565 617 589"><b>Returns</b></p> <ul data-bbox="541 610 800 634" style="list-style-type: none"> <li>• a new location request</li> </ul> <p data-bbox="499 651 1787 683"><a href="https://developers.google.com/android/reference/com/google/android/gms/location/LocationRequest">https://developers.google.com/android/reference/com/google/android/gms/location/LocationRequest</a></p>

**Exhibit C for U.S. Patent No. 9,445,251 Against HTC Accused Products**

US9445251B2	HTC
	<p><b>public static final int PRIORITY_BALANCED_POWER_ACCURACY</b></p> <p>Used with <code>setPriority(int)</code> to request "block" level accuracy.</p> <p>Block level accuracy is considered to be about 100 meter accuracy. Using a coarse accuracy such as this often consumes less power.</p> <p>Constant Value: 102</p> <hr/> <p><b>public static final int PRIORITY_HIGH_ACCURACY</b></p> <p>Used with <code>setPriority(int)</code> to request the most accurate locations available.</p> <p>This will return the finest location available.</p> <p>Constant Value: 100</p> <hr/> <p><b>public static final int PRIORITY_LOW_POWER</b></p> <p>Used with <code>setPriority(int)</code> to request "city" level accuracy.</p> <p>City level accuracy is considered to be about 10km accuracy. Using a coarse accuracy such as this often consumes less power.</p> <p>Constant Value: 104</p> <p><a href="https://developers.google.com/android/reference/com/google/android/gms/location/LocationRequest">https://developers.google.com/android/reference/com/google/android/gms/location/LocationRequest</a></p>

**Exhibit C for U.S. Patent No. 9,445,251 Against HTC Accused Products**

US9445251B2	HTC
	<p data-bbox="520 253 1738 285"><code>public Task&lt;Location&gt; getLastLocation ()</code></p> <p data-bbox="520 318 1094 342">Returns the best most recent location currently available.</p> <p data-bbox="520 375 1686 431">If a location is not available, which should happen very rarely, null will be returned. The best accuracy available while respecting the location permissions will be returned.</p> <p data-bbox="520 464 1724 521">This method provides a simplified way to get location. It is particularly well suited for applications that do not require an accurate location and that do not want to maintain extra logic for location updates.</p> <p data-bbox="520 578 1738 610"><code>public Task&lt;LocationAvailability&gt; getLocationAvailability ()</code></p> <p data-bbox="520 643 1682 699">Returns the availability of location data. When <code>isLocationAvailable()</code> returns true, then the location returned by <code>getLastLocation()</code> will be reasonably up to date within the hints specified by the active <code>LocationRequest</code> s.</p> <p data-bbox="520 732 1461 756">If the client isn't connected to Google Play services and the request times out, null is returned.</p> <p data-bbox="520 789 1661 846">Note it's always possible for <code>getLastLocation()</code> to return null even when this method returns true (e.g. location settings were disabled between calls).</p> <p data-bbox="520 862 1906 927"><a href="https://developers.google.com/android/reference/com/google/android/gms/location/FusedLocationProviderClient">https://developers.google.com/android/reference/com/google/android/gms/location/FusedLocationProviderClient</a></p>

**Exhibit C for U.S. Patent No. 9,445,251 Against HTC Accused Products**

US9445251B2	HTC						
	<p data-bbox="514 245 1734 326"> <code>public Task&lt;Void&gt; requestLocationUpdates (LocationRequest request, LocationCallback callback, Looper looper)</code> </p> <p data-bbox="514 354 1255 378">Requests location updates with a callback on the specified Looper thread.</p> <p data-bbox="514 412 1671 472">This method is suited for the foreground use cases. For background use cases, the <code>PendingIntent</code> version of the method is recommended, see <code>requestLocationUpdates(LocationRequest, PendingIntent)</code>.</p> <p data-bbox="514 506 1356 531">Any previous LocationRequests registered on this LocationListener will be replaced.</p> <p data-bbox="514 565 1671 656">This call will keep the Google Play services connection active, so make sure to call <code>removeLocationUpdates(LocationCallback)</code> when you no longer need it, otherwise you lose the benefits of the automatic connection management.</p> <p data-bbox="514 690 1728 714">Callbacks for <code>LocationCallback</code> will be made on the specified thread, which must already be a prepared looper thread.</p> <p data-bbox="531 738 653 763"><b>Parameters</b></p> <table border="1" data-bbox="514 792 1734 1008"> <tbody> <tr> <td data-bbox="514 792 617 857"><b>request</b></td> <td data-bbox="617 792 1734 857">The location request for the updates.</td> </tr> <tr> <td data-bbox="514 857 617 922"><b>callback</b></td> <td data-bbox="617 857 1734 922">The callback for the location updates.</td> </tr> <tr> <td data-bbox="514 922 617 1008"><b>looper</b></td> <td data-bbox="617 922 1734 1008">The Looper object whose message queue will be used to implement the callback mechanism, or null to make callbacks on the calling thread.</td> </tr> </tbody> </table> <p data-bbox="499 1024 1906 1089"> <a href="https://developers.google.com/android/reference/com/google/android/gms/location/FusedLocationProviderClient">https://developers.google.com/android/reference/com/google/android/gms/location/FusedLocationProviderClient</a> </p>	<b>request</b>	The location request for the updates.	<b>callback</b>	The callback for the location updates.	<b>looper</b>	The Looper object whose message queue will be used to implement the callback mechanism, or null to make callbacks on the calling thread.
<b>request</b>	The location request for the updates.						
<b>callback</b>	The callback for the location updates.						
<b>looper</b>	The Looper object whose message queue will be used to implement the callback mechanism, or null to make callbacks on the calling thread.						

## Exhibit C for U.S. Patent No. 9,445,251 Against HTC Accused Products

US9445251B2	HTC				
	<pre>public Task&lt;Void&gt; requestLocationUpdates (LocationRequest request, PendingIntent callbackIntent)</pre> <p>Requests location updates with a callback on the specified PendingIntent.</p> <p>This method is suited for the background use cases, more specifically for receiving location updates, even when the app has been killed by the system. In order to do so, use a <code>PendingIntent</code> for a started service. For foreground use cases, the <code>LocationCallback</code> version of the method is recommended, see <code>requestLocationUpdates(LocationRequest, LocationCallback, Looper)</code>.</p> <p>Any previously registered requests that have the same PendingIntent (as defined by <code>equals(Object)</code>) will be replaced by this request.</p> <p>Both <code>LocationResult</code> and <code>LocationAvailability</code> are sent to the given PendingIntent. You can extract data from an Intent using <code>hasResult(Intent)</code>, <code>extractResult(Intent)</code>, <code>hasLocationAvailability(Intent)</code>, and <code>extractLocationAvailability(Intent)</code>.</p> <p><b>Parameters</b></p> <table border="1"> <tr> <td><code>request</code></td> <td>The location request for the updates.</td> </tr> <tr> <td><code>callbackIntent</code></td> <td>A pending intent to be sent for each location update.</td> </tr> </table> <p><b>Returns</b></p> <ul style="list-style-type: none"> <li>a Task for the call, check <code>isSuccessful()</code> to determine if it was successful.</li> </ul> <p><a href="https://developers.google.com/android/reference/com/google/android/gms/location/FusedLocationProviderClient">https://developers.google.com/android/reference/com/google/android/gms/location/FusedLocationProviderClient</a></p>	<code>request</code>	The location request for the updates.	<code>callbackIntent</code>	A pending intent to be sent for each location update.
<code>request</code>	The location request for the updates.				
<code>callbackIntent</code>	A pending intent to be sent for each location update.				

## Exhibit C for U.S. Patent No. 9,445,251 Against HTC Accused Products

US9445251B2	HTC						
	<p><b>public void onLocationAvailability (LocationAvailability locationAvailability)</b></p> <p>Called when there is a change in the availability of location data.</p> <p>When <code>isLocationAvailable()</code> returns <code>false</code> you can assume that location will not be returned in <code>onLocationResult(LocationResult)</code> until something changes in the device's settings or environment. Even when <code>isLocationAvailable()</code> returns <code>true</code> the <code>onLocationResult(LocationResult)</code> may not always be called regularly, however the device location is known and both the most recently delivered location and <code>getLastLocation(GoogleApiClient)</code> will be reasonably up to date given the hints specified by the active <code>LocationRequest</code>s.</p> <p><b>Parameters</b></p> <table border="1"> <tr> <td><code>locationAvailability</code></td> <td>The current status of location availability.</td> </tr> </table> <p><b>public void onLocationResult (LocationResult result)</b></p> <p>Called when device location information is available.</p> <p>The most recent location returned by <code>getLastLocation()</code> is not guaranteed to be immediately fresh, but will be reasonably up to date given the hints specified by the active <code>LocationRequest</code>s.</p> <p><b>Parameters</b></p> <table border="1"> <tr> <td><code>result</code></td> <td>The latest location result available.</td> </tr> </table> <p><a href="https://developers.google.com/android/reference/com/google/android/gms/location/LocationCallback">https://developers.google.com/android/reference/com/google/android/gms/location/LocationCallback</a></p> <p><b>public abstract void onLocationChanged (Location location)</b></p> <p>Called when the location has changed.</p> <p><b>Parameters</b></p> <table border="1"> <tr> <td><code>location</code></td> <td>The updated location.</td> </tr> </table>	<code>locationAvailability</code>	The current status of location availability.	<code>result</code>	The latest location result available.	<code>location</code>	The updated location.
<code>locationAvailability</code>	The current status of location availability.						
<code>result</code>	The latest location result available.						
<code>location</code>	The updated location.						



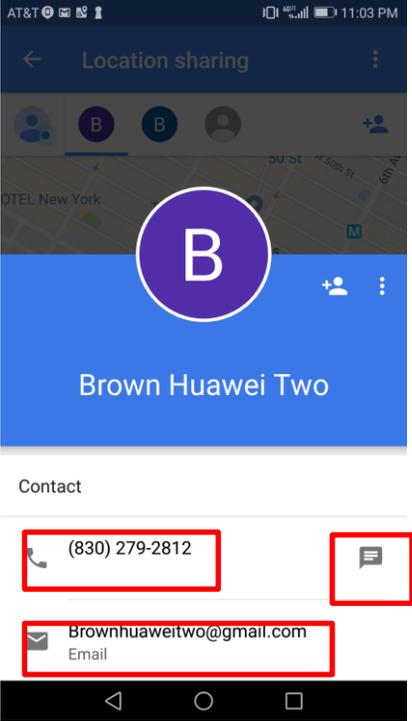
**Exhibit C for U.S. Patent No. 9,445,251 Against HTC Accused Products**

US9445251B2	HTC
	<p><a href="https://developers.google.com/android/reference/com/google/android/gms/location/LocationListener">https://developers.google.com/android/reference/com/google/android/gms/location/LocationListener</a></p> <p>Public Constructors</p> <hr/> <p>public <b>MapView</b> (<a href="#">Context</a> context)</p> <p>public <b>MapView</b> (<a href="#">Context</a> context, <a href="#">AttributeSet</a> attrs)</p> <p>public <b>MapView</b> (<a href="#">Context</a> context, <a href="#">AttributeSet</a> attrs, int defStyle)</p> <p>public <b>MapView</b> (<a href="#">Context</a> context, <a href="#">GoogleMapOptions</a> options)</p> <p><a href="https://developers.google.com/android/reference/com/google/android/gms/maps/MapView">https://developers.google.com/android/reference/com/google/android/gms/maps/MapView</a></p>

**Exhibit C for U.S. Patent No. 9,445,251 Against HTC Accused Products**

US9445251B2	HTC		
	<p><code>public void <b>getMapAsync</b> (<a href="#">OnMapReadyCallback</a> callback)</code></p> <p>Returns a non-null instance of the <code>GoogleMap</code>, ready to be used.</p> <p>Note that:</p> <ul style="list-style-type: none"> <li>• This method must be called from the main thread.</li> <li>• The callback will be executed in the main thread.</li> <li>• In the case where Google Play services is not installed on the user's device, the callback will not be triggered until the user installs it.</li> <li>• The <code>GoogleMap</code> object provided by the callback is non-null.</li> </ul> <p><b>Parameters</b></p> <table border="1" data-bbox="520 690 1724 755"> <tr> <td><code>callback</code></td> <td>The callback object that will be triggered when the map is ready to be used.</td> </tr> </table> <p><code>public final void <b>onCreate</b> (<a href="#">Bundle</a> savedInstanceState)</code></p> <p>You must call this method from the parent Activity/Fragment's corresponding method.</p> <p><a href="https://developers.google.com/android/reference/com/google/android/gms/maps/MapView">https://developers.google.com/android/reference/com/google/android/gms/maps/MapView</a></p>	<code>callback</code>	The callback object that will be triggered when the map is ready to be used.
<code>callback</code>	The callback object that will be triggered when the map is ready to be used.		
<p>2. The method of claim 1, wherein the data includes a short message service message, a text message, an image, or a video.</p>	<p>Each of the Accused Products perform the method of claim 1, wherein the data includes a short message service message, a text message, an image, or a video. As set forth above, each Accused Product can send the following communications.</p> <p>Exemplary Source Code:</p>		

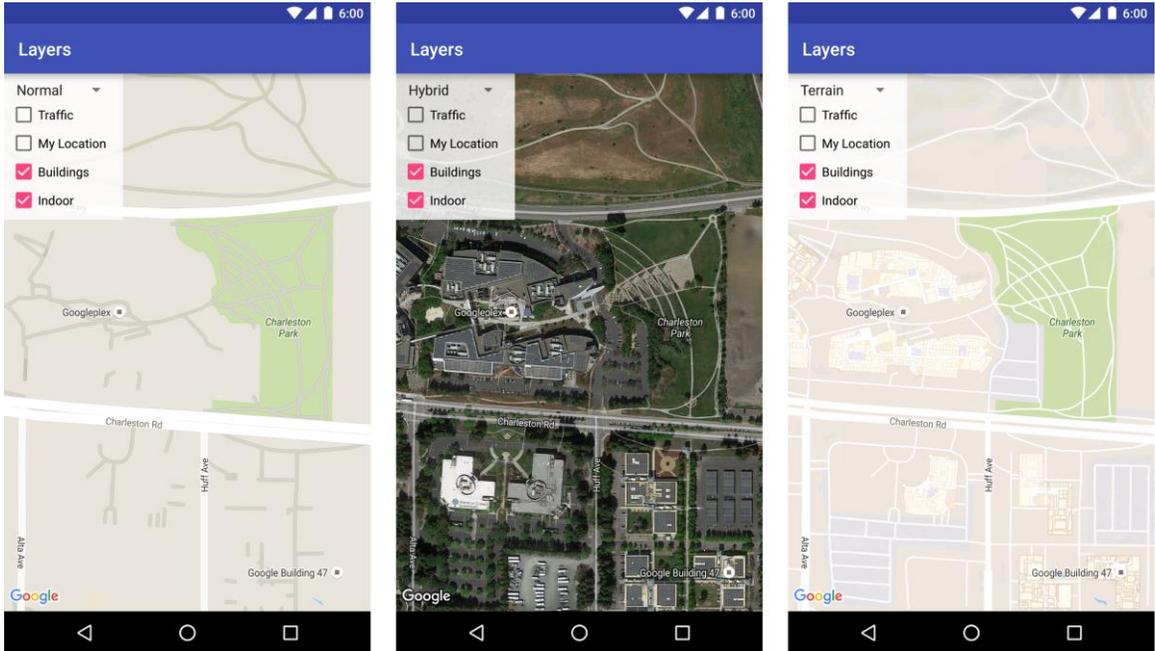
**Exhibit C for U.S. Patent No. 9,445,251 Against HTC Accused Products**

US9445251B2	HTC
	 <p>Additionally, within the messages and Hangouts apps, the Accused Products may send videos, images, sound, and text. Additionally within the Find My Device app, the Accused Products can communicate sound and text.</p>
<p>3. The method of claim 1, wherein the first device is a personal digital assistant (PDA) or a personal computer (PC).</p>	<p>Each of the Accused Products performs the method of claim 1, wherein the first device is a personal digital assistant (PDA) or a personal computer (PC).</p> <p>For example, one of skill in the art would understand that the Accused Products are at least personal digital assistants (PDA) within the ordinary meaning of the term because the Accused Products perform all of the functions of a PDA.</p>

**Exhibit C for U.S. Patent No. 9,445,251 Against HTC Accused Products**

US9445251B2	HTC
<p>4. The method of claim 1, wherein the second map is a satellite image.</p>	<p>Each of the Accused Products performs the method of claim 1, wherein the second map is a satellite image.</p> <p>Each of the Accused Products performs this method when the “satellite” button is selected:</p> <p><b>Map types</b></p> <p>There are many types of maps available within the Google Maps Android API. A map's type governs the overall representation of the map. For example, an atlas usually contains <b>political</b> maps that focus on showing boundaries, and <b>road</b> maps that show all of the roads for a city or region. The Google Maps Android API offers four types of maps, as well as an option to have no map at all:</p> <p><b>Normal</b> Typical road map. Shows roads, some features built by humans, and important natural features like rivers. Road and feature labels are also visible.</p> <p><b>Hybrid</b> Satellite photograph data with road maps added. Road and feature labels are also visible.</p> <p><b>Satellite</b> Satellite photograph data. Road and feature labels are not visible.</p> <p><b>Terrain</b> Topographic data. The map includes colors, contour lines and labels, and perspective shading. Some roads and labels are also visible.</p> <p><b>None</b> No tiles. The map will be rendered as an empty grid with no tiles loaded.</p>

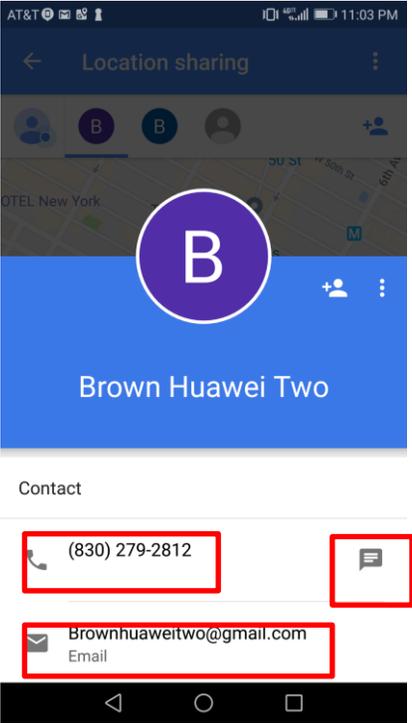
## Exhibit C for U.S. Patent No. 9,445,251 Against HTC Accused Products

US9445251B2	HTC
	<p data-bbox="499 240 1010 289"><b>Change the map type</b></p> <p data-bbox="499 298 1881 435">To set the type of a map, call the <code>GoogleMap</code> object's <code>setMapType()</code> method, passing one of the type constants defined in <code>GoogleMap</code>. For example, to display a satellite map:</p> <p data-bbox="499 485 787 524"><code>GoogleMap</code> map;</p> <p data-bbox="499 548 533 565">...</p> <pre data-bbox="499 573 1386 651">// Sets the map type to be "hybrid" map.setMapType(GoogleMap.MAP_TYPE_HYBRID);</pre> <p data-bbox="499 659 1881 743">The image below shows a comparison of normal, hybrid and terrain maps for the same location:</p> <div data-bbox="499 748 1654 1399"></div>

**Exhibit C for U.S. Patent No. 9,445,251 Against HTC Accused Products**

US9445251B2	HTC
	<p><a href="https://developers.google.com/maps/documentation/android-api/map">https://developers.google.com/maps/documentation/android-api/map</a></p>
<p>5. The method of claim 1, further comprising sending, by the first device, updated location information comprising an updated location of the first device, the updated location information being sent based on passage of a predetermined time interval since sending previous location information comprising a previous location of the first device, displacement of the first device by a predetermined distance relative to a previous location of the first device, or both.</p>	<p>Each of the Accused Products performs the method of claim 1, further comprising sending, by the first device, updated location information comprising an updated location of the first device, the updated location information being sent based on passage of a predetermined time interval since sending previous location information comprising a previous location of the first device, displacement of the first device by a predetermined distance relative to a previous location of the first device, or both.</p> <p>For example the Accused Products utilize a location manager service such as LocationServices that monitors, among other things, periods of time that are predetermined, or displacement from a predetermined distance.</p> <p>This method is thus performed at least when a user passes a geographic threshold that results in a location update (e.g., “leaves” notification). For example, Android utilizes the following framework that meets this limitation as implemented on the Accused Products:</p> <p><a href="https://developers.google.com/android/reference/com/google/android/gms/location/LocationServices">https://developers.google.com/android/reference/com/google/android/gms/location/LocationServices</a>  <a href="https://developer.android.com/reference/android/location/LocationManager.html">https://developer.android.com/reference/android/location/LocationManager.html</a></p> <p>e.g. <code>requestLocationUpdates(String provider, long minTime, float minDistance, LocationListener listener)</code>  Register for location updates using the named provider, and a pending intent.</p> <p>Android also makes use of “geotagging” that invokes time-based updating.</p>
<p>6. The method of claim 1, further comprising identifying second</p>	<p>Each of the Accused Products performs the method of claim 1, further comprising identifying second user interaction with the interactive display selecting at least one of the user-selectable symbols corresponding to at least one of the second devices and user interaction with the display specifying an action and, based thereon, initiating a phone call or phone conference with the at least one second device.</p>

**Exhibit C for U.S. Patent No. 9,445,251 Against HTC Accused Products**

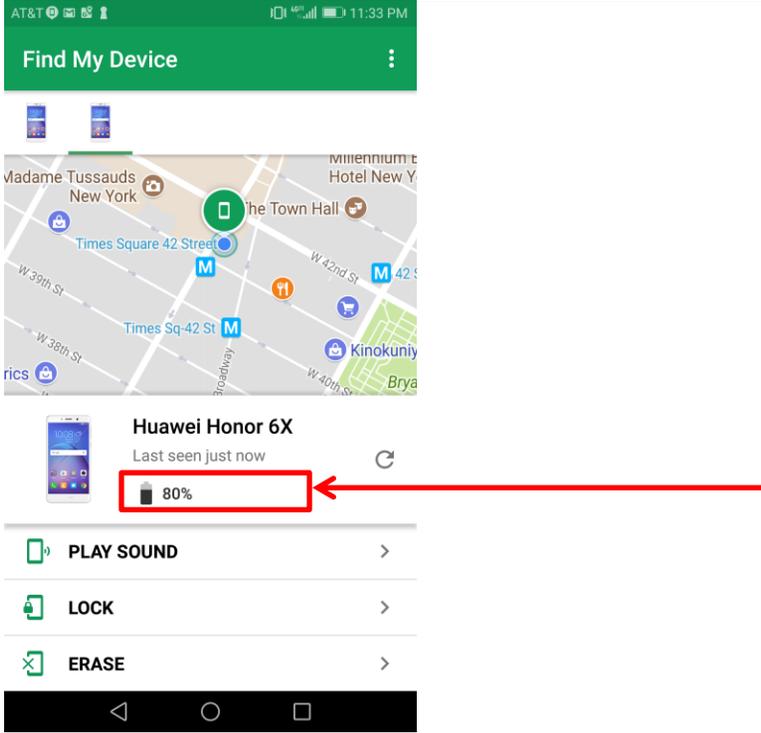
US9445251B2	HTC
<p>user interaction with the interactive display selecting at least one of the user-selectable symbols corresponding to at least one of the second devices and user interaction with the display specifying an action and, based thereon, initiating a phone call or phone conference with the at least one second device.</p>	<p>For example, the Accused Products include software that registers touch events with an interactive display, where touching a “call” button starts a phone call. Furthermore, these phone calls can merge multiple parties into a conference call.</p>  <p>See also:  <a href="https://www.wikihow.com/Conference-Call-on-an-Android">https://www.wikihow.com/Conference-Call-on-an-Android</a></p>
<p>7. The method of claim 1, wherein the message from the</p>	<p>Each of the Accused Products performs the method of claim 1, wherein the message from the second device is a Short Message Service (SMS) message or a text message.</p>

**Exhibit C for U.S. Patent No. 9,445,251 Against HTC Accused Products**

<b>US9445251B2</b>	<b>HTC</b>
second device is a Short Message Service (SMS) message or a text message.	For example, the Accused Products each include the Messages app that allows for messaging including SMS messaging. The Accused Products also include Hangouts (including later versions) that include text messaging.
8. The method of claim 1, wherein participating in the group further includes sending first status information to the server and receiving second status information from the server, the first status information comprising a battery level of the first device, a signal strength of a wireless signal of the first device, a status of a Global Positioning Satellite (GPS) receiver of the first device, or a combination thereof, the second location information comprising a plurality of battery levels of the respective plurality of second devices included in the group, a plurality of signal strengths of wireless signals of the respective plurality of second devices included in the group, a plurality of statuses of GPS receivers of the respective plurality of second devices included in the group, or a combination thereof.	<p>Each of the Accused Products performs the method of claim 1, wherein participating in the group further includes sending first status information to the server and receiving second status information from the server, the first status information comprising a battery level of the first device, a signal strength of a wireless signal of the first device, a status of a Global Positioning Satellite (GPS) receiver of the first device, or a combination thereof, the second location information comprising a plurality of battery levels of the respective plurality of second devices included in the group, a plurality of signal strengths of wireless signals of the respective plurality of second devices included in the group, a plurality of statuses of GPS receivers of the respective plurality of second devices included in the group, or a combination thereof.</p> <p>For example, the Accused Products each the ability to communicate battery level and wireless networks to servers, where that battery level and wireless networks are communicated to the first device.</p> <p>Additionally, the status of the GPS receivers is also communicated within the apps. For example, the second devices communicate the accuracy of their GPS signal, as well as status so that the first device can view both the accuracy of the signal (represented by a circle around the user) as well as the status (online vs. offline).</p> <p>Additionally, status of a GPS receiver may be communicated through notifications that communicate when a device has passed a location threshold.</p>



**Exhibit C for U.S. Patent No. 9,445,251 Against HTC Accused Products**

US9445251B2	HTC
<p>plurality of second devices included in the group, a plurality of signal strengths of wireless signals of the respective plurality of second devices included in the group, a plurality of statuses of GPS receivers of the respective plurality of second devices included in the group, or a combination thereof.</p>	
<p>9. The method of claim 1, wherein the first device is a smart phone.</p>	<p>Each of the Accused Android phone devices are smartphones. The Android tablet devices with cellular capabilities are at least the equivalent of smartphones because they perform the limitations in the same way, with the same functionality, and achieve substantially the same result.</p>
<p>10[A]. The method of claim 1, further comprising: with the first device, transmitting a group identifier associated with a second group,</p>	<p>The Accused Products perform the method of claim 1, wherein the method further includes: with the first device, transmitting a group identifier associated with a second group, the second group including a second plurality of second devices; and based on transmitting the group identifier associated with the second group, participating in the second group, wherein participating in the second group includes receiving third location information from the server, the third location information comprising a plurality of locations of the respective second plurality of second devices included in the second group.</p>

**Exhibit C for U.S. Patent No. 9,445,251 Against HTC Accused Products**

<b>US9445251B2</b>	<b>HTC</b>
<p>the second group including a second plurality of second devices; and based on transmitting the group identifier associated with the second group, participating in the second group, wherein participating in the second group includes receiving third location information from the server, the third location information comprising a plurality of locations of the respective second plurality of second devices included in the second group.</p>	<p>As set forth above with regard to grouping, second groups are established in several ways. As a first example, as between the Maps and Find My Device applications, each constitutes at least one separate group.</p> <p>As a further example, within the Maps app, devices may be grouped based on pre-determined groupings from Google Hangouts or other apps. Moreover, on information and belief, an identifier must exist on the servers that correlates each group, and which is communicated as between the servers and devices.</p>
<p>11. The method of claim 1, wherein the data includes a voice recording.</p>	<p>Each of the Accused Products performs the method of claim 1, wherein the data includes a voice recording.</p> <p>For example, each Accused Product can send a recording.</p> <p><a href="https://www.techrepublic.com/article/how-to-send-audio-clips-via-sms-in-android/">https://www.techrepublic.com/article/how-to-send-audio-clips-via-sms-in-android/</a></p>
<p>12. The method of claim 1, further comprising: using a Global Positioning</p>	<p>Each of the Accused Products that includes a cellular connection meets this claim by performing the method of claim 1, further using a Global Positioning Satellite (GPS) receiver of the first device to obtain data indicative of the location of the first device, wherein sending the first location information to the server comprises using the Internet Protocol (IP) to send the first location information to the server.</p>

**Exhibit C for U.S. Patent No. 9,445,251 Against HTC Accused Products**

US9445251B2	HTC
<p>Satellite (GPS) receiver of the first device to obtain data indicative of the location of the first device, wherein sending the first location information to the server comprises using the Internet Protocol (IP) to send the first location information to the server.</p>	<p>For example, location information used for Find My Device and Maps are transmitted per IP as set forth above in claim 1. Furthermore, on cellular models, this information is gathered from on a GPS receiver within the Android device.</p>
<p>13[A]. The method of claim 1, further comprising identifying, by the first device, user interaction with the display selecting a particular user-selectable symbol positioned on the second georeferenced map and corresponding to a particular second device, wherein identifying the user interaction selecting the particular user-</p>	<p>On information and belief, each of the Accused Products performs the method of claim 1 and performs this method for at least the same reasons as discussed above with regard to Claim 6 when utilizing a second map, e.g., a satellite image map.</p>

**Exhibit C for U.S. Patent No. 9,445,251 Against HTC Accused Products**

US9445251B2	HTC
<p>selectable symbol comprises: detecting user selection of a portion of the interactive display corresponding to a position on the second georeferenced map;</p>	
<p>[13B] based at least in part on coordinates of the selected position on the second georeferenced map and on the data relating positions on the second georeferenced map to spatial coordinates, determining spatial coordinates of a location represented by the selected position on the second georeferenced map;</p>	<p>On information and belie, the Accused Products include a database locations and users at least within the Find My Device and Maps apps.</p> <p>For example, when markers, such as symbols, are added to maps, they are added based on longitude and latitude.</p> <p><b>Add a map</b></p> <p>Display a map, using the Google Maps Android API.</p> <p>Add a <code>&lt;fragment&gt;</code> element to your activity's layout file, <code>activity_maps.xml</code>. This element defines a <code>SupportMapFragment</code> to act as a container for the map and to provide access to the <code>GoogleMap</code> object. The tutorial uses the Android support library version of the map fragment, to ensure backward compatibility with earlier versions of the Android framework.</p> <pre data-bbox="499 1159 1915 1386">&lt;fragment xmlns:android="http://schemas.android.com/apk/res/android"     xmlns:tools="http://schemas.android.com/tools"     android:id="@+id/map"     android:name="com.google.android.gms.maps.SupportMapFragment"     android:layout_width="match_parent"     android:layout_height="match_parent"</pre>

**Exhibit C for U.S. Patent No. 9,445,251 Against HTC Accused Products**

US9445251B2	HTC
	<pre data-bbox="499 237 1917 313">tools:context="com.example.mapwithmarker.MapsMarkerActivity" /&gt;</pre> <p data-bbox="499 347 1871 456">In your activity's <code>onCreate()</code> method, set the layout file as the content view. Get a handle to the map fragment by calling <code>FragmentManager.findFragmentById()</code>. Then use <code>getMapAsync()</code> to register for the map callback:</p> <pre data-bbox="499 492 1917 967"> @Override protected void onCreate(Bundle savedInstanceState) {     super.onCreate(savedInstanceState);     // Retrieve the content view that renders the map.     setContentView(R.layout.activity_maps);     // Get the SupportMapFragment and request notification     // when the map is ready to be used.     SupportMapFragment mapFragment = (SupportMapFragment) getSupportFragmentManager()         .findFragmentById(R.id.map);     mapFragment.getMapAsync(this); } </pre> <p data-bbox="499 1003 1871 1073">Implement the <code>OnMapReadyCallback</code> interface and override the <code>onMapReady()</code> method, to set up the map when the <code>GoogleMap</code> object is available:</p> <pre data-bbox="499 1109 1917 1422"> public class MapsMarkerActivity extends AppCompatActivity     implements OnMapReadyCallback {     // Include the onCreate() method here too, as described above.     @Override     public void onMapReady(GoogleMap googleMap) {         // Add a marker in Sydney, Australia,         // and move the map's camera to the same location.         LatLng sydney = new LatLng(-33.852, 151.211); </pre>

**Exhibit C for U.S. Patent No. 9,445,251 Against HTC Accused Products**

US9445251B2	HTC
	<pre data-bbox="499 240 1900 430"> googleMap.addMarker(new MarkerOptions().position(sydney)     .title("Marker in Sydney")); googleMap.moveCamera(CameraUpdateFactory.newLatLng(sydney)); } } </pre> <p data-bbox="499 467 1539 495">see, also, <a href="https://developers.google.com/maps/documentation/android-api/marker">https://developers.google.com/maps/documentation/android-api/marker</a></p>
<p data-bbox="184 506 476 792">[13C] and identifying the particular user-selectable symbol based, at least in part, on the spatial coordinates represented by the selected position.</p>	<p data-bbox="499 506 1833 571">On information and belie, the Accused Products identify the symbol based, at least in part, on the spatial coordinates.</p> <p data-bbox="499 613 722 646">See [13B] above.</p> <p data-bbox="499 688 1864 753">Each marker corresponds to a spatial coordinate (i.e. longitude/latitude) as well as an x/y coordinate on the map display.</p> <p data-bbox="499 795 621 828">See, e.g.,</p> <p data-bbox="499 870 1549 902"><a href="https://developers.google.com/maps/documentation/android-api/map-with-marker">https://developers.google.com/maps/documentation/android-api/map-with-marker</a></p> <p data-bbox="499 906 1415 938"><a href="https://developers.google.com/maps/documentation/android-api/marker">https://developers.google.com/maps/documentation/android-api/marker</a></p>
<p data-bbox="184 945 476 1409">14[A]. The method of claim 13, wherein identifying the particular user-selectable symbol based, at least in part, on the spatial coordinates represented by the selected position comprises: searching a database of entities for an entity located</p>	<p data-bbox="499 945 1906 1123">On information and belief, each of the Accused Products performs the method of claim 1 and performs this method for at least the same reasons as discussed above with regard to Claims 1, 6 , and 13. Additionally, the Accused Products identify symbols based on spatial coordinates in that the Accused Products include symbols corresponding to entities and/or a user is able to search for entities, which are retrieved and identified by symbols on the display.</p> <p data-bbox="499 1166 1094 1198">See, e.g, Android Maps and Geotagging APIs:</p> <p data-bbox="499 1286 1083 1318"><b>Instantiate the Places API clients</b></p> <p data-bbox="499 1360 1829 1393">The following interfaces provide the primary entry points to the Google Places API for Android:</p>

**Exhibit C for U.S. Patent No. 9,445,251 Against HTC Accused Products**

US9445251B2	HTC
<p>nearest to the spatial coordinates represented by the selected position, wherein the entities represented by data in the database include the second devices, wherein the database data include locations of the respective entities, and wherein the database is searchable by location;</p>	<p>The <code>GeoDataClient</code> provides access to Google's database of local place and business information.</p> <p>The <code>PlaceDetectionClient</code> provides quick access to the device's current place, and offers the opportunity to report the location of the device at a particular place.</p> <p>The <code>LocationServices</code> interface is the main entry point for Android location services.</p> <p>To use the APIs, instantiate <code>GeoDataClient</code>, <code>PlaceDetectionClient</code>, and <code>FusedLocationProviderClient</code> in your fragment's or activity's <code>onCreate()</code> method, as shown in the following code sample:</p> <pre data-bbox="495 607 1913 1127">protected void onCreate(Bundle savedInstanceState) {     super.onCreate(savedInstanceState);     setContentView(R.layout.activity_main);      // Construct a GeoDataClient.     mGeoDataClient = Places.getGeoDataClient(this, null);      // Construct a PlaceDetectionClient.     mPlaceDetectionClient = Places.getPlaceDetectionClient(this, null);      // Construct a FusedLocationProviderClient.     mFusedLocationProviderClient =     LocationServices.getFusedLocationProviderClient(this); }</pre> <p>See, e.g., <a href="https://developers.google.com/maps/documentation/android-api/current-place-tutorial">https://developers.google.com/maps/documentation/android-api/current-place-tutorial</a>  <a href="https://developers.google.com/android/reference/com/google/android/gms/location/places/GeoDataClient">https://developers.google.com/android/reference/com/google/android/gms/location/places/GeoDataClient</a></p>
<p>[14B] and based on a result of searching the database, identifying the particular second device as the entity</p>	<p>On information and belief, each of the Accused Products performs the method of claim 1 and performs this method for at least the same reasons as discussed above with regard to Claims 1, 6, and 13. Additionally, the Accused Products identify symbols based on spatial coordinates in that the Accused Products include symbols corresponding to entities and/or a user is able to search for entities, which are retrieved and identified by symbols on the display.</p>

**Exhibit C for U.S. Patent No. 9,445,251 Against HTC Accused Products**

US9445251B2	HTC
located nearest to the spatial coordinates represented by the selected position, wherein the particular user-selectable symbol corresponds to the particular second device.	<p>See, e.g.,</p> <p><a href="https://developers.google.com/android/reference/com/google/android/gms/location/places/GeoDataClient">https://developers.google.com/android/reference/com/google/android/gms/location/places/GeoDataClient</a></p>
15[A]. The method of claim 14, wherein the entity is a first entity, and wherein the method further comprises performing by the first device: receiving user input via user interaction with the interactive display of the first device, the user input specifying a location and a symbol corresponding to a second entity other than the first device and the second devices;	<p>On information and belief, each of the Accused Products performs the method of claim 1 and performs this method for at least the same reasons as discussed above with regard to Claims 1, 6, 13 and 14. Additionally, the Accused Products identify symbols based on spatial coordinates in that the Accused Products include symbols corresponding to entities and/or a user is able to search for entities, which are retrieved and identified by symbols on the display.</p> <p>See, e.g.,</p> <p>Selection with Markers:  <a href="https://developers.google.com/maps/documentation/android-api/marker">https://developers.google.com/maps/documentation/android-api/marker</a></p> <p>Queries with GeoTagging database:  <a href="https://developers.google.com/android/reference/com/google/android/gms/location/places/GeoDataClient">https://developers.google.com/android/reference/com/google/android/gms/location/places/GeoDataClient</a></p>
[15b] and based on the user input, adding the user-specified symbol to the	<p>On information and belief, each of the Accused Products performs the method of claim 1 and performs this method for at least the same reasons as discussed above with regard to Claims 1, 6, and 13. Additionally, the Accused Products identify symbols based on spatial coordinates in that the Accused Products include symbols corresponding to entities and/or a user is able to search for entities, which are retrieved and identified by</p>



**Exhibit C for U.S. Patent No. 9,445,251 Against HTC Accused Products**

US9445251B2	HTC
<p>interactive display at a position on the second georeferenced map corresponding to the user-specified location of the second entity.</p>	<p>symbols on the display.</p> <p>Placing a Marker:  <a href="https://developers.google.com/maps/documentation/android-api/marker">https://developers.google.com/maps/documentation/android-api/marker</a></p> <p>based on queries with GeoTagging database:  <a href="https://developers.google.com/android/reference/com/google/android/gms/location/places/GeoDataClient">https://developers.google.com/android/reference/com/google/android/gms/location/places/GeoDataClient</a></p>
<p>16. The method of claim 15, further comprising performing by the first device: transmitting the user-specified symbol and location of the second entity to the second devices for addition of the user-specified symbol to respective interactive displays of the second devices at respective positions on respective georeferenced maps corresponding to the user-specified location of the second entity.</p>	<p>On information and belief, each of the Accused Products performs the method of claim 1 and performs this method for at least the same reasons as discussed above with regard to Claims 1, 6, 13, 14, and 15. Additionally, the Accused Products provide the functionality of adding a symbol to the display and sending the symbol over Hangouts or Messages app for display on the second device (i.e. pin drop, share location features).</p> <p>See, e.g., Placing a Marker:  <a href="https://developers.google.com/maps/documentation/android-api/marker">https://developers.google.com/maps/documentation/android-api/marker</a></p> <p>based on queries with GeoTagging database:  <a href="https://developers.google.com/android/reference/com/google/android/gms/location/places/GeoDataClient">https://developers.google.com/android/reference/com/google/android/gms/location/places/GeoDataClient</a></p> <p>Sharing a link:  <a href="https://support.google.com/maps/answer/144361?co=GENIE.Platform%3DDesktop&amp;hl=en">https://support.google.com/maps/answer/144361?co=GENIE.Platform%3DDesktop&amp;hl=en</a></p>
<p>17. The method of claim 16, wherein the user input further</p>	<p>On information and belief, each of the Accused Products performs the method of claim 1 and performs this method for at least the same reasons as discussed above with regard to Claims 1, 6, 13, 14, 15, and 16. Additionally, the Accused Products provide the functionality of adding a symbol to the display and sending</p>

**Exhibit C for U.S. Patent No. 9,445,251 Against HTC Accused Products**

US9445251B2	HTC
<p>specifies information associated with the second entity, and wherein the method further comprises performing, by the first device: transmitting the user-specified information associated with the second entity to the second devices.</p>	<p>the symbol over the Messages app for display on the second device (i.e. drop pin, share location features).</p> <p>See, e.g., Sharing a link:  <a href="https://support.google.com/maps/answer/144361?co=GENIE.Platform%3DDesktop&amp;hl=en">https://support.google.com/maps/answer/144361?co=GENIE.Platform%3DDesktop&amp;hl=en</a></p>
<p>18. The method of claim 17, further comprising performing by the first device: adding data representing the spatial coordinates of the location of the second entity and data representing the information associated with the second entity to the database.</p>	<p>On information and belief, each of the Accused Products performs the method of claim 1 and performs this method for at least the same reasons as discussed above with regard to Claims 1, 6, 13, 14, 15, 16, and 17. Additionally, the Accused Products provide the functionality of adding a symbol to the display and sending the symbol over Hangouts or Messages app for display on the second device (i.e. drop pin, share location features). On information and belief, the Accused Products add spatial coordinates and metadata with entities.</p> <p>Sharing a link:  <a href="https://support.google.com/maps/answer/144361?co=GENIE.Platform%3DDesktop&amp;hl=en">https://support.google.com/maps/answer/144361?co=GENIE.Platform%3DDesktop&amp;hl=en</a></p> <p>Markers (adding location information to the link associated with the database):</p> <pre>static final LatLng PERTH = new LatLng(-31.90, 115.86); Marker perth = mMap.addMarker(new MarkerOptions()     .position(PERTH)     .draggable(true));</pre>
19[A]. The method of	On information and belief, each of the Accused Products performs the method of claim 1 and performs this

**Exhibit C for U.S. Patent No. 9,445,251 Against HTC Accused Products**

US9445251B2	HTC
<p>claim 15, wherein the portion of the interactive display is a first portion, wherein the position of the symbol corresponding to the particular second device is a first position, and wherein receiving the user input specifying the location of the second entity comprises: detecting user selection of a second portion of the interactive display corresponding to a second position on the second georeferenced map;</p>	<p>method for at least the same reasons as discussed above with regard to Claims 1, 6, 13, 14, and 15. Additionally, the Accused Products identify symbols based on spatial coordinates in that the Accused Products include symbols corresponding to entities and/or a user is able to search for entities, which are retrieved and identified by symbols on the display. On information and belief, the Accused Products include spatial coordinates and metadata with entities.</p> <p>See claims 13-18 above wherein the second entity is an additional selectable marker.</p> <p>See, e.g., Placing a Marker:  <a href="https://developers.google.com/maps/documentation/android-api/marker">https://developers.google.com/maps/documentation/android-api/marker</a></p> <p>Markers may be selectable:</p> <p><b>Marker click events</b></p> <p>You can use an <code>OnMarkerClickListener</code> to listen for click events on the marker. To set this listener on the map, call <code>GoogleMap.setOnMarkerClickListener(OnMarkerClickListener)</code>. When a user clicks on a marker, <code>onMarkerClick(Marker)</code> will be called and the marker will be passed through as an argument. This method returns a boolean that indicates whether you have consumed the event (i.e., you want to suppress the default behavior). If it returns <code>false</code>, then the default behavior will occur in addition to your custom behavior. The default behavior for a marker click event is to show its <code>info window</code> (if available) and move the camera such that the marker is centered on the map.</p>
<p>[19B] and based at least in part on coordinates of the second position on the second georeferenced map</p>	<p>On information and belief, the Accused Products identify the symbol based, at least in part, on the spatial coordinates.</p> <p>See claims 13-18 above wherein the second entity is an additional selectable marker. A pin may be dropped on the “location of the second entity” by sharing a location.</p>

**Exhibit C for U.S. Patent No. 9,445,251 Against HTC Accused Products**

US9445251B2	HTC
<p>and on the data relating positions on the second georeferenced map to spatial coordinates, determining spatial coordinates of a location represented by the second position on the second georeferenced map, wherein the location represented by the second position is the location of the second entity.</p>	
<p>20. The method of claim 14, wherein the database is stored on the first device.</p>	<p>On information and belief, each of the Accused Products includes at least a database on the device that stores information related to each of the apps on the Android device, including a database storing information relevant to claim 14.</p>
<p>21. The method of claim 14, wherein the database is stored on the server.</p>	<p>On information and belief, each of the Accused Products interfaces with a database on one of either Google or an ISP's servers that stores information related to that stored locally on the device.</p>
<p>22. The method of claim 1, wherein the spatial coordinates comprise latitude and longitude coordinates.</p>	<p>Each of the Accused Products performs the limitations of Claim 1, and further performs these steps wherein the spatial coordinates comprise latitude and longitude coordinates.</p> <p>This limitation is met because the location on the map corresponds to a latitude and longitude location, or at least the equivalent thereof. This relationship is corroborated by the Maps functionality which establishes the relationship between a point on the map and a latitude/longitude.</p>

**Exhibit C for U.S. Patent No. 9,445,251 Against HTC Accused Products**

US9445251B2	HTC
<p>23. The method of claim 1, further comprising performing, by the first device: identifying user interaction with the interactive display selecting a particular user-selectable symbol corresponding to a particular second device and user interaction with the display specifying an action and, based thereon, initiating voice-over-IP (VOIP) communication with the particular second device.</p>	<p>Each of the Accused Products performs the method of claim 1, wherein the steps further comprise performing, by the first device: identifying user interaction with the interactive display selecting a particular user-selectable symbol corresponding to a particular second device and user interaction with the display specifying an action and, based thereon, initiating voice-over-IP (VOIP) communication with the particular second device.</p> <p>For example, the Accused Products include software that registers touch events with an interactive display, where touching a “Hangouts” button starts an IP based video and audio call. Furthermore, the user may initiate a Hangouts Audio call, which further satisfies this limitation.</p>
<p>24[P]. A system comprising: a first device programmed to perform operations comprising:</p>	<p><b>HTC</b> directly and/or indirectly infringes by providing a system comprising a first device programmed to perform operations [of claim 24]. See 1P above. Each <b>HTC</b> device includes software programs, such as the Android operating system including Android APIs and associated apps, that are programmed to perform the operations of claim 24 [See e.g., claim 1 above]. Additionally, to the extent that any of the accused functionality is not provided with a particular <b>HTC</b> device, <b>HTC</b> contributes to infringement by providing the device hardware and capability to download software. <b>HTC</b> also induces its customers to make and use the accused system by instructing and actively inducing its customers to download software and to use the accused system in an infringing manner.</p>
<p>[24A] receiving a message from a second device,</p>	<p>See [1A] above.</p>

**Exhibit C for U.S. Patent No. 9,445,251 Against HTC Accused Products**

US9445251B2	HTC
wherein the message relates to joining a group;	
[24B] based on receiving the message from the second device, participating in the group, wherein participating in the group includes sending first location information to a server and receiving second location information from the server, the first location information comprising a location of the first device, the second location information comprising a plurality of locations of a respective plurality of second devices included in the group;	<i>See [1B] above.</i>
[24C] presenting, via an interactive display of the first device, a first interactive, georeferenced map and a plurality of	<i>See [1C] above.</i>

**Exhibit C for U.S. Patent No. 9,445,251 Against HTC Accused Products**

US9445251B2	HTC
<p>user-selectable symbols corresponding to the plurality of second devices, wherein the symbols are positioned on the first georeferenced map at respective positions corresponding to the locations of the second devices, and wherein the first georeferenced map includes data relating positions on the first georeferenced map to spatial coordinates;</p>	
<p>[24D] sending, from the first device to the server, a request for a second georeferenced map different from the first georeferenced map, wherein the request specifies a map location;</p>	<p><i>See [1D] above.</i></p>
<p>[24E] receiving, from the server, the second georeferenced map, wherein the second</p>	<p><i>See [1E] above.</i></p>

**Exhibit C for U.S. Patent No. 9,445,251 Against HTC Accused Products**

US9445251B2	HTC
georeferenced map includes the requested location and data relating positions on the second georeferenced map to spatial coordinates;	
[24F] presenting, via the interactive display of the first device, the second georeferenced map and the plurality of user-selectable symbols corresponding to the plurality of second devices, wherein the symbols are positioned on the second georeferenced map at respective positions corresponding to the locations of the second devices;	<i>See [1F] above.</i>
[24G] and identifying user interaction with the interactive display selecting one or more of the user-selectable symbols corresponding to one	<i>See [1G] above.</i>



**Exhibit C for U.S. Patent No. 9,445,251 Against HTC Accused Products**

US9445251B2	HTC
<p>or more of the second devices and positioned on the second georeferenced map and user interaction with the display specifying an action and, based thereon, using an Internet Protocol to send data to the one or more second devices via the server, wherein the first device does not have access to respective Internet Protocol addresses of the second devices.</p>	
<p>25. The system of claim 24, wherein the data includes a short message service message, a text message, an image, or a video.</p>	<p><i>See [2] above.</i></p>
<p>26. The system of claim 24, wherein the first device is a personal digital assistant (PDA) or a personal computer</p>	<p><i>See [3] above.</i></p>

**Exhibit C for U.S. Patent No. 9,445,251 Against HTC Accused Products**

US9445251B2	HTC
(PC).	
27. The system of claim 24, wherein the second map is a satellite image.	<i>See [4] above.</i>
28. The system of claim 24, wherein the operations further comprise sending updated location information comprising an updated location of the first device, the updated location information being sent based on passage of a predetermined time interval since sending previous location information comprising a previous location of the first device, displacement of the first device by a predetermined distance relative to a previous location of the first device, or both.	<i>See [5] above.</i>
29. The system of claim 24, wherein the	<i>See [6] above.</i>

**Exhibit C for U.S. Patent No. 9,445,251 Against HTC Accused Products**

US9445251B2	HTC
<p>operations further comprise identifying second user interaction with the interactive display selecting at least one of the user-selectable symbols corresponding to at least one of the second devices and user interaction with the display specifying an action and, based thereon, initiating a phone call or phone conference with the at least one second device.</p>	
<p>30. The system of claim 24, wherein the message from the second device is a Short Message Service (SMS) message or a text message.</p>	<p><i>See [7] above.</i></p>
<p>31. The system of claim 24, wherein participating in the group further includes sending first status</p>	<p><i>See [8] above.</i></p>

**Exhibit C for U.S. Patent No. 9,445,251 Against HTC Accused Products**

US9445251B2	HTC
<p>information to the server and receiving second status information from the server, the first status information comprising a battery level of the first device, a signal strength of a wireless signal of the first device, a status of a Global Positioning Satellite (GPS) receiver of the first device, or a combination thereof, the second location information comprising a plurality of battery levels of the respective plurality of second devices included in the group, a plurality of signal strengths of wireless signals of the respective plurality of second devices included in the group, a plurality of statuses of GPS receivers of</p>	

**Exhibit C for U.S. Patent No. 9,445,251 Against HTC Accused Products**

US9445251B2	HTC
<p>the respective plurality of second devices included in the group, or a combination thereof.</p>	
<p>32. The system of claim 24, wherein the first device is a smart phone.</p>	<p><i>See [9] above.</i></p>
<p>33. The system of claim 24, wherein the operations further include: with the first device, transmitting a group identifier associated with a second group, the second group including a second plurality of second devices; and based on transmitting the group identifier associated with the second group, participating in the second group, wherein participating in the second group includes receiving third location information from the server, the third</p>	<p><i>See [10] above.</i></p>

**Exhibit C for U.S. Patent No. 9,445,251 Against HTC Accused Products**

US9445251B2	HTC
location information comprising a plurality of locations of the respective second plurality of second devices included in the second group.	
34. The system of claim 24, wherein the data includes a voice recording.	<i>See [11] above.</i>
35. The system of claim 24, wherein the first device includes a Global Positioning Satellite (GPS) receiver, and wherein the operations further include: using the GPS receiver to obtain data indicative of the location of the first device, wherein sending the first location information to the server comprises using the Internet Protocol (IP) to send the first location information to the server.	<i>See [12] above.</i>

**Exhibit D - Claim Chart for U.S. Patent No. 9,467,838 Against HTC**

In these Infringement Contentions, AGIS Software Development LLC (“AGIS”) contends that at least the following claims of U.S. Patent No. 9,467,838 (the “’838 Patent”) identified below are infringed by the Accused Products (*e.g.*, phones and tablets) which are manufactured, sold, offered for sale, and/or used by HTC Corporation (“HTC”).

The Accused Products comprise HTC products running the Android mobile operating system and manufactured, used, or sold during and after 2011. For example, the Accused Products comprise the following Android-based phones and tablets: 10, 10 evo, 10 Lifestyle, 2125 / 2100 (Faraday), 3125 / Smartflip / 8500 (Star Trek), 5800 / Fusion / S720, 7 Mozart, 7 Pro, 7 Surround, 7 Trophy, 8125 / 8100 / MDA (USA) / K-JAM / P4300 (Wizard), 8XT, Amaze 4G, Aria, Arrive, Arrive / 7 Pro (CDMA), Bolt, Butterfly, Butterfly 2, Butterfly 3, Butterfly S, ChaCha, Dash / S620 / S621 (Excalibur), Dash 3G / Snap (GSM), Desire, Desire (CDMA), Desire / Desire 601 (CDMA), Desire 10 Compact, Desire 10 Lifestyle, Desire 10 Pro, Desire 200, Desire 210 dual sim, Desire 300, Desire 310, Desire 310 dual sim, Desire 320, Desire 326G dual sim, Desire 400 dual sim, Desire 500, Desire 501, Desire 501 dual sim, Desire 510, Desire 510 (CDMA), Desire 510 (GSM), Desire 516 dual sim, Desire 520, Desire 526, Desire 526 (CDMA), Desire 526G+ dual sim, Desire 530, Desire 555, Desire 600 dual sim, Desire 601, Desire 601 dual sim, Desire 610, Desire 610 (GSM), Desire 612, Desire 612 (CDMA), Desire 616 dual sim, Desire 620, Desire 620G dual sim, Desire 625, Desire 626, Desire 626 (CDMA), Desire 626 (GSM), Desire 626 (USA), Desire 626G+, Desire 626s, Desire 626s (CDMA), Desire 626s (GSM), Desire 628, Desire 630, Desire 650, Desire 700, Desire 700 dual sim, Desire 728 dual sim, Desire 728 Ultra, Desire 816, Desire 816 dual sim, Desire 816G dual sim, Desire 820, Desire 820 dual sim, Desire 820G+ dual sim, Desire 820q dual sim, Desire 820s dual sim, Desire 825, Desire 826 dual sim, Desire 828 dual sim, Desire 830, Desire C, Desire C (CDMA), Desire Eye, Desire HD, Desire L, Desire P, Desire Q, Desire S, Desire SV, Desire U, Desire V, Desire VC, Desire VT, Desire X, Desire XC, Desire Z, Dream, DROID DNA, DROID ERIS, Droid Incredible, DROID Incredible 2, DROID Incredible 4G LTE, EVO 3D, EVO 3D CDMA, Evo 4G, Evo 4G LTE, Evo 4G+, EVO Design 4G, EVO Design 4G / Hero S (CDMA), EVO Shift 4G, EVO V 4G / EVO 3D (CDMA), EVO View 4G, Explorer, First, Flyer, Flyer Wi-Fi, Freestyle, Fuze / Touch Pro (GSM), G1, G2, Glacier, Gratia, HD mini, HD2, HD7, HD7 / HD7S, HD7S, Hero, Hero (CDMA), Hero S, Imagio, Incredible S, Inspire 4G, J, JAMin / S200 (Prophet), Jetstream, Lead, Legend, Magic, MAX 4G, MDA Compact / xda II mini / JAM (Magician), Merge, Mogul / XV6800 / PPC6800 / P4000, myTouch 3G / Magic, myTouch 3G Slide, myTouch 4G, myTouch 4G Slide, One, One (E8), One (E8) CDMA, One (M7 / CDMA), One (M7 / GSM), One (M8 Eye), One (M8), One (M8) (CDMA), One (M8) (GSM), One (M8) CDMA, One (M8) dual sim, One A9, One A9s, One Dual Sim, One E9, One E9+, One M8s, One M9, One M9 (CDMA), One M9 (GSM), One M9 Prime Camera, One M9+, One M9+ Supreme Camera, One M9s, One Max, One max (CDMA), One ME, One mini, One mini 2, One mini 2 (GSM), One Remix, One Remix / One mini 2 (CDMA), One S, One S C2, One S9, One SC, One ST, One SV, One SV CDMA, One V, One VX, One X, One X

**Exhibit D - Claim Chart for U.S. Patent No. 9,467,838 Against HTC**

AT&T, One X+, One X10, One X9, One XC, One XL, Ozone, Ozone XV6175, Panache, Paradise, Prime, Pure, Pure / Touch Diamond2, Radar, Raider 4G, Rezound, Rhyme, Rhyme CDMA, Rider, S710 (Vox), S730, S740, Salsa, Schubert, SDA (USA) / SP5m (Tornado), Sensation, Sensation 4G, Sensation XE, Sensation XL, Shadow, Shadow (2009), Smart, Snap, Snap S511 (CDMA), SP3i / SDA (Europe) (Feeler), SPV C550 (Hurricane), SPV E200 / XPhone (Voyager), Status, Surround, Tattoo, ThunderBolt, ThunderBolt 4G, Tilt 8925 / TyTN II, Tilt2, Titan, Titan II, Touch (CDMA) / XV6900, Touch 3G, Touch Cruise, Touch Cruise 09, Touch Diamond (CDMA), Touch Diamond2, Touch Diamond2 CDMA, Touch Dual, Touch HD, Touch HD T8285, Touch Pro, Touch Pro (CDMA), Touch Pro2, Touch Pro2 (CDMA), Touch Pro2 (GSM) / Tilt 2, Touch Viva, Touch2, Trophy, Trophy (CDMA), TyTN / 8525 / JasJam (Hermes), U Play, U Ultra, U Ultra , U11, U11, U11 Eyes, U11 Life, U11 Plus, U11+, Velocity 4G, Vivid, Wildfire, Wildfire (CDMA), Wildfire S, Wildfire S (CDMA), Wildfire S (GSM), Wing / P4350 (Herald), xda II / MDA II, FLYER, JETSTREAM, FLYER WI-FI, EVO View 4G, FLYER CDMA, and any variants thereof. AGIS reserves the right to amend this list of Accused Products as discovery progresses. For example, AGIS reviewed Android-based products from multiple Android-based handset manufacturers, including two HTC phones (serial numbers FA73J1500645, FA73E1500899) which are available for inspection at HTC's request. For example, the Accused Products comprise HTC products, including but not limited to the phones and tablets as described herein, running the following versions (and all intervening updates and sub-versions) of the Android mobile operating system: Android 2.3, 4.0, 4.1, 4.2, 4.3, 4.4, 5.0, 5.1, 6.0, 7.0, 7.1, 8.0, and 8.1. For example, the Accused Products comprise HTC products, including but not limited to the phones and tablets as described herein, running any versions of the following Android-based applications and/or software: Android Device Manager, Find My Phone, Find My Device, Google Latitude, Google Plus, Google Hangouts, Google Maps, Google Assistant, Google Search, Google Messages, Android Messenger, Google Allo, Google Duo, Gmail, and Google Chrome. For example, the Accused Products comprise HTC products, including but not limited to the phones and tablets described herein, participating in any networks and/or services related to the execution and/or use of the Android mobile operating system versions and Android-based applications and/or software described herein.

AGIS does not concede that any claims of the '838 Patent that are not listed below are not infringed by the identified products. Moreover, the citations to certain documents and other information below are intended to be exemplary only and in no way foreclose AGIS from citing or relying on additional documents, information, source code, and/or testimony at a later time. These contentions are preliminary in nature, and an analysis of HTC's products, internal documentation, source code, and/or testimony from relevant witnesses may more fully and accurately describe the infringing features of its accused products. Accordingly, AGIS reserves the right to supplement, correct, modify, and/or amend these contentions once such additional information is made available to AGIS. Furthermore, AGIS reserves the right to supplement, correct, modify, and/or amend these contentions as discovery in this case



**Exhibit D - Claim Chart for U.S. Patent No. 9,467,838 Against HTC**

progresses; in view of the Court's claim construction order(s); in view of any positions taken by HTC, including but not limited to positions on claim construction, invalidity, and/or non-infringement; and in connection with the preparation and exchange of expert reports.

<b>US9467838</b>	<b>Exemplary Supporting Evidence Regarding Accused Products</b>
<p>1[P]. A computer-implemented method comprising:</p>	<p>HTC infringes directly and/or indirectly by performing, inducing others to perform, and/or contributing to the performance of: a computer-implemented method [of claim 1].</p> <p>The Accused Products meet the claim limitations by providing device-location tracking features such as those features described below. For example, the Accused Products meet the claim limitations because they are pre-installed with Android mobile operating systems containing code for providing device-location tracking features as provided in the claims limitations herein. For example, the Accused Products run applications and/or software that run within the Android mobile operating system and that use components of the Android mobile operating system to provide device-location tracking features. Upon information and belief, in addition to the components and features of the Android mobile operating system itself, the following applications and/or software run within the Android mobile operating system and use components of the Android mobile operating system to provide device-location tracking features: Android Device Manager, Find My Phone, Find My Device, Google Latitude, Google Plus, Google Hangouts, Google Maps, Google Messages, Android Messenger, Google Allo, Google Duo, and Google Chrome.</p> <p>The Accused Products meet the claim limitations under at least two exemplary methods and/or systems, as shown below, which may be useable together or separately.</p> <p><b><u>Find My Device:</u></b> Android Device Manager is the predecessor to Find My Device and has been available as a standard, pre-installed feature since 2013 and downloadable as a software application. The current iteration, Find My Device, often called the “new and improved Android Device Manager” or “rebranded Android Device Manager” is now part of the standard Google Play Protect suite which is “built in and enabled on all devices,” <i>i.e.</i>, the Accused Products running Android OS. Upon information and belief, the Find My Device method also uses and/or works in conjunction with functionalities associated with Google</p>

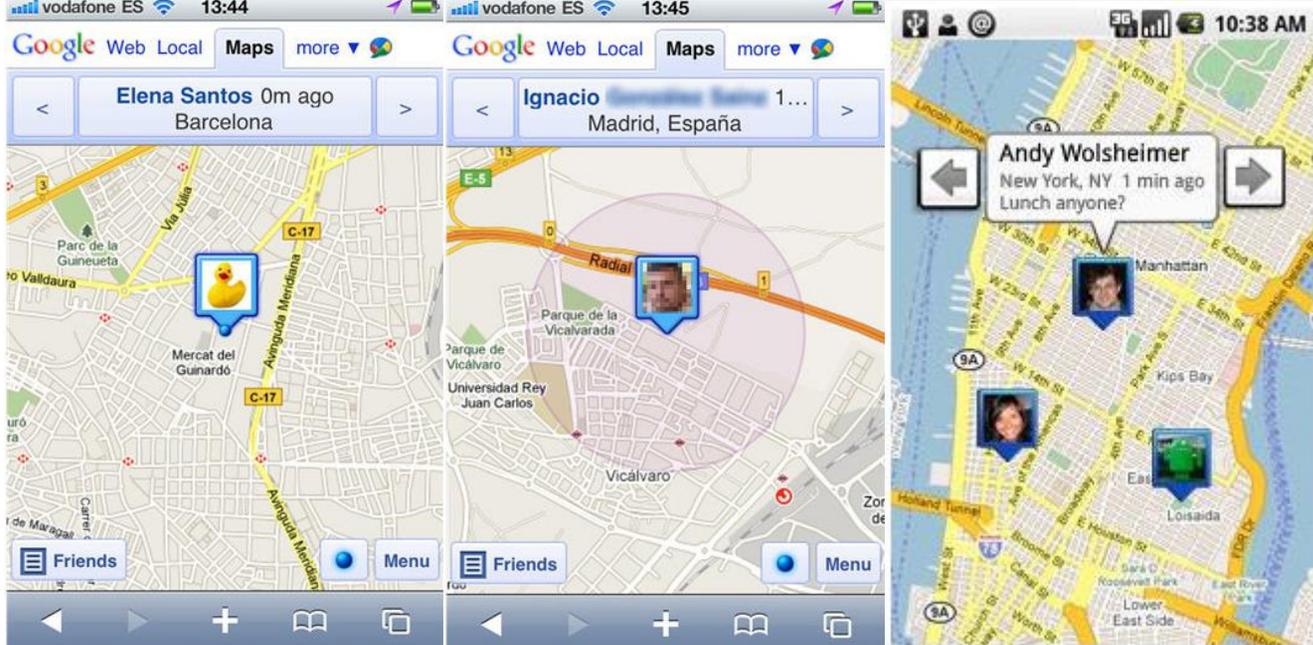
**Exhibit D - Claim Chart for U.S. Patent No. 9,467,838 Against HTC**

US9467838	Exemplary Supporting Evidence Regarding Accused Products
	<p>Maps, Google Messages, Android Messenger, Location Access, Google Chrome, and other features which come pre-installed on the Accused Products. For the purposes of avoiding needlessly presenting cumulative and duplicative evidence, AGIS sets forth the Find My Device feature of the Accused Products as representative of this first exemplary method. AGIS reserves the right to supplement these contentions to the extent that defendant requires additional information in accordance with P.R. 3-1 and for any other reason for which it may deem necessary.</p> <p><i>See, e.g.</i>, <a href="https://www.androidcentral.com/find-my-device;">https://www.androidcentral.com/find-my-device;</a>  <a href="https://support.google.com/android/answer/6160491?hl=en">https://support.google.com/android/answer/6160491?hl=en;</a> <a href="https://android.googleblog.com/2013/08/find-your-lost-phone-with-android.html">https://android.googleblog.com/2013/08/find-your-lost-phone-with-android.html;</a>  <a href="https://play.google.com/store/apps/details?id=com.google.android.apps.adm&amp;hl=en;">https://play.google.com/store/apps/details?id=com.google.android.apps.adm&amp;hl=en;</a>  <a href="https://www.blog.google/products/android/google-play-protect/">https://www.blog.google/products/android/google-play-protect/</a></p> <p><b><u>Google Maps Share Location:</u></b> Share Location is currently included as a standard feature on the Accused Devices operating as a feature of Google Maps. Google Maps is a pre-installed software application in Android OS. The Accused Devices have included the Share Location functionalities since 2009 as part of Google Latitude which was an opt-in feature for Google Maps on Android OS-based mobile devices, such as the Accused Products. Share Location functionalities were briefly shifted from Latitude for Google Maps to Google Plus and Google Hangouts, until reappearing as a standard feature in Google Maps. Upon information and belief, the Share Location method also uses and/or works in conjunction with functionalities associated with Google Maps, Google Messages, Android Messenger, Location Access, Google Allo, Google Duo, Google Chrome, and other features which come pre-installed on the Accused Products. For the purposes of avoiding needlessly presenting cumulative and duplicative evidence, AGIS sets forth Google Maps' Share Location feature of the Accused Products as representative of this second exemplary method. AGIS reserves the right to supplement these contentions to the extent that defendant requires additional information in accordance with P.R. 3-1 and for any other reason for which it may deem necessary.</p> <p><i>See, e.g.</i>, <a href="https://techcrunch.com/2017/03/22/google-maps-now-lets-you-share-your-location-with-friends-and-family-for-a-specific-period-of-time/">https://techcrunch.com/2017/03/22/google-maps-now-lets-you-share-your-location-with-friends-and-family-for-a-specific-period-of-time/;</a> <a href="https://googleblog.blogspot.com/2009/02/see-where-your-friends-are-with-google.html">https://googleblog.blogspot.com/2009/02/see-where-your-friends-are-with-google.html;</a> <a href="https://googleblog.blogspot.jp/2013/07/a-new-google-maps-app-for-smartphones.html">https://googleblog.blogspot.jp/2013/07/a-new-google-maps-app-for-smartphones.html;</a> <a href="http://googleplusproject.blogspot.com/2013/05/google-for-android-42.html">http://googleplusproject.blogspot.com/2013/05/google-for-android-42.html;</a></p>

**Exhibit D - Claim Chart for U.S. Patent No. 9,467,838 Against HTC**

US9467838	Exemplary Supporting Evidence Regarding Accused Products
	<p><a href="https://googleblog.blogspot.com/2013/10/google-hangouts-and-photos-save-some.html">https://googleblog.blogspot.com/2013/10/google-hangouts-and-photos-save-some.html</a></p> <p><b>Control within reach, even when your device isn't</b></p> <p>One of the biggest security risks you're likely to face is simply losing your phone. To help in these times of need, we're launching <a href="#">Find My Device</a> as part of Google Play Protect. With Find My Device you can locate, ring, lock and erase your Android devices—phones, tablets, and even watches. This feature is built in and enabled on all devices; visit <a href="http://android.com/find">android.com/find</a> or check out <a href="#">the app</a>.</p> <p><i>See, e.g.,</i> <a href="https://www.blog.google/products/android/google-play-protect/">https://www.blog.google/products/android/google-play-protect/</a></p> <p><b>Android Device Manager</b></p> <p>If you lose your phone or tablet, the Android Device Manager can help you find its approximate location, make it ring, lock it, or erase its data.</p> <p>These options are turned on by default. To view or change them, open the  Google Settings app (found in  All Apps) and touch <b>Android Device Manager</b>.</p> <hr/> <p>ANDROID QUICK START GUIDE                      ESSENTIALS                      28</p> <p><a href="https://static.googleusercontent.com/media/www.google.com/en/us/help/hc/images/android/android_ug_50/Android-Lollipop-Quick-Start-Guide.pdf">https://static.googleusercontent.com/media/www.google.com/en/us/help/hc/images/android/android_ug_50/Android-Lollipop-Quick-Start-Guide.pdf</a></p> <p>As discussed above and shown below, Google Latitude provided similar features to Google Maps' location sharing feature since 2009:</p>

**Exhibit D - Claim Chart for U.S. Patent No. 9,467,838 Against HTC**

US9467838	Exemplary Supporting Evidence Regarding Accused Products
	 <p><a href="http://googlemobile.blogspot.com/2009/02/locate-your-friends-in-real-time-with.html">http://googlemobile.blogspot.com/2009/02/locate-your-friends-in-real-time-with.html</a></p>
<p>[1A] performing, by a first device:</p>	<p>HTC infringes directly and/or indirectly by performing, inducing others to perform, and/or contributing to the performance, by a first device [the operations of claim 1].</p> <p>For example, the HTC products, such as Android-based phones and tablets run device-location tracking features and/or software such as Android Device Manager, Find My Phone, Find My Device, Google Latitude, Google Plus, Google Hangouts, Google Maps, Google Messages, and Android Messenger, Google Allo, Google Duo, and Google Chrome.</p>
<p>[1B] joining a communication network</p>	<p>HTC infringes directly and/or indirectly by performing, inducing others to perform, and/or contributing to the performance of: joining a communication network corresponding to a group, wherein joining the</p>

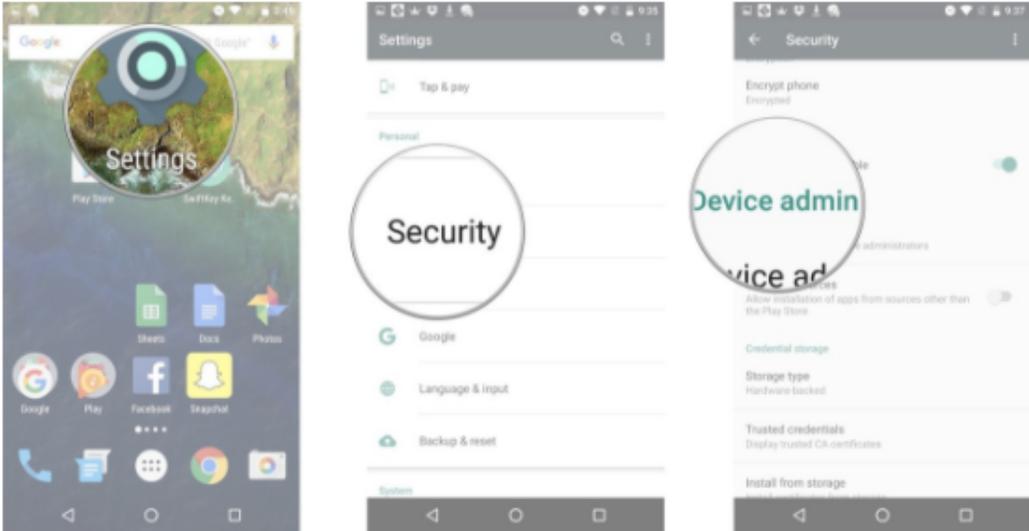
**Exhibit D - Claim Chart for U.S. Patent No. 9,467,838 Against HTC**

US9467838	Exemplary Supporting Evidence Regarding Accused Products
<p>corresponding to a group, wherein joining the communication network comprises transmitting a message including an identifier corresponding to the group;</p>	<p>communication network comprises transmitting a message including an identifier corresponding to the group.</p> <p><b><u>Regarding Find My Device</u></b> and Android Device Manager, the Accused Products require a user to join the corresponding network by: signing-in to the device with an identifier (e.g., Google Account) or linking the device to the identifier by remote means. The sign-in process takes place within the Find My Device software on the Accused Product; by navigating to android.com/find within the Google Chrome browser on the Accused Product; by navigating to google.com and typing “find my phone” within the Google Chrome browser on the Accused Product; or by searching “find my phone” within the search bar of the Google home screen and/or Google application on the Accused Product. Alternatively, the sign-in process may partially or completely take place using credentials already provided when the user associates a Google Account with the Accused Product, e.g., during initial setup of the Accused Product. Subject to discovery, one or more additional or substitute identifiers may to correspond to the group. The sign-in process involves a user entering its Google Account and additional authentication data on the interface of the Accused Product and sending a message containing the Google Account and additional authentication data over a network to the group. The group comprises the multiple devices linked to the identifier.</p> <p><b><u>Regarding Google Maps</u></b>, Google Latitude, Google Plus, Google Hangouts, Google Messages, and Android Messenger, Google Duo, Google Allo, and Google Chrome, the Accused Products require a user to join the corresponding network by signing-in to the device with an identifier (e.g., Google Account). The sign-in process takes place within the Google Maps software on the Accused Product or by navigating to maps.google.com within the Google Chrome browser on the Accused Product. Alternatively, the sign-in process may partially or completely take place using credentials already provided when the user associates a Google Account with the Accused Product, e.g., during initial setup of the Accused Product. Subject to discovery, one or more additional or substitute identifiers may to correspond to the group. The sign-in process involves a user entering its Google Account and additional authentication data on the interface of the Accused Product and sending a message containing the Google Account and additional authentication data over a network to the group. The group comprises the multiple identifiers, individuals, profiles, and/or devices associated with the group.</p> <p>Further regarding Google Maps, Google Latitude, Google Plus, Google Hangouts, Google Messages, and Android Messenger, Google Duo, Google Allo, and Google Chrome, the Accused Products alternatively</p>

**Exhibit D - Claim Chart for U.S. Patent No. 9,467,838 Against HTC**

US9467838	<b>Exemplary Supporting Evidence Regarding Accused Products</b>
	<p>require a user to send a request containing the identifier (e.g., Google Account). Subject to discovery, additional identifiers may be assigned or used to correspond to the group. The request may be an invitation or message that associates a Google Account with one or more Google Accounts for the purposes of sharing locations within the group. The group comprises the multiple identifiers, individuals, profiles, and/or devices associated with the group</p> <p><b><u>Exemplary Support for Find My Device:</u></b></p>

**Exhibit D - Claim Chart for U.S. Patent No. 9,467,838 Against HTC**

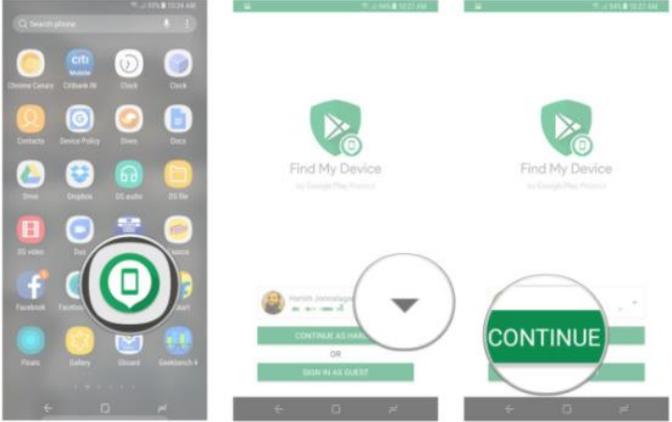
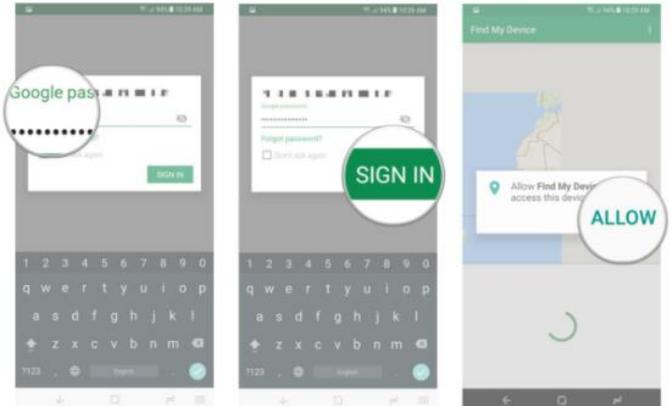
US9467838	Exemplary Supporting Evidence Regarding Accused Products
	<p data-bbox="548 245 1365 282"><b>How to enable Find My Device on your phone</b></p> <p data-bbox="548 321 1564 440">In newer Android phones, the Find My Device service is already located conveniently in your Settings app, but if you can't find it you can always <a href="#">download Find My Device from the Google Play Store</a>. This locating service has essentially amalgamated with Google to make finding your phone easier. There are just a couple of things you'll need to activate.</p> <ol data-bbox="548 488 856 618" style="list-style-type: none"><li>1. Launch Settings.</li><li>2. Tap Security.</li><li>3. Tap Device Administration.</li></ol> <div data-bbox="579 656 1608 1187"></div> <ol data-bbox="548 1243 1262 1268" style="list-style-type: none"><li>4. Tap Find My Device so that a checkmark appears in the checkbox.</li></ol> <p data-bbox="537 1317 1285 1349"><a href="https://www.androidcentral.com/how-track-android-phone">https://www.androidcentral.com/how-track-android-phone</a></p>

**Exhibit D - Claim Chart for U.S. Patent No. 9,467,838 Against HTC**

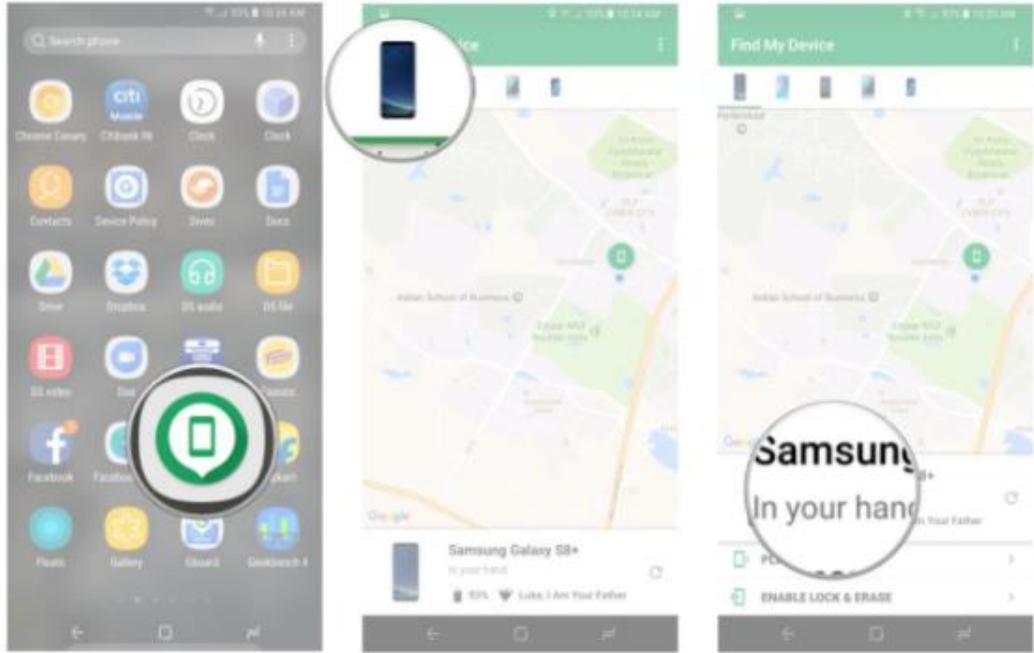
US9467838	Exemplary Supporting Evidence Regarding Accused Products
	<p data-bbox="558 240 1203 277"><b>How to locate your phone with Google</b></p> <p data-bbox="558 310 1520 363">Should you happen to lose your phone, you can locate its whereabouts by logging into your Google account from any computer or even from another phone.</p> <ol data-bbox="558 407 1276 532" style="list-style-type: none"> <li data-bbox="558 407 1136 431">1. Launch a web browser from a phone, tablet, or computer.</li> <li data-bbox="558 456 1276 480">2. Navigate to Google if it is not your default search engine or home page.</li> <li data-bbox="558 505 1104 529">3. Type find my phone android in the Google search bar.</li> </ol> <div data-bbox="585 565 1549 1133"> <p>The image contains three sequential screenshots from an Android phone. The first screenshot shows the home screen with various app icons; the Chrome browser icon is circled in red. The second screenshot shows the Google search page with the search bar containing the text 'google' circled in red. The third screenshot shows the search results for 'find my phone android', with the first result, 'find my phone android app', circled in red.</p> </div> <ol data-bbox="558 1182 1545 1317" style="list-style-type: none"> <li data-bbox="558 1182 1171 1206">4. Tap on Find My Device (usually the first option in the search).</li> <li data-bbox="558 1230 1545 1317">5. Enter your email address and password just as though you were checking your email. If you have 2-step verification set up on your Google account (and you most certainly should), you'll need to complete that process as well.</li> </ol> <p data-bbox="537 1325 1283 1357"><a href="https://www.androidcentral.com/how-track-android-phone">https://www.androidcentral.com/how-track-android-phone</a></p>



**Exhibit D - Claim Chart for U.S. Patent No. 9,467,838 Against HTC**

US9467838	Exemplary Supporting Evidence Regarding Accused Products
	<ol style="list-style-type: none"> <li>1. Open Find Device from your home screen or app drawer.</li> <li>2. Select the Google account you want to use the service with.</li> <li>3. Hit the Continue as button.</li> </ol>  <ol style="list-style-type: none"> <li>4. Enter your Google account password.</li> <li>5. Tap Sign in.</li> <li>6. Give location access to the service.</li> </ol>  <p><a href="https://www.androidcentral.com/find-my-device">https://www.androidcentral.com/find-my-device</a></p>

**Exhibit D - Claim Chart for U.S. Patent No. 9,467,838 Against HTC**

US9467838	Exemplary Supporting Evidence Regarding Accused Products
	<p>Once you're signed in to Find My Device, you'll see a map with your current location as well as the make and model of your phone, and two options — Play Sound, and Enable Lock &amp; Erase. Hitting the latter option will allow you to start using the Lock and Erase functions.</p> <p>If you've signed into more than one phone, you can select a particular device by browsing the list at the top of the screen.</p> <ol style="list-style-type: none"> <li>1. Open <b>Find My Device</b> from your home screen or app drawer.</li> <li>2. Select your phone from the list of devices at the top of the screen.</li> <li>3. See if your phone is discoverable.</li> </ol>  <p><a href="https://www.androidcentral.com/find-my-device">https://www.androidcentral.com/find-my-device</a></p>

**Exhibit D - Claim Chart for U.S. Patent No. 9,467,838 Against HTC**

US9467838	Exemplary Supporting Evidence Regarding Accused Products
	<h2 data-bbox="548 272 1686 326">How to locate your phone over the internet</h2> <p data-bbox="548 370 1692 511">If you've lost your phone, you can remotely locate it through the <a href="#">Find My Device website</a>. You'll need to sign in to the Google account that was used to set up Find My Device. It takes a few seconds, but the service should be able to track your phone. Alternatively, you can also do a Google search for "find my phone" to locate your handset.</p> <ol data-bbox="539 570 1031 722" style="list-style-type: none"><li>1. Head to the <a href="#">Find My Device website</a>.</li><li>2. Sign in to your <a href="#">Google account</a>.</li><li>3. Check if your device is visible.</li></ol> <div data-bbox="583 764 1793 1219"></div> <p data-bbox="533 1252 1157 1279"><a href="https://www.androidcentral.com/find-my-device">https://www.androidcentral.com/find-my-device</a></p> <p data-bbox="533 1321 1052 1349"><b><u>Exemplary Support for Google Maps:</u></b></p>

**Exhibit D - Claim Chart for U.S. Patent No. 9,467,838 Against HTC**

US9467838	Exemplary Supporting Evidence Regarding Accused Products
	<p data-bbox="573 240 989 261">COMPUTER    <b>ANDROID</b>    IPHONE &amp; IPAD</p> <hr data-bbox="560 289 1591 293"/> <h3 data-bbox="560 345 1045 383">If they have a Google Account</h3> <ol data-bbox="569 404 1440 695" style="list-style-type: none"> <li data-bbox="569 404 1241 425">1. If you haven't already, add their Gmail address to your <a href="#">Google Contacts</a> .</li> <li data-bbox="569 440 1440 461">2. On your Android phone or tablet, open the Google Maps app  and sign in. <a href="#">Learn how to sign in.</a></li> <li data-bbox="569 475 1056 496">3. Tap Menu ≡ &gt; <b>Location sharing</b> &gt; Add People +.</li> <li data-bbox="569 511 1026 532">4. Choose how long you want to share your location.</li> <li data-bbox="569 547 1161 610">5. Tap <b>Select People</b>.             <ul data-bbox="590 586 1161 607" style="list-style-type: none"> <li data-bbox="590 586 1161 607">• If you're asked about your contacts, give Google Maps access.</li> </ul> </li> <li data-bbox="569 634 905 656">6. Choose who you want to share with.</li> <li data-bbox="569 670 684 691">7. Tap <b>Share</b>.</li> </ol> <h3 data-bbox="560 756 1131 794">If they don't have a Google Account</h3> <ol data-bbox="569 815 1581 938" style="list-style-type: none"> <li data-bbox="569 815 1440 836">1. On your Android phone or tablet, open the Google Maps app  and sign in. <a href="#">Learn how to sign in.</a></li> <li data-bbox="569 850 1056 872">2. Tap Menu ≡ &gt; <b>Location sharing</b> &gt; Add People +.</li> <li data-bbox="569 886 1581 938">3. Tap More ... &gt; <b>Copy to clipboard</b>. People with this link can see your location for as long as you choose, up to 72 hours.</li> </ol> <h3 data-bbox="560 987 890 1024">Share using another app</h3> <p data-bbox="560 1040 1226 1062">You can also share through messaging apps. Tap More ... &gt; select an app.</p> <h3 data-bbox="560 1122 764 1159">Stop sharing</h3> <ol data-bbox="569 1180 1226 1278" style="list-style-type: none"> <li data-bbox="569 1180 863 1201">1. Open the Google Maps app .</li> <li data-bbox="569 1216 890 1237">2. Tap Menu ≡ &gt; <b>Location sharing</b>.</li> <li data-bbox="569 1252 1226 1273">3. Next to the person with whom you want to stop sharing, tap Remove ✕ .</li> </ol> <p data-bbox="537 1295 1724 1326"><a href="https://support.google.com/maps/answer/7326816?co=GENIE.Platform%3DAndroid&amp;oco=1">https://support.google.com/maps/answer/7326816?co=GENIE.Platform%3DAndroid&amp;oco=1</a></p>

**Exhibit D - Claim Chart for U.S. Patent No. 9,467,838 Against HTC**

US9467838	Exemplary Supporting Evidence Regarding Accused Products
	<p data-bbox="556 253 863 297"><b>Share your E.T.A</b></p> <p data-bbox="556 321 1686 347">After you start your drive, you can share your destination, estimated arrival time, and where you are on the route.</p> <ol data-bbox="556 375 1381 618" style="list-style-type: none"> <li data-bbox="556 375 909 401">1. Open the Google Maps app .</li> <li data-bbox="556 418 1182 444">2. Set a driving destination. <a href="#">Learn how to navigate to a place.</a></li> <li data-bbox="556 462 1224 488">3. After you start navigation, tap More ^ &gt; <b>Share trip progress.</b></li> <li data-bbox="556 506 898 532">4. Choose a person from the list.</li> <li data-bbox="556 550 701 576">5. Tap <b>Share.</b></li> <li data-bbox="556 594 1381 620">6. Location Sharing will stop when you reach your destination or stop navigating.</li> </ol> <ul data-bbox="556 646 1224 672" style="list-style-type: none"> <li data-bbox="556 646 1224 672">• To stop sharing before you arrive, tap More ^ &gt; <b>Stop sharing.</b></li> </ul> <p data-bbox="556 740 978 784"><b>See where someone is</b></p> <p data-bbox="556 808 1289 834">If someone shares their location with you, you can see them on the map.</p> <ol data-bbox="556 862 940 976" style="list-style-type: none"> <li data-bbox="556 862 909 888">1. Open the Google Maps app .</li> <li data-bbox="556 906 940 932">2. Tap Menu ≡ &gt; <b>Location sharing.</b></li> <li data-bbox="556 950 772 976">3. Choose someone.</li> </ol> <ul data-bbox="556 1002 1325 1027" style="list-style-type: none"> <li data-bbox="556 1002 1325 1027">• To see an updated location, tap on a friend's icon &gt; More ⋮ &gt; <b>Refresh.</b></li> </ul> <p data-bbox="556 1084 1056 1128"><b>Stop seeing someone's location</b></p> <ol data-bbox="556 1146 1482 1300" style="list-style-type: none"> <li data-bbox="556 1146 909 1172">1. Open the Google Maps app .</li> <li data-bbox="556 1190 856 1216">2. On the map, tap their icon.</li> <li data-bbox="556 1234 877 1260">3. At the bottom, tap More ^ .</li> <li data-bbox="556 1278 1482 1304">4. To temporarily hide someone, tap <b>Hide from map.</b> You can stop hiding them at any time.</li> </ol> <p data-bbox="556 1328 1766 1354"><b>Note:</b> You can stop someone's location from ever appearing on your map. <a href="#">Learn how to block another person's account.</a></p> <p data-bbox="533 1378 1724 1404"><a href="https://support.google.com/maps/answer/7326816?co=GENIE.Platform%3DAndroid&amp;oco=1">https://support.google.com/maps/answer/7326816?co=GENIE.Platform%3DAndroid&amp;oco=1</a></p>

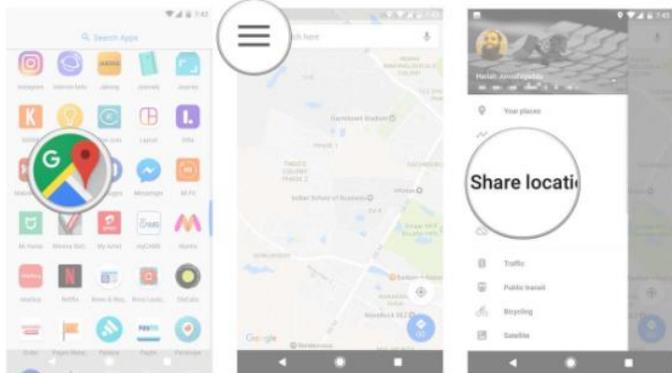
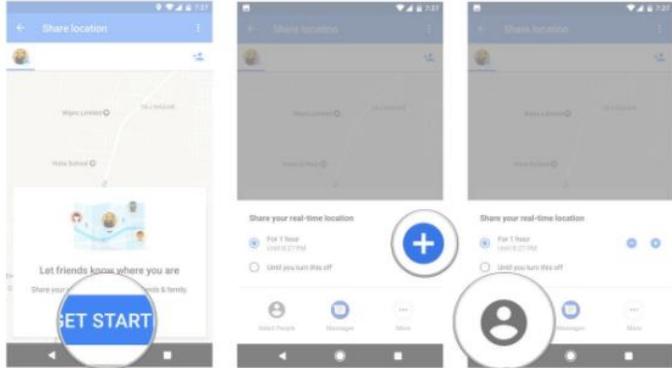
**Exhibit D - Claim Chart for U.S. Patent No. 9,467,838 Against HTC**

US9467838	Exemplary Supporting Evidence Regarding Accused Products
	<p data-bbox="548 313 1020 362"><b>Create a list of places</b></p> <p data-bbox="548 380 1367 402">In Google Maps, you can create a list of places, like your favorite places or places you want to visit.</p> <p data-bbox="562 475 957 495">COMPUTER <b>ANDROID</b> IPHONE &amp; IPAD</p> <hr data-bbox="548 521 1394 524"/> <p data-bbox="548 573 789 605"><b>Make a new list</b></p> <ol data-bbox="558 630 1108 789" style="list-style-type: none"><li>1. On your Android phone or tablet, open the Google Maps app .</li><li>2. Tap Menu  &gt; <b>Your places</b> &gt; <b>Saved</b>.</li><li>3. In the bottom right, tap Add .</li><li>4. Enter a name and description.</li><li>5. Tap <b>Save</b>.</li></ol> <p data-bbox="548 846 867 878"><b>Save a place to a list</b></p> <ol data-bbox="558 902 999 1062" style="list-style-type: none"><li>1. Open the Google Maps app .</li><li>2. Search for a place or tap it on the map.</li><li>3. At the bottom, tap the place's name or address.</li><li>4. Tap <b>Save</b>.</li><li>5. Choose a list. To create a new list, tap <b>New list</b> .</li></ol> <p data-bbox="548 1118 758 1151"><b>See your lists</b></p> <ol data-bbox="558 1175 905 1230" style="list-style-type: none"><li>1. Open the Google Maps app .</li><li>2. Tap Menu  &gt; <b>Your places</b> &gt; <b>Saved</b>.</li></ol> <p data-bbox="533 1239 1906 1304"><a href="https://support.google.com/maps/answer/7280933?hl=en&amp;ref_topic=7301134&amp;co=GENIE.Platform%3DAndroid&amp;oco=1">https://support.google.com/maps/answer/7280933?hl=en&amp;ref_topic=7301134&amp;co=GENIE.Platform%3DAndroid&amp;oco=1</a></p>

**Exhibit D - Claim Chart for U.S. Patent No. 9,467,838 Against HTC**

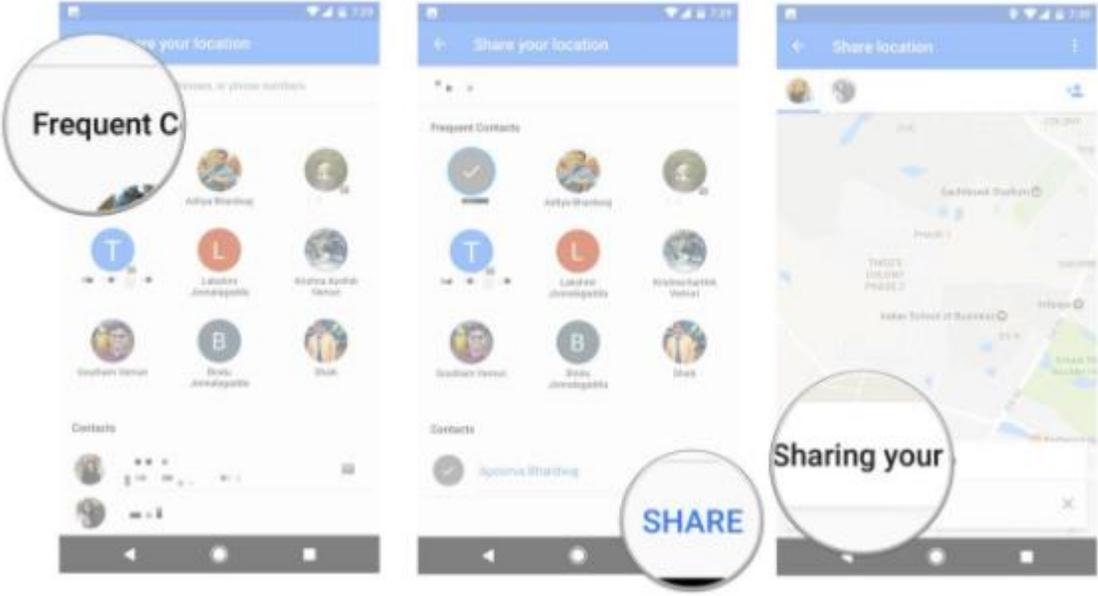
US9467838	Exemplary Supporting Evidence Regarding Accused Products
	<p data-bbox="567 243 903 284"><b>Hide or share lists</b></p> <p data-bbox="567 308 934 332"><b>Note:</b> You can't share starred places.</p> <ol data-bbox="577 365 1711 633" style="list-style-type: none"> <li data-bbox="577 365 913 389">1. Open the Google Maps app .</li> <li data-bbox="577 406 997 430">2. Tap Menu  &gt; <b>Your places</b> &gt; <b>Saved</b>.</li> <li data-bbox="577 446 1711 633">3. Next to the list you want to share, tap More  &gt; choose an option: <ul data-bbox="598 487 1711 633" style="list-style-type: none"> <li data-bbox="598 487 1470 511">• <b>Hide/Show on your map:</b> Display or hide your saved places when looking at the map.</li> <li data-bbox="598 527 1081 552">• <b>Share list:</b> Allow others to see your saved list.</li> <li data-bbox="598 568 1711 633">• <b>Sharing options:</b> You can make your list public, private, or shared. To let anyone with the link see your list, tap <b>Shared</b>. To let anyone find and follow your list, tap <b>Public</b>.</li> </ul> </li> </ol> <p data-bbox="567 698 787 738"><b>Follow a list</b></p> <p data-bbox="567 763 1753 820">If you follow a list made by someone else, their saved places will show up in Your Places. The places will also appear as suggested locations in Google Maps.</p> <p data-bbox="567 876 934 917"><b>Follow a list using a link</b></p> <ol data-bbox="577 933 1375 1047" style="list-style-type: none"> <li data-bbox="577 933 976 958">1. Tap on the link you received to open it.</li> <li data-bbox="577 974 1291 998">2. Tap <b>Follow</b>. This list will now be added to the group of lists you follow.</li> <li data-bbox="577 1015 1375 1047">3. <b>Optional:</b> To unfollow a list someone shared with you, tap the list &gt; <b>Following</b>.</li> </ol> <p data-bbox="567 1096 945 1136"><b>See lists made by others</b></p> <p data-bbox="567 1153 1354 1177">If a user has any Google Maps lists that were made public, you can follow them.</p> <ol data-bbox="577 1209 1155 1323" style="list-style-type: none"> <li data-bbox="577 1209 1155 1234">1. Tap on the name of a user whose list you want to follow.</li> <li data-bbox="577 1250 703 1274">2. Tap <b>Lists</b>.</li> <li data-bbox="577 1291 1155 1323">3. Tap on the list you want to follow &gt; More  &gt; <b>Follow</b>.</li> </ol> <p data-bbox="535 1331 1911 1396"><a href="https://support.google.com/maps/answer/7280933?hl=en&amp;ref_topic=7301134&amp;co=GENIE.Platform%3DAndroid&amp;oco=1">https://support.google.com/maps/answer/7280933?hl=en&amp;ref_topic=7301134&amp;co=GENIE.Platform%3DAndroid&amp;oco=1</a></p>

**Exhibit D - Claim Chart for U.S. Patent No. 9,467,838 Against HTC**

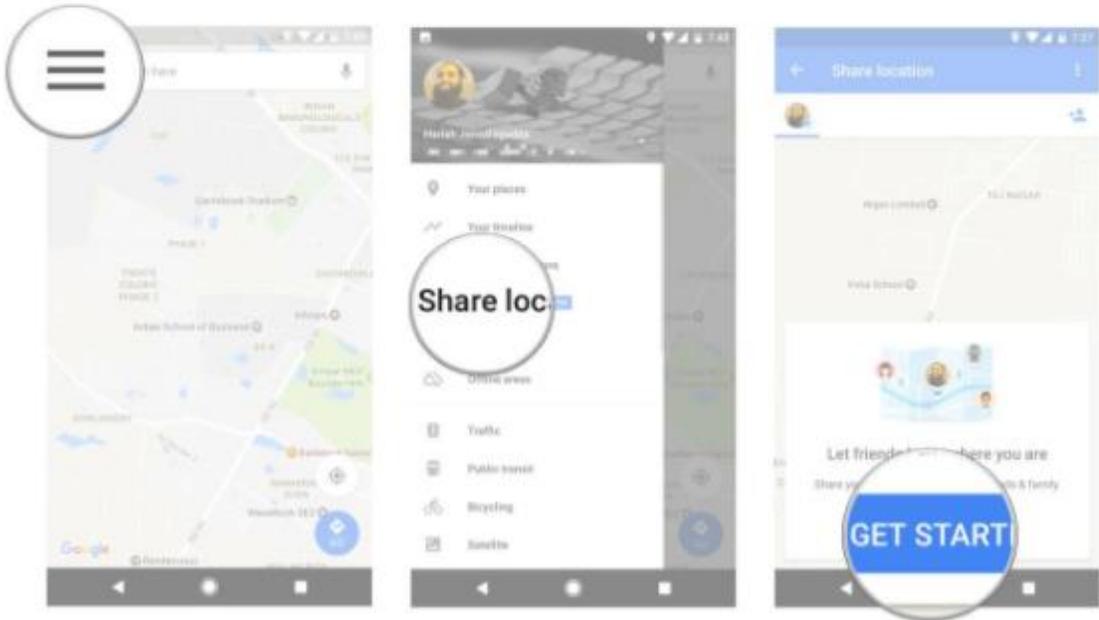
US9467838	Exemplary Supporting Evidence Regarding Accused Products
	<p data-bbox="541 272 1178 310"><b>How to share your location in Google Maps</b></p> <ol data-bbox="541 337 1157 423" style="list-style-type: none"> <li>1. Open Google Maps from the app drawer or the home screen.</li> <li>2. Tap the <b>hamburger menu</b> (the three horizontal lines) on the top left corner of the screen.</li> <li>3. Select <b>Share location</b>.</li> </ol>  <ol data-bbox="541 862 1192 971" style="list-style-type: none"> <li>4. Tap <b>Get Started</b>.</li> <li>5. Use the <b>+</b> icon to select a time period or select the <b>Until you turn this off</b> setting to share your location indefinitely.</li> <li>6. Tap <b>Select People</b>.</li> </ol>  <p data-bbox="541 1377 1381 1409"><a href="https://www.androidcentral.com/how-share-location-google-maps">https://www.androidcentral.com/how-share-location-google-maps</a></p>



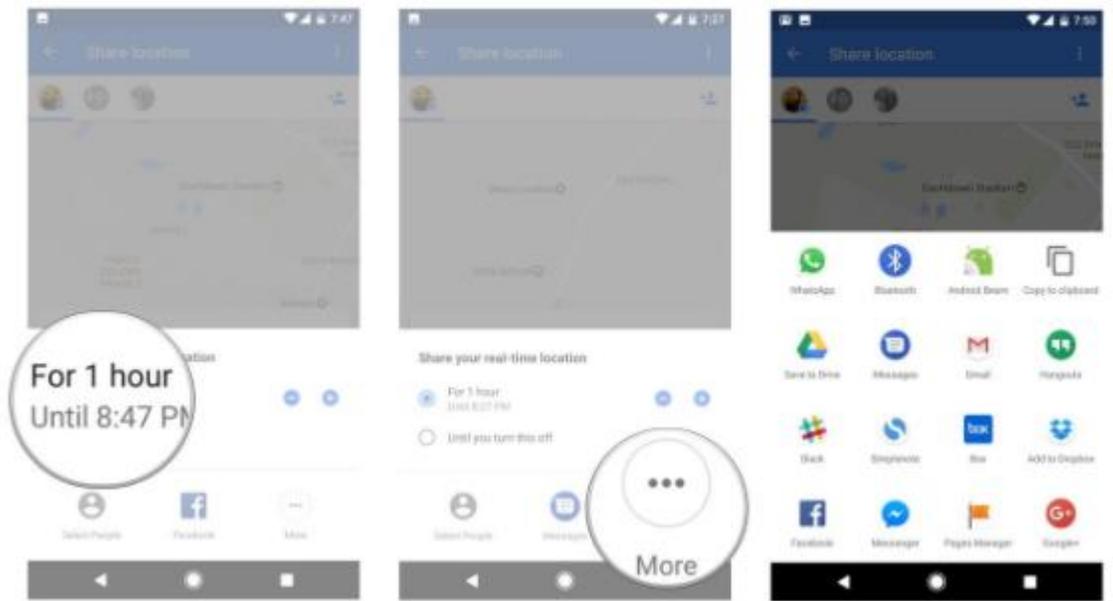
**Exhibit D - Claim Chart for U.S. Patent No. 9,467,838 Against HTC**

US9467838	Exemplary Supporting Evidence Regarding Accused Products
	<p data-bbox="552 289 1602 462">7. You'll see a list of your frequent contacts at the top, along with a full list of contacts. Pick the contacts by tapping their name. 8. Once you've selected the contacts you want to share your location to, tap Share. 9. You'll see a message saying that the selected contact can view your location.</p>  <p data-bbox="533 1138 1381 1170"><a href="https://www.androidcentral.com/how-share-location-google-maps">https://www.androidcentral.com/how-share-location-google-maps</a></p>

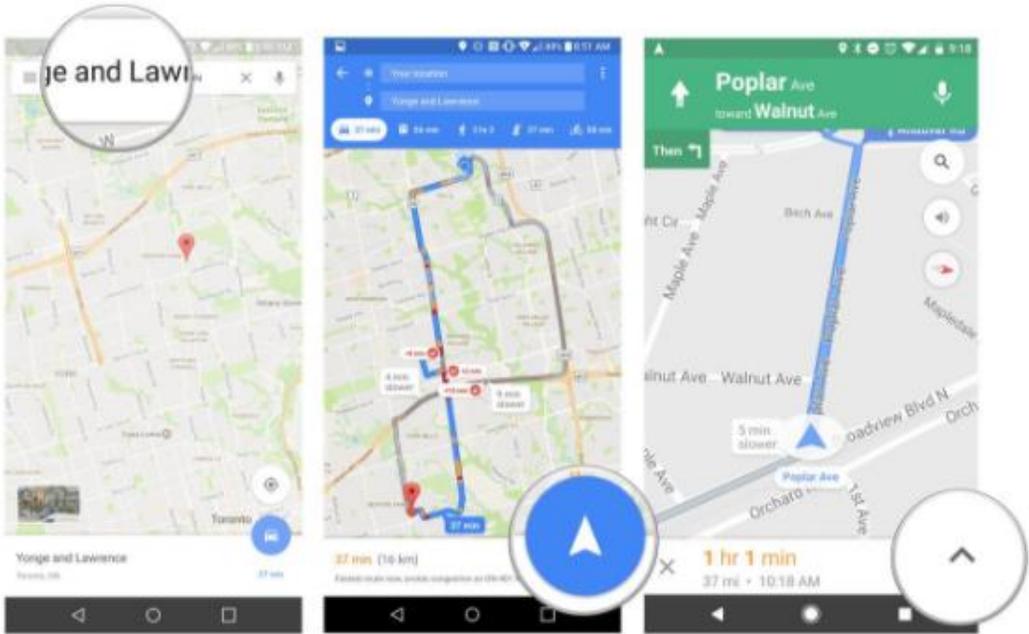
**Exhibit D - Claim Chart for U.S. Patent No. 9,467,838 Against HTC**

US9467838	Exemplary Supporting Evidence Regarding Accused Products
	<h2 data-bbox="548 240 1281 289">How to create a shareable link</h2> <p data-bbox="548 329 1486 358">You can also create a link and use it to share your location easily. Here's how to do it:</p> <ol data-bbox="541 407 1260 548" style="list-style-type: none"><li>1. Tap the <b>hamburger menu</b> on the top left corner of the screen.</li><li>2. Select <b>Share location</b>.</li><li>3. Tap <b>Get Started</b>.</li></ol>  <p data-bbox="533 1230 1381 1263"><a href="https://www.androidcentral.com/how-share-location-google-maps">https://www.androidcentral.com/how-share-location-google-maps</a></p>

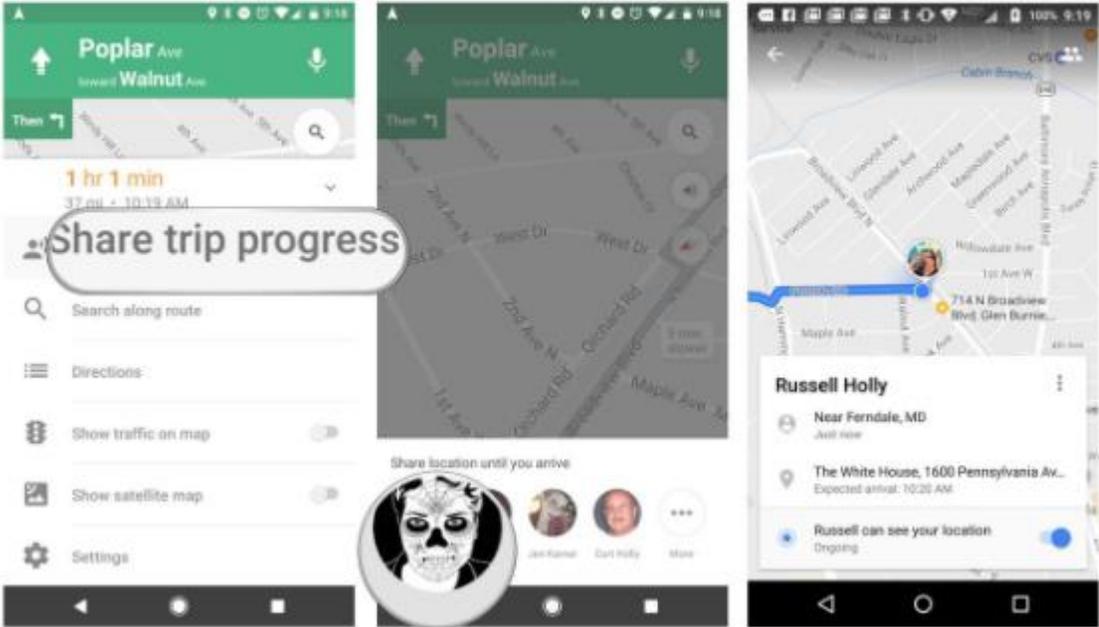
**Exhibit D - Claim Chart for U.S. Patent No. 9,467,838 Against HTC**

US9467838	Exemplary Supporting Evidence Regarding Accused Products
	<p>4. Select the amount of time you want to share your location.</p> <p>5. Tap More.</p> <p>6. Select your app of choice to create and send a unique URL that broadcasts your current location. You can email it, send the link via Messenger, or even tweet it to the intended recipient.</p>  <p><a href="https://www.androidcentral.com/how-share-location-google-maps">https://www.androidcentral.com/how-share-location-google-maps</a></p>

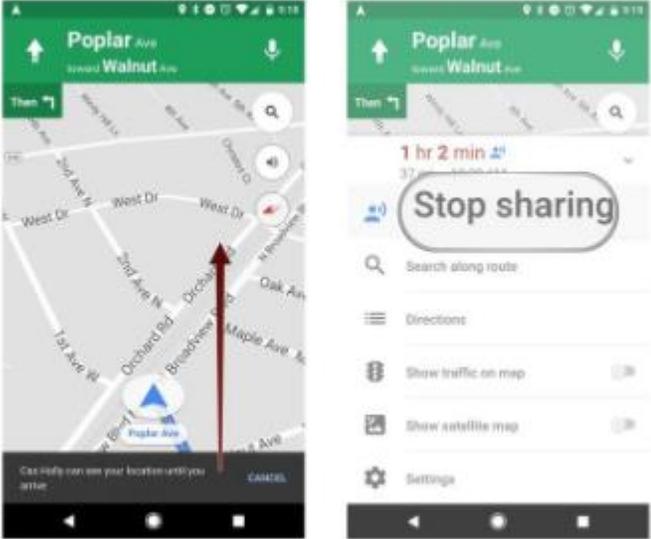
## Exhibit D - Claim Chart for U.S. Patent No. 9,467,838 Against HTC

US9467838	Exemplary Supporting Evidence Regarding Accused Products
	<h2 data-bbox="550 240 1451 337">How to share your navigation directions while you walk, drive or transit</h2> <p data-bbox="550 375 1577 464">One of the best ways to share your location with someone is during a drive, walk or transit. If you're meeting a friend or family member somewhere, or navigating towards their home, Google Maps lets you share your location with them for the duration of the trip. It's magic!</p> <ol data-bbox="550 513 1419 643" style="list-style-type: none"><li data-bbox="550 513 995 537">1. In the <b>search bar</b> enter your destination.</li><li data-bbox="550 565 1419 589">2. Pick your navigation type (drive, transit, walk) and press the <b>blue navigate button</b>.</li><li data-bbox="550 617 1419 641">3. Tap the arrow next to the <b>time-to-destination</b> number at the bottom of the screen.</li></ol>  <p data-bbox="535 1328 1381 1360"><a href="https://www.androidcentral.com/how-share-location-google-maps">https://www.androidcentral.com/how-share-location-google-maps</a></p>

**Exhibit D - Claim Chart for U.S. Patent No. 9,467,838 Against HTC**

US9467838	Exemplary Supporting Evidence Regarding Accused Products
	<p>4. Tap Share trip progress.</p> <p>5. Choose one or more contacts to share trip progress.</p>  <p>You can also stop sharing your location with someone before a trip ends.  <a href="https://www.androidcentral.com/how-share-location-google-maps">https://www.androidcentral.com/how-share-location-google-maps</a></p>

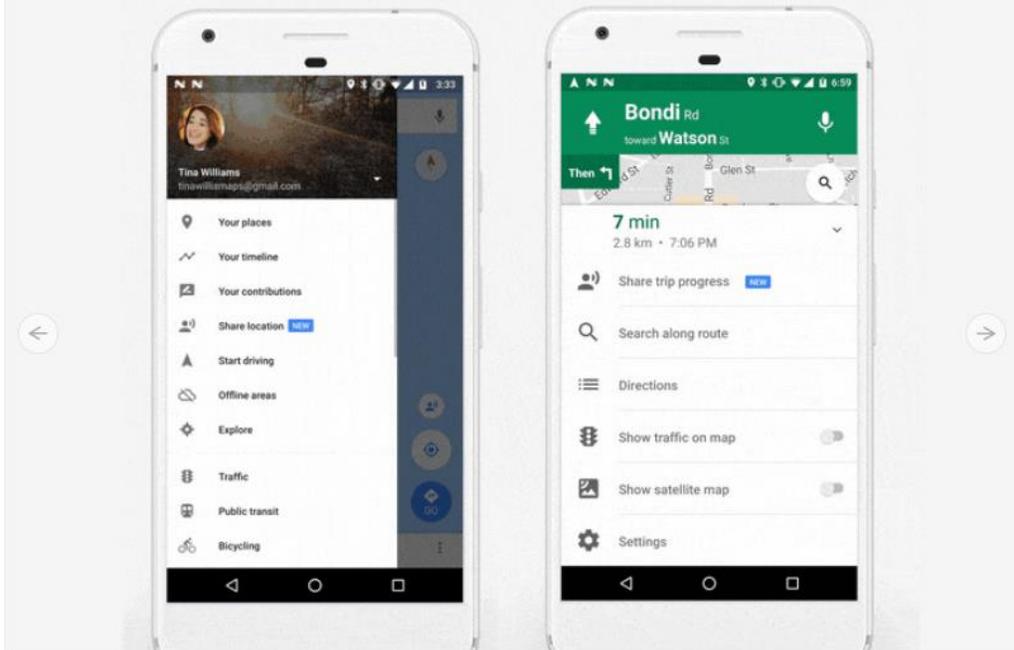
**Exhibit D - Claim Chart for U.S. Patent No. 9,467,838 Against HTC**

US9467838	Exemplary Supporting Evidence Regarding Accused Products
	<ol style="list-style-type: none"><li data-bbox="562 245 1493 272">1. Tap the arrow next to the time-to-destination number at the bottom of the screen.</li><li data-bbox="562 302 793 329">2. Tap Stop sharing.</li></ol> <div data-bbox="772 383 1423 922"></div> <p data-bbox="562 976 659 1003">That's It!</p> <p data-bbox="562 1045 1633 1073">Are you excited that location sharing is back in Google Maps? How often do you use the feature?</p> <p data-bbox="537 1086 1381 1114"><a href="https://www.androidcentral.com/how-share-location-google-maps">https://www.androidcentral.com/how-share-location-google-maps</a></p> <p data-bbox="537 1195 1444 1222">As shown below, a group may also be defined within Google Contacts.</p>

**Exhibit D - Claim Chart for U.S. Patent No. 9,467,838 Against HTC**

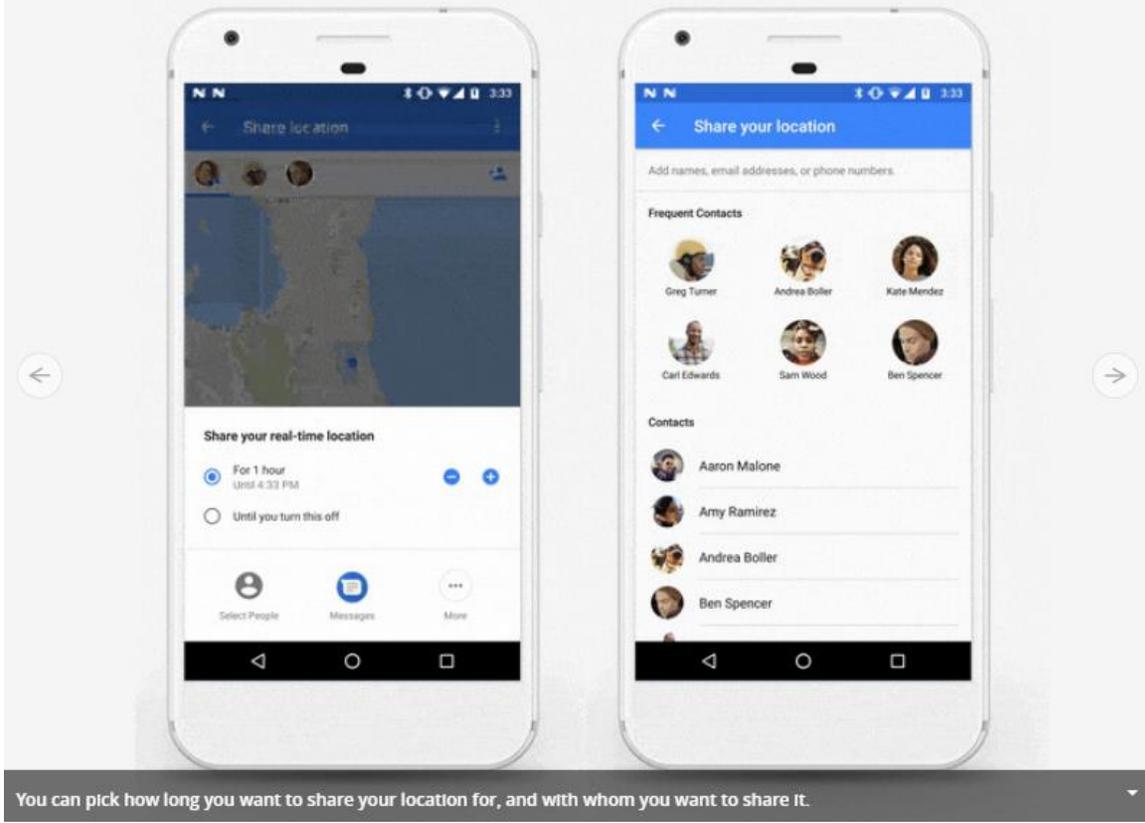
US9467838	Exemplary Supporting Evidence Regarding Accused Products
	<p><b>See your contacts</b></p> <ol style="list-style-type: none"> <li>1. Open your device's Contacts app .</li> <li>2. Tap Menu .</li> </ol> <ul style="list-style-type: none"> <li>• <b>See contacts by label:</b> Choose a label from the list.</li> <li>• <b>See contacts for another account:</b> Tap Down arrow  &gt; pick an account.</li> <li>• <b>See the contacts for all your accounts:</b> Choose <b>All contacts</b>.</li> </ul> <p><b>Tip:</b> If you have multiple contacts with the same information, the information will be grouped into one contact.</p> <ul style="list-style-type: none"> <li>• <b>See your Google Account contacts on the web:</b> Go to <a href="#">Google Contacts</a> .</li> </ul> <p><a href="https://support.google.com/android/answer/6118731?hl=en&amp;ref_topic=6118711">https://support.google.com/android/answer/6118731?hl=en&amp;ref_topic=6118711</a></p> <p><b>Label your contacts</b></p> <p>You can group contacts together using labels.</p> <ol style="list-style-type: none"> <li>1. Open your device's Contacts app .</li> <li>2. Tap Menu  &gt; <b>Create label</b>.</li> <li>3. Enter a label name and tap <b>Ok</b>.</li> </ol> <ul style="list-style-type: none"> <li>• <b>Add one contact to a label:</b> Tap Add contact  &gt; choose a contact.</li> <li>• <b>Add multiple contacts to a label:</b> Tap Add contact  &gt; touch and hold a contact &gt; tap the other contacts &gt; tap <b>Add</b>.</li> </ul> <p><a href="https://support.google.com/android/answer/6118731?hl=en&amp;ref_topic=6118711">https://support.google.com/android/answer/6118731?hl=en&amp;ref_topic=6118711</a></p> <p><b>Share your contacts</b></p> <ol style="list-style-type: none"> <li>1. Open your device's Contacts app .</li> <li>2. Tap a contact in the list.</li> <li>3. Tap More  &gt; <b>Share</b>.</li> <li>4. Choose how you want to share the contact.</li> </ol> <p><a href="https://support.google.com/android/answer/6118731?hl=en&amp;ref_topic=6118711">https://support.google.com/android/answer/6118731?hl=en&amp;ref_topic=6118711</a></p>

**Exhibit D - Claim Chart for U.S. Patent No. 9,467,838 Against HTC**

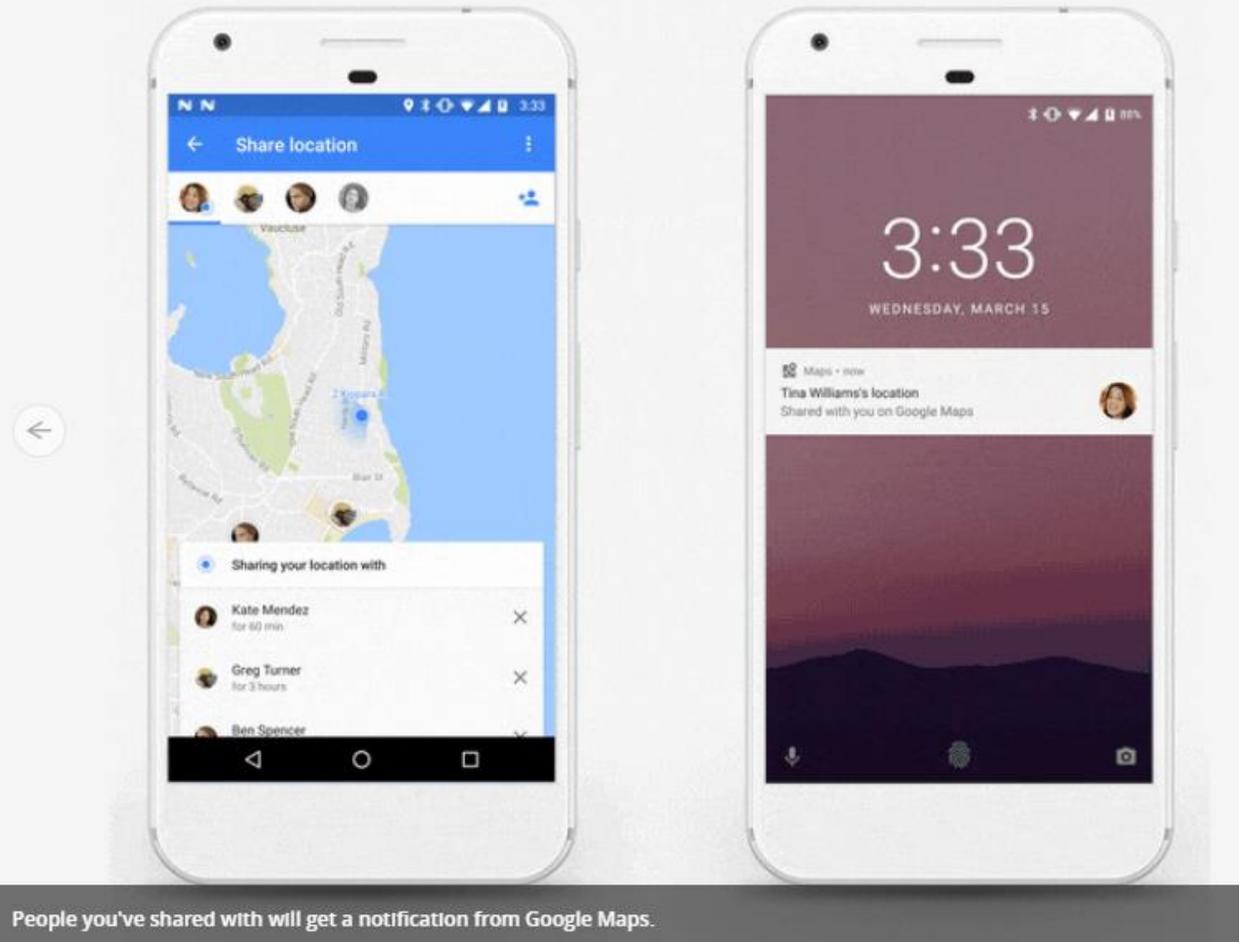
US9467838	Exemplary Supporting Evidence Regarding Accused Products
	 <p>Ways to start location sharing: (left) from the app navigation drawer, and (right) from the menu, you can drag up while navigating somewhere.</p> <p><a href="https://arstechnica.com/gadgets/2017/03/location-sharing-finally-returns-to-google-maps/">https://arstechnica.com/gadgets/2017/03/location-sharing-finally-returns-to-google-maps/</a></p>



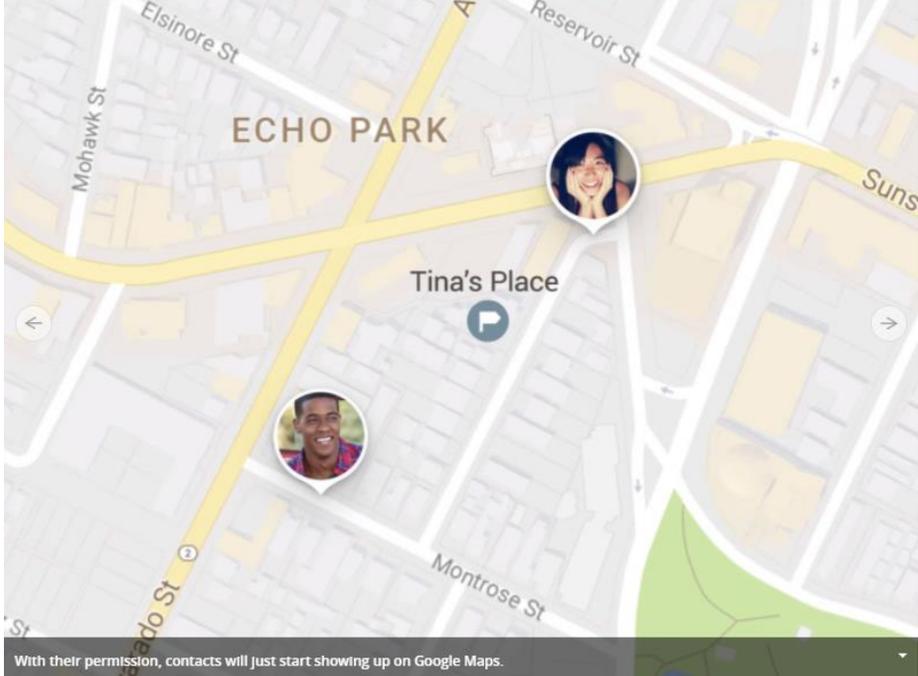
**Exhibit D - Claim Chart for U.S. Patent No. 9,467,838 Against HTC**

US9467838	Exemplary Supporting Evidence Regarding Accused Products
	 <p data-bbox="535 1023 1680 1063">You can pick how long you want to share your location for, and with whom you want to share it.</p> <p data-bbox="535 1063 1680 1096"><a href="https://arstechnica.com/gadgets/2017/03/location-sharing-finally-returns-to-google-maps/">https://arstechnica.com/gadgets/2017/03/location-sharing-finally-returns-to-google-maps/</a></p>

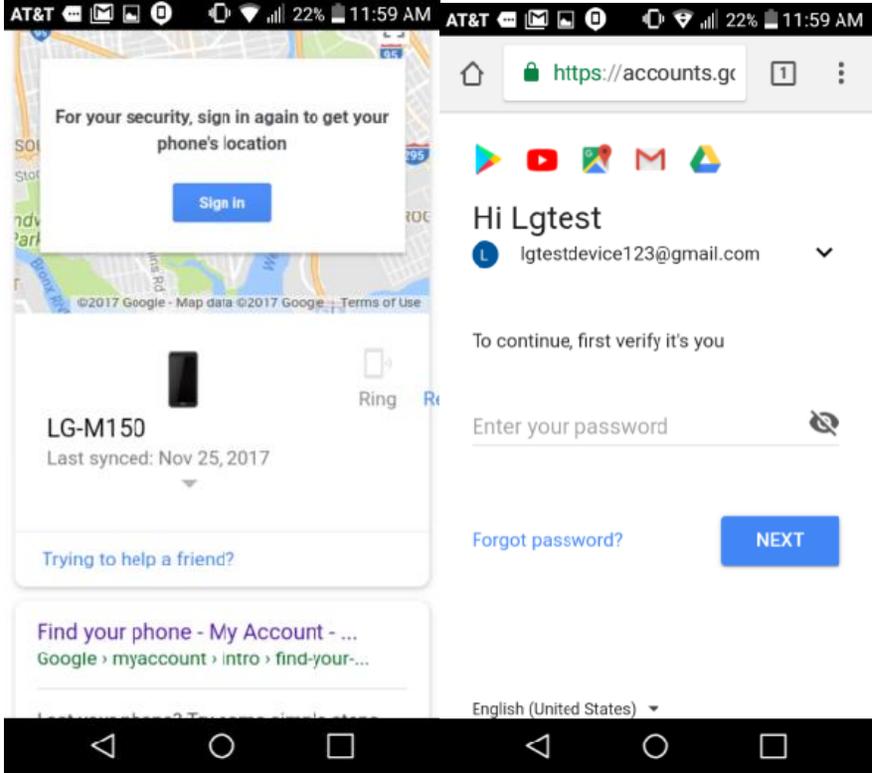
**Exhibit D - Claim Chart for U.S. Patent No. 9,467,838 Against HTC**

US9467838	Exemplary Supporting Evidence Regarding Accused Products
	 <p data-bbox="541 1144 1197 1177">People you've shared with will get a notification from Google Maps.</p> <p data-bbox="541 1188 1680 1221"><a href="https://arstechnica.com/gadgets/2017/03/location-sharing-finally-returns-to-google-maps/">https://arstechnica.com/gadgets/2017/03/location-sharing-finally-returns-to-google-maps/</a></p>

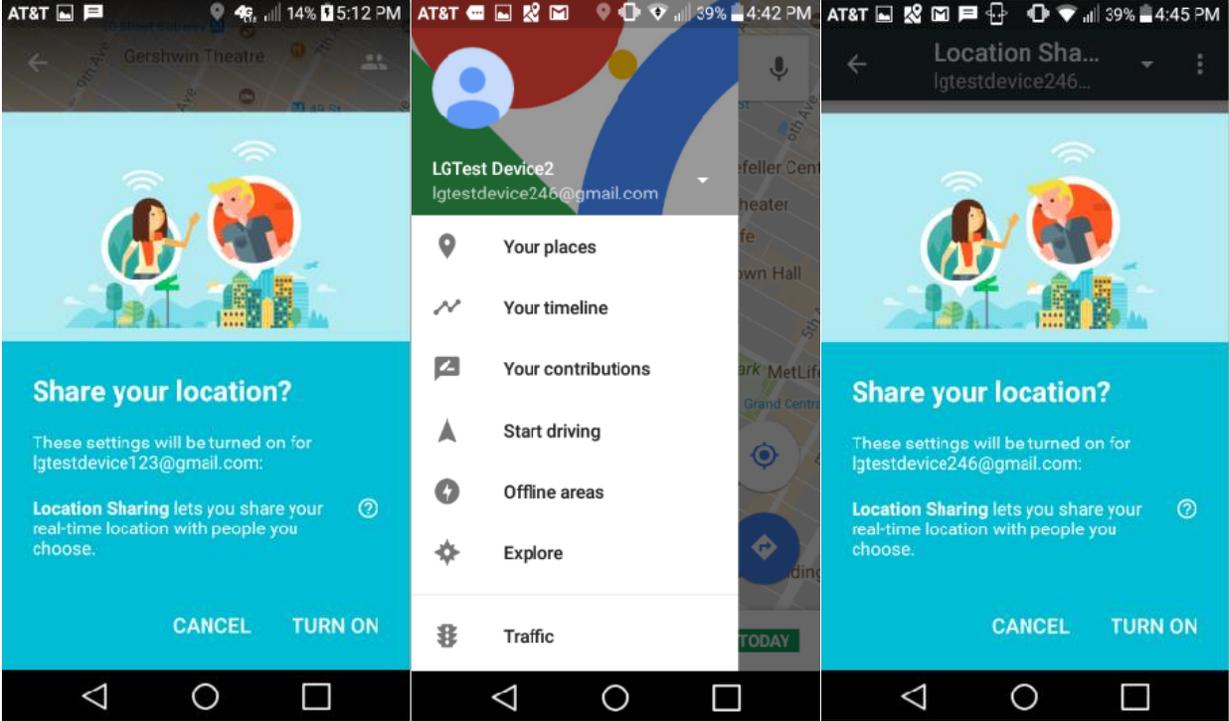
**Exhibit D - Claim Chart for U.S. Patent No. 9,467,838 Against HTC**

US9467838	Exemplary Supporting Evidence Regarding Accused Products
	 <p data-bbox="535 876 1453 909">With their permission, contacts will just start showing up on Google Maps.</p> <p data-bbox="535 917 1680 950"><a href="https://arstechnica.com/gadgets/2017/03/location-sharing-finally-returns-to-google-maps/">https://arstechnica.com/gadgets/2017/03/location-sharing-finally-returns-to-google-maps/</a></p> <p data-bbox="535 1023 1092 1055"><b><u>Exemplary Find My Device Screenshots:</u></b></p>

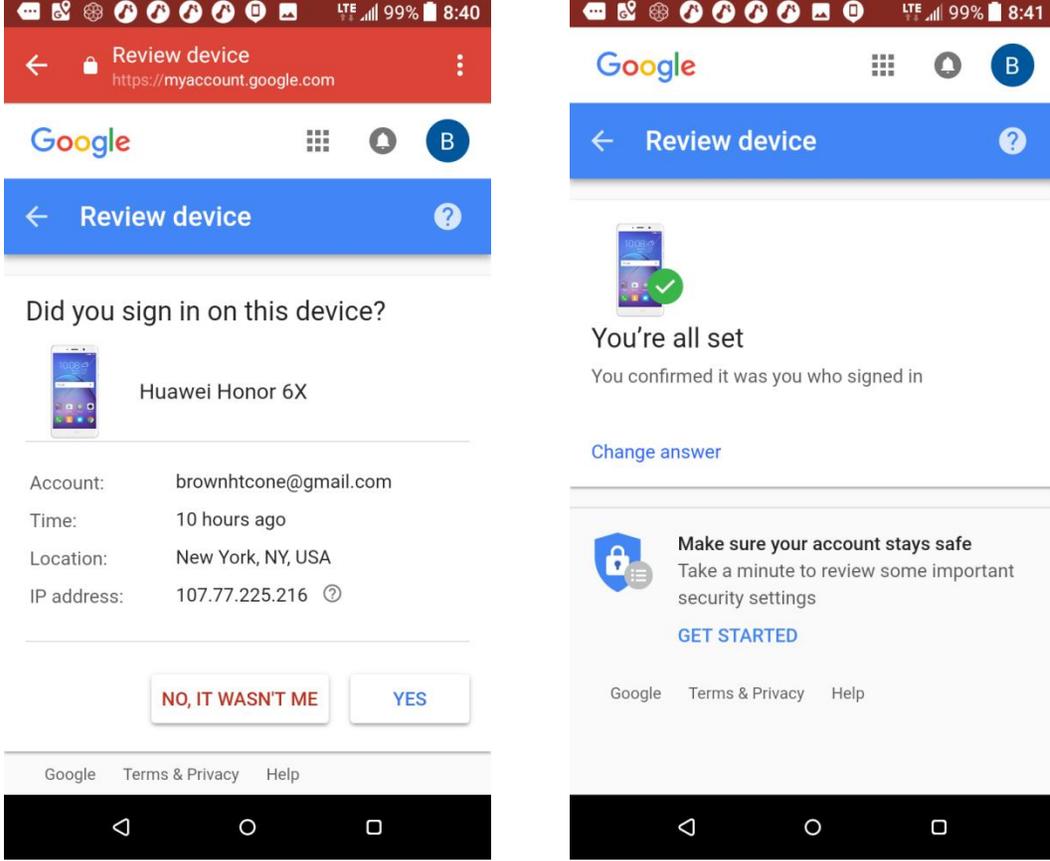
**Exhibit D - Claim Chart for U.S. Patent No. 9,467,838 Against HTC**

US9467838	Exemplary Supporting Evidence Regarding Accused Products
	 <p><b><u>Exemplary Google Maps Screenshots:</u></b></p>

**Exhibit D - Claim Chart for U.S. Patent No. 9,467,838 Against HTC**

US9467838	Exemplary Supporting Evidence Regarding Accused Products
	 <p>The Accused Products require a user to join the corresponding network by: signing-in to the device with an identifier (e.g., Google Account) or linking the device to the identifier by remote means. The message received by the first device relates to the second device joining into a group with the first device.</p>

**Exhibit D - Claim Chart for U.S. Patent No. 9,467,838 Against HTC**

US9467838	Exemplary Supporting Evidence Regarding Accused Products
	 <p>e.g., above, the device receives a message indicating that the second device has joined the group, i.e. the google account.</p> <p><b><u>Exemplary Source Code:</u></b></p>

**Exhibit D - Claim Chart for U.S. Patent No. 9,467,838 Against HTC**

US9467838	Exemplary Supporting Evidence Regarding Accused Products
	<p>The above functionality is performed at least in part by the following publicly available source code and/or source code that invokes or is invoked by the following source code (or a substantially similar copy compiled and loaded onto the Accused Products by HTC): AGIS reserves the right to amend these contentions to include additional source code as discovery progresses and as additional source code is made available.</p> <h2 data-bbox="554 581 1041 634">Contacts Provider</h2> <p>The Contacts Provider is a powerful and flexible Android component that manages the device's central repository of data about people. The Contacts Provider is the source of data you see in the device's contacts application, and you can also access its data in your own application and transfer data between the device and online services. The provider accommodates a wide range of data sources and tries to manage as much data as possible for each person, with the result that its organization is complex. Because of this, the provider's API includes an extensive set of contract classes and interfaces that facilitate both data retrieval and modification.</p> <p>This guide describes the following:</p> <ul data-bbox="554 1019 1398 1192" style="list-style-type: none"><li>• The basic provider structure.</li><li>• How to retrieve data from the provider.</li><li>• How to modify data in the provider.</li><li>• How to write a sync adapter for synchronizing data from your server to the Contacts Provider.</li></ul> <p data-bbox="533 1239 1514 1268"><a href="https://developer.android.com/guide/topics/providers/contacts-provider.html">https://developer.android.com/guide/topics/providers/contacts-provider.html</a></p>

**Exhibit D - Claim Chart for U.S. Patent No. 9,467,838 Against HTC**

US9467838	Exemplary Supporting Evidence Regarding Accused Products
	<p><b>Overview</b></p> <p>ContactsContract defines an extensible database of contact-related information. Contact information is stored in a three-tier data model:</p> <ul style="list-style-type: none"> <li>• A row in the <code>ContactsContract.Data</code> table can store any kind of personal data, such as a phone number or email addresses. The set of data kinds that can be stored in this table is open-ended. There is a predefined set of common kinds, but any application can add its own data kinds.</li> <li>• A row in the <code>ContactsContract.RawContacts</code> table represents a set of data describing a person and associated with a single account (for example, one of the user's Gmail accounts).</li> <li>• A row in the <code>ContactsContract.Contacts</code> table represents an aggregate of one or more RawContacts presumably describing the same person. When data in or associated with the RawContacts table is changed, the affected aggregate contacts are updated as necessary.</li> </ul> <p>Other tables include:</p> <ul style="list-style-type: none"> <li>• <code>ContactsContract.Groups</code>, which contains information about raw contact groups such as Gmail contact groups. The current API does not support the notion of groups spanning multiple accounts.</li> <li>• <code>ContactsContract.StatusUpdates</code>, which contains social status updates including IM availability.</li> <li>• <code>ContactsContract.AggregationExceptions</code>, which is used for manual aggregation and disaggregation of raw contacts</li> <li>• <code>ContactsContract.Settings</code>, which contains visibility and sync settings for accounts and groups.</li> <li>• <code>ContactsContract.SyncState</code>, which contains free-form data maintained on behalf of sync adapters</li> <li>• <code>ContactsContract.PhoneLookup</code>, which is used for quick caller-ID lookup</li> </ul> <p><a href="https://developer.android.com/reference/android/provider/ContactsContract.html">https://developer.android.com/reference/android/provider/ContactsContract.html</a></p> <p><b>Data</b></p> <p>As noted previously, the data for a raw contact is stored in a <code>ContactsContract.Data</code> row that is linked to the raw contact's <code>_ID</code> value. This allows a single raw contact to have multiple instances of the same type of data such as email addresses or phone numbers. For example, if "Thomas Higginson" for <code>emilyd@gmail.com</code> (the raw contact row for Thomas Higginson associated with the Google account <code>emilyd@gmail.com</code>) has a home email address of <code>thigg@gmail.com</code> and a work email address of <code>thomas.higginson@gmail.com</code>, the Contacts Provider stores the two email address rows and links them both to the raw contact.</p> <p>Notice that different types of data are stored in this single table. Display name, phone number, email, postal address, photo, and website detail rows are all found in the <code>ContactsContract.Data</code> table. To help manage this, the <code>ContactsContract.Data</code> table has some columns with descriptive names, and others with generic names. The contents of a descriptive-name column have the same meaning regardless of the type of data in the row, while the contents of a generic-name column have different meanings depending on the type of data.</p>



**Exhibit D - Claim Chart for U.S. Patent No. 9,467,838 Against HTC**

US9467838	Exemplary Supporting Evidence Regarding Accused Products														
	<p data-bbox="535 233 1514 264"><a href="https://developer.android.com/guide/topics/providers/contacts-provider.html">https://developer.android.com/guide/topics/providers/contacts-provider.html</a></p> <table border="1" data-bbox="541 370 1768 1042"> <thead> <tr> <th data-bbox="550 375 642 412">Task</th> <th data-bbox="648 375 879 412">Action</th> <th data-bbox="886 375 1213 412">Data</th> <th data-bbox="1220 375 1499 412">MIME type</th> <th data-bbox="1505 375 1759 412">Notes</th> </tr> </thead> <tbody> <tr> <td data-bbox="550 417 642 545">Pick a contact from a list</td> <td data-bbox="648 417 879 545">ACTION_PICK</td> <td data-bbox="886 417 1213 854">                     One of:                     <ul style="list-style-type: none"> <li data-bbox="898 456 1178 513">• <code>Contacts.CONTENT_URI</code>, which displays a list of contacts.</li> <li data-bbox="898 537 1192 626">• <code>Phone.CONTENT_URI</code>, which displays a list of phone numbers for a raw contact.</li> <li data-bbox="898 651 1199 740">• <code>StructuredPostal.CONTENT_URI</code>, which displays a list of postal addresses for a raw contact.</li> <li data-bbox="898 764 1146 854">• <code>Email.CONTENT_URI</code>, which displays a list of email addresses for a raw contact.</li> </ul> </td> <td data-bbox="1220 417 1499 545">Not used</td> <td data-bbox="1505 417 1759 1032">                     Displays a list of raw contacts or a list of data from a raw contact, depending on the content URI type you supply. Call <code>startActivityForResult()</code>, which returns the content URI of the selected row. The form of the URI is the table's content URI with the row's <code>LOOKUP_ID</code> appended to it. The device's contacts app delegates read and write permissions to this content URI for the life of your activity. See the <a href="#">Content Provider Basics</a> guide for more details.                 </td> </tr> </tbody> </table> <p data-bbox="535 1081 1514 1112"><a href="https://developer.android.com/guide/topics/providers/contacts-provider.html">https://developer.android.com/guide/topics/providers/contacts-provider.html</a></p>					Task	Action	Data	MIME type	Notes	Pick a contact from a list	ACTION_PICK	One of: <ul style="list-style-type: none"> <li data-bbox="898 456 1178 513">• <code>Contacts.CONTENT_URI</code>, which displays a list of contacts.</li> <li data-bbox="898 537 1192 626">• <code>Phone.CONTENT_URI</code>, which displays a list of phone numbers for a raw contact.</li> <li data-bbox="898 651 1199 740">• <code>StructuredPostal.CONTENT_URI</code>, which displays a list of postal addresses for a raw contact.</li> <li data-bbox="898 764 1146 854">• <code>Email.CONTENT_URI</code>, which displays a list of email addresses for a raw contact.</li> </ul>	Not used	Displays a list of raw contacts or a list of data from a raw contact, depending on the content URI type you supply. Call <code>startActivityForResult()</code> , which returns the content URI of the selected row. The form of the URI is the table's content URI with the row's <code>LOOKUP_ID</code> appended to it. The device's contacts app delegates read and write permissions to this content URI for the life of your activity. See the <a href="#">Content Provider Basics</a> guide for more details.
Task	Action	Data	MIME type	Notes											
Pick a contact from a list	ACTION_PICK	One of: <ul style="list-style-type: none"> <li data-bbox="898 456 1178 513">• <code>Contacts.CONTENT_URI</code>, which displays a list of contacts.</li> <li data-bbox="898 537 1192 626">• <code>Phone.CONTENT_URI</code>, which displays a list of phone numbers for a raw contact.</li> <li data-bbox="898 651 1199 740">• <code>StructuredPostal.CONTENT_URI</code>, which displays a list of postal addresses for a raw contact.</li> <li data-bbox="898 764 1146 854">• <code>Email.CONTENT_URI</code>, which displays a list of email addresses for a raw contact.</li> </ul>	Not used	Displays a list of raw contacts or a list of data from a raw contact, depending on the content URI type you supply. Call <code>startActivityForResult()</code> , which returns the content URI of the selected row. The form of the URI is the table's content URI with the row's <code>LOOKUP_ID</code> appended to it. The device's contacts app delegates read and write permissions to this content URI for the life of your activity. See the <a href="#">Content Provider Basics</a> guide for more details.											

## Exhibit D - Claim Chart for U.S. Patent No. 9,467,838 Against HTC

US9467838	Exemplary Supporting Evidence Regarding Accused Products
	<pre>25  * Parsed form of the intent sent to the Contacts application. 26  */ 27  public class ContactsRequest { 28 29      /** Default mode: browse contacts */ 30      public static final int ACTION_DEFAULT = 10; 31 32      /** Show all contacts */ 33      public static final int ACTION_ALL_CONTACTS = 15; 34 35      /** Show all contacts with phone numbers */ 36      public static final int ACTION_CONTACTS_WITH_PHONES = 17; 37 38      /** Show contents of a specific group */ 39      public static final int ACTION_GROUP = 20; 40 41      /** Show all starred contacts */ 42      public static final int ACTION_STARRED = 30; 43 44      /** Show frequently contacted contacts */ 45      public static final int ACTION_FREQUENT = 40; 46 47      /** Show starred and the frequent */ 48      public static final int ACTION_STREQUENT = 50; 49 50      /** Show all contacts and pick them when clicking */ 51      public static final int ACTION_PICK_CONTACT = 60; 52 53      /** Show all contacts as well as the option to create a new one */ 54      public static final int ACTION_PICK_OR_CREATE_CONTACT = 70; 55 56      /** Show all contacts and pick them for edit when clicking, and allow creating a new contact */ 57      public static final int ACTION_INSERT_OR_EDIT_CONTACT = 80;</pre> <p><a href="https://android.googlesource.com/platform/packages/apps/Contacts/+nougat-mr1-release/src/com/android/contacts/list/ContactsRequest.java">https://android.googlesource.com/platform/packages/apps/Contacts/+nougat-mr1-release/src/com/android/contacts/list/ContactsRequest.java</a></p>

**Exhibit D - Claim Chart for U.S. Patent No. 9,467,838 Against HTC**

US9467838	Exemplary Supporting Evidence Regarding Accused Products
	<pre>59      /** Show all phone numbers and pick them when clicking */ 60      public static final int ACTION_PICK_PHONE = 90; 61 62      /** Show all postal addresses and pick them when clicking */ 63      public static final int ACTION_PICK_POSTAL = 100; 64 65      /** Show all postal addresses and pick them when clicking */ 66      public static final int ACTION_PICK_EMAIL = 105; 67 68      /** Show all contacts and create a shortcut for the picked contact */ 69      public static final int ACTION_CREATE_SHORTCUT_CONTACT = 110; 70 71      /** Show all phone numbers and create a call shortcut for the picked number */ 72      public static final int ACTION_CREATE_SHORTCUT_CALL = 120; 73 74      /** Show all phone numbers and create an SMS shortcut for the picked number */ 75      public static final int ACTION_CREATE_SHORTCUT_SMS = 130; 76 77      /** Show all contacts and activate the specified one */ 78      public static final int ACTION_VIEW_CONTACT = 140; 79 80      /** Show contacts recommended for joining with a specified target contact */ 81      public static final int ACTION_PICK_JOIN = 150;</pre> <p><a href="https://android.gogglesource.com/platform/packages/apps/Contacts/+/nougat-mr1-release/src/com/android/contacts/list/ContactsRequest.java">https://android.gogglesource.com/platform/packages/apps/Contacts/+/nougat-mr1-release/src/com/android/contacts/list/ContactsRequest.java</a></p>

**Exhibit D - Claim Chart for U.S. Patent No. 9,467,838 Against HTC**

US9467838	Exemplary Supporting Evidence Regarding Accused Products
	<pre> 104  * Displays a list to browse contacts. 105  */ 106  public class PeopleActivity extends ContactsActivity implements 107      View.OnCreateContextMenuListener, 108      View.OnClickListener, 109      ActionBarAdapter.Listener, 110      DialogManager.DialogShowingViewActivity, 111      ContactListFilterController.ContactListFilterListener, 112      ProviderStatusListener, 113      MultiContactDeleteListener, 114      JoinContactsListener { </pre> <p data-bbox="531 691 1591 764"><a href="https://android.googlesource.com/platform/packages/apps/Contacts/+/nougat-mr1-release/src/com/android/contacts/activities/PeopleActivity.java">https://android.googlesource.com/platform/packages/apps/Contacts/+/nougat-mr1-release/src/com/android/contacts/activities/PeopleActivity.java</a></p> <pre> 145      * Showing a list of Contacts. Also used for showing search results in search mode. 146      */ 147      private MultiSelectContactsListFragment mAllFragment; 148      private ContactTileListFragment mFavoritesFragment; </pre> <p data-bbox="531 971 1591 1044"><a href="https://android.googlesource.com/platform/packages/apps/Contacts/+/nougat-mr1-release/src/com/android/contacts/activities/PeopleActivity.java">https://android.googlesource.com/platform/packages/apps/Contacts/+/nougat-mr1-release/src/com/android/contacts/activities/PeopleActivity.java</a></p>

## Exhibit D - Claim Chart for U.S. Patent No. 9,467,838 Against HTC

US9467838	Exemplary Supporting Evidence Regarding Accused Products
	<pre>458     private void configureFragments(boolean fromRequest) { 459         if (fromRequest) { 460             ContactListFilter filter = null; 461             int actionCode = mRequest.getActionCode(); 462             boolean searchMode = mRequest.isSearchMode(); 463             final int tabToOpen; 464             switch (actionCode) { 465                 case ContactsRequest.ACTION_ALL_CONTACTS: 466                     filter = ContactListFilter.createFilterWithType( 467                         ContactListFilter.FILTER_TYPE_ALL_ACCOUNTS); 468                     tabToOpen = TabState.ALL; 469                     break; 470                 case ContactsRequest.ACTION_CONTACTS_WITH_PHONES: 471                     filter = ContactListFilter.createFilterWithType( 472                         ContactListFilter.FILTER_TYPE_WITH_PHONE_NUMBERS_ONLY); 473                     tabToOpen = TabState.ALL; 474                     break; 475 476                 case ContactsRequest.ACTION_FREQUENT: 477                 case ContactsRequest.ACTION_STREQUENT: 478                 case ContactsRequest.ACTION_STARRED: 479                     tabToOpen = TabState.FAVORITES; 480                     break; 481                 case ContactsRequest.ACTION_VIEW_CONTACT: 482                     tabToOpen = TabState.ALL; 483                     break; 484                 default: 485                     tabToOpen = -1; 486                     break; 487             } </pre> <p data-bbox="535 1352 1591 1385"><a href="https://android.googlesource.com/platform/packages/apps/Contacts/+/nougat-mr1-">https://android.googlesource.com/platform/packages/apps/Contacts/+/nougat-mr1-</a></p>

## Exhibit D - Claim Chart for U.S. Patent No. 9,467,838 Against HTC

US9467838	Exemplary Supporting Evidence Regarding Accused Products
	<pre>release/src/com/android/contacts/activities/PeopleActivity.java  488         if (tabToOpen != -1) { 489             mActionBarAdapter.setCurrentTab(tabToOpen); 490         } 491 492         if (filter != null) { 493             mContactListFilterController.setContactListFilter(filter, false); 494             searchMode = false; 495         } 496 497         if (mRequest.getContactUri() != null) { 498             searchMode = false; 499         } 500 501         mActionBarAdapter.setSearchMode(searchMode); 502         configureContactListFragmentForRequest(); 503     } 504 505     configureContactListFragment(); 506 507     invalidateOptionsMenuIfNeeded(); 508 }</pre> <p><a href="https://android.googlesource.com/platform/packages/apps/Contacts/+/nougat-mr1-release/src/com/android/contacts/activities/PeopleActivity.java">https://android.googlesource.com/platform/packages/apps/Contacts/+/nougat-mr1-release/src/com/android/contacts/activities/PeopleActivity.java</a></p>

**Exhibit D - Claim Chart for U.S. Patent No. 9,467,838 Against HTC**

US9467838	Exemplary Supporting Evidence Regarding Accused Products
	<pre data-bbox="548 237 1480 792">35 public class ProfileAndContactsLoader extends CursorLoader { 36 37     private boolean mLoadProfile; 38 39     private String[] mProjection; 40 41     private Uri mExtraUri; 42     private String[] mExtraProjection; 43     private String mExtraSelection; 44     private String[] mExtraSelectionArgs; 45     private boolean mMergeExtraContactsAfterPrimary; 46 47     public ProfileAndContactsLoader(Context context) { 48         super(context); 49     }</pre> <p data-bbox="537 841 1709 911"><a href="https://android.googlesource.com/platform/packages/apps/ContactsCommon/+nougat-mr1-release/src/com/android/contacts/common/list/ProfileAndContactsLoader.java">https://android.googlesource.com/platform/packages/apps/ContactsCommon/+nougat-mr1-release/src/com/android/contacts/common/list/ProfileAndContactsLoader.java</a></p>

**Exhibit D - Claim Chart for U.S. Patent No. 9,467,838 Against HTC**

US9467838	Exemplary Supporting Evidence Regarding Accused Products
	<pre>35 public final class GroupMemberLoader extends CursorLoader { 36 37     public static class GroupEditorQuery { 38         private static final String[] PROJECTION = new String[] { 39             Data.CONTACT_ID,           // 0 40             Data.RAW_CONTACT_ID,      // 1 41             Data.DISPLAY_NAME_PRIMARY, // 2 42             Data.PHOTO_URI,           // 3 43             Data.LOOKUP_KEY,          // 4 44         }; 45 46         public static final int CONTACT_ID           = 0; 47         public static final int RAW_CONTACT_ID      = 1; 48         public static final int CONTACT_DISPLAY_NAME_PRIMARY = 2; 49         public static final int CONTACT_PHOTO_URI   = 3; 50         public static final int CONTACT_LOOKUP_KEY  = 4; 51     } 52 53     public static class GroupDetailQuery { 54         private static final String[] PROJECTION = new String[] { 55             Data.CONTACT_ID,           // 0 56             Data.PHOTO_URI,           // 1 57             Data.LOOKUP_KEY,          // 2 58             Data.DISPLAY_NAME_PRIMARY, // 3 59             Data.CONTACT_PRESENCE,     // 4 60             Data.CONTACT_STATUS,      // 5 61         }; 62 63         public static final int CONTACT_ID           = 0; 64         public static final int CONTACT_PHOTO_URI   = 1; 65         public static final int CONTACT_LOOKUP_KEY  = 2; 66         public static final int CONTACT_DISPLAY_NAME_PRIMARY = 3; 67         public static final int CONTACT_PRESENCE_STATUS = 4; 68         public static final int CONTACT_STATUS      = 5; 69     } 70 71     private final long mGroupId;</pre> <p><a href="https://android.googlesource.com/platform/packages/apps/Contacts/+/nougat-mr1-release/src/com/android/contacts/GroupMemberLoader.java">https://android.googlesource.com/platform/packages/apps/Contacts/+/nougat-mr1-release/src/com/android/contacts/GroupMemberLoader.java</a></p>



## Exhibit D - Claim Chart for U.S. Patent No. 9,467,838 Against HTC

US9467838	Exemplary Supporting Evidence Regarding Accused Products
	<pre>24 * Group loader for the group list that includes details such as the number of contacts per group 25 * and number of groups per account. This list is sorted by account type, account name, where the 26 * group names are in alphabetical order. Note that the list excludes default, favorite, and deleted 27 * groups. 28 */ 29 public final class GroupListLoader extends CursorLoader { 30 31     private final static String[] COLUMNS = new String[] { 32         Groups.ACCOUNT_NAME, 33         Groups.ACCOUNT_TYPE, 34         Groups.DATA_SET, 35         Groups._ID, 36         Groups.TITLE, 37         Groups.SUMMARY_COUNT, 38     }; 39 40     public final static int ACCOUNT_NAME = 0; 41     public final static int ACCOUNT_TYPE = 1; 42     public final static int DATA_SET = 2; 43     public final static int GROUP_ID = 3; 44     public final static int TITLE = 4; 45     public final static int MEMBER_COUNT = 5; 46 47     private static final Uri GROUP_LIST_URI = Groups.CONTENT_SUMMARY_URI; 48 49     public GroupListLoader(Context context) { 50         super(context, GROUP_LIST_URI, COLUMNS, Groups.ACCOUNT_TYPE + " NOT NULL AND " 51             + Groups.ACCOUNT_NAME + " NOT NULL AND " + Groups.AUTO_ADD + "=0 AND " + 52             Groups.FAVORITES + "=0 AND " + Groups.DELETED + "=0", null, 53             Groups.ACCOUNT_TYPE + ", " + Groups.ACCOUNT_NAME + ", " + Groups.DATA_SET + ", " + 54             Groups.TITLE + " COLLATE LOCALIZED ASC"); 55     } 56 }</pre> <p><a href="https://android.googlesource.com/platform/packages/apps/Contacts/+/nougat-mr1-release/src/com/android/contacts/GroupListLoader.java">https://android.googlesource.com/platform/packages/apps/Contacts/+/nougat-mr1-release/src/com/android/contacts/GroupListLoader.java</a></p>

**Exhibit D - Claim Chart for U.S. Patent No. 9,467,838 Against HTC**

US9467838	Exemplary Supporting Evidence Regarding Accused Products
	<pre> 24  * Group meta-data loader. Loads all groups or just a single group from the 25  * database (if given a {@link Uri}). 26  */ 27  public final class GroupMetaLoader extends CursorLoader { 28 29      private final static String[] COLUMNS = new String[] { 30          Groups.ACCOUNT_NAME, 31          Groups.ACCOUNT_TYPE, 32          Groups.DATA_SET, 33          Groups._ID, 34          Groups.TITLE, 35          Groups.AUTO_ADD, 36          Groups.FAVORITES, 37          Groups.GROUP_IS_READ_ONLY, 38          Groups.DELETED, 39      }; 40 41      public final static int ACCOUNT_NAME = 0; 42      public final static int ACCOUNT_TYPE = 1; 43      public final static int DATA_SET = 2; 44      public final static int GROUP_ID = 3; 45      public final static int TITLE = 4; 46      public final static int AUTO_ADD = 5; 47      public final static int FAVORITES = 6; 48      public final static int IS_READ_ONLY = 7; 49      public final static int DELETED = 8; 50 51      public GroupMetaLoader(Context context, Uri groupUri) { 52          super(context, ensureIsGroupUri(groupUri), COLUMNS, Groups.ACCOUNT_TYPE + " NOT NULL AND " 53              + Groups.ACCOUNT_NAME + " NOT NULL", null, null); 54      } 55 56      /** 57       * Ensures that this is a valid group URI. If invalid, then an exception is 58       * thrown. Otherwise, the original URI is returned. 59       */ 60      private static Uri ensureIsGroupUri(final Uri groupUri) { 61          // TODO: Fix ContactsProvider2 getType method to resolve the group Uri 62          if (groupUri == null) { 63              throw new IllegalArgumentException("Uri must not be null"); 64          } 65          if (!groupUri.toString().startsWith(Groups.CONTENT_URI.toString())) { 66              throw new IllegalArgumentException("Invalid group Uri: " + groupUri); 67          } 68          return groupUri; 69      } 70  } </pre> <p><a href="https://android.goesource.com/platform/packages/apps/Contacts/+nougat-mr1-release/src/com/android/contacts/GroupMetaLoader.java">https://android.goesource.com/platform/packages/apps/Contacts/+nougat-mr1-release/src/com/android/contacts/GroupMetaLoader.java</a></p>

## Exhibit D - Claim Chart for U.S. Patent No. 9,467,838 Against HTC

US9467838	Exemplary Supporting Evidence Regarding Accused Products
	<pre>19  * Meta-data for a contact group. We load all groups associated with the contact's 20  * constituent accounts. 21  */ 22  public final class GroupMetaData { 23      private String mAccountName; 24      private String mAccountType; 25      private String mDataSet; 26      private long mGroupId; 27      private String mTitle; 28      private boolean mDefaultGroup; 29      private boolean mFavorites; 30 31      public GroupMetaData(String accountName, String accountType, String dataSet, long groupId, 32                          String title, boolean defaultGroup, boolean favorites) { 33          this.mAccountName = accountName; 34          this.mAccountType = accountType; 35          this.mDataSet = dataSet; 36          this.mGroupId = groupId; 37          this.mTitle = title; 38          this.mDefaultGroup = defaultGroup; 39          this.mFavorites = favorites; 40      } 41 42      public String getAccountName() { 43          return mAccountName; 44      } 45 46      public String getAccountType() { 47          return mAccountType; 48      } 49 50      public String getDataSet() { 51          return mDataSet; 52      } 53 54      public long getGroupId() { 55          return mGroupId; 56      } 57 58      public String getTitle() { 59          return mTitle; 60      } 61 62      public boolean isDefaultGroup() { 63          return mDefaultGroup; 64      } 65 66      public boolean isFavorites() { 67          return mFavorites;</pre>

## Exhibit D - Claim Chart for U.S. Patent No. 9,467,838 Against HTC

US9467838	Exemplary Supporting Evidence Regarding Accused Products
	<p data-bbox="537 235 1709 305"><a href="https://android.googlesource.com/platform/packages/apps/ContactsCommon/+nougat-mr1-release/src/com/android/contacts/common/GroupMetaData.java">https://android.googlesource.com/platform/packages/apps/ContactsCommon/+nougat-mr1-release/src/com/android/contacts/common/GroupMetaData.java</a></p> <pre data-bbox="558 553 1625 1372">44 * Class that sends chat message via SMS. 45 * 46 * The interface emulates a blocking sending similar to making an HTTP request. 47 * It calls the SmsManager to send a (potentially multipart) message and waits 48 * on the sent status on each part. The waiting has a timeout so it won't wait 49 * forever. Once the sent status of all parts received, the call returns. 50 * A successful sending requires success status for all parts. Otherwise, we 51 * pick the highest level of failure as the error for the whole message, which 52 * is used to determine if we need to retry the sending. 53 */ 54 public class SmsSender { 55     private static final String TAG = LogUtil.BUGLE_TAG; 56 57     public static final String EXTRA_PART_ID = "part_id"; 58 59     /* 60      * A map for pending sms messages. The key is the random request UUID. 61      */ 62     private static ConcurrentHashMap&lt;Uri, SendResult&gt; sPendingMessageMap = 63         new ConcurrentHashMap&lt;Uri, SendResult&gt;(); 64 65     private static final Random RANDOM = new Random(); 66 67     // Whether we should send multipart SMS as separate messages 68     private static Boolean sSendMultipartSmsAsSeparateMessages = null; 69</pre>

## Exhibit D - Claim Chart for U.S. Patent No. 9,467,838 Against HTC

US9467838	Exemplary Supporting Evidence Regarding Accused Products
	<p data-bbox="537 235 1619 305"><a href="https://android.googlesource.com/platform/packages/apps/Messaging+/nougat-mr1-release/src/com/android/messaging/sms/SmsSender.java">https://android.googlesource.com/platform/packages/apps/Messaging+/nougat-mr1-release/src/com/android/messaging/sms/SmsSender.java</a></p> <pre data-bbox="552 349 1770 1349"> 253 // Actually sending the message using SmsManager 254 private static void sendInternal(final Context context, final int subId, String dest, 255     final ArrayList&lt;String&gt; messages, final String serviceCenter, 256     final boolean requireDeliveryReport, final Uri messageUri) throws SmsException { 257     Assert.notNull(context); 258     final SmsManager smsManager = PhoneUtils.get(subId).getSmsManager(); 259     final int messageCount = messages.size(); 260     final ArrayList&lt;PendingIntent&gt; deliveryIntents = new ArrayList&lt;PendingIntent&gt;(messageCount); 261     final ArrayList&lt;PendingIntent&gt; sentIntents = new ArrayList&lt;PendingIntent&gt;(messageCount); 262     for (int i = 0; i &lt; messageCount; i++) { 263         // Make pending intents different for each message part 264         final int partId = (messageCount &lt;= 1 ? 0 : i + 1); 265         if (requireDeliveryReport &amp;&amp; (i == (messageCount - 1))) { 266             // TODO we only care about the delivery status of the last part 267             // Shall we have better tracking of delivery status of all parts? 268             deliveryIntents.add(PendingIntent.getBroadcast( 269                 context, 270                 partId, 271                 getSendStatusIntent(context, SendStatusReceiver.MESSAGE_DELIVERED_ACTION, 272                     messageUri, partId, subId), 273                 0/*flag*/)); 274         } else { 275             deliveryIntents.add(null); 276         } 277         sentIntents.add(PendingIntent.getBroadcast( 278             context, 279             partId, 280             getSendStatusIntent(context, SendStatusReceiver.MESSAGE_SENT_ACTION, 281                 messageUri, partId, subId), 282             0/*flag*/)); 283     } 284     if (sSendMultipartSmsAsSeparateMessages == null) { 285         sSendMultipartSmsAsSeparateMessages = MmsConfig.get(subId) 286             .getSendMultipartSmsAsSeparateMessages(); 287     } </pre> <p data-bbox="537 1390 1619 1419"><a href="https://android.googlesource.com/platform/packages/apps/Messaging+/nougat-mr1-">https://android.googlesource.com/platform/packages/apps/Messaging+/nougat-mr1-</a></p>

## Exhibit D - Claim Chart for U.S. Patent No. 9,467,838 Against HTC

US9467838	Exemplary Supporting Evidence Regarding Accused Products
	<p data-bbox="535 233 1260 264"><a href="https://android.googlesource.com/platform/packages/apps/Messaging/+/nougat-mr1-release/src/com/android/messaging/sms/SmsSender.java">release/src/com/android/messaging/sms/SmsSender.java</a></p> <pre data-bbox="535 300 1711 868">288     try { 289         if (sSendMultipartSmsAsSeparateMessages) { 290             // If multipart sms is not supported, send them as separate messages 291             for (int i = 0; i &lt; messageCount; i++) { 292                 smsManager.sendTextMessage(dest, 293                     serviceCenter, 294                     messages.get(i), 295                     sentIntents.get(i), 296                     deliveryIntents.get(i)); 297             } 298         } else { 299             smsManager.sendMultipartTextMessage( 300                 dest, serviceCenter, messages, sentIntents, deliveryIntents); 301         } 302     } catch (final Exception e) { 303         throw new SmsException("SmsSender: caught exception in sending " + e); 304     } 305 }</pre> <p data-bbox="535 909 1627 987"><a href="https://android.googlesource.com/platform/packages/apps/Messaging/+/nougat-mr1-release/src/com/android/messaging/sms/SmsSender.java">https://android.googlesource.com/platform/packages/apps/Messaging/+/nougat-mr1-release/src/com/android/messaging/sms/SmsSender.java</a></p>

## Exhibit D - Claim Chart for U.S. Patent No. 9,467,838 Against HTC

US9467838	Exemplary Supporting Evidence Regarding Accused Products
	<pre>56 * Class that receives incoming SMS messages through android.provider.Telephony.SMS_RECEIVED 57 * 58 * This class serves two purposes: 59 * - Process phone verification SMS messages 60 * - Handle SMS messages when the user has enabled us to be the default SMS app (Pre-KLP) 61 */ 62 public final class SmsReceiver extends BroadcastReceiver { 63     private static final String TAG = LogUtil.BUGLE_TAG; 64 65     private static ArrayList&lt;Pattern&gt; sIgnoreSmsPatterns; 66</pre> <p><a href="https://android.googlesource.com/platform/packages/apps/Messaging/+nougat-mr1-release/src/com/android/messaging/receiver/SmsReceiver.java">https://android.googlesource.com/platform/packages/apps/Messaging/+nougat-mr1-release/src/com/android/messaging/receiver/SmsReceiver.java</a></p>

## Exhibit D - Claim Chart for U.S. Patent No. 9,467,838 Against HTC

US9467838	Exemplary Supporting Evidence Regarding Accused Products
	<pre> 200     public static void deliverSmsMessages(final Context context, final int subId, 201         final int errorCode, final android.telephony.SmsMessage[] messages) { 202         final ContentValues messageValues = 203             MmsUtils.parseReceivedSmsMessage(context, messages, errorCode); 204 205         LogUtil.v(TAG, "SmsReceiver.deliverSmsMessages"); 206 207         final long nowInMillis = System.currentTimeMillis(); 208         final long receivedTimestampMs = MmsUtils.getMessageDate(messages[0], nowInMillis); 209 210         messageValues.put(Sms.Inbox.DATE, receivedTimestampMs); 211         // Default to unread and unseen for us but ReceiveSmsMessageAction will override 212         // seen for the telephony db. 213         messageValues.put(Sms.Inbox.READ, 0); 214         messageValues.put(Sms.Inbox.SEEN, 0); 215         if (OsUtil.isAtLeastLMR1()) { 216             messageValues.put(Sms.SUBSCRIPTION_ID, subId); 217         } 218 219         if (messages[0].getMessageClass() == android.telephony.SmsMessage.MessageClass.CLASS_0    220             DebugUtils.debugClassZeroSmsEnabled()) { 221             Factory.get().getUIIntents().launchClassZeroActivity(context, messageValues); 222         } else { 223             final ReceiveSmsMessageAction action = new ReceiveSmsMessageAction(messageValues); 224             action.start(); 225         } 226     } </pre> <p><a href="https://android.googlesource.com/platform/packages/apps/Messaging/+/nougat-mr1-release/src/com/android/messaging/receiver/SmsReceiver.java">https://android.googlesource.com/platform/packages/apps/Messaging/+/nougat-mr1-release/src/com/android/messaging/receiver/SmsReceiver.java</a></p>



## Exhibit D - Claim Chart for U.S. Patent No. 9,467,838 Against HTC

US9467838	Exemplary Supporting Evidence Regarding Accused Products
	<pre>228     @Override 229     public void onReceive(final Context context, final Intent intent) { 230         LogUtil.v(TAG, "SmsReceiver.onReceive " + intent); 231         // On KLP+ we only take delivery of SMS messages in SmsDeliverReceiver. 232         if (PhoneUtils.getDefault().isSmsEnabled()) { 233             final String action = intent.getAction(); 234             if (OsUtil.isSecondaryUser() &amp;&amp; 235                 (Telephony.Sms.Intents.SMS_RECEIVED_ACTION.equals(action)    236                  // TODO: update this with the actual constant from Telephony 237                   "android.provider.Telephony.MMS_DOWNLOADED".equals(action))) { 238                 postNewMessageSecondaryUserNotification(); 239             } else if (!OsUtil.isAtLeastKLP()) { 240                 deliverSmsIntent(context, intent); 241             } 242         } 243     }</pre> <p><a href="https://android.googlesource.com/platform/packages/apps/Messaging/+/nougat-mr1-release/src/com/android/messaging/receiver/SmsReceiver.java">https://android.googlesource.com/platform/packages/apps/Messaging/+/nougat-mr1-release/src/com/android/messaging/receiver/SmsReceiver.java</a></p>

## Exhibit D - Claim Chart for U.S. Patent No. 9,467,838 Against HTC

US9467838	Exemplary Supporting Evidence Regarding Accused Products
	<pre>52  * Class that sends chat message via MMS. 53  * 54  * The interface emulates a blocking send similar to making an HTTP request. 55  */ 56  public class MmsSender { 57      private static final String TAG = LogUtil.BUGLE_TAG; 58 59      /** 60       * Send an MMS message. 61       * 62       * @param context Context 63       * @param messageUri The unique URI of the message for identifying it during sending 64       * @param sendReq The SendReq PDU of the message 65       * @throws MmsFailureException 66       */ 67      public static void sendMms(final Context context, final int subId, final Uri messageUri, 68                               final SendReq sendReq, final Bundle sentIntentExtras) throws MmsFailureException { 69          sendMms(context, 70                 subId, 71                 messageUri, 72                 null /* locationUri */, 73                 sendReq, 74                 true /* responseImportant */, 75                 sentIntentExtras); 76      }</pre> <p><a href="https://android.googlesource.com/platform/packages/apps/Messaging/+nougat-mr1-release/src/com/android/messaging/sms/MmsSender.java">https://android.googlesource.com/platform/packages/apps/Messaging/+nougat-mr1-release/src/com/android/messaging/sms/MmsSender.java</a></p>

## Exhibit D - Claim Chart for U.S. Patent No. 9,467,838 Against HTC

US9467838	Exemplary Supporting Evidence Regarding Accused Products
	<pre> 240      * Download an MMS message. 241      * 242      * @param context Context 243      * @param contentLocation The url of the MMS message 244      * @throws MmsFailureException 245      * @throws InvalidHeaderValueException 246      */ 247      public static void downloadMms(final Context context, final int subId, 248          final String contentLocation, Bundle extras) throws MmsFailureException, 249          InvalidHeaderValueException { 250          final Uri requestUri = Uri.parse(contentLocation); 251          final Uri contentUri = MmsFileProvider.buildRawMmsUri(); 252 253          final Intent downloadedIntent = new Intent(SendStatusReceiver.MMS_DOWNLOADED_ACTION, 254              requestUri, 255              context, 256              SendStatusReceiver.class); 257          downloadedIntent.putExtra(SendMessageAction.EXTRA_CONTENT_URI, contentUri); 258          if (extras != null) { 259              downloadedIntent.putExtras(extras); 260          } 261          final PendingIntent downloadedPendingIntent = PendingIntent.getBroadcast( 262              context, 263              0 /*request code*/, 264              downloadedIntent, 265              PendingIntent.FLAG_UPDATE_CURRENT); 266 267          MmsManager.downloadMultimediaMessage(subId, context, contentLocation, contentUri, 268              downloadedPendingIntent); 269      } </pre> <p><a href="https://android.googlesource.com/platform/packages/apps/Messaging/+/nougat-mr1-release/src/com/android/messaging/sms/MmsSender.java">https://android.googlesource.com/platform/packages/apps/Messaging/+/nougat-mr1-release/src/com/android/messaging/sms/MmsSender.java</a></p>

## Exhibit D - Claim Chart for U.S. Patent No. 9,467,838 Against HTC

US9467838	Exemplary Supporting Evidence Regarding Accused Products
	<pre> 97      * Execute an MMS HTTP request, either a POST (sending) or a GET (downloading) 98      * 99      * @param urlString The request URL, for sending it is usually the MMSC, and for downloading 100     *           it is the message URL 101     * @param pdu For POST (sending) only, the PDU to send 102     * @param method HTTP method, POST for sending and GET for downloading 103     * @param isProxySet Is there a proxy for the MMSC 104     * @param proxyHost The proxy host 105     * @param proxyPort The proxy port 106     * @param mmsConfig The MMS config to use 107     * @param userAgent The user agent header value 108     * @param uaProfUrl The UA Prof URL header value 109     * @return The HTTP response body 110     * @throws MmsHttpException For any failures 111     */ 112     public byte[] execute(String urlString, byte[] pdu, String method, boolean isProxySet, 113                          String proxyHost, int proxyPort, Bundle mmsConfig, String userAgent, String uaProfUrl) 114                          throws MmsHttpException { 115         Log.d(MmsService.TAG, "HTTP: " + method + " " + Utils.redactUrlForNonVerbose(urlString) 116             + (isProxySet ? (" proxy=" + proxyHost + ":" + proxyPort) : "") 117             + ", PDU size=" + (pdu != null ? pdu.length : 0)); 118         checkMethod(method); 119         HttpURLConnection connection = null; 120         try { 121             Proxy proxy = Proxy.NO_PROXY; 122             if (isProxySet) { 123                 proxy = new Proxy(Proxy.Type.HTTP, new InetSocketAddress(proxyHost, proxyPort)); 124             } 125             final URL url = new URL(urlString); 126             // Now get the connection 127             connection = (HttpURLConnection) url.openConnection(proxy); 128             connection.setDoInput(true); 129             connection.setConnectTimeout( 130                 mmsConfig.getInt(CarrierConfigValuesLoader.CONFIG_HTTP_SOCKET_TIMEOUT, 131                                 CarrierConfigValuesLoader.CONFIG_HTTP_SOCKET_TIMEOUT_DEFAULT)); </pre> <p data-bbox="531 1252 1619 1328"> <a href="https://android.googlesource.com/platform/packages/apps/Messaging/+/nougat-mr1-release/src/android/support/v7/mms/MmsHttpClient.java">https://android.googlesource.com/platform/packages/apps/Messaging/+/nougat-mr1-release/src/android/support/v7/mms/MmsHttpClient.java</a> </p>

## Exhibit D - Claim Chart for U.S. Patent No. 9,467,838 Against HTC

US9467838	Exemplary Supporting Evidence Regarding Accused Products
	<pre> 132 // ----- COMMON HEADERS ----- 133 // Header: Accept 134 connection.setRequestProperty(HEADER_ACCEPT, HEADER_VALUE_ACCEPT); 135 // Header: Accept-Language 136 connection.setRequestProperty( 137     HEADER_ACCEPT_LANGUAGE, getCurrentAcceptLanguage(Locale.getDefault())); 138 // Header: User-Agent 139 Log.i(MmsService.TAG, "HTTP: User-Agent=" + userAgent); 140 connection.setRequestProperty(HEADER_USER_AGENT, userAgent); 141 // Header: x-wap-profile 142 final String uaProfUrlTagName = mmsConfig.getString( 143     CarrierConfigValuesLoader.CONFIG_UA_PROF_TAG_NAME, 144     CarrierConfigValuesLoader.CONFIG_UA_PROF_TAG_NAME_DEFAULT); 145 if (uaProfUrl != null) { 146     Log.i(MmsService.TAG, "HTTP: UaProfUrl=" + uaProfUrl); 147     connection.setRequestProperty(uaProfUrlTagName, uaProfUrl); 148 } 149 // Add extra headers specified by mms_config.xml's httpparams 150 addExtraHeaders(connection, mmsConfig); 151 // Different stuff for GET and POST 152 if (METHOD_POST.equals(method)) { 153     if (pdu == null    pdu.length &lt; 1) { 154         Log.e(MmsService.TAG, "HTTP: empty pdu"); 155         throw new MmsHttpException(0/*statusCode*/, "Sending empty PDU"); 156     } 157     connection.setDoOutput(true); 158     connection.setRequestMethod(METHOD_POST); 159     if (mmsConfig.getBoolean( 160         CarrierConfigValuesLoader.CONFIG_SUPPORT_HTTP_CHARSET_HEADER, 161         CarrierConfigValuesLoader.CONFIG_SUPPORT_HTTP_CHARSET_HEADER_DEFAULT)) { 162         connection.setRequestProperty(HEADER_CONTENT_TYPE, 163             HEADER_VALUE_CONTENT_TYPE_WITH_CHARSET); 164     } else { 165         connection.setRequestProperty(HEADER_CONTENT_TYPE, 166             HEADER_VALUE_CONTENT_TYPE_WITHOUT_CHARSET); </pre> <p data-bbox="531 1289 1619 1365"> <a href="https://android.globalsources.com/platform/packages/apps/Messaging/+nougat-mr1-release/src/android/support/v7/mms/MmsHttpClient.java">https://android.globalsources.com/platform/packages/apps/Messaging/+nougat-mr1-release/src/android/support/v7/mms/MmsHttpClient.java</a> </p>

## Exhibit D - Claim Chart for U.S. Patent No. 9,467,838 Against HTC

US9467838	Exemplary Supporting Evidence Regarding Accused Products
	<pre> 167         } 168         if (Log.isLoggable(MmsService.TAG, Log.VERBOSE)) { 169             logHttpHeaders(connection.getRequestProperties()); 170         } 171         connection.setFixedLengthStreamingMode(pdu.length); 172         // Sending request body 173         final OutputStream out = 174             new BufferedOutputStream(connection.getOutputStream()); 175         out.write(pdu); 176         out.flush(); 177         out.close(); 178     } else if (METHOD_GET.equals(method)) { 179         if (Log.isLoggable(MmsService.TAG, Log.VERBOSE)) { 180             logHttpHeaders(connection.getRequestProperties()); 181         } 182         connection.setRequestMethod(METHOD_GET); 183     } 184     // Get response 185     final int responseCode = connection.getResponseCode(); 186     final String responseMessage = connection.getResponseMessage(); 187     Log.d(MmsService.TAG, "HTTP: " + responseCode + " " + responseMessage); 188     if (Log.isLoggable(MmsService.TAG, Log.VERBOSE)) { 189         logHttpHeaders(connection.getHeaderFields()); 190     } 191     if (responseCode / 100 != 2) { 192         throw new MmsHttpException(responseCode, responseMessage); 193     } 194     final InputStream in = new BufferedInputStream(connection.getInputStream()); 195     final ByteArrayOutputStream byteOut = new ByteArrayOutputStream(); 196     final byte[] buf = new byte[4096]; 197     int count = 0; 198     while ((count = in.read(buf)) &gt; 0) { 199         byteOut.write(buf, 0, count); 200     } 201     in.close(); 202     final byte[] responseBody = byteOut.toByteArray(); 203     Log.d(MmsService.TAG, "HTTP: response size=" 204         + (responseBody != null ? responseBody.length : 0)); 205     return responseBody; </pre> <p><a href="https://android.gogglesource.com/platform/packages/apps/Messaging/+nougat-mr1-">https://android.gogglesource.com/platform/packages/apps/Messaging/+nougat-mr1-</a></p>

**Exhibit D - Claim Chart for U.S. Patent No. 9,467,838 Against HTC**

US9467838	Exemplary Supporting Evidence Regarding Accused Products
	<a href="#">release/src/android/support/v7/mms/MmsHttpClient.java</a>

## Exhibit D - Claim Chart for U.S. Patent No. 9,467,838 Against HTC

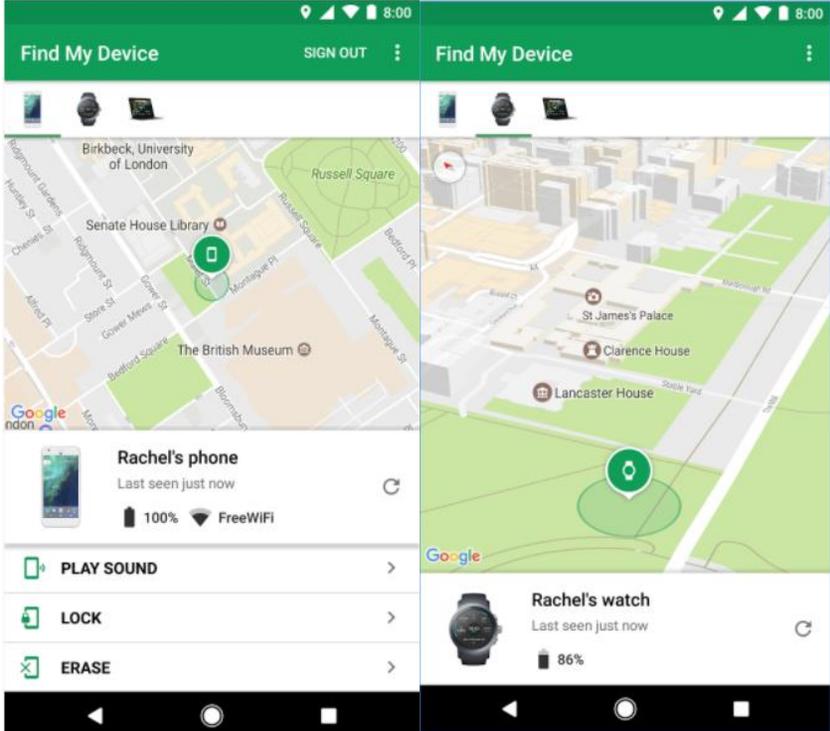
US9467838	Exemplary Supporting Evidence Regarding Accused Products
	<pre> 38  * Request to send an MMS 39  */ 40  class SendRequest extends MmsRequest { 41      // Max send response PDU size in bytes (exceeding this may cause problem with 42      // system intent delivery). 43      private static final int MAX_SEND_RESPONSE_SIZE = 1000 * 1024; 44 45      private byte[] mPduData; 46 47      SendRequest(final String locationUri, final Uri pduUri, final PendingIntent sentIntent) { 48          super(locationUri, pduUri, sentIntent); 49      } 50 51      @Override 52      protected boolean loadRequest(final Context context, final Bundle mmsConfig) { 53          mPduData = readPduFromContentUri( 54              context, 55              mPduUri, 56              mmsConfig.getInt( 57                  CarrierConfigValuesLoader.CONFIG_MAX_MESSAGE_SIZE, 58                  CarrierConfigValuesLoader.CONFIG_MAX_MESSAGE_SIZE_DEFAULT)); 59          return (mPduData != null); 60      } 61 62      @Override 63      protected boolean transferResponse(final Context context, final Intent fillIn, 64          final byte[] response) { 65          // SendConf pdus are always small and can be included in the intent 66          if (response != null &amp;&amp; fillIn != null) { 67              if (response.length &gt; MAX_SEND_RESPONSE_SIZE) { 68                  // If the response PDU is too large, it won't be able to fit in 69                  // the PendingIntent to be transferred via system IPC. 70                  return false; 71              } 72              fillIn.putExtra(SmsManager.EXTRA_MMS_DATA, response); 73          } 74          return true; 75      } </pre>
	<p><a href="https://android.googlesource.com/platform/packages/apps/Messaging/+/nougat-mr1-release/src/android/support/v7/mms/SendRequest.java">https://android.googlesource.com/platform/packages/apps/Messaging/+/nougat-mr1-release/src/android/support/v7/mms/SendRequest.java</a></p>



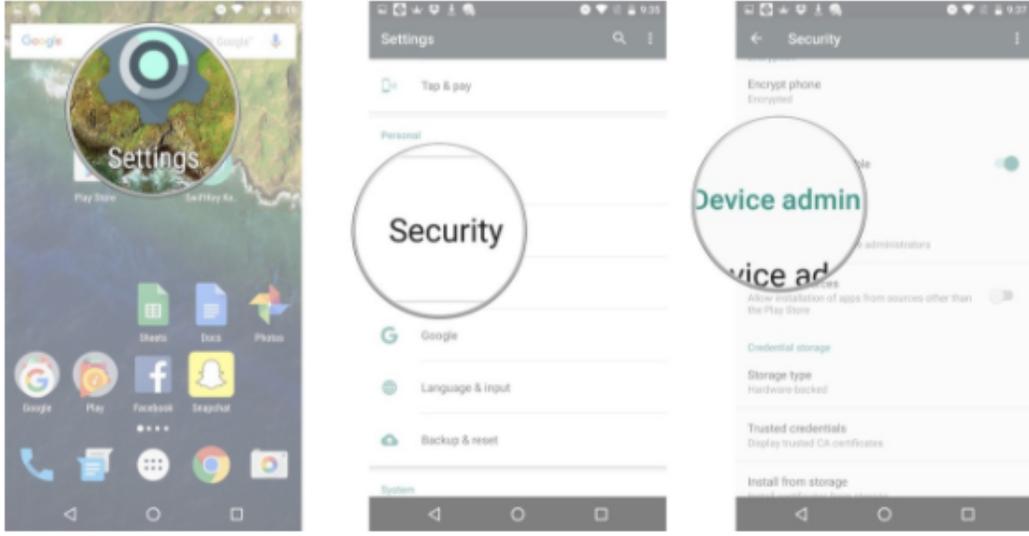
**Exhibit D - Claim Chart for U.S. Patent No. 9,467,838 Against HTC**

<b>US9467838</b>	<b>Exemplary Supporting Evidence Regarding Accused Products</b>
<p>[1C] participating in the group, wherein participating in the group includes sending first location information to a first server and receiving second location information from the first server, the first location information comprising a location of the first device, the second location information comprising one or more locations of one or more respective second devices included in the group;</p>	<p>HTC infringes directly and/or indirectly by performing, inducing others to perform, and/or contributing to the performance of: participating in the group, wherein participating in the group includes sending first location information to a first server and receiving second location information from the first server, the first location information comprising a location of the first device, the second location information comprising one or more locations of one or more respective second devices included in the group.</p> <p><b><u>Regarding Find My Device</u></b> and Android Device Manager, the Accused Products require a user to join the corresponding network by: signing-in to the device with an identifier (e.g., Google Account) or linking the device to the identifier by remote means. The Accused Products require that a signed-in user of a first device to share its location by enabling Location Access or Location Services. When a user of a first device associated with an identity (e.g. Google Account) enables the Find My Device feature and Location Services on the first device, the user shares its location and the first device's location is sent to a server. When the user makes a request to view a map of device locations associated with the identity, the device receives one or more locations corresponding to one or more second devices associated with the identity. Alternatively, the locations corresponding to one or more second devices associated with the identity are sent to the first device on a rolling basis.</p> <p><b><u>Regarding Google Maps</u></b>, Google Latitude, Google Plus, Google Hangouts, Google Messages, Google Allo, Google Duo, Google Chrome, and Android Messenger, the Accused Products require a user to join the corresponding network by signing-in to the device with an identifier (e.g., Google Account). When the signed-in user enables Google Maps Share Location and Location Services on the first device, the user shares its location and the first device's location is sent to a server. When the user enables sharing to one or more contacts (of respective devices) and the one or more contacts enable sharing their location to the user of the first device, the user of the first device receives the locations of the one or more contacts.</p> <p><b><u>Further regarding Google Maps</u></b>, Google Latitude, Google Plus, Google Hangouts, Google Messages, Google Allo, Google Duo, Google Chrome, and Android Messenger, the Accused Products alternatively require a user to send a request containing the identifier (e.g., Google Account). When the signed-in user enables Google Maps Share Location and Location Services on the first device, the user shares its location and the first device's location is sent to a server. When the user sends a message to another contact through Google Maps, Google Messages, and/or another means from within the Google Maps application, the message including location information are sent to a server before transmission to the intended contact.</p>

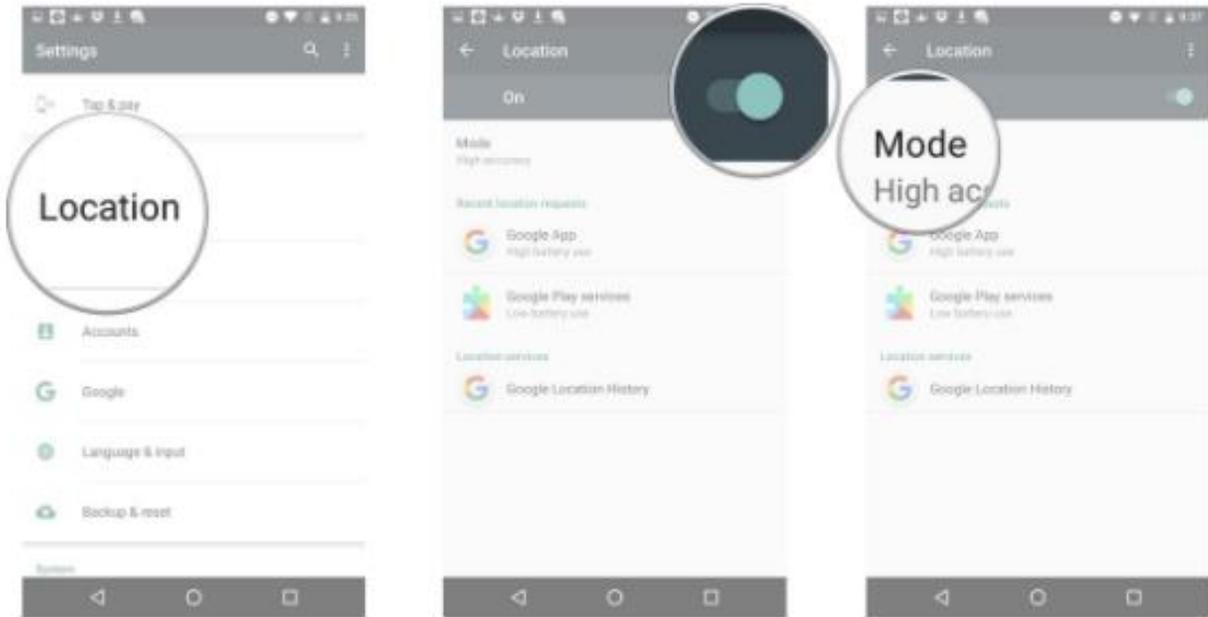
**Exhibit D - Claim Chart for U.S. Patent No. 9,467,838 Against HTC**

US9467838	Exemplary Supporting Evidence Regarding Accused Products
	<p>When one or more contacts enable sharing their location to the user of the first device, or alternatively send a message containing location information, or alternatively accept a request to share their location with the first user, the user of the first device receives the locations of the one or more contacts.</p> <p><b><u>Exemplary Support for Find My Device:</u></b></p>  <p><a href="https://play.google.com/store/apps/details?id=com.google.android.apps.adm&amp;hl=en">https://play.google.com/store/apps/details?id=com.google.android.apps.adm&amp;hl=en</a></p>

**Exhibit D - Claim Chart for U.S. Patent No. 9,467,838 Against HTC**

US9467838	Exemplary Supporting Evidence Regarding Accused Products
	<p><b>How to enable Find My Device on your phone</b></p> <p>In newer Android phones, the Find My Device service is already located conveniently in your Settings app, but if you can't find it you can always <a href="#">download Find My Device from the Google Play Store</a>. This locating service has essentially amalgamated with Google to make finding your phone easier. There are just a couple of things you'll need to activate.</p> <ol style="list-style-type: none"> <li>1. Launch Settings.</li> <li>2. Tap Security.</li> <li>3. Tap Device Administration.</li> </ol>  <ol style="list-style-type: none"> <li>4. Tap Find My Device so that a checkmark appears in the checkbox.</li> </ol> <p><a href="https://www.androidcentral.com/how-track-android-phone">https://www.androidcentral.com/how-track-android-phone</a></p>

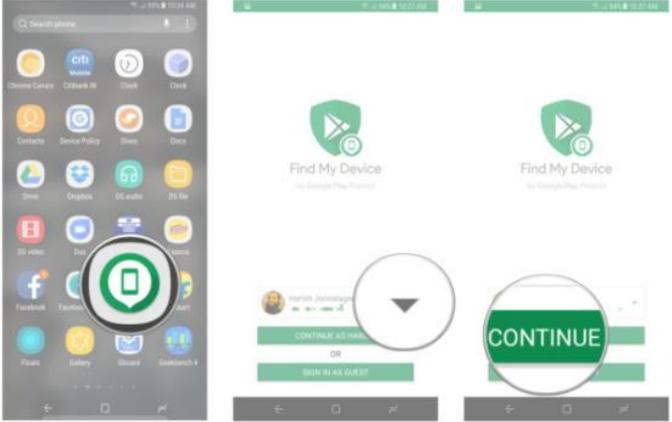
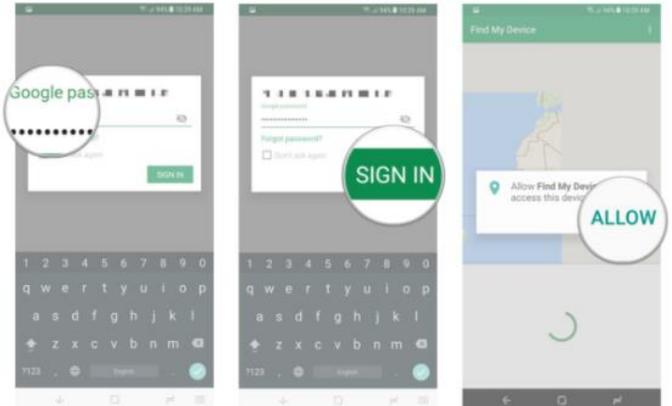
**Exhibit D - Claim Chart for U.S. Patent No. 9,467,838 Against HTC**

US9467838	Exemplary Supporting Evidence Regarding Accused Products
	<p data-bbox="556 245 1081 277">7. Tap <b>Location</b> in the main Settings menu.</p> <p data-bbox="556 305 1480 337">8. Tap the <b>switch</b> beside <b>Location</b> at the top of the screen so that it turns on.</p> <p data-bbox="556 365 724 397">9. Tap <b>Mode</b>.</p>  <p>The image contains three sequential screenshots of an Android phone's settings interface. The first screenshot shows the main 'Settings' menu with 'Location' circled in white. The second screenshot shows the 'Location' settings page with the 'On' toggle switch circled in white. The third screenshot shows the 'Location' settings page with the 'Mode' dropdown menu circled in white, and the 'High accuracy' option selected.</p> <p data-bbox="535 1096 1291 1128"><a href="https://www.androidcentral.com/how-track-android-phone">https://www.androidcentral.com/how-track-android-phone</a></p>

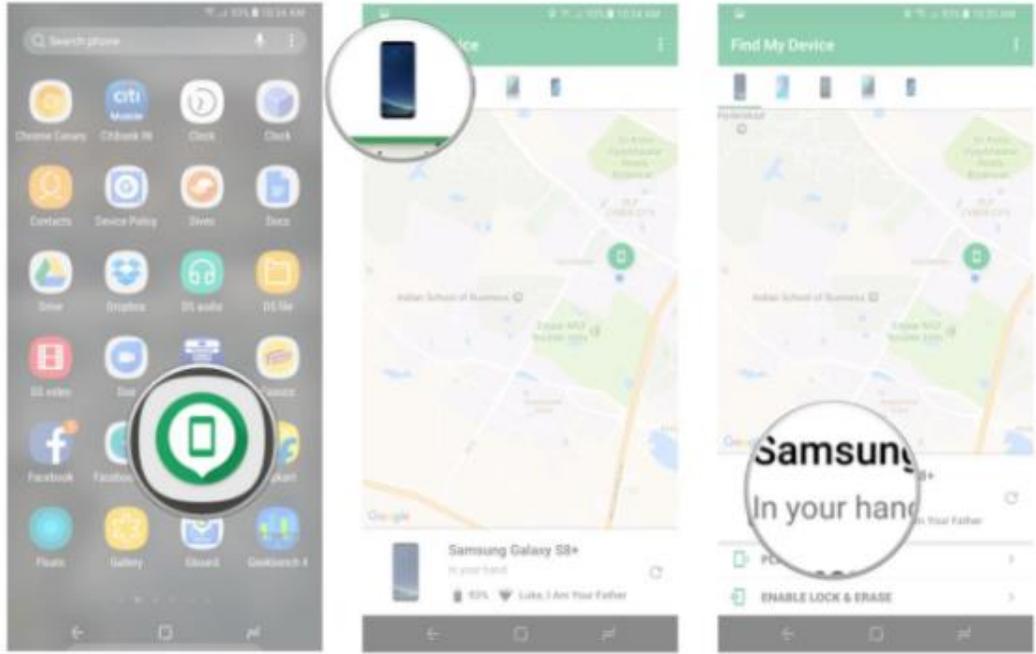
## Exhibit D - Claim Chart for U.S. Patent No. 9,467,838 Against HTC

US9467838	Exemplary Supporting Evidence Regarding Accused Products
	<p data-bbox="558 240 1205 277"><b>How to locate your phone with Google</b></p> <p data-bbox="558 310 1520 363">Should you happen to lose your phone, you can locate its whereabouts by logging into your Google account from any computer or even from another phone.</p> <ol data-bbox="558 407 1276 532" style="list-style-type: none"> <li data-bbox="558 407 1136 431">1. Launch a web browser from a phone, tablet, or computer.</li> <li data-bbox="558 459 1276 483">2. Navigate to Google if it is not your default search engine or home page.</li> <li data-bbox="558 511 1108 535">3. Type find my phone android in the Google search bar.</li> </ol> <div data-bbox="585 565 1549 1133" style="text-align: center;"> </div> <ol data-bbox="558 1182 1545 1317" style="list-style-type: none"> <li data-bbox="558 1182 1171 1206">4. Tap on Find My Device (usually the first option in the search).</li> <li data-bbox="558 1230 1545 1317">5. Enter your email address and password just as though you were checking your email. If you have 2-step verification set up on your Google account (and you most certainly should), you'll need to complete that process as well.</li> </ol> <p data-bbox="537 1325 1283 1357"><a href="https://www.androidcentral.com/how-track-android-phone">https://www.androidcentral.com/how-track-android-phone</a></p>

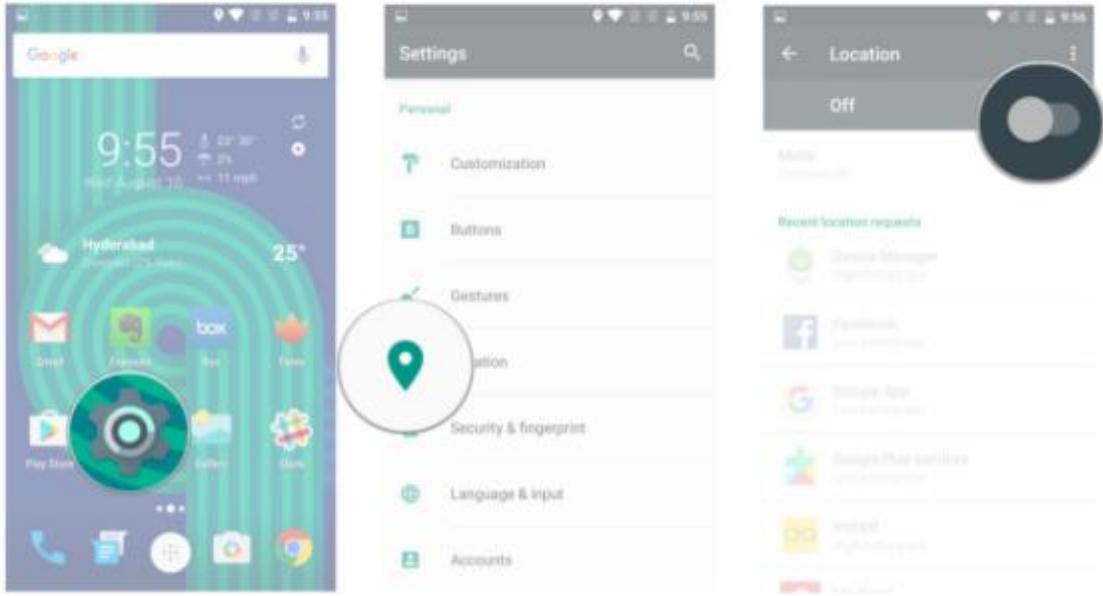
**Exhibit D - Claim Chart for U.S. Patent No. 9,467,838 Against HTC**

US9467838	Exemplary Supporting Evidence Regarding Accused Products
	<ol style="list-style-type: none"> <li>1. Open Find Device from your home screen or app drawer.</li> <li>2. Select the Google account you want to use the service with.</li> <li>3. Hit the Continue as button.</li> </ol>  <ol style="list-style-type: none"> <li>4. Enter your Google account password.</li> <li>5. Tap Sign in.</li> <li>6. Give location access to the service.</li> </ol>  <p><a href="https://www.androidcentral.com/find-my-device">https://www.androidcentral.com/find-my-device</a></p>

**Exhibit D - Claim Chart for U.S. Patent No. 9,467,838 Against HTC**

US9467838	Exemplary Supporting Evidence Regarding Accused Products
	<p>Once you're signed in to Find My Device, you'll see a map with your current location as well as the make and model of your phone, and two options — Play Sound, and Enable Lock &amp; Erase. Hitting the latter option will allow you to start using the Lock and Erase functions.</p> <p>If you've signed into more than one phone, you can select a particular device by browsing the list at the top of the screen.</p> <ol style="list-style-type: none"> <li>1. Open <b>Find My Device</b> from your home screen or app drawer.</li> <li>2. Select your phone from the list of devices at the top of the screen.</li> <li>3. See if your phone is discoverable.</li> </ol>  <p><a href="https://www.androidcentral.com/find-my-device">https://www.androidcentral.com/find-my-device</a></p>

**Exhibit D - Claim Chart for U.S. Patent No. 9,467,838 Against HTC**

US9467838	Exemplary Supporting Evidence Regarding Accused Products
	<p>If you're not able to find your phone or if it says that the device is unavailable, it is likely that the location services are disabled. Find My Device relies on GPS to track your phone, so now would be a good time to enable location services.</p> <ol style="list-style-type: none"> <li>1. Open <b>Settings</b> from your home screen or app drawer.</li> <li>2. Tap <b>Location</b>.</li> <li>3. Toggle <b>Enable location services</b>.</li> </ol>  <p><a href="https://www.androidcentral.com/find-my-device">https://www.androidcentral.com/find-my-device</a></p> <p><b><u>Exemplary Support for Google Maps:</u></b></p>



**Exhibit D - Claim Chart for U.S. Patent No. 9,467,838 Against HTC**

US9467838	Exemplary Supporting Evidence Regarding Accused Products
	<p data-bbox="573 240 989 261">COMPUTER    <b>ANDROID</b>    IPHONE &amp; IPAD</p> <hr data-bbox="560 289 1593 293"/> <h3 data-bbox="560 345 1045 383">If they have a Google Account</h3> <ol data-bbox="569 404 1440 695" style="list-style-type: none"> <li data-bbox="569 404 1241 425">1. If you haven't already, add their Gmail address to your <a href="#">Google Contacts</a> .</li> <li data-bbox="569 440 1440 461">2. On your Android phone or tablet, open the Google Maps app  and sign in. <a href="#">Learn how to sign in.</a></li> <li data-bbox="569 475 1056 496">3. Tap Menu ≡ &gt; <b>Location sharing</b> &gt; Add People +.</li> <li data-bbox="569 511 1026 532">4. Choose how long you want to share your location.</li> <li data-bbox="569 547 1163 610">5. Tap <b>Select People</b>. <ul data-bbox="590 586 1163 607" style="list-style-type: none"> <li data-bbox="590 586 1163 607">• If you're asked about your contacts, give Google Maps access.</li> </ul> </li> <li data-bbox="569 634 909 656">6. Choose who you want to share with.</li> <li data-bbox="569 670 684 691">7. Tap <b>Share</b>.</li> </ol> <h3 data-bbox="560 756 1131 794">If they don't have a Google Account</h3> <ol data-bbox="569 815 1583 938" style="list-style-type: none"> <li data-bbox="569 815 1440 836">1. On your Android phone or tablet, open the Google Maps app  and sign in. <a href="#">Learn how to sign in.</a></li> <li data-bbox="569 850 1056 872">2. Tap Menu ≡ &gt; <b>Location sharing</b> &gt; Add People +.</li> <li data-bbox="569 886 1583 938">3. Tap More ... &gt; <b>Copy to clipboard</b>. People with this link can see your location for as long as you choose, up to 72 hours.</li> </ol> <h3 data-bbox="560 987 892 1024">Share using another app</h3> <p data-bbox="560 1040 1226 1062">You can also share through messaging apps. Tap More ... &gt; select an app.</p> <h3 data-bbox="560 1122 764 1159">Stop sharing</h3> <ol data-bbox="569 1180 1226 1279" style="list-style-type: none"> <li data-bbox="569 1180 867 1201">1. Open the Google Maps app .</li> <li data-bbox="569 1216 892 1237">2. Tap Menu ≡ &gt; <b>Location sharing</b>.</li> <li data-bbox="569 1252 1226 1273">3. Next to the person with whom you want to stop sharing, tap Remove ✕ .</li> </ol> <p data-bbox="537 1295 1724 1326"><a href="https://support.google.com/maps/answer/7326816?co=GENIE.Platform%3DAndroid&amp;oco=1">https://support.google.com/maps/answer/7326816?co=GENIE.Platform%3DAndroid&amp;oco=1</a></p>

**Exhibit D - Claim Chart for U.S. Patent No. 9,467,838 Against HTC**

US9467838	Exemplary Supporting Evidence Regarding Accused Products
	<p data-bbox="556 253 863 297"><b>Share your E.T.A</b></p> <p data-bbox="556 321 1686 347">After you start your drive, you can share your destination, estimated arrival time, and where you are on the route.</p> <ol data-bbox="556 375 1381 618" style="list-style-type: none"> <li data-bbox="556 375 911 401">1. Open the Google Maps app .</li> <li data-bbox="556 418 1184 444">2. Set a driving destination. <a href="#">Learn how to navigate to a place.</a></li> <li data-bbox="556 462 1224 488">3. After you start navigation, tap More ^ &gt; <b>Share trip progress.</b></li> <li data-bbox="556 506 898 532">4. Choose a person from the list.</li> <li data-bbox="556 550 701 576">5. Tap <b>Share.</b></li> <li data-bbox="556 594 1381 620">6. Location Sharing will stop when you reach your destination or stop navigating.</li> </ol> <ul data-bbox="556 646 1226 672" style="list-style-type: none"> <li data-bbox="556 646 1226 672">• To stop sharing before you arrive, tap More ^ &gt; <b>Stop sharing.</b></li> </ul> <p data-bbox="556 740 978 784"><b>See where someone is</b></p> <p data-bbox="556 808 1289 834">If someone shares their location with you, you can see them on the map.</p> <ol data-bbox="556 862 940 976" style="list-style-type: none"> <li data-bbox="556 862 911 888">1. Open the Google Maps app .</li> <li data-bbox="556 906 940 932">2. Tap Menu ≡ &gt; <b>Location sharing.</b></li> <li data-bbox="556 950 774 976">3. Choose someone.</li> </ol> <ul data-bbox="556 1002 1325 1027" style="list-style-type: none"> <li data-bbox="556 1002 1325 1027">• To see an updated location, tap on a friend's icon &gt; More ⋮ &gt; <b>Refresh.</b></li> </ul> <p data-bbox="556 1084 1056 1128"><b>Stop seeing someone's location</b></p> <ol data-bbox="556 1146 1482 1300" style="list-style-type: none"> <li data-bbox="556 1146 911 1172">1. Open the Google Maps app .</li> <li data-bbox="556 1190 858 1216">2. On the map, tap their icon.</li> <li data-bbox="556 1234 877 1260">3. At the bottom, tap More ^ .</li> <li data-bbox="556 1278 1482 1304">4. To temporarily hide someone, tap <b>Hide from map.</b> You can stop hiding them at any time.</li> </ol> <p data-bbox="556 1330 1766 1356"><b>Note:</b> You can stop someone's location from ever appearing on your map. <a href="#">Learn how to block another person's account.</a></p> <p data-bbox="533 1377 1724 1403"><a href="https://support.google.com/maps/answer/7326816?co=GENIE.Platform%3DAndroid&amp;oco=1">https://support.google.com/maps/answer/7326816?co=GENIE.Platform%3DAndroid&amp;oco=1</a></p>

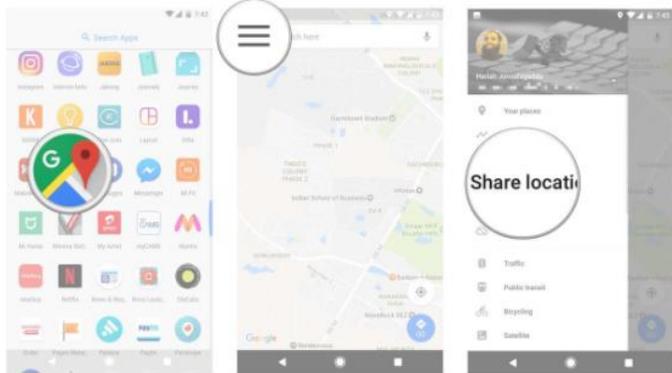
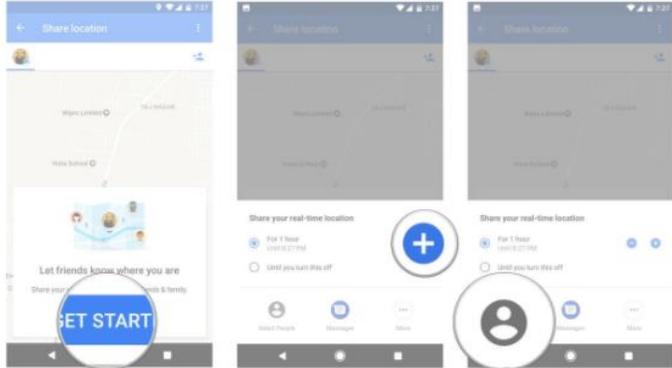
**Exhibit D - Claim Chart for U.S. Patent No. 9,467,838 Against HTC**

US9467838	Exemplary Supporting Evidence Regarding Accused Products
	<p data-bbox="548 315 1020 362"><b>Create a list of places</b></p> <p data-bbox="548 383 1367 402">In Google Maps, you can create a list of places, like your favorite places or places you want to visit.</p> <p data-bbox="562 477 957 496">COMPUTER   <b>ANDROID</b>   IPHONE &amp; IPAD</p> <hr data-bbox="548 521 1394 524"/> <p data-bbox="548 574 789 605"><b>Make a new list</b></p> <ol data-bbox="558 631 1108 789" style="list-style-type: none"><li>1. On your Android phone or tablet, open the Google Maps app .</li><li>2. Tap Menu  &gt; <b>Your places</b> &gt; <b>Saved</b>.</li><li>3. In the bottom right, tap Add .</li><li>4. Enter a name and description.</li><li>5. Tap <b>Save</b>.</li></ol> <p data-bbox="548 846 867 876"><b>Save a place to a list</b></p> <ol data-bbox="558 902 999 1060" style="list-style-type: none"><li>1. Open the Google Maps app .</li><li>2. Search for a place or tap it on the map.</li><li>3. At the bottom, tap the place's name or address.</li><li>4. Tap <b>Save</b>.</li><li>5. Choose a list. To create a new list, tap <b>New list</b> .</li></ol> <p data-bbox="548 1117 758 1148"><b>See your lists</b></p> <ol data-bbox="558 1174 905 1227" style="list-style-type: none"><li>1. Open the Google Maps app .</li><li>2. Tap Menu  &gt; <b>Your places</b> &gt; <b>Saved</b>.</li></ol> <p data-bbox="533 1239 1906 1304"><a href="https://support.google.com/maps/answer/7280933?hl=en&amp;ref_topic=7301134&amp;co=GENIE.Platform%3DAndroid&amp;oco=1">https://support.google.com/maps/answer/7280933?hl=en&amp;ref_topic=7301134&amp;co=GENIE.Platform%3DAndroid&amp;oco=1</a></p>

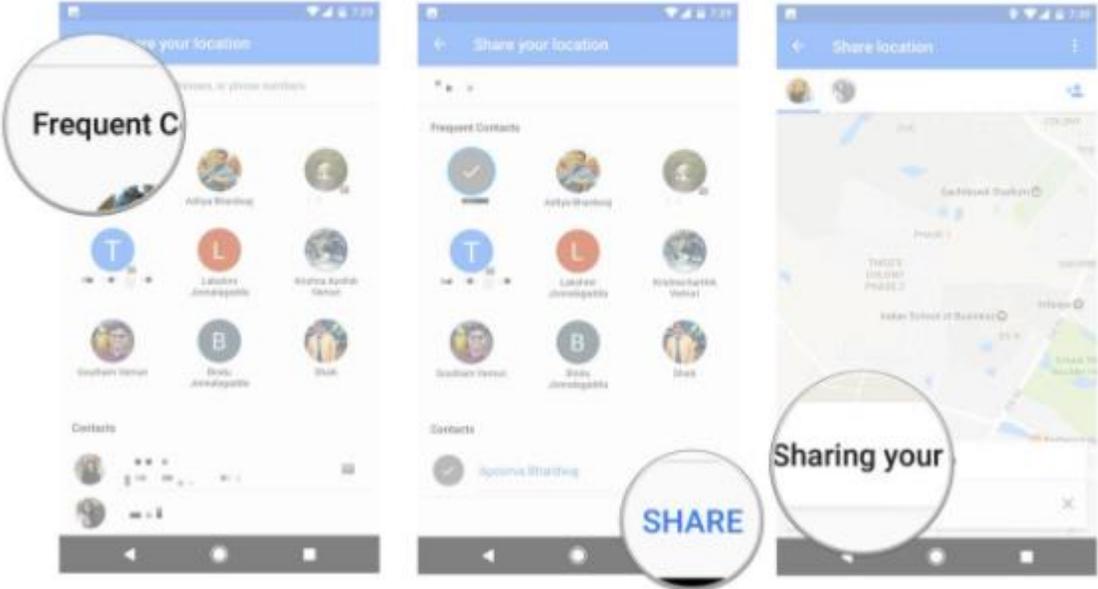
**Exhibit D - Claim Chart for U.S. Patent No. 9,467,838 Against HTC**

US9467838	Exemplary Supporting Evidence Regarding Accused Products
	<p><b>Hide or share lists</b></p> <p><b>Note:</b> You can't share starred places.</p> <ol style="list-style-type: none"> <li>1. Open the Google Maps app .</li> <li>2. Tap Menu  &gt; <b>Your places</b> &gt; <b>Saved</b>.</li> <li>3. Next to the list you want to share, tap More  &gt; choose an option: <ul style="list-style-type: none"> <li>• <b>Hide/Show on your map:</b> Display or hide your saved places when looking at the map.</li> <li>• <b>Share list:</b> Allow others to see your saved list.</li> <li>• <b>Sharing options:</b> You can make your list public, private, or shared. To let anyone with the link see your list, tap <b>Shared</b>. To let anyone find and follow your list, tap <b>Public</b>.</li> </ul> </li> </ol> <p><b>Follow a list</b></p> <p>If you follow a list made by someone else, their saved places will show up in Your Places. The places will also appear as suggested locations in Google Maps.</p> <p><b>Follow a list using a link</b></p> <ol style="list-style-type: none"> <li>1. Tap on the link you received to open it.</li> <li>2. Tap <b>Follow</b>. This list will now be added to the group of lists you follow.</li> <li>3. <b>Optional:</b> To unfollow a list someone shared with you, tap the list &gt; <b>Following</b>.</li> </ol> <p><b>See lists made by others</b></p> <p>If a user has any Google Maps lists that were made public, you can follow them.</p> <ol style="list-style-type: none"> <li>1. Tap on the name of a user whose list you want to follow.</li> <li>2. Tap <b>Lists</b>.</li> <li>3. Tap on the list you want to follow &gt; More  &gt; <b>Follow</b>.</li> </ol> <p><a href="https://support.google.com/maps/answer/7280933?hl=en&amp;ref_topic=7301134&amp;co=GENIE.Platform%3DAndroid&amp;oco=1">https://support.google.com/maps/answer/7280933?hl=en&amp;ref_topic=7301134&amp;co=GENIE.Platform%3DAndroid&amp;oco=1</a></p>

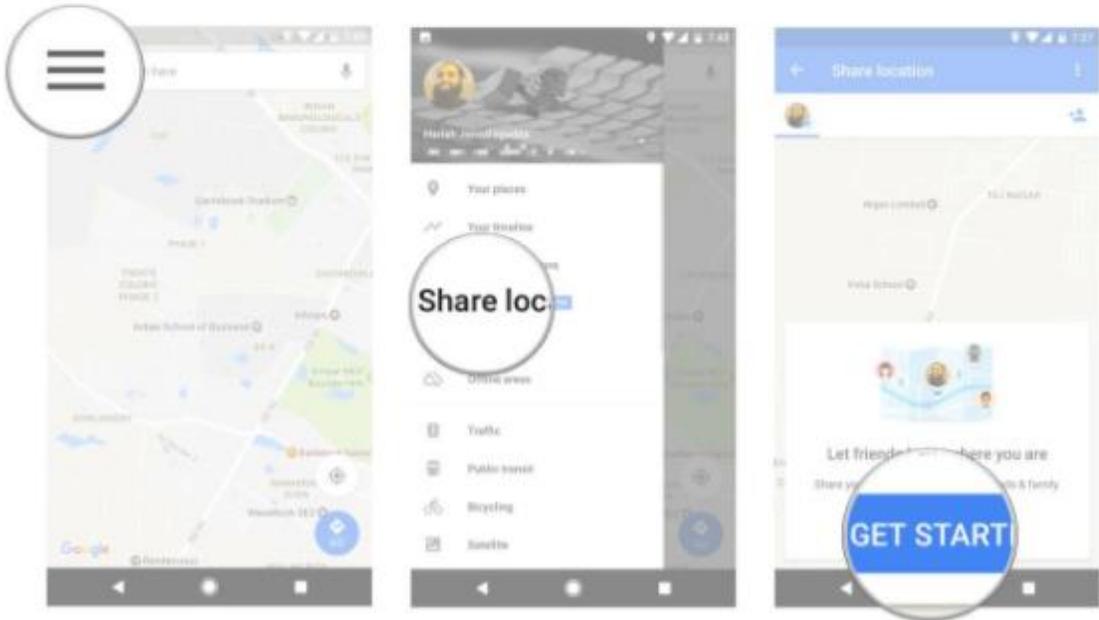
**Exhibit D - Claim Chart for U.S. Patent No. 9,467,838 Against HTC**

US9467838	Exemplary Supporting Evidence Regarding Accused Products
	<p data-bbox="541 272 1178 310"><b>How to share your location in Google Maps</b></p> <ol data-bbox="541 337 1157 423" style="list-style-type: none"> <li>1. Open Google Maps from the app drawer or the home screen.</li> <li>2. Tap the <b>hamburger menu</b> (the three horizontal lines) on the top left corner of the screen.</li> <li>3. Select <b>Share location</b>.</li> </ol>  <ol data-bbox="541 862 1192 971" style="list-style-type: none"> <li>4. Tap <b>Get Started</b>.</li> <li>5. Use the <b>+</b> icon to select a time period or select the <b>Until you turn this off</b> setting to share your location indefinitely.</li> <li>6. Tap <b>Select People</b>.</li> </ol>  <p data-bbox="541 1377 1381 1408"><a href="https://www.androidcentral.com/how-share-location-google-maps">https://www.androidcentral.com/how-share-location-google-maps</a></p>

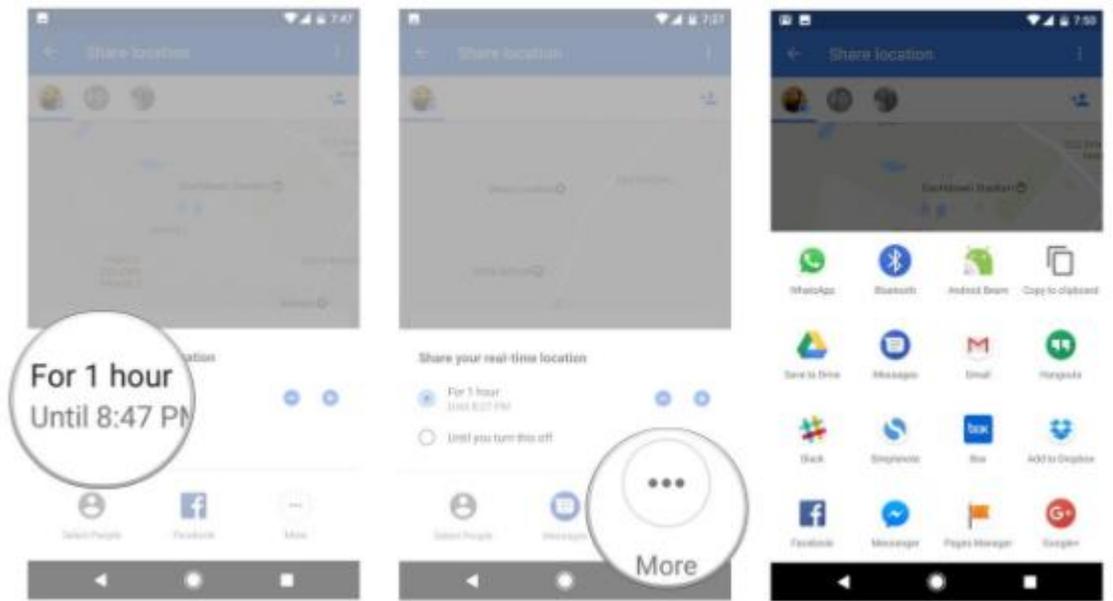
**Exhibit D - Claim Chart for U.S. Patent No. 9,467,838 Against HTC**

US9467838	Exemplary Supporting Evidence Regarding Accused Products
	<p data-bbox="552 289 1602 462">7. You'll see a list of your frequent contacts at the top, along with a full list of contacts. Pick the contacts by tapping their name. 8. Once you've selected the contacts you want to share your location to, tap Share. 9. You'll see a message saying that the selected contact can view your location.</p>  <p data-bbox="533 1138 1381 1170"><a href="https://www.androidcentral.com/how-share-location-google-maps">https://www.androidcentral.com/how-share-location-google-maps</a></p>

**Exhibit D - Claim Chart for U.S. Patent No. 9,467,838 Against HTC**

US9467838	Exemplary Supporting Evidence Regarding Accused Products
	<h2 data-bbox="548 240 1281 289">How to create a shareable link</h2> <p data-bbox="548 329 1486 358">You can also create a link and use it to share your location easily. Here's how to do it:</p> <ol data-bbox="537 407 1260 548" style="list-style-type: none"><li>1. Tap the <b>hamburger menu</b> on the top left corner of the screen.</li><li>2. Select <b>Share location</b>.</li><li>3. Tap <b>Get Started</b>.</li></ol>  <p data-bbox="531 1230 1381 1263"><a href="https://www.androidcentral.com/how-share-location-google-maps">https://www.androidcentral.com/how-share-location-google-maps</a></p>

**Exhibit D - Claim Chart for U.S. Patent No. 9,467,838 Against HTC**

US9467838	Exemplary Supporting Evidence Regarding Accused Products
	<p>4. Select the amount of time you want to share your location.</p> <p>5. Tap More.</p> <p>6. Select your app of choice to create and send a unique URL that broadcasts your current location. You can email it, send the link via Messenger, or even tweet it to the intended recipient.</p>  <p><a href="https://www.androidcentral.com/how-share-location-google-maps">https://www.androidcentral.com/how-share-location-google-maps</a></p>



**Exhibit D - Claim Chart for U.S. Patent No. 9,467,838 Against HTC**

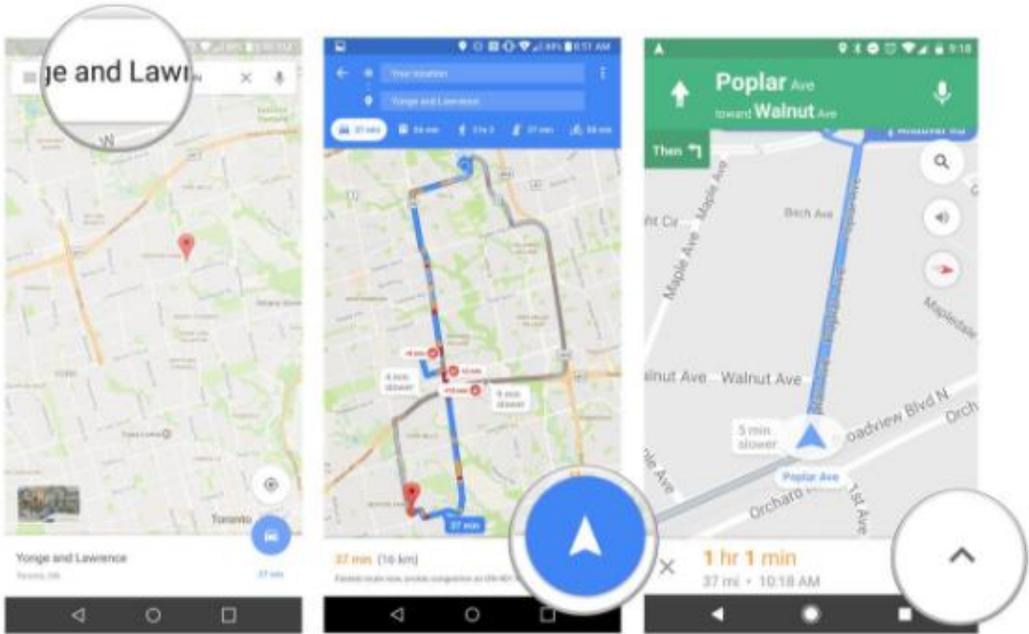
US9467838	Exemplary Supporting Evidence Regarding Accused Products
	<p data-bbox="550 240 1451 337"><b>How to share your navigation directions while you walk, drive or transit</b></p> <p data-bbox="550 375 1577 464">One of the best ways to share your location with someone is during a drive, walk or transit. If you're meeting a friend or family member somewhere, or navigating towards their home, Google Maps lets you share your location with them for the duration of the trip. It's magic!</p> <ol data-bbox="550 513 1419 643" style="list-style-type: none"><li>1. In the <b>search bar</b> enter your destination.</li><li>2. Pick your navigation type (drive, transit, walk) and press the <b>blue navigate button</b>.</li><li>3. Tap the arrow next to the <b>time-to-destination</b> number at the bottom of the screen.</li></ol>  <p data-bbox="535 1328 1381 1360"><a href="https://www.androidcentral.com/how-share-location-google-maps">https://www.androidcentral.com/how-share-location-google-maps</a></p>

Exhibit D - Claim Chart for U.S. Patent No. 9,467,838 Against HTC

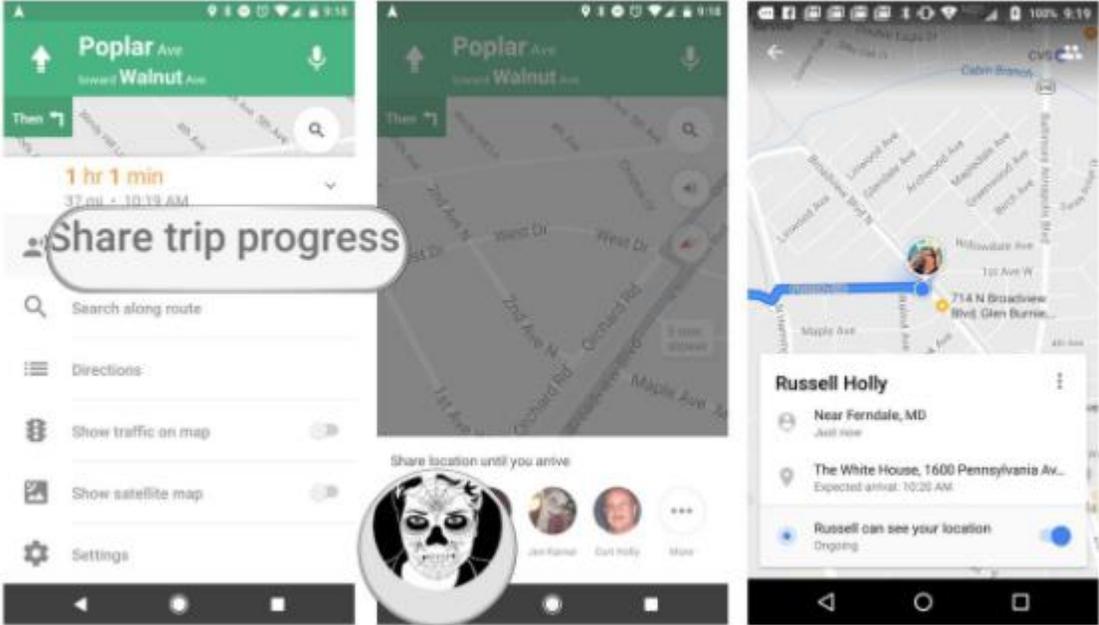
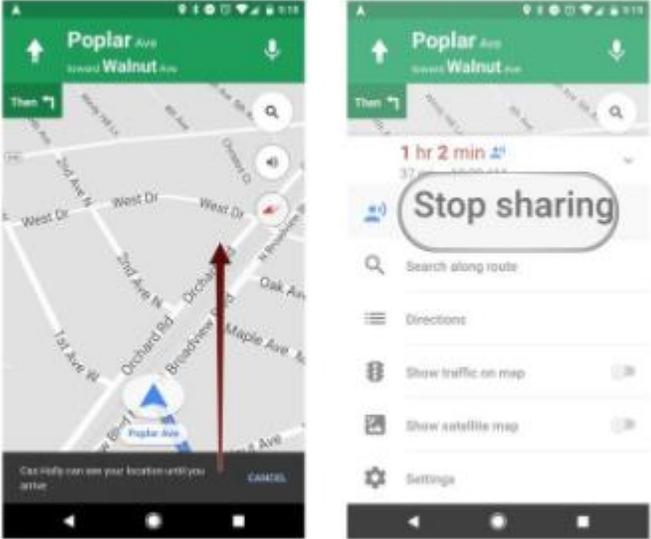
US9467838	Exemplary Supporting Evidence Regarding Accused Products
	<p data-bbox="552 277 861 305">4. Tap Share trip progress.</p> <p data-bbox="552 334 1171 362">5. Choose one or more contacts to share trip progress.</p>  <p data-bbox="562 1065 1381 1092">You can also stop sharing your location with someone before a trip ends.</p> <p data-bbox="537 1102 1381 1130"><a href="https://www.androidcentral.com/how-share-location-google-maps">https://www.androidcentral.com/how-share-location-google-maps</a></p>

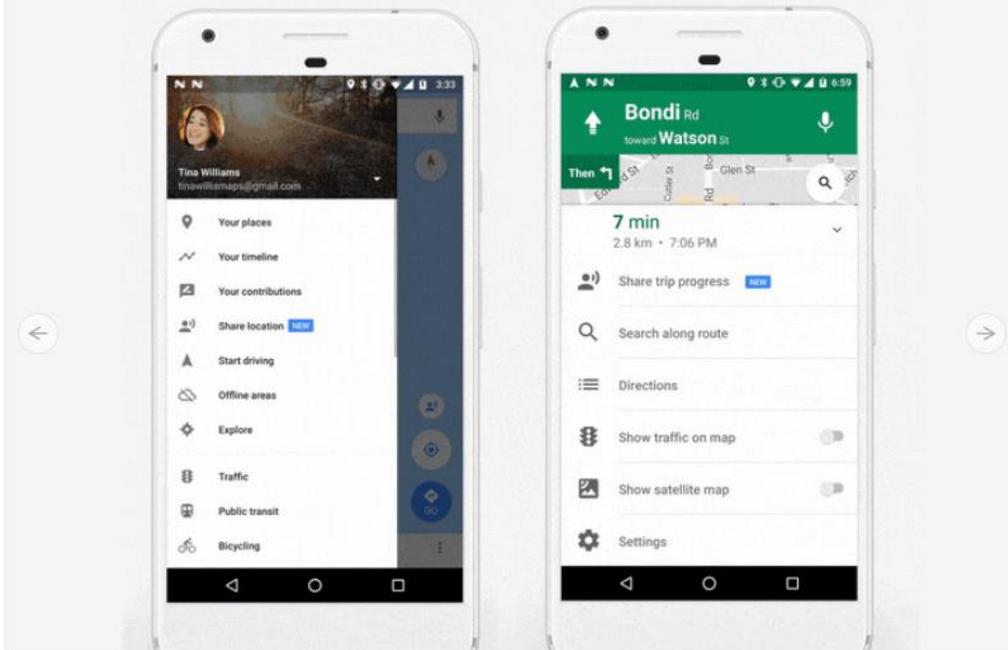
Exhibit D - Claim Chart for U.S. Patent No. 9,467,838 Against HTC

US9467838	Exemplary Supporting Evidence Regarding Accused Products
	<ol style="list-style-type: none"><li data-bbox="562 245 1493 272">1. Tap the arrow next to the time-to-destination number at the bottom of the screen.</li><li data-bbox="562 302 793 329">2. Tap Stop sharing.</li></ol> <div data-bbox="772 383 1423 922"></div> <p data-bbox="569 976 659 1003">That's It!</p> <p data-bbox="537 1045 1633 1073">Are you excited that location sharing is back in Google Maps? How often do you use the feature?</p> <p data-bbox="537 1086 1381 1114"><a href="https://www.androidcentral.com/how-share-location-google-maps">https://www.androidcentral.com/how-share-location-google-maps</a></p> <p data-bbox="537 1159 1444 1187">As shown below, a group may also be defined within Google Contacts.</p>

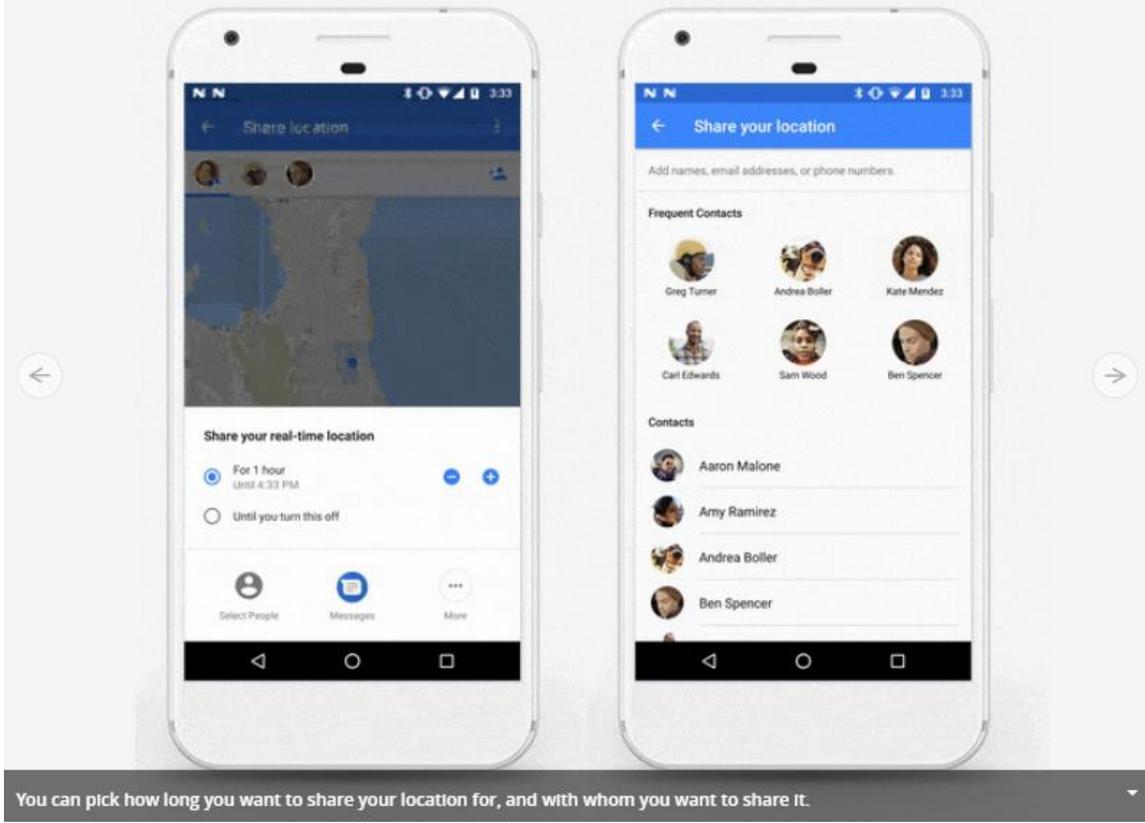
**Exhibit D - Claim Chart for U.S. Patent No. 9,467,838 Against HTC**

US9467838	Exemplary Supporting Evidence Regarding Accused Products
	<p><b>See your contacts</b></p> <ol style="list-style-type: none"> <li>1. Open your device's Contacts app .</li> <li>2. Tap Menu .</li> </ol> <ul style="list-style-type: none"> <li>• <b>See contacts by label:</b> Choose a label from the list.</li> <li>• <b>See contacts for another account:</b> Tap Down arrow  &gt; pick an account.</li> <li>• <b>See the contacts for all your accounts:</b> Choose <b>All contacts</b>.</li> </ul> <p><b>Tip:</b> If you have multiple contacts with the same information, the information will be grouped into one contact.</p> <ul style="list-style-type: none"> <li>• <b>See your Google Account contacts on the web:</b> Go to <a href="#">Google Contacts</a> .</li> </ul> <p><a href="https://support.google.com/android/answer/6118731?hl=en&amp;ref_topic=6118711">https://support.google.com/android/answer/6118731?hl=en&amp;ref_topic=6118711</a></p> <p><b>Label your contacts</b></p> <p>You can group contacts together using labels.</p> <ol style="list-style-type: none"> <li>1. Open your device's Contacts app .</li> <li>2. Tap Menu  &gt; <b>Create label</b>.</li> <li>3. Enter a label name and tap <b>Ok</b>.</li> </ol> <ul style="list-style-type: none"> <li>• <b>Add one contact to a label:</b> Tap Add contact  &gt; choose a contact.</li> <li>• <b>Add multiple contacts to a label:</b> Tap Add contact  &gt; touch and hold a contact &gt; tap the other contacts &gt; tap <b>Add</b>.</li> </ul> <p><a href="https://support.google.com/android/answer/6118731?hl=en&amp;ref_topic=6118711">https://support.google.com/android/answer/6118731?hl=en&amp;ref_topic=6118711</a></p> <p><b>Share your contacts</b></p> <ol style="list-style-type: none"> <li>1. Open your device's Contacts app .</li> <li>2. Tap a contact in the list.</li> <li>3. Tap More  &gt; <b>Share</b>.</li> <li>4. Choose how you want to share the contact.</li> </ol> <p><a href="https://support.google.com/android/answer/6118731?hl=en&amp;ref_topic=6118711">https://support.google.com/android/answer/6118731?hl=en&amp;ref_topic=6118711</a></p>

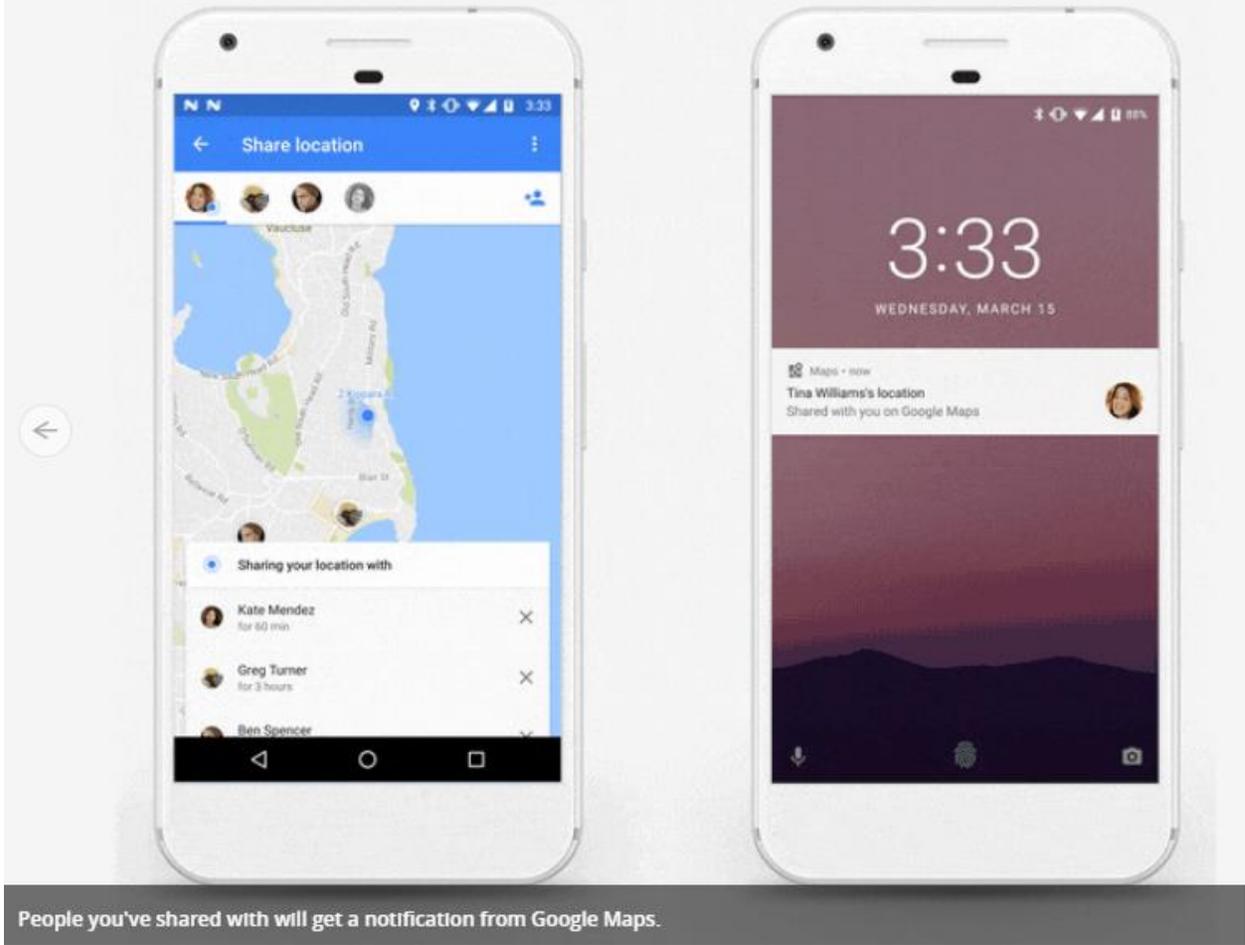
**Exhibit D - Claim Chart for U.S. Patent No. 9,467,838 Against HTC**

US9467838	Exemplary Supporting Evidence Regarding Accused Products
	 <p data-bbox="535 933 1543 990">Ways to start location sharing: (left) from the app navigation drawer, and (right) from the menu, you can drag up while navigating somewhere.</p> <p data-bbox="535 998 1543 1031"><a href="https://arstechnica.com/gadgets/2017/03/location-sharing-finally-returns-to-google-maps/">https://arstechnica.com/gadgets/2017/03/location-sharing-finally-returns-to-google-maps/</a></p>

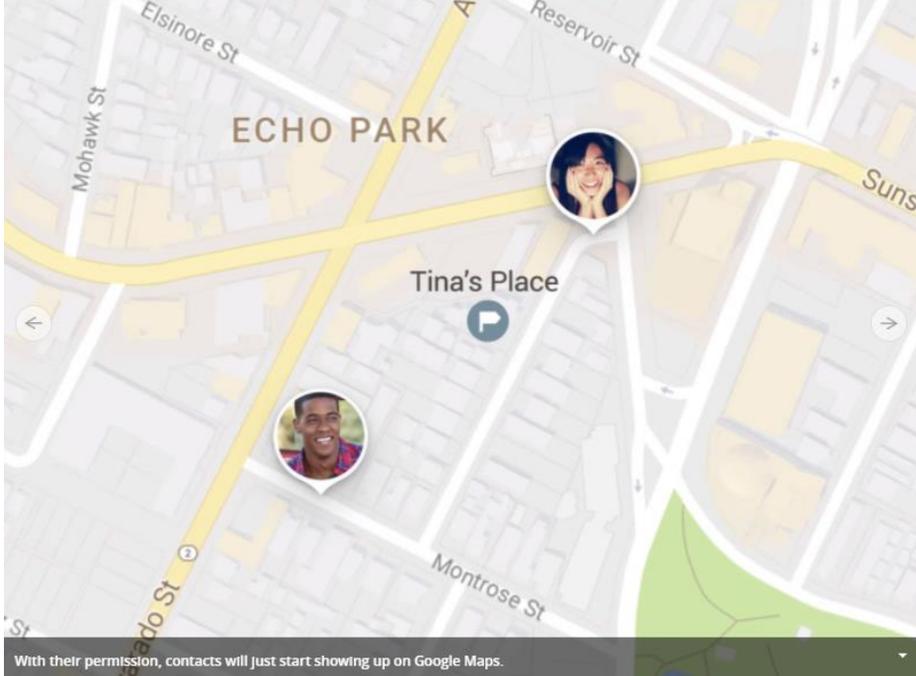
**Exhibit D - Claim Chart for U.S. Patent No. 9,467,838 Against HTC**

US9467838	Exemplary Supporting Evidence Regarding Accused Products
	 <p data-bbox="535 1023 1680 1063">You can pick how long you want to share your location for, and with whom you want to share it.</p> <p data-bbox="535 1063 1680 1096"><a href="https://arstechnica.com/gadgets/2017/03/location-sharing-finally-returns-to-google-maps/">https://arstechnica.com/gadgets/2017/03/location-sharing-finally-returns-to-google-maps/</a></p>

**Exhibit D - Claim Chart for U.S. Patent No. 9,467,838 Against HTC**

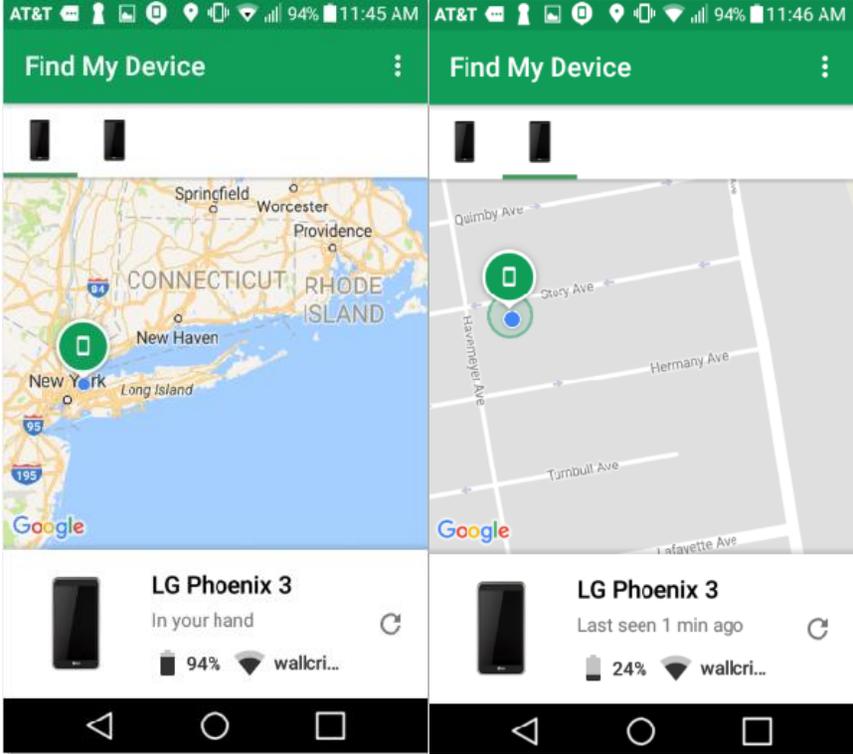
US9467838	Exemplary Supporting Evidence Regarding Accused Products
	 <p data-bbox="535 1144 1197 1177">People you've shared with will get a notification from Google Maps.</p> <p data-bbox="535 1188 1680 1221"><a href="https://arstechnica.com/gadgets/2017/03/location-sharing-finally-returns-to-google-maps/">https://arstechnica.com/gadgets/2017/03/location-sharing-finally-returns-to-google-maps/</a></p>

**Exhibit D - Claim Chart for U.S. Patent No. 9,467,838 Against HTC**

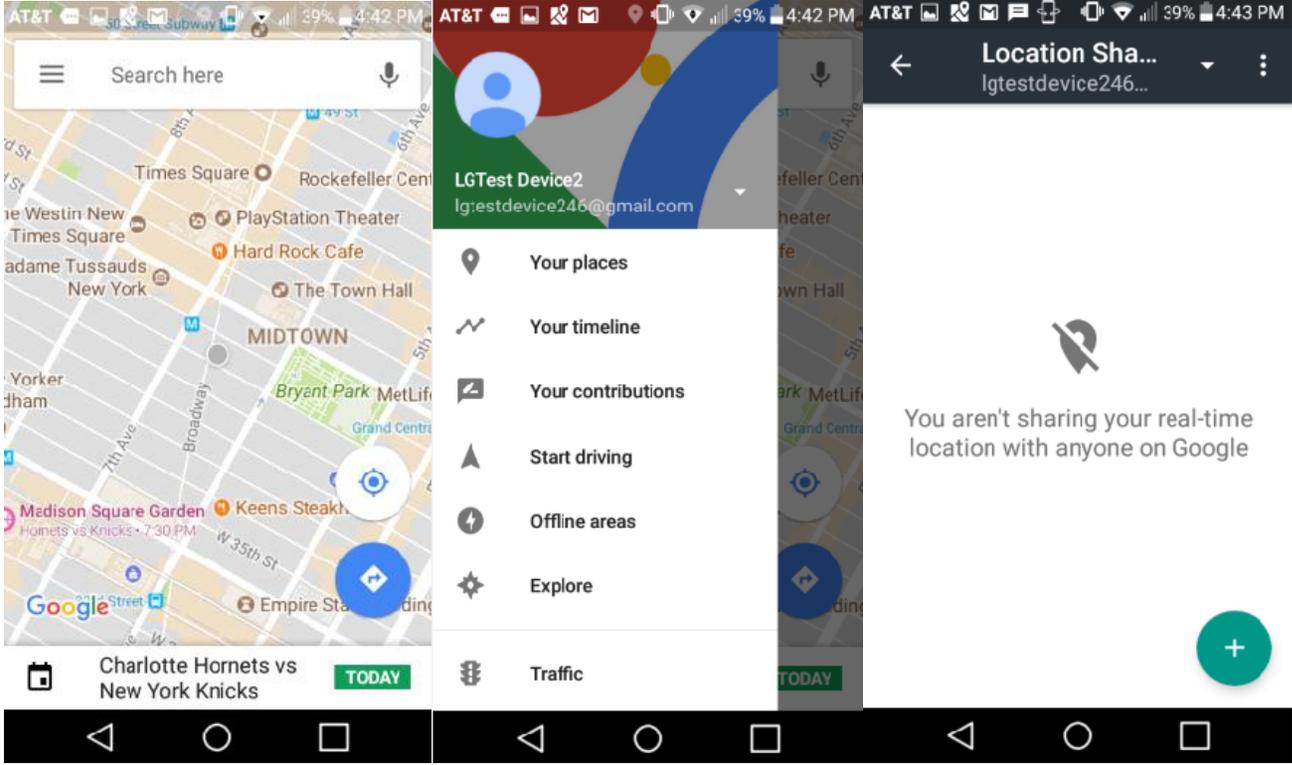
US9467838	Exemplary Supporting Evidence Regarding Accused Products
	 <p data-bbox="535 909 1451 950">With their permission, contacts will just start showing up on Google Maps.</p> <p data-bbox="535 950 1451 982"><a href="https://arstechnica.com/gadgets/2017/03/location-sharing-finally-returns-to-google-maps/">https://arstechnica.com/gadgets/2017/03/location-sharing-finally-returns-to-google-maps/</a></p> <p data-bbox="535 982 1451 1053"><b><u>Exemplary Find My Device Screenshots:</u></b></p>



**Exhibit D - Claim Chart for U.S. Patent No. 9,467,838 Against HTC**

US9467838	Exemplary Supporting Evidence Regarding Accused Products
	 <p><b>Exemplary Google Maps Screenshots:</b></p>

**Exhibit D - Claim Chart for U.S. Patent No. 9,467,838 Against HTC**

US9467838	Exemplary Supporting Evidence Regarding Accused Products
	 <p>The image displays three sequential screenshots from a mobile phone, illustrating the location sharing interface in Google Maps. The first screenshot shows the Google Maps application with a search bar at the top and a map of Midtown Manhattan. A location sharing notification is visible at the top right, indicating that location sharing is currently off. The second screenshot shows the location sharing settings menu, which includes options such as 'Your places', 'Your timeline', 'Your contributions', 'Start driving', 'Offline areas', 'Explore', and 'Traffic'. The third screenshot shows the 'Location Sharing' settings page, where the user is notified that they are not sharing their real-time location with anyone on Google.</p>

**Exhibit D - Claim Chart for U.S. Patent No. 9,467,838 Against HTC**

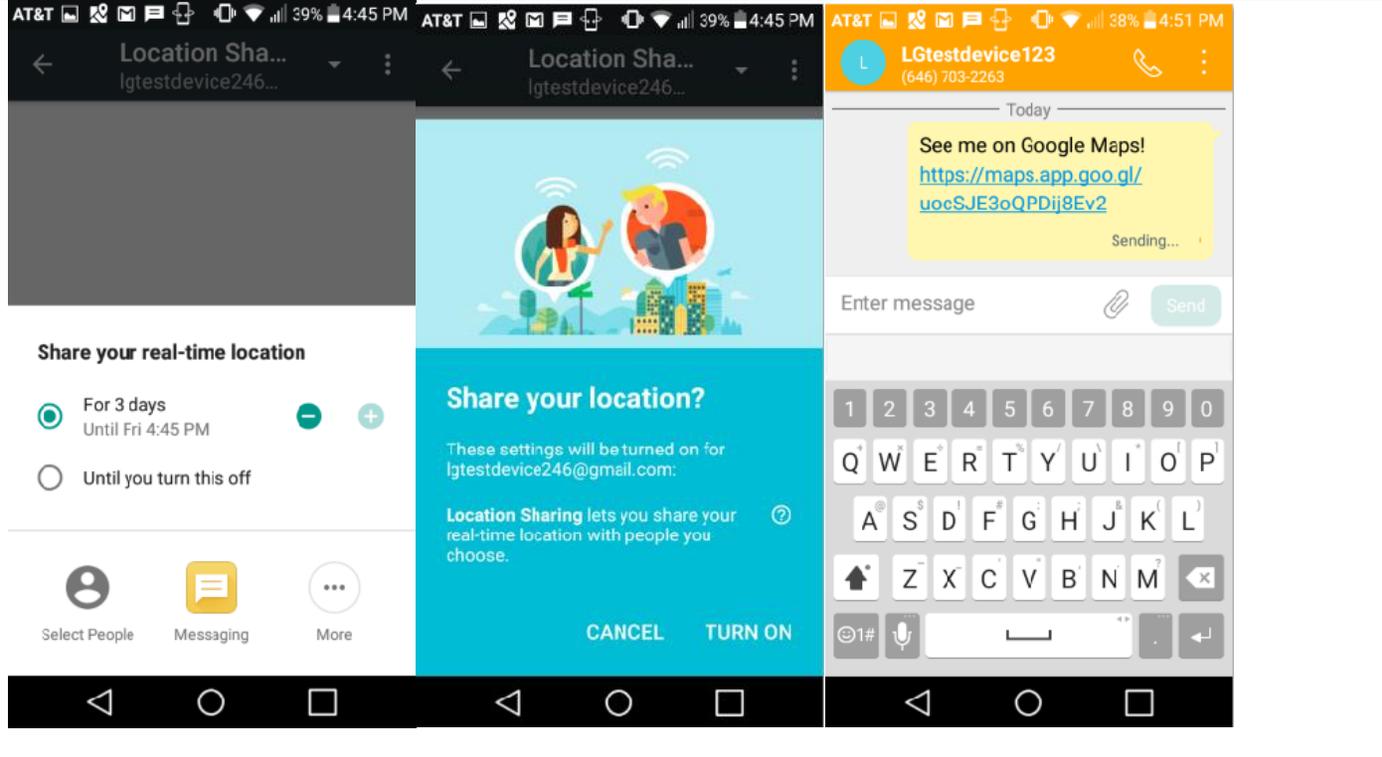
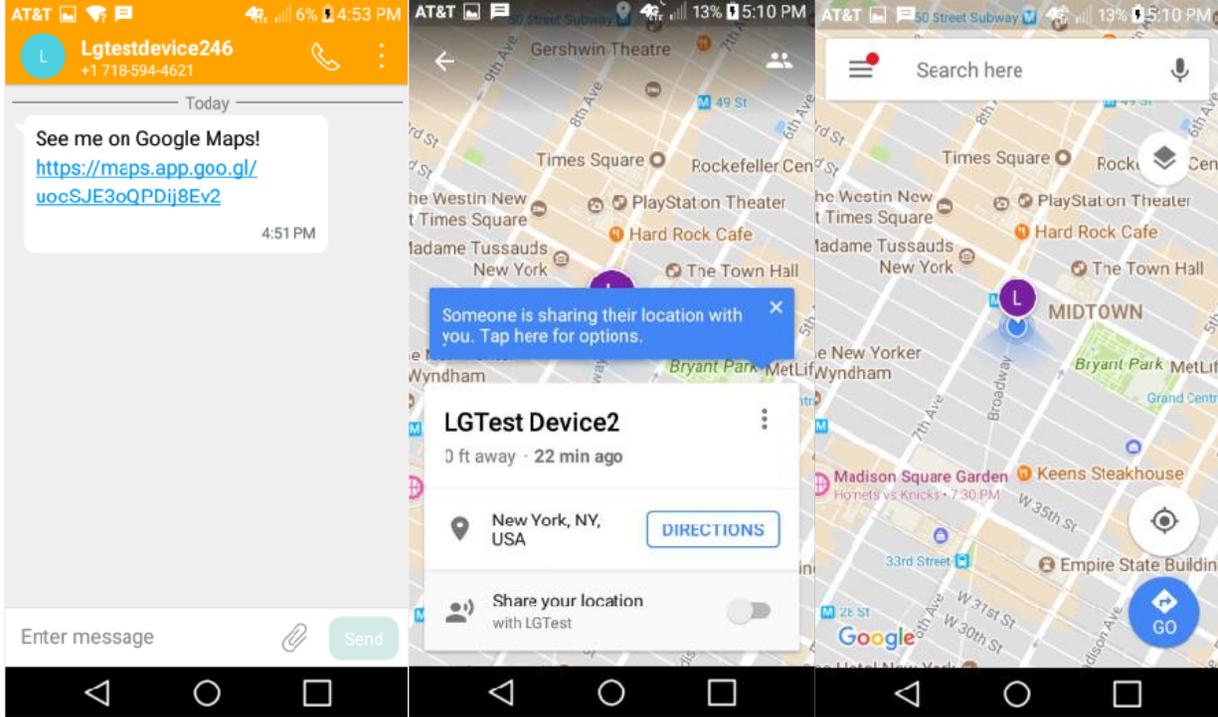
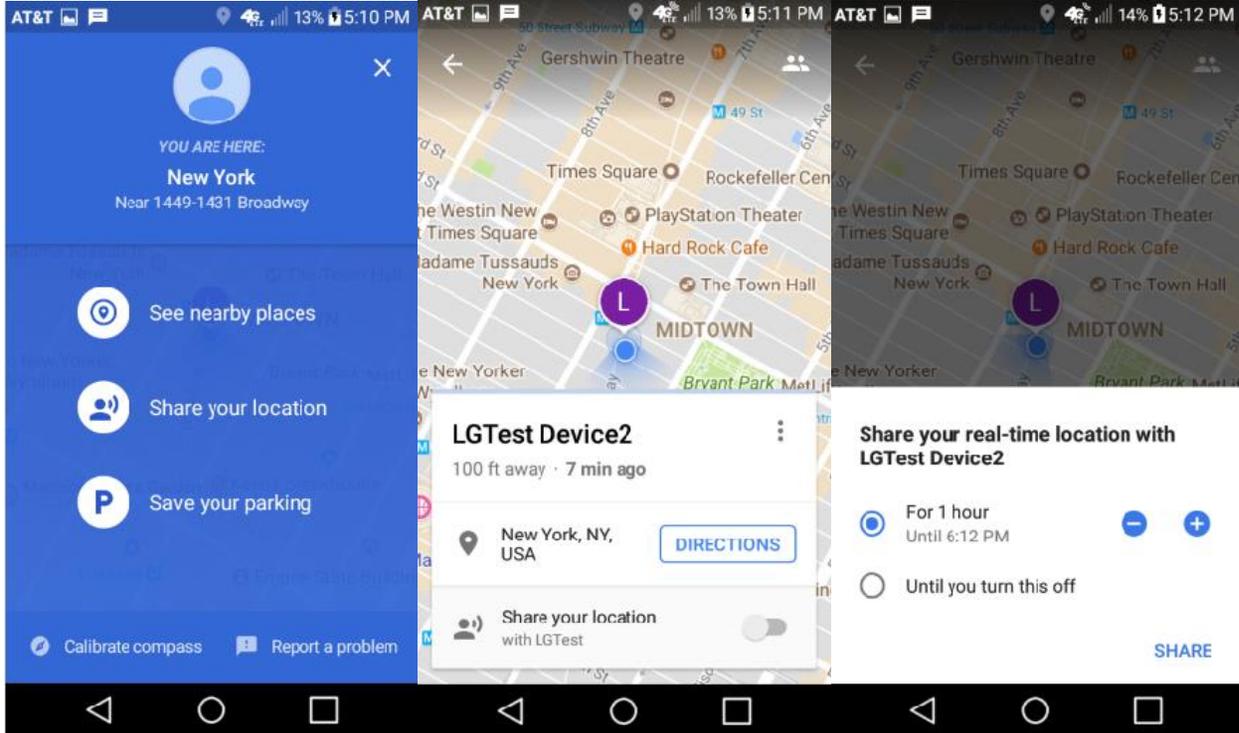
US9467838	Exemplary Supporting Evidence Regarding Accused Products
	 <p>The evidence consists of three screenshots from an Android phone. The first screenshot shows the 'Location Sharing' settings for the contact 'lgtestdevice246...'. It displays two options: 'For 3 days Until Fri 4:45 PM' (selected) and 'Until you turn this off'. Below the settings are icons for 'Select People', 'Messaging', and 'More'. The second screenshot is a confirmation dialog titled 'Share your location?' with a blue background. It states: 'These settings will be turned on for lgtestdevice246@gmail.com. Location Sharing lets you share your real-time location with people you choose.' At the bottom are 'CANCEL' and 'TURN ON' buttons. The third screenshot shows a text message from 'LGtestdevice123 (646) 703-2263' with the text: 'See me on Google Maps! https://maps.app.goo.gl/uocSJE3oQPDij8Ev2'. The message is in the process of being sent.</p>

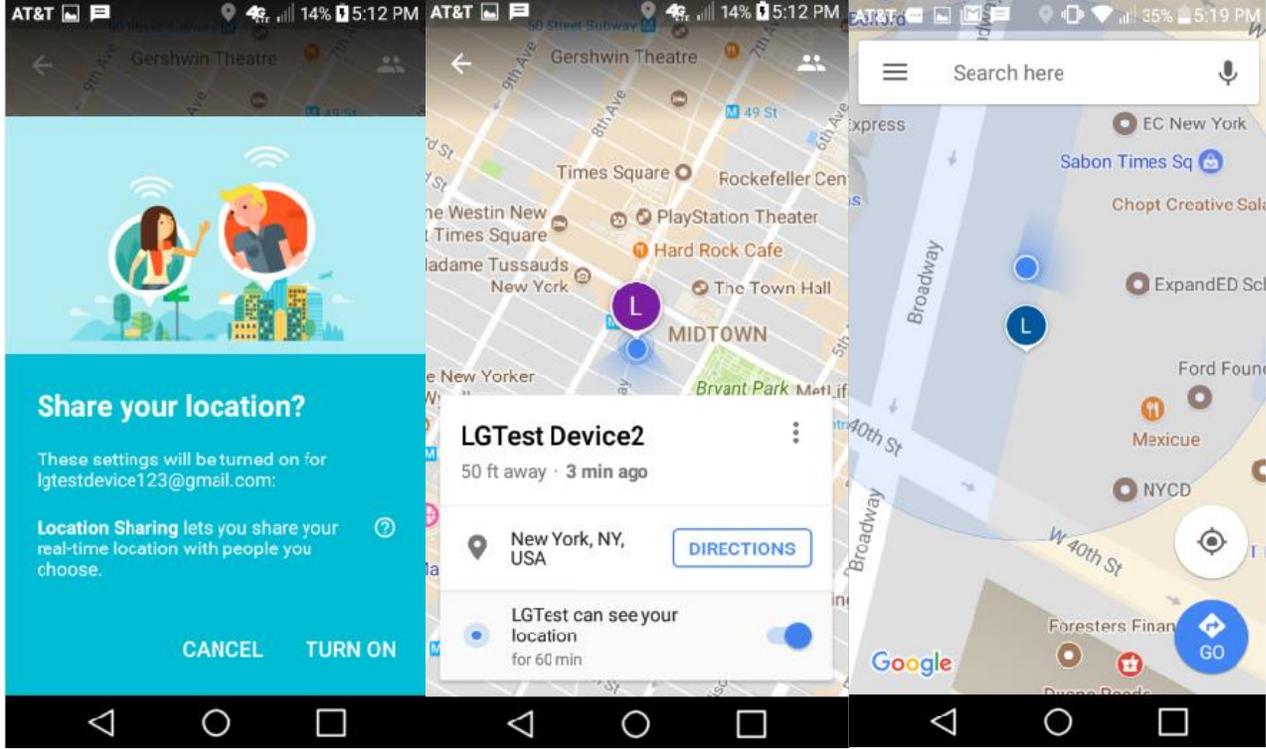
Exhibit D - Claim Chart for U.S. Patent No. 9,467,838 Against HTC

US9467838	Exemplary Supporting Evidence Regarding Accused Products
	 <p>The figure consists of three side-by-side screenshots from a mobile phone. The leftmost screenshot is a text message from a contact named 'Lgtestdevice246' (+1 718-594-4621). The message says 'See me on Google Maps!' and includes a URL: <a href="https://maps.app.goo.gl/uocSJE3oQPDij8Ev2">https://maps.app.goo.gl/uocSJE3oQPDij8Ev2</a>. The middle screenshot shows a location sharing notification for 'LGTest Device2' which is '0 ft away · 22 min ago' in 'New York, NY, USA'. A blue box above it says 'Someone is sharing their location with you. Tap here for options.' The rightmost screenshot shows a Google Maps interface with a location pin in Midtown Manhattan, New York City. The map shows landmarks like Times Square, Hard Rock Cafe, and Bryant Park.</p>

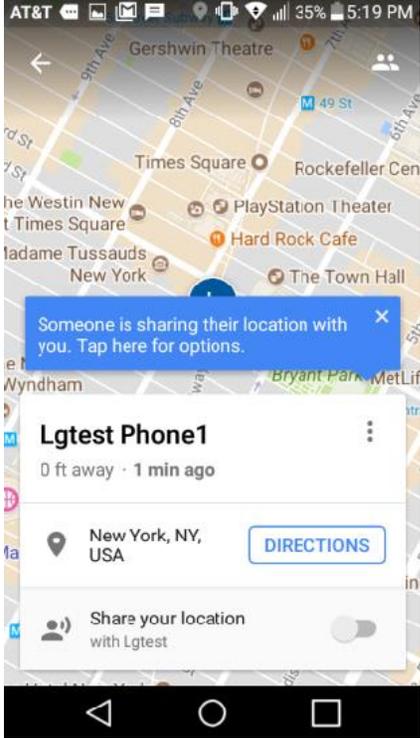
**Exhibit D - Claim Chart for U.S. Patent No. 9,467,838 Against HTC**

US9467838	Exemplary Supporting Evidence Regarding Accused Products
	 <p>The evidence consists of three sequential screenshots from an Android mobile application. The first screenshot shows a location card for 'New York' with the text 'YOU ARE HERE: New York Near 1449-1431 Broadway'. Below the card are three options: 'See nearby places', 'Share your location', and 'Save your parking'. The second screenshot shows a map with a location card for 'LGTest Device2' at 'New York, NY, USA' and a 'SHARE' button. The third screenshot shows the sharing options menu with 'For 1 hour' selected and a 'SHARE' button.</p>

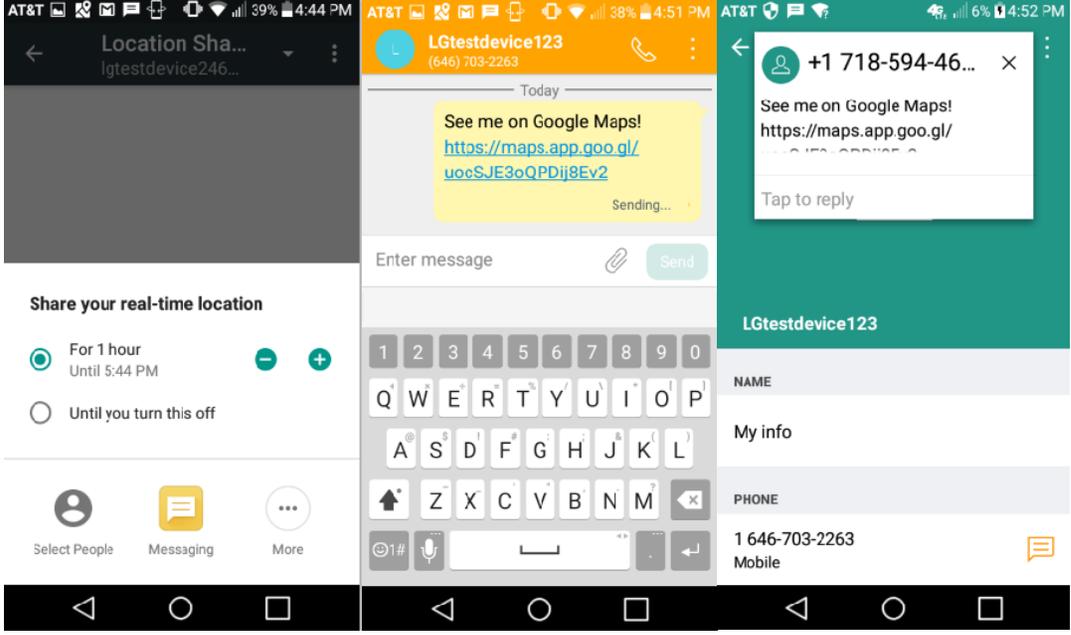
**Exhibit D - Claim Chart for U.S. Patent No. 9,467,838 Against HTC**

US9467838	Exemplary Supporting Evidence Regarding Accused Products
	 <p>The image displays three sequential screenshots from a mobile phone interface, likely an Android device, showing location sharing settings and a notification. The first screenshot shows a 'Share your location?' dialog box with the text: 'These settings will be turned on for lgtestdevice123@gmail.com: Location Sharing lets you share your real-time location with people you choose.' The second screenshot shows a notification for 'LGTest Device2' with the text: '50 ft away · 3 min ago' and 'New York, NY, USA'. The third screenshot shows a map of Midtown Manhattan with a search bar and a 'GO' button.</p>

**Exhibit D - Claim Chart for U.S. Patent No. 9,467,838 Against HTC**

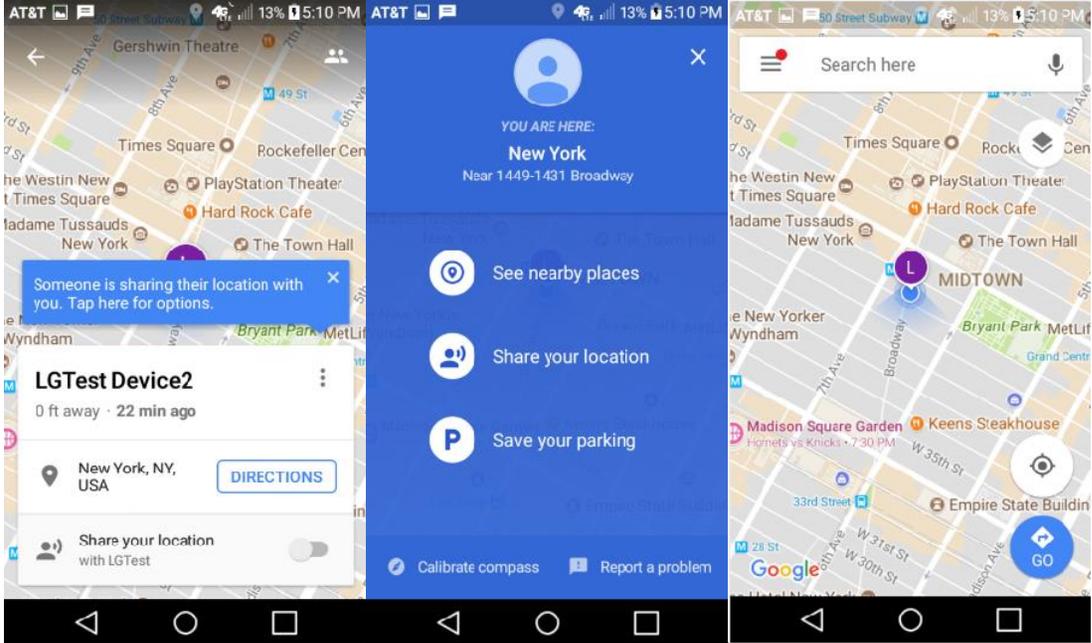
US9467838	Exemplary Supporting Evidence Regarding Accused Products
	 <p>The screenshot shows a mobile phone interface with a map of Times Square, New York. A blue notification bubble at the top of the map reads: "Someone is sharing their location with you. Tap here for options." Below the map, a white information card for "Lgtest Phone1" is displayed. The card shows the location as "New York, NY, USA" and includes a "DIRECTIONS" button. At the bottom of the card, there is a toggle switch for "Share your location with Lgtest", which is currently turned off. The phone's status bar at the top shows AT&amp;T service, 35% battery, and 5:19 PM. The bottom of the screen shows the standard Android navigation bar.</p>

**Exhibit D - Claim Chart for U.S. Patent No. 9,467,838 Against HTC**

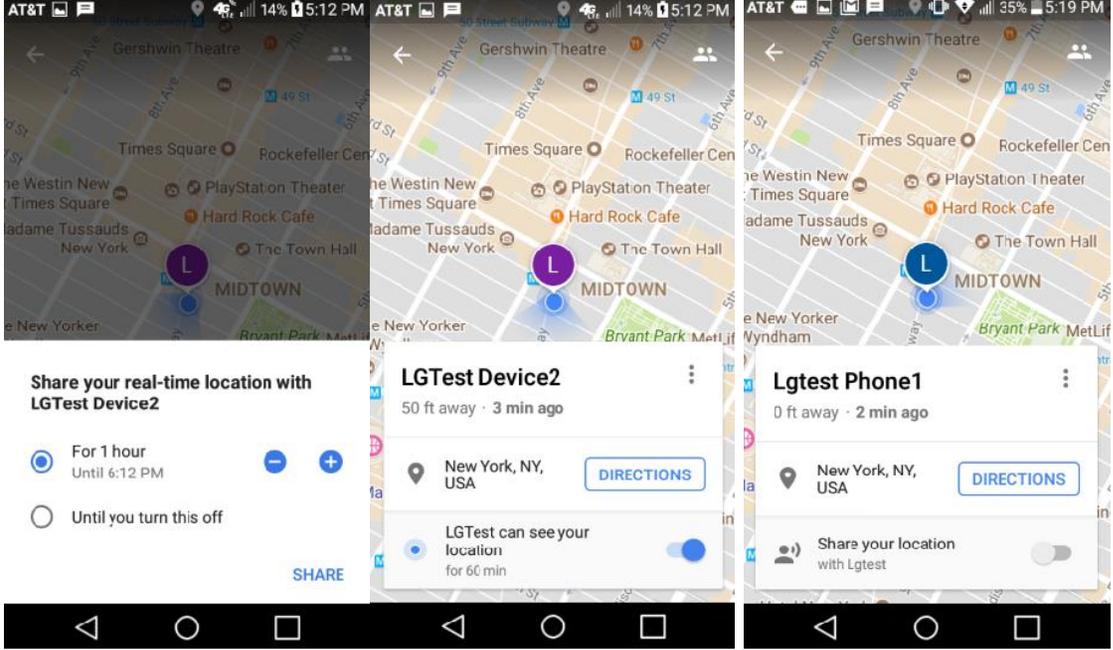
US9467838	Exemplary Supporting Evidence Regarding Accused Products
	 <p>The figure displays three sequential screenshots from a mobile messaging application, illustrating the process of sharing location information and displaying contact details.</p> <ul style="list-style-type: none"><li><b>Left Screenshot (4:44 PM):</b> Shows a conversation with a contact named 'Location Sha...' (lgtestdevice246...). A 'Share your real-time location' dialog is open, with the option 'For 1 hour Until 5:44 PM' selected. The bottom navigation bar includes 'Select People', 'Messaging', and 'More'.</li><li><b>Middle Screenshot (4:51 PM):</b> Shows the same contact, 'LGtestdevice123' (646) 703-2263. A yellow message bubble is being sent: 'See me on Google Maps! <a href="https://maps.app.goo.gl/uocSJE3oQPDij8Ev2">https://maps.app.goo.gl/uocSJE3oQPDij8Ev2</a>'. The input field contains 'Enter message' and a 'Send' button.</li><li><b>Right Screenshot (4:52 PM):</b> Shows the contact's profile card for 'LGtestdevice123'. The 'PHONE' field is visible, displaying '1 646-703-2263 Mobile'. A message bubble is partially visible at the top: 'See me on Google Maps! <a href="https://maps.app.goo.gl/...">https://maps.app.goo.gl/...</a> Tap to reply'.</li></ul>



**Exhibit D - Claim Chart for U.S. Patent No. 9,467,838 Against HTC**

US9467838	Exemplary Supporting Evidence Regarding Accused Products
	 <p>The figure consists of three side-by-side screenshots of an Android mobile phone interface. The leftmost screenshot shows a Google Maps application with a location sharing notification for 'LCTest Device2' at 'New York, NY, USA'. The middle screenshot shows a blue overlay menu with options: 'See nearby places', 'Share your location', and 'Save your parking'. The rightmost screenshot shows the same Google Maps application with a search bar at the top and a 'GO' button at the bottom right.</p>

**Exhibit D - Claim Chart for U.S. Patent No. 9,467,838 Against HTC**

US9467838	Exemplary Supporting Evidence Regarding Accused Products
	 <p><b><u>Exemplary Source Code:</u></b></p> <p>The above functionality is performed at least in part by the following publicly available source code and/or source code that invokes or is invoked by the following source code (or a substantially similar copy compiled and loaded onto the Accused Products by HTC). AGIS reserves the right to amend these contentions to include additional source code as discovery progresses and as additional source code is made available.</p>

**Exhibit D - Claim Chart for U.S. Patent No. 9,467,838 Against HTC**

US9467838	Exemplary Supporting Evidence Regarding Accused Products
	<h2 data-bbox="554 240 1041 293">Contacts Provider</h2> <p data-bbox="554 329 1495 589">The Contacts Provider is a powerful and flexible Android component that manages the device's central repository of data about people. The Contacts Provider is the source of data you see in the device's contacts application, and you can also access its data in your own application and transfer data between the device and online services. The provider accommodates a wide range of data sources and tries to manage as much data as possible for each person, with the result that its organization is complex. Because of this, the provider's API includes an extensive set of contract classes and interfaces that facilitate both data retrieval and modification.</p> <p data-bbox="554 625 863 646">This guide describes the following:</p> <ul data-bbox="554 675 1398 850" style="list-style-type: none"><li data-bbox="554 675 831 696">• The basic provider structure.</li><li data-bbox="554 725 919 747">• How to retrieve data from the provider.</li><li data-bbox="554 776 890 797">• How to modify data in the provider.</li><li data-bbox="554 826 1398 847">• How to write a sync adapter for synchronizing data from your server to the Contacts Provider.</li></ul> <p data-bbox="537 862 1514 889"><a href="https://developer.android.com/guide/topics/providers/contacts-provider.html">https://developer.android.com/guide/topics/providers/contacts-provider.html</a></p>

**Exhibit D - Claim Chart for U.S. Patent No. 9,467,838 Against HTC**

US9467838	Exemplary Supporting Evidence Regarding Accused Products
	<p><b>Overview</b></p> <p>ContactsContract defines an extensible database of contact-related information. Contact information is stored in a three-tier data model:</p> <ul style="list-style-type: none"> <li>• A row in the <code>ContactsContract.Data</code> table can store any kind of personal data, such as a phone number or email addresses. The set of data kinds that can be stored in this table is open-ended. There is a predefined set of common kinds, but any application can add its own data kinds.</li> <li>• A row in the <code>ContactsContract.RawContacts</code> table represents a set of data describing a person and associated with a single account (for example, one of the user's Gmail accounts).</li> <li>• A row in the <code>ContactsContract.Contacts</code> table represents an aggregate of one or more RawContacts presumably describing the same person. When data in or associated with the RawContacts table is changed, the affected aggregate contacts are updated as necessary.</li> </ul> <p>Other tables include:</p> <ul style="list-style-type: none"> <li>• <code>ContactsContract.Groups</code>, which contains information about raw contact groups such as Gmail contact groups. The current API does not support the notion of groups spanning multiple accounts.</li> <li>• <code>ContactsContract.StatusUpdates</code>, which contains social status updates including IM availability.</li> <li>• <code>ContactsContract.AggregationExceptions</code>, which is used for manual aggregation and disaggregation of raw contacts</li> <li>• <code>ContactsContract.Settings</code>, which contains visibility and sync settings for accounts and groups.</li> <li>• <code>ContactsContract.SyncState</code>, which contains free-form data maintained on behalf of sync adapters</li> <li>• <code>ContactsContract.PhoneLookup</code>, which is used for quick caller-ID lookup</li> </ul> <p><a href="https://developer.android.com/reference/android/provider/ContactsContract.html">https://developer.android.com/reference/android/provider/ContactsContract.html</a></p> <p><b>Data</b></p> <p>As noted previously, the data for a raw contact is stored in a <code>ContactsContract.Data</code> row that is linked to the raw contact's <code>_ID</code> value. This allows a single raw contact to have multiple instances of the same type of data such as email addresses or phone numbers. For example, if "Thomas Higginson" for <code>emilyd@gmail.com</code> (the raw contact row for Thomas Higginson associated with the Google account <code>emilyd@gmail.com</code>) has a home email address of <code>thigg@gmail.com</code> and a work email address of <code>thomas.higginson@gmail.com</code>, the Contacts Provider stores the two email address rows and links them both to the raw contact.</p> <p>Notice that different types of data are stored in this single table. Display name, phone number, email, postal address, photo, and website detail rows are all found in the <code>ContactsContract.Data</code> table. To help manage this, the <code>ContactsContract.Data</code> table has some columns with descriptive names, and others with generic names. The contents of a descriptive-name column have the same meaning regardless of the type of data in the row, while the contents of a generic-name column have different meanings depending on the type of data.</p> <p><a href="https://developer.android.com/guide/topics/providers/contacts-provider.html">https://developer.android.com/guide/topics/providers/contacts-provider.html</a></p>

**Exhibit D - Claim Chart for U.S. Patent No. 9,467,838 Against HTC**

US9467838	Exemplary Supporting Evidence Regarding Accused Products														
	<table border="1"> <thead> <tr> <th data-bbox="535 237 630 274">Task</th> <th data-bbox="630 237 877 274">Action</th> <th data-bbox="877 237 1213 274">Data</th> <th data-bbox="1213 237 1507 274">MIME type</th> <th data-bbox="1507 237 1770 274">Notes</th> </tr> </thead> <tbody> <tr> <td data-bbox="535 274 630 906">Pick a contact from a list</td> <td data-bbox="630 274 877 906">ACTION_PICK</td> <td data-bbox="877 274 1213 906">                     One of:                     <ul style="list-style-type: none"> <li>• <code>Contacts.CONTENT_URI</code>, which displays a list of contacts.</li> <li>• <code>Phone.CONTENT_URI</code>, which displays a list of phone numbers for a raw contact.</li> <li>• <code>StructuredPostal.CONTENT_URI</code>, which displays a list of postal addresses for a raw contact.</li> <li>• <code>Email.CONTENT_URI</code>, which displays a list of email addresses for a raw contact.</li> </ul> </td> <td data-bbox="1213 274 1507 906">Not used</td> <td data-bbox="1507 274 1770 906">                     Displays a list of raw contacts or a list of data from a raw contact, depending on the content URI type you supply.                      Call <code>startActivityForResult()</code>, which returns the content URI of the selected row. The form of the URI is the table's content URI with the row's <code>LOOKUP_ID</code> appended to it. The device's contacts app delegates read and write permissions to this content URI for the life of your activity. See the <a href="#">Content Provider Basics</a> guide for more details.                 </td> </tr> </tbody> </table>	Task	Action	Data	MIME type	Notes	Pick a contact from a list	ACTION_PICK	One of: <ul style="list-style-type: none"> <li>• <code>Contacts.CONTENT_URI</code>, which displays a list of contacts.</li> <li>• <code>Phone.CONTENT_URI</code>, which displays a list of phone numbers for a raw contact.</li> <li>• <code>StructuredPostal.CONTENT_URI</code>, which displays a list of postal addresses for a raw contact.</li> <li>• <code>Email.CONTENT_URI</code>, which displays a list of email addresses for a raw contact.</li> </ul>	Not used	Displays a list of raw contacts or a list of data from a raw contact, depending on the content URI type you supply. Call <code>startActivityForResult()</code> , which returns the content URI of the selected row. The form of the URI is the table's content URI with the row's <code>LOOKUP_ID</code> appended to it. The device's contacts app delegates read and write permissions to this content URI for the life of your activity. See the <a href="#">Content Provider Basics</a> guide for more details.				<p><a href="https://developer.android.com/guide/topics/providers/contacts-provider.html">https://developer.android.com/guide/topics/providers/contacts-provider.html</a></p>
Task	Action	Data	MIME type	Notes											
Pick a contact from a list	ACTION_PICK	One of: <ul style="list-style-type: none"> <li>• <code>Contacts.CONTENT_URI</code>, which displays a list of contacts.</li> <li>• <code>Phone.CONTENT_URI</code>, which displays a list of phone numbers for a raw contact.</li> <li>• <code>StructuredPostal.CONTENT_URI</code>, which displays a list of postal addresses for a raw contact.</li> <li>• <code>Email.CONTENT_URI</code>, which displays a list of email addresses for a raw contact.</li> </ul>	Not used	Displays a list of raw contacts or a list of data from a raw contact, depending on the content URI type you supply. Call <code>startActivityForResult()</code> , which returns the content URI of the selected row. The form of the URI is the table's content URI with the row's <code>LOOKUP_ID</code> appended to it. The device's contacts app delegates read and write permissions to this content URI for the life of your activity. See the <a href="#">Content Provider Basics</a> guide for more details.											

## Exhibit D - Claim Chart for U.S. Patent No. 9,467,838 Against HTC

US9467838	Exemplary Supporting Evidence Regarding Accused Products
	<pre>25  * Parsed form of the intent sent to the Contacts application. 26  */ 27  public class ContactsRequest { 28 29      /** Default mode: browse contacts */ 30      public static final int ACTION_DEFAULT = 10; 31 32      /** Show all contacts */ 33      public static final int ACTION_ALL_CONTACTS = 15; 34 35      /** Show all contacts with phone numbers */ 36      public static final int ACTION_CONTACTS_WITH_PHONES = 17; 37 38      /** Show contents of a specific group */ 39      public static final int ACTION_GROUP = 20; 40 41      /** Show all starred contacts */ 42      public static final int ACTION_STARRED = 30; 43 44      /** Show frequently contacted contacts */ 45      public static final int ACTION_FREQUENT = 40; 46 47      /** Show starred and the frequent */ 48      public static final int ACTION_STREQUENT = 50; 49 50      /** Show all contacts and pick them when clicking */ 51      public static final int ACTION_PICK_CONTACT = 60; 52 53      /** Show all contacts as well as the option to create a new one */ 54      public static final int ACTION_PICK_OR_CREATE_CONTACT = 70; 55 56      /** Show all contacts and pick them for edit when clicking, and allow creating a new contact */ 57      public static final int ACTION_INSERT_OR_EDIT_CONTACT = 80;</pre> <p><a href="https://android.googlesource.com/platform/packages/apps/Contacts/+nougat-mr1-release/src/com/android/contacts/list/ContactsRequest.java">https://android.googlesource.com/platform/packages/apps/Contacts/+nougat-mr1-release/src/com/android/contacts/list/ContactsRequest.java</a></p>

## Exhibit D - Claim Chart for U.S. Patent No. 9,467,838 Against HTC

US9467838	Exemplary Supporting Evidence Regarding Accused Products
	<pre>59     /** Show all phone numbers and pick them when clicking */ 60     public static final int ACTION_PICK_PHONE = 90; 61 62     /** Show all postal addresses and pick them when clicking */ 63     public static final int ACTION_PICK_POSTAL = 100; 64 65     /** Show all postal addresses and pick them when clicking */ 66     public static final int ACTION_PICK_EMAIL = 105; 67 68     /** Show all contacts and create a shortcut for the picked contact */ 69     public static final int ACTION_CREATE_SHORTCUT_CONTACT = 110; 70 71     /** Show all phone numbers and create a call shortcut for the picked number */ 72     public static final int ACTION_CREATE_SHORTCUT_CALL = 120; 73 74     /** Show all phone numbers and create an SMS shortcut for the picked number */ 75     public static final int ACTION_CREATE_SHORTCUT_SMS = 130; 76 77     /** Show all contacts and activate the specified one */ 78     public static final int ACTION_VIEW_CONTACT = 140; 79 80     /** Show contacts recommended for joining with a specified target contact */ 81     public static final int ACTION_PICK_JOIN = 150;</pre> <p><a href="https://android.googlesource.com/platform/packages/apps/Contacts/+/nougat-mr1-release/src/com/android/contacts/list/ContactsRequest.java">https://android.googlesource.com/platform/packages/apps/Contacts/+/nougat-mr1-release/src/com/android/contacts/list/ContactsRequest.java</a></p>

**Exhibit D - Claim Chart for U.S. Patent No. 9,467,838 Against HTC**

US9467838	Exemplary Supporting Evidence Regarding Accused Products
	<pre> 104  * Displays a list to browse contacts. 105  */ 106  public class PeopleActivity extends ContactsActivity implements 107      View.OnCreateContextMenuListener, 108      View.OnClickListener, 109      ActionBarAdapter.Listener, 110      DialogManager.DialogShowingViewActivity, 111      ContactListFilterController.ContactListFilterListener, 112      ProviderStatusListener, 113      MultiContactDeleteListener, 114      JoinContactsListener { </pre> <p data-bbox="535 657 1591 727"><a href="https://android.googlesource.com/platform/packages/apps/Contacts/+/nougat-mr1-release/src/com/android/contacts/activities/PeopleActivity.java">https://android.googlesource.com/platform/packages/apps/Contacts/+/nougat-mr1-release/src/com/android/contacts/activities/PeopleActivity.java</a></p> <pre> 145      * Showing a list of Contacts. Also used for showing search results in search mode. 146      */ 147      private MultiSelectContactsListFragment mAllFragment; 148      private ContactTileListFragment mFavoritesFragment; </pre> <p data-bbox="535 868 1591 935"><a href="https://android.googlesource.com/platform/packages/apps/Contacts/+/nougat-mr1-release/src/com/android/contacts/activities/PeopleActivity.java">https://android.googlesource.com/platform/packages/apps/Contacts/+/nougat-mr1-release/src/com/android/contacts/activities/PeopleActivity.java</a></p>



## Exhibit D - Claim Chart for U.S. Patent No. 9,467,838 Against HTC

US9467838	Exemplary Supporting Evidence Regarding Accused Products
	<pre>458     private void configureFragments(boolean fromRequest) { 459         if (fromRequest) { 460             ContactListFilter filter = null; 461             int actionCode = mRequest.getActionCode(); 462             boolean searchMode = mRequest.isSearchMode(); 463             final int tabToOpen; 464             switch (actionCode) { 465                 case ContactsRequest.ACTION_ALL_CONTACTS: 466                     filter = ContactListFilter.createFilterWithType( 467                         ContactListFilter.FILTER_TYPE_ALL_ACCOUNTS); 468                     tabToOpen = TabState.ALL; 469                     break; 470                 case ContactsRequest.ACTION_CONTACTS_WITH_PHONES: 471                     filter = ContactListFilter.createFilterWithType( 472                         ContactListFilter.FILTER_TYPE_WITH_PHONE_NUMBERS_ONLY); 473                     tabToOpen = TabState.ALL; 474                     break; 475 476                 case ContactsRequest.ACTION_FREQUENT: 477                 case ContactsRequest.ACTION_STREQUENT: 478                 case ContactsRequest.ACTION_STARRED: 479                     tabToOpen = TabState.FAVORITES; 480                     break; 481                 case ContactsRequest.ACTION_VIEW_CONTACT: 482                     tabToOpen = TabState.ALL; 483                     break; 484                 default: 485                     tabToOpen = -1; 486                     break; 487             } </pre> <p data-bbox="531 1321 1591 1390"><a href="https://android.gogglesource.com/platform/packages/apps/Contacts/+nougat-mr1-release/src/com/android/contacts/activities/PeopleActivity.java">https://android.gogglesource.com/platform/packages/apps/Contacts/+nougat-mr1-release/src/com/android/contacts/activities/PeopleActivity.java</a></p>

## Exhibit D - Claim Chart for U.S. Patent No. 9,467,838 Against HTC

US9467838	Exemplary Supporting Evidence Regarding Accused Products
	<pre>488         if (tabToOpen != -1) { 489             mActionBarAdapter.setCurrentTab(tabToOpen); 490         } 491 492         if (filter != null) { 493             mContactListFilterController.setContactListFilter(filter, false); 494             searchMode = false; 495         } 496 497         if (mRequest.getContactUri() != null) { 498             searchMode = false; 499         } 500 501         mActionBarAdapter.setSearchMode(searchMode); 502         configureContactListFragmentForRequest(); 503     } 504 505     configureContactListFragment(); 506 507     invalidateOptionsMenuIfNeeded(); 508 }</pre> <p data-bbox="531 1019 1591 1084"><a href="https://android.googlesource.com/platform/packages/apps/Contacts/+/nougat-mr1-release/src/com/android/contacts/activities/PeopleActivity.java">https://android.googlesource.com/platform/packages/apps/Contacts/+/nougat-mr1-release/src/com/android/contacts/activities/PeopleActivity.java</a></p>

**Exhibit D - Claim Chart for U.S. Patent No. 9,467,838 Against HTC**

US9467838	Exemplary Supporting Evidence Regarding Accused Products
	<pre>35 public class ProfileAndContactsLoader extends CursorLoader { 36 37     private boolean mLoadProfile; 38 39     private String[] mProjection; 40 41     private Uri mExtraUri; 42     private String[] mExtraProjection; 43     private String mExtraSelection; 44     private String[] mExtraSelectionArgs; 45     private boolean mMergeExtraContactsAfterPrimary; 46 47     public ProfileAndContactsLoader(Context context) { 48         super(context); 49     }</pre> <p data-bbox="531 808 1709 873"><a href="https://android.googlesource.com/platform/packages/apps/ContactsCommon/+nougat-mr1-release/src/com/android/contacts/common/list/ProfileAndContactsLoader.java">https://android.googlesource.com/platform/packages/apps/ContactsCommon/+nougat-mr1-release/src/com/android/contacts/common/list/ProfileAndContactsLoader.java</a></p>

## Exhibit D - Claim Chart for U.S. Patent No. 9,467,838 Against HTC

US9467838	Exemplary Supporting Evidence Regarding Accused Products
	<pre>35 public final class GroupMemberLoader extends CursorLoader { 36 37     public static class GroupEditorQuery { 38         private static final String[] PROJECTION = new String[] { 39             Data.CONTACT_ID,           // 0 40             Data.RAW_CONTACT_ID,      // 1 41             Data.DISPLAY_NAME_PRIMARY, // 2 42             Data.PHOTO_URI,           // 3 43             Data.LOOKUP_KEY,          // 4 44         }; 45 46         public static final int CONTACT_ID           = 0; 47         public static final int RAW_CONTACT_ID      = 1; 48         public static final int CONTACT_DISPLAY_NAME_PRIMARY = 2; 49         public static final int CONTACT_PHOTO_URI   = 3; 50         public static final int CONTACT_LOOKUP_KEY  = 4; 51     } 52 53     public static class GroupDetailQuery { 54         private static final String[] PROJECTION = new String[] { 55             Data.CONTACT_ID,           // 0 56             Data.PHOTO_URI,           // 1 57             Data.LOOKUP_KEY,          // 2 58             Data.DISPLAY_NAME_PRIMARY, // 3 59             Data.CONTACT_PRESENCE,    // 4 60             Data.CONTACT_STATUS,      // 5 61         }; 62 63         public static final int CONTACT_ID           = 0; 64         public static final int CONTACT_PHOTO_URI   = 1; 65         public static final int CONTACT_LOOKUP_KEY  = 2; 66         public static final int CONTACT_DISPLAY_NAME_PRIMARY = 3; 67         public static final int CONTACT_PRESENCE_STATUS = 4; 68         public static final int CONTACT_STATUS      = 5; 69     } 70 71     private final long mGroupId;</pre> <p><a href="https://android.googlesource.com/platform/packages/apps/Contacts/+/nougat-mr1-release/src/com/android/contacts/GroupMemberLoader.java">https://android.googlesource.com/platform/packages/apps/Contacts/+/nougat-mr1-release/src/com/android/contacts/GroupMemberLoader.java</a></p>

## Exhibit D - Claim Chart for U.S. Patent No. 9,467,838 Against HTC

US9467838	Exemplary Supporting Evidence Regarding Accused Products
	<pre>24 * Group loader for the group list that includes details such as the number of contacts per group 25 * and number of groups per account. This list is sorted by account type, account name, where the 26 * group names are in alphabetical order. Note that the list excludes default, favorite, and deleted 27 * groups. 28 */ 29 public final class GroupListLoader extends CursorLoader { 30 31     private final static String[] COLUMNS = new String[] { 32         Groups.ACCOUNT_NAME, 33         Groups.ACCOUNT_TYPE, 34         Groups.DATA_SET, 35         Groups._ID, 36         Groups.TITLE, 37         Groups.SUMMARY_COUNT, 38     }; 39 40     public final static int ACCOUNT_NAME = 0; 41     public final static int ACCOUNT_TYPE = 1; 42     public final static int DATA_SET = 2; 43     public final static int GROUP_ID = 3; 44     public final static int TITLE = 4; 45     public final static int MEMBER_COUNT = 5; 46 47     private static final Uri GROUP_LIST_URI = Groups.CONTENT_SUMMARY_URI; 48 49     public GroupListLoader(Context context) { 50         super(context, GROUP_LIST_URI, COLUMNS, Groups.ACCOUNT_TYPE + " NOT NULL AND " 51             + Groups.ACCOUNT_NAME + " NOT NULL AND " + Groups.AUTO_ADD + "=0 AND " + 52             Groups.FAVORITES + "=0 AND " + Groups.DELETED + "=0", null, 53             Groups.ACCOUNT_TYPE + ", " + Groups.ACCOUNT_NAME + ", " + Groups.DATA_SET + ", " + 54             Groups.TITLE + " COLLATE LOCALIZED ASC"); 55     } 56 }</pre> <p><a href="https://android.googlesource.com/platform/packages/apps/Contacts/+/nougat-mr1-release/src/com/android/contacts/GroupListLoader.java">https://android.googlesource.com/platform/packages/apps/Contacts/+/nougat-mr1-release/src/com/android/contacts/GroupListLoader.java</a></p>

## Exhibit D - Claim Chart for U.S. Patent No. 9,467,838 Against HTC

US9467838	Exemplary Supporting Evidence Regarding Accused Products
	<pre> 24  * Group meta-data loader. Loads all groups or just a single group from the 25  * database (if given a {@link Uri}). 26  */ 27  public final class GroupMetaDataLoader extends CursorLoader { 28 29      private final static String[] COLUMNS = new String[] { 30          Groups.ACCOUNT_NAME, 31          Groups.ACCOUNT_TYPE, 32          Groups.DATA_SET, 33          Groups._ID, 34          Groups.TITLE, 35          Groups.AUTO_ADD, 36          Groups.FAVORITES, 37          Groups.GROUP_IS_READ_ONLY, 38          Groups.DELETED, 39      }; 40 41      public final static int ACCOUNT_NAME = 0; 42      public final static int ACCOUNT_TYPE = 1; 43      public final static int DATA_SET = 2; 44      public final static int GROUP_ID = 3; 45      public final static int TITLE = 4; 46      public final static int AUTO_ADD = 5; 47      public final static int FAVORITES = 6; 48      public final static int IS_READ_ONLY = 7; 49      public final static int DELETED = 8; 50 51      public GroupMetaDataLoader(Context context, Uri groupUri) { 52          super(context, ensureIsGroupUri(groupUri), COLUMNS, Groups.ACCOUNT_TYPE + " NOT NULL AND " 53              + Groups.ACCOUNT_NAME + " NOT NULL", null, null); 54      } 55 56      /** 57       * Ensures that this is a valid group URI. If invalid, then an exception is 58       * thrown. Otherwise, the original URI is returned. 59       */ 60      private static Uri ensureIsGroupUri(final Uri groupUri) { 61          // TODO: Fix ContactsProvider2 getType method to resolve the group Uri 62          if (groupUri == null) { 63              throw new IllegalArgumentException("Uri must not be null"); 64          } 65          if (!groupUri.toString().startsWith(Groups.CONTENT_URI.toString())) { 66              throw new IllegalArgumentException("Invalid group Uri: " + groupUri); 67          } 68          return groupUri; 69      } 70  } </pre> <p><a href="https://android.googlesource.com/platform/packages/apps/Contacts/+/nougat-mr1-release/src/com/android/contacts/GroupMetaDataLoader.java">https://android.googlesource.com/platform/packages/apps/Contacts/+/nougat-mr1-release/src/com/android/contacts/GroupMetaDataLoader.java</a></p>

## Exhibit D - Claim Chart for U.S. Patent No. 9,467,838 Against HTC

US9467838	Exemplary Supporting Evidence Regarding Accused Products
	<pre>19 * Meta-data for a contact group. We load all groups associated with the contact's 20 * constituent accounts. 21 */ 22 public final class GroupMetaData { 23     private String mAccountName; 24     private String mAccountType; 25     private String mDataSet; 26     private long mGroupId; 27     private String mTitle; 28     private boolean mDefaultGroup; 29     private boolean mFavorites; 30 31     public GroupMetaData(String accountName, String accountType, String dataSet, long groupId, 32         String title, boolean defaultGroup, boolean favorites) { 33         this.mAccountName = accountName; 34         this.mAccountType = accountType; 35         this.mDataSet = dataSet; 36         this.mGroupId = groupId; 37         this.mTitle = title; 38         this.mDefaultGroup = defaultGroup; 39         this.mFavorites = favorites; 40     } 41 42     public String getAccountName() { 43         return mAccountName; 44     } 45 46     public String getAccountType() { 47         return mAccountType; 48     } 49 50     public String getDataSet() { 51         return mDataSet; 52     } 53 54     public long getGroupId() { 55         return mGroupId; 56     } 57 58     public String getTitle() { 59         return mTitle; 60     } 61 62     public boolean isDefaultGroup() { 63         return mDefaultGroup; 64     } 65 66     public boolean isFavorites() { 67         return mFavorites; 68     } 69 }</pre> <p><a href="https://android.googlesource.com/platform/packages/apps/ContactsCommon/+nougat-mr1-release/src/com/android/contacts/common/GroupMetaData.java">https://android.googlesource.com/platform/packages/apps/ContactsCommon/+nougat-mr1-release/src/com/android/contacts/common/GroupMetaData.java</a></p>

## Exhibit D - Claim Chart for U.S. Patent No. 9,467,838 Against HTC

US9467838	Exemplary Supporting Evidence Regarding Accused Products
	<pre>44 * Class that sends chat message via SMS. 45 * 46 * The interface emulates a blocking sending similar to making an HTTP request. 47 * It calls the SmsManager to send a (potentially multipart) message and waits 48 * on the sent status on each part. The waiting has a timeout so it won't wait 49 * forever. Once the sent status of all parts received, the call returns. 50 * A successful sending requires success status for all parts. Otherwise, we 51 * pick the highest level of failure as the error for the whole message, which 52 * is used to determine if we need to retry the sending. 53 */ 54 public class SmsSender { 55     private static final String TAG = LogUtil.BUGLE_TAG; 56 57     public static final String EXTRA_PART_ID = "part_id"; 58 59     /* 60     * A map for pending sms messages. The key is the random request UUID. 61     */ 62     private static ConcurrentHashMap&lt;Uri, SendResult&gt; sPendingMessageMap = 63         new ConcurrentHashMap&lt;Uri, SendResult&gt;(); 64 65     private static final Random RANDOM = new Random(); 66 67     // Whether we should send multipart SMS as separate messages 68     private static Boolean sSendMultipartSmsAsSeparateMessages = null; 69</pre> <p><a href="https://android.googlesource.com/platform/packages/apps/Messaging/+/nougat-mr1-release/src/com/android/messaging/sms/SmsSender.java">https://android.googlesource.com/platform/packages/apps/Messaging/+/nougat-mr1-release/src/com/android/messaging/sms/SmsSender.java</a></p>



## Exhibit D - Claim Chart for U.S. Patent No. 9,467,838 Against HTC

US9467838	Exemplary Supporting Evidence Regarding Accused Products
	<pre> 253 // Actually sending the message using SmsManager 254 private static void sendInternal(final Context context, final int subId, String dest, 255     final ArrayList&lt;String&gt; messages, final String serviceCenter, 256     final boolean requireDeliveryReport, final Uri messageUri) throws SmsException { 257     Assert.notNull(context); 258     final SmsManager smsManager = PhoneUtils.get(subId).getSmsManager(); 259     final int messageCount = messages.size(); 260     final ArrayList&lt;PendingIntent&gt; deliveryIntents = new ArrayList&lt;PendingIntent&gt;(messageCount); 261     final ArrayList&lt;PendingIntent&gt; sentIntents = new ArrayList&lt;PendingIntent&gt;(messageCount); 262     for (int i = 0; i &lt; messageCount; i++) { 263         // Make pending intents different for each message part 264         final int partId = (messageCount &lt;= 1 ? 0 : i + 1); 265         if (requireDeliveryReport &amp;&amp; (i == (messageCount - 1))) { 266             // TODO we only care about the delivery status of the last part 267             // Shall we have better tracking of delivery status of all parts? 268             deliveryIntents.add(PendingIntent.getBroadcast( 269                 context, 270                 partId, 271                 getSendStatusIntent(context, SendStatusReceiver.MESSAGE_DELIVERED_ACTION, 272                     messageUri, partId, subId), 273                 0/*flag*/)); 274         } else { 275             deliveryIntents.add(null); 276         } 277         sentIntents.add(PendingIntent.getBroadcast( 278             context, 279             partId, 280             getSendStatusIntent(context, SendStatusReceiver.MESSAGE_SENT_ACTION, 281                 messageUri, partId, subId), 282             0/*flag*/)); 283     } 284     if (sSendMultipartSmsAsSeparateMessages == null) { 285         sSendMultipartSmsAsSeparateMessages = MmsConfig.get(subId) 286             .getSendMultipartSmsAsSeparateMessages(); 287     } </pre> <p><a href="https://android.googlesource.com/platform/packages/apps/Messaging/+nougat-mr1-release/src/com/android/messaging/sms/SmsSender.java">https://android.googlesource.com/platform/packages/apps/Messaging/+nougat-mr1-release/src/com/android/messaging/sms/SmsSender.java</a></p>

## Exhibit D - Claim Chart for U.S. Patent No. 9,467,838 Against HTC

US9467838	Exemplary Supporting Evidence Regarding Accused Products
	<pre>288     try { 289         if (sSendMultipartSmsAsSeparateMessages) { 290             // If multipart sms is not supported, send them as separate messages 291             for (int i = 0; i &lt; messageCount; i++) { 292                 smsManager.sendTextMessage(dest, 293                     serviceCenter, 294                     messages.get(i), 295                     sentIntents.get(i), 296                     deliveryIntents.get(i)); 297             } 298         } else { 299             smsManager.sendMultipartTextMessage( 300                 dest, serviceCenter, messages, sentIntents, deliveryIntents); 301         } 302     } catch (final Exception e) { 303         throw new SmsException("SmsSender: caught exception in sending " + e); 304     } 305 }</pre> <p><a href="https://android.googlesource.com/platform/packages/apps/Messaging/+nougat-mr1-release/src/com/android/messaging/sms/SmsSender.java">https://android.googlesource.com/platform/packages/apps/Messaging/+nougat-mr1-release/src/com/android/messaging/sms/SmsSender.java</a></p> <pre>56  * Class that receives incoming SMS messages through android.provider.Telephony.SMS_RECEIVED 57  * 58  * This class serves two purposes: 59  * - Process phone verification SMS messages 60  * - Handle SMS messages when the user has enabled us to be the default SMS app (Pre-KLP) 61  */ 62  public final class SmsReceiver extends BroadcastReceiver { 63      private static final String TAG = LogUtil.BUGLE_TAG; 64 65      private static ArrayList&lt;Pattern&gt; sIgnoreSmsPatterns; 66  }</pre> <p><a href="https://android.googlesource.com/platform/packages/apps/Messaging/+nougat-mr1-release/src/com/android/messaging/receiver/SmsReceiver.java">https://android.googlesource.com/platform/packages/apps/Messaging/+nougat-mr1-release/src/com/android/messaging/receiver/SmsReceiver.java</a></p>

## Exhibit D - Claim Chart for U.S. Patent No. 9,467,838 Against HTC

US9467838	Exemplary Supporting Evidence Regarding Accused Products
	<pre> 200     public static void deliverSmsMessages(final Context context, final int subId, 201         final int errorCode, final android.telephony.SmsMessage[] messages) { 202         final ContentValues messageValues = 203             MmsUtils.parseReceivedSmsMessage(context, messages, errorCode); 204 205         LogUtil.v(TAG, "SmsReceiver.deliverSmsMessages"); 206 207         final long nowInMillis = System.currentTimeMillis(); 208         final long receivedTimestampMs = MmsUtils.getMessageDate(messages[0], nowInMillis); 209 210         messageValues.put(Sms.Inbox.DATE, receivedTimestampMs); 211         // Default to unread and unseen for us but ReceiveSmsMessageAction will override 212         // seen for the telephony db. 213         messageValues.put(Sms.Inbox.READ, 0); 214         messageValues.put(Sms.Inbox.SEEN, 0); 215         if (OsUtil.isAtLeastL_MR1()) { 216             messageValues.put(Sms.SUBSCRIPTION_ID, subId); 217         } 218 219         if (messages[0].getMessageClass() == android.telephony.SmsMessage.MessageClass.CLASS_0    220             DebugUtils.debugClassZeroSmsEnabled()) { 221             Factory.get().getUIIntents().launchClassZeroActivity(context, messageValues); 222         } else { 223             final ReceiveSmsMessageAction action = new ReceiveSmsMessageAction(messageValues); 224             action.start(); 225         } 226     } </pre> <p><a href="https://android.googlesource.com/platform/packages/apps/Messaging/+nougat-mr1-release/src/com/android/messaging/receiver/SmsReceiver.java">https://android.googlesource.com/platform/packages/apps/Messaging/+nougat-mr1-release/src/com/android/messaging/receiver/SmsReceiver.java</a></p>

## Exhibit D - Claim Chart for U.S. Patent No. 9,467,838 Against HTC

US9467838	Exemplary Supporting Evidence Regarding Accused Products
	<pre>228     @Override 229     public void onReceive(final Context context, final Intent intent) { 230         LogUtil.v(TAG, "SmsReceiver.onReceive " + intent); 231         // On KLP+ we only take delivery of SMS messages in SmsDeliverReceiver. 232         if (PhoneUtils.getDefault().isSmsEnabled()) { 233             final String action = intent.getAction(); 234             if (OsUtil.isSecondaryUser() &amp;&amp; 235                 (Telephony.Sms.Intents.SMS_RECEIVED_ACTION.equals(action)    236                  // TODO: update this with the actual constant from Telephony 237                  "android.provider.Telephony.MMS_DOWNLOADED".equals(action))) { 238                 postNewMessageSecondaryUserNotification(); 239             } else if (!OsUtil.isAtLeastKLP()) { 240                 deliverSmsIntent(context, intent); 241             } 242         } 243     }</pre> <p><a href="https://android.googlesource.com/platform/packages/apps/Messaging/+nougat-mr1-release/src/com/android/messaging/receiver/SmsReceiver.java">https://android.googlesource.com/platform/packages/apps/Messaging/+nougat-mr1-release/src/com/android/messaging/receiver/SmsReceiver.java</a></p>

## Exhibit D - Claim Chart for U.S. Patent No. 9,467,838 Against HTC

US9467838	Exemplary Supporting Evidence Regarding Accused Products
	<pre> 52  * Class that sends chat message via MMS. 53  * 54  * The interface emulates a blocking send similar to making an HTTP request. 55  */ 56  public class MmsSender { 57      private static final String TAG = LogUtil.BUGLE_TAG; 58 59      /** 60       * Send an MMS message. 61       * 62       * @param context Context 63       * @param messageUri The unique URI of the message for identifying it during sending 64       * @param sendReq The SendReq PDU of the message 65       * @throws MmsFailureException 66       */ 67      public static void sendMms(final Context context, final int subId, final Uri messageUri, 68                               final SendReq sendReq, final Bundle sentIntentExtras) throws MmsFailureException { 69          sendMms(context, 70                subId, 71                messageUri, 72                null /* locationUri */, 73                sendReq, 74                true /* responseImportant */, 75                sentIntentExtras); 76      } </pre> <p><a href="https://android.googlesource.com/platform/packages/apps/Messaging/+/nougat-mr1-release/src/com/android/messaging/sms/MmsSender.java">https://android.googlesource.com/platform/packages/apps/Messaging/+/nougat-mr1-release/src/com/android/messaging/sms/MmsSender.java</a></p>

## Exhibit D - Claim Chart for U.S. Patent No. 9,467,838 Against HTC

US9467838	Exemplary Supporting Evidence Regarding Accused Products
	<pre> 240      * Download an MMS message. 241      * 242      * @param context Context 243      * @param contentLocation The url of the MMS message 244      * @throws MmsFailureException 245      * @throws InvalidHeaderValueException 246      */ 247      public static void downloadMms(final Context context, final int subId, 248          final String contentLocation, Bundle extras) throws MmsFailureException, 249          InvalidHeaderValueException { 250          final Uri requestUri = Uri.parse(contentLocation); 251          final Uri contentUri = MmsFileProvider.buildRawMmsUri(); 252 253          final Intent downloadedIntent = new Intent(SendStatusReceiver.MMS_DOWNLOADED_ACTION, 254              requestUri, 255              context, 256              SendStatusReceiver.class); 257          downloadedIntent.putExtra(SendMessageAction.EXTRA_CONTENT_URI, contentUri); 258          if (extras != null) { 259              downloadedIntent.putExtras(extras); 260          } 261          final PendingIntent downloadedPendingIntent = PendingIntent.getBroadcast( 262              context, 263              0 /*request code*/, 264              downloadedIntent, 265              PendingIntent.FLAG_UPDATE_CURRENT); 266 267          MmsManager.downloadMultimediaMessage(subId, context, contentLocation, contentUri, 268              downloadedPendingIntent); 269      } </pre> <p><a href="https://android.googlesource.com/platform/packages/apps/Messaging/+nougat-mr1-release/src/com/android/messaging/sms/MmsSender.java">https://android.googlesource.com/platform/packages/apps/Messaging/+nougat-mr1-release/src/com/android/messaging/sms/MmsSender.java</a></p>

## Exhibit D - Claim Chart for U.S. Patent No. 9,467,838 Against HTC

US9467838	Exemplary Supporting Evidence Regarding Accused Products
	<pre> 97      * Execute an MMS HTTP request, either a POST (sending) or a GET (downloading) 98      * 99      * @param urlString The request URL, for sending it is usually the MMSC, and for downloading 100     *           it is the message URL 101     * @param pdu For POST (sending) only, the PDU to send 102     * @param method HTTP method, POST for sending and GET for downloading 103     * @param isProxySet Is there a proxy for the MMSC 104     * @param proxyHost The proxy host 105     * @param proxyPort The proxy port 106     * @param mmsConfig The MMS config to use 107     * @param userAgent The user agent header value 108     * @param uaProfUrl The UA Prof URL header value 109     * @return The HTTP response body 110     * @throws MmsHttpException For any failures 111     */ 112     public byte[] execute(String urlString, byte[] pdu, String method, boolean isProxySet, 113         String proxyHost, int proxyPort, Bundle mmsConfig, String userAgent, String uaProfUrl) 114         throws MmsHttpException { 115         Log.d(MmsService.TAG, "HTTP: " + method + " " + Utils.redactUrlForNonVerbose(urlString) 116             + (isProxySet ? (" proxy=" + proxyHost + ":" + proxyPort) : "") 117             + ", PDU size=" + (pdu != null ? pdu.length : 0)); 118         checkMethod(method); 119         HttpURLConnection connection = null; 120         try { 121             Proxy proxy = Proxy.NO_PROXY; 122             if (isProxySet) { 123                 proxy = new Proxy(Proxy.Type.HTTP, new InetSocketAddress(proxyHost, proxyPort)); 124             } 125             final URL url = new URL(urlString); 126             // Now get the connection 127             connection = (HttpURLConnection) url.openConnection(proxy); 128             connection.setDoInput(true); 129             connection.setConnectTimeout( 130                 mmsConfig.getInt(CarrierConfigValuesLoader.CONFIG_HTTP_SOCKET_TIMEOUT, 131                     CarrierConfigValuesLoader.CONFIG_HTTP_SOCKET_TIMEOUT_DEFAULT)); </pre> <p><a href="https://android.googlesource.com/platform/packages/apps/Messaging/+nougat-mr1-release/src/android/support/v7/mms/MmsHttpClient.java">https://android.googlesource.com/platform/packages/apps/Messaging/+nougat-mr1-release/src/android/support/v7/mms/MmsHttpClient.java</a></p>

## Exhibit D - Claim Chart for U.S. Patent No. 9,467,838 Against HTC

US9467838	Exemplary Supporting Evidence Regarding Accused Products
	<pre> 132 // ----- COMMON HEADERS ----- 133 // Header: Accept 134 connection.setRequestProperty(HEADER_ACCEPT, HEADER_VALUE_ACCEPT); 135 // Header: Accept-Language 136 connection.setRequestProperty( 137     HEADER_ACCEPT_LANGUAGE, getCurrentAcceptLanguage(Locale.getDefault())); 138 // Header: User-Agent 139 Log.i(MmsService.TAG, "HTTP: User-Agent=" + userAgent); 140 connection.setRequestProperty(HEADER_USER_AGENT, userAgent); 141 // Header: x-wap-profile 142 final String uaProfUrlTagName = mmsConfig.getString( 143     CarrierConfigValuesLoader.CONFIG_UA_PROF_TAG_NAME, 144     CarrierConfigValuesLoader.CONFIG_UA_PROF_TAG_NAME_DEFAULT); 145 if (uaProfUrl != null) { 146     Log.i(MmsService.TAG, "HTTP: UaProfUrl=" + uaProfUrl); 147     connection.setRequestProperty(uaProfUrlTagName, uaProfUrl); 148 } 149 // Add extra headers specified by mms_config.xml's httpparams 150 addExtraHeaders(connection, mmsConfig); 151 // Different stuff for GET and POST 152 if (METHOD_POST.equals(method)) { 153     if (pdu == null    pdu.length &lt; 1) { 154         Log.e(MmsService.TAG, "HTTP: empty pdu"); 155         throw new MmsHttpException(0/*statusCode*/, "Sending empty PDU"); 156     } 157     connection.setDoOutput(true); 158     connection.setRequestMethod(METHOD_POST); 159     if (mmsConfig.getBoolean( 160         CarrierConfigValuesLoader.CONFIG_SUPPORT_HTTP_CHARSET_HEADER, 161         CarrierConfigValuesLoader.CONFIG_SUPPORT_HTTP_CHARSET_HEADER_DEFAULT)) { 162         connection.setRequestProperty(HEADER_CONTENT_TYPE, 163             HEADER_VALUE_CONTENT_TYPE_WITH_CHARSET); 164     } else { 165         connection.setRequestProperty(HEADER_CONTENT_TYPE, 166             HEADER_VALUE_CONTENT_TYPE_WITHOUT_CHARSET); </pre> <p><a href="https://android.googlesource.com/platform/packages/apps/Messaging+/nougat-mr1-release/src/android/support/v7/mms/MmsHttpClient.java">https://android.googlesource.com/platform/packages/apps/Messaging+/nougat-mr1-release/src/android/support/v7/mms/MmsHttpClient.java</a></p>



## Exhibit D - Claim Chart for U.S. Patent No. 9,467,838 Against HTC

US9467838	Exemplary Supporting Evidence Regarding Accused Products
	<pre> 167         } 168         if (Log.isLoggable(MmsService.TAG, Log.VERBOSE)) { 169             logHttpHeaders(connection.getRequestProperties()); 170         } 171         connection.setFixedLengthStreamingMode(pdu.length); 172         // Sending request body 173         final OutputStream out = 174             new BufferedOutputStream(connection.getOutputStream()); 175         out.write(pdu); 176         out.flush(); 177         out.close(); 178     } else if (METHOD_GET.equals(method)) { 179         if (Log.isLoggable(MmsService.TAG, Log.VERBOSE)) { 180             logHttpHeaders(connection.getRequestProperties()); 181         } 182         connection.setRequestMethod(METHOD_GET); 183     } 184     // Get response 185     final int responseCode = connection.getResponseCode(); 186     final String responseMessage = connection.getResponseMessage(); 187     Log.d(MmsService.TAG, "HTTP: " + responseCode + " " + responseMessage); 188     if (Log.isLoggable(MmsService.TAG, Log.VERBOSE)) { 189         logHttpHeaders(connection.getHeaderFields()); 190     } 191     if (responseCode / 100 != 2) { 192         throw new MmsHttpException(responseCode, responseMessage); 193     } 194     final InputStream in = new BufferedInputStream(connection.getInputStream()); 195     final ByteArrayOutputStream byteOut = new ByteArrayOutputStream(); 196     final byte[] buf = new byte[4096]; 197     int count = 0; 198     while ((count = in.read(buf)) &gt; 0) { 199         byteOut.write(buf, 0, count); 200     } 201     in.close(); 202     final byte[] responseBody = byteOut.toByteArray(); 203     Log.d(MmsService.TAG, "HTTP: response size=" 204         + (responseBody != null ? responseBody.length : 0)); 205     return responseBody; </pre> <p><a href="https://android.googlesource.com/platform/packages/apps/Messaging/+/nougat-mr1-release/src/android/support/v7/mms/MmsHttpClient.java">https://android.googlesource.com/platform/packages/apps/Messaging/+/nougat-mr1-release/src/android/support/v7/mms/MmsHttpClient.java</a></p>

## Exhibit D - Claim Chart for U.S. Patent No. 9,467,838 Against HTC

US9467838	Exemplary Supporting Evidence Regarding Accused Products
	<pre> 38  * Request to send an MMS 39  */ 40  class SendRequest extends MmsRequest { 41      // Max send response PDU size in bytes (exceeding this may cause problem with 42      // system intent delivery). 43      private static final int MAX_SEND_RESPONSE_SIZE = 1000 * 1024; 44 45      private byte[] mPduData; 46 47      SendRequest(final String locationUri, final Uri pduUri, final PendingIntent sentIntent) { 48          super(locationUri, pduUri, sentIntent); 49      } 50 51      @Override 52      protected boolean loadRequest(final Context context, final Bundle mmsConfig) { 53          mPduData = readPduFromContentUri( 54              context, 55              mPduUri, 56              mmsConfig.getInt( 57                  CarrierConfigValuesLoader.CONFIG_MAX_MESSAGE_SIZE, 58                  CarrierConfigValuesLoader.CONFIG_MAX_MESSAGE_SIZE_DEFAULT)); 59          return (mPduData != null); 60      } 61 62      @Override 63      protected boolean transferResponse(final Context context, final Intent fillIn, 64          final byte[] response) { 65          // SendConf pdus are always small and can be included in the intent 66          if (response != null &amp;&amp; fillIn != null) { 67              if (response.length &gt; MAX_SEND_RESPONSE_SIZE) { 68                  // If the response PDU is too large, it won't be able to fit in 69                  // the PendingIntent to be transferred via system IPC. 70                  return false; 71              } 72              fillIn.putExtra(SmsManager.EXTRA_MMS_DATA, response); 73          } 74          return true; 75      } </pre>

## Exhibit D - Claim Chart for U.S. Patent No. 9,467,838 Against HTC

US9467838	Exemplary Supporting Evidence Regarding Accused Products
	<p data-bbox="537 235 1619 302"><a href="https://android.googlesource.com/platform/packages/apps/Messaging+/nougat-mr1-release/src/android/support/v7/mms/SendRequest.java">https://android.googlesource.com/platform/packages/apps/Messaging+/nougat-mr1-release/src/android/support/v7/mms/SendRequest.java</a></p> <pre data-bbox="562 358 1003 391">public static LocationRequest create ()</pre> <p data-bbox="548 423 1052 448">Create a location request with default parameters.</p> <p data-bbox="548 480 1661 540">Default parameters are for a block accuracy, slowly updated location. It can then be adjusted as required by the applications before passing to the <code>FusedLocationProviderApi</code>.</p> <p data-bbox="573 565 657 589"><b>Returns</b></p> <ul data-bbox="579 610 835 634" style="list-style-type: none"><li>• a new location request</li></ul> <p data-bbox="537 651 1818 680"><a href="https://developers.google.com/android/reference/com/google/android/gms/location/LocationRequest">https://developers.google.com/android/reference/com/google/android/gms/location/LocationRequest</a></p>

**Exhibit D - Claim Chart for U.S. Patent No. 9,467,838 Against HTC**

US9467838	Exemplary Supporting Evidence Regarding Accused Products
	<p><b>public static final int PRIORITY_BALANCED_POWER_ACCURACY</b></p> <p>Used with <code>setPriority(int)</code> to request "block" level accuracy.</p> <p>Block level accuracy is considered to be about 100 meter accuracy. Using a coarse accuracy such as this often consumes less power.</p> <p>Constant Value: 102</p> <p><b>public static final int PRIORITY_HIGH_ACCURACY</b></p> <p>Used with <code>setPriority(int)</code> to request the most accurate locations available.</p> <p>This will return the finest location available.</p> <p>Constant Value: 100</p> <p><b>public static final int PRIORITY_LOW_POWER</b></p> <p>Used with <code>setPriority(int)</code> to request "city" level accuracy.</p> <p>City level accuracy is considered to be about 10km accuracy. Using a coarse accuracy such as this often consumes less power.</p> <p>Constant Value: 104</p> <p><a href="https://developers.google.com/android/reference/com/google/android/gms/location/LocationRequest">https://developers.google.com/android/reference/com/google/android/gms/location/LocationRequest</a></p>

## Exhibit D - Claim Chart for U.S. Patent No. 9,467,838 Against HTC

US9467838	Exemplary Supporting Evidence Regarding Accused Products
	<p data-bbox="562 248 1774 285"><code>public Task&lt;Location&gt; getLastLocation ()</code></p> <p data-bbox="548 313 1129 337">Returns the best most recent location currently available.</p> <p data-bbox="548 370 1719 427">If a location is not available, which should happen very rarely, null will be returned. The best accuracy available while respecting the location permissions will be returned.</p> <p data-bbox="548 459 1759 516">This method provides a simplified way to get location. It is particularly well suited for applications that do not require an accurate location and that do not want to maintain extra logic for location updates.</p> <p data-bbox="562 578 1774 615"><code>public Task&lt;LocationAvailability&gt; getLocationAvailability ()</code></p> <p data-bbox="548 643 1713 699">Returns the availability of location data. When <code>isLocationAvailable()</code> returns true, then the location returned by <code>getLastLocation()</code> will be reasonably up to date within the hints specified by the active <code>LocationRequest</code>s.</p> <p data-bbox="548 732 1495 756">If the client isn't connected to Google Play services and the request times out, null is returned.</p> <p data-bbox="548 789 1696 846">Note it's always possible for <code>getLastLocation()</code> to return null even when this method returns true (e.g. location settings were disabled between calls).</p> <p data-bbox="537 862 1902 928"><a href="https://developers.google.com/android/reference/com/google/android/gms/location/FusedLocationProviderClient">https://developers.google.com/android/reference/com/google/android/gms/location/FusedLocationProviderClient</a></p>

**Exhibit D - Claim Chart for U.S. Patent No. 9,467,838 Against HTC**

US9467838	Exemplary Supporting Evidence Regarding Accused Products						
	<p data-bbox="546 240 1774 326"> <code>public Task&lt;Void&gt; requestLocationUpdates (LocationRequest request, LocationCallback callback, Looper looper)</code> </p> <p data-bbox="546 354 1291 380">Requests location updates with a callback on the specified Looper thread.</p> <p data-bbox="546 410 1711 472">This method is suited for the foreground use cases. For background use cases, the <code>PendingIntent</code> version of the method is recommended, see <code>requestLocationUpdates(LocationRequest, PendingIntent)</code>.</p> <p data-bbox="546 503 1396 529">Any previous LocationRequests registered on this LocationListener will be replaced.</p> <p data-bbox="546 560 1711 654">This call will keep the Google Play services connection active, so make sure to call <code>removeLocationUpdates(LocationCallback)</code> when you no longer need it, otherwise you lose the benefits of the automatic connection management.</p> <p data-bbox="546 685 1774 711">Callbacks for <code>LocationCallback</code> will be made on the specified thread, which must already be a prepared looper thread.</p> <p data-bbox="546 735 682 761"><b>Parameters</b></p> <table border="1" data-bbox="546 789 1774 1008"> <tbody> <tr> <td data-bbox="546 789 655 857"><b>request</b></td> <td data-bbox="655 789 1774 857">The location request for the updates.</td> </tr> <tr> <td data-bbox="546 857 655 925"><b>callback</b></td> <td data-bbox="655 857 1774 925">The callback for the location updates.</td> </tr> <tr> <td data-bbox="546 925 655 1008"><b>looper</b></td> <td data-bbox="655 925 1774 1008">The Looper object whose message queue will be used to implement the callback mechanism, or null to make callbacks on the calling thread.</td> </tr> </tbody> </table> <p data-bbox="546 1023 1911 1089"> <a href="https://developers.google.com/android/reference/com/google/android/gms/location/FusedLocationProviderClient">https://developers.google.com/android/reference/com/google/android/gms/location/FusedLocationProviderClient</a> </p>	<b>request</b>	The location request for the updates.	<b>callback</b>	The callback for the location updates.	<b>looper</b>	The Looper object whose message queue will be used to implement the callback mechanism, or null to make callbacks on the calling thread.
<b>request</b>	The location request for the updates.						
<b>callback</b>	The callback for the location updates.						
<b>looper</b>	The Looper object whose message queue will be used to implement the callback mechanism, or null to make callbacks on the calling thread.						

## Exhibit D - Claim Chart for U.S. Patent No. 9,467,838 Against HTC

US9467838	Exemplary Supporting Evidence Regarding Accused Products				
	<p data-bbox="556 240 1766 321">public Task&lt;Void&gt; requestLocationUpdates (LocationRequest request, PendingIntent callbackIntent)</p> <p data-bbox="556 354 1291 378">Requests location updates with a callback on the specified PendingIntent.</p> <p data-bbox="556 410 1759 540">This method is suited for the background use cases, more specifically for receiving location updates, even when the app has been killed by the system. In order to do so, use a PendingIntent for a started service. For foreground use cases, the LocationCallback version of the method is recommended, see requestLocationUpdates(LocationRequest, LocationCallback, Looper).</p> <p data-bbox="556 573 1749 630">Any previously registered requests that have the same PendingIntent (as defined by equals(Object)) will be replaced by this request.</p> <p data-bbox="556 662 1755 751">Both LocationResult and LocationAvailability are sent to the given PendingIntent. You can extract data from an Intent using hasResult(Intent), extractResult(Intent), hasLocationAvailability(Intent), and extractLocationAvailability(Intent).</p> <p data-bbox="569 784 695 808"><b>Parameters</b></p> <table border="1" data-bbox="556 833 1766 971"> <tbody> <tr> <td data-bbox="556 833 863 906">request</td> <td data-bbox="863 833 1766 906">The location request for the updates.</td> </tr> <tr> <td data-bbox="556 906 863 971">callbackIntent</td> <td data-bbox="863 906 1766 971">A pending intent to be sent for each location update.</td> </tr> </tbody> </table> <p data-bbox="569 995 653 1019"><b>Returns</b></p> <ul data-bbox="577 1044 1381 1068" style="list-style-type: none"> <li>• a Task for the call, check isSuccessful() to determine if it was successful.</li> </ul> <p data-bbox="535 1084 1906 1141"><a href="https://developers.google.com/android/reference/com/google/android/gms/location/FusedLocationProviderClient">https://developers.google.com/android/reference/com/google/android/gms/location/FusedLocationProviderClient</a></p>	request	The location request for the updates.	callbackIntent	A pending intent to be sent for each location update.
request	The location request for the updates.				
callbackIntent	A pending intent to be sent for each location update.				

**Exhibit D - Claim Chart for U.S. Patent No. 9,467,838 Against HTC**

US9467838	Exemplary Supporting Evidence Regarding Accused Products						
	<p><code>public void onLocationAvailability (LocationAvailability locationAvailability)</code></p> <p>Called when there is a change in the availability of location data.</p> <p>When <code>isLocationAvailable()</code> returns <code>false</code> you can assume that location will not be returned in <code>onLocationResult(LocationResult)</code> until something changes in the device's settings or environment. Even when <code>isLocationAvailable()</code> returns <code>true</code> the <code>onLocationResult(LocationResult)</code> may not always be called regularly, however the device location is known and both the most recently delivered location and <code>getLastLocation(GoogleApiClient)</code> will be reasonably up to date given the hints specified by the active <code>LocationRequest</code>s.</p> <p><b>Parameters</b></p> <table border="1" data-bbox="552 639 1766 704"> <tr> <td><code>locationAvailability</code></td> <td>The current status of location availability.</td> </tr> </table> <p><code>public void onLocationResult (LocationResult result)</code></p> <p>Called when device location information is available.</p> <p>The most recent location returned by <code>getLastLocation()</code> is not guaranteed to be immediately fresh, but will be reasonably up to date given the hints specified by the active <code>LocationRequest</code>s.</p> <p><b>Parameters</b></p> <table border="1" data-bbox="552 1019 1766 1084"> <tr> <td><code>result</code></td> <td>The latest location result available.</td> </tr> </table> <p><a href="https://developers.google.com/android/reference/com/google/android/gms/location/LocationCallback">https://developers.google.com/android/reference/com/google/android/gms/location/LocationCallback</a></p> <p><code>public abstract void onLocationChanged (Location location)</code></p> <p>Called when the location has changed.</p> <p><b>Parameters</b></p> <table border="1" data-bbox="552 1318 1766 1383"> <tr> <td><code>location</code></td> <td>The updated location.</td> </tr> </table>	<code>locationAvailability</code>	The current status of location availability.	<code>result</code>	The latest location result available.	<code>location</code>	The updated location.
<code>locationAvailability</code>	The current status of location availability.						
<code>result</code>	The latest location result available.						
<code>location</code>	The updated location.						



## Exhibit D - Claim Chart for U.S. Patent No. 9,467,838 Against HTC

US9467838	Exemplary Supporting Evidence Regarding Accused Products
	<p data-bbox="541 237 1822 264"><a href="https://developers.google.com/android/reference/com/google/android/gms/location/LocationListener">https://developers.google.com/android/reference/com/google/android/gms/location/LocationListener</a></p> <p data-bbox="541 318 835 345">Public Constructors</p> <hr data-bbox="541 358 1766 363"/> <p data-bbox="552 410 947 438">public <b>MapView</b> (<b>Context</b> context)</p> <p data-bbox="552 508 1150 535">public <b>MapView</b> (<b>Context</b> context, <b>AttributeSet</b> attrs)</p> <p data-bbox="552 605 1289 633">public <b>MapView</b> (<b>Context</b> context, <b>AttributeSet</b> attrs, int defStyle)</p> <p data-bbox="552 703 1268 730">public <b>MapView</b> (<b>Context</b> context, <b>GoogleMapOptions</b> options)</p> <p data-bbox="541 751 1696 779"><a href="https://developers.google.com/android/reference/com/google/android/gms/maps/MapView">https://developers.google.com/android/reference/com/google/android/gms/maps/MapView</a></p>

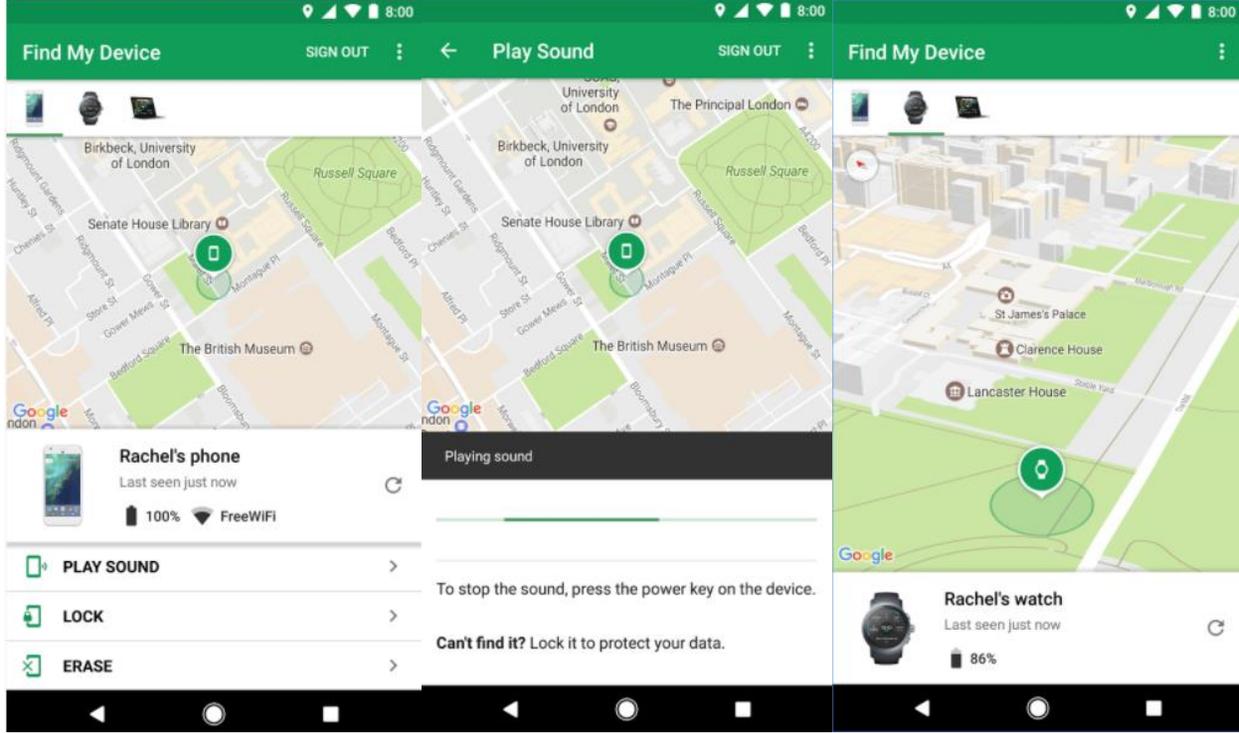
**Exhibit D - Claim Chart for U.S. Patent No. 9,467,838 Against HTC**

US9467838	Exemplary Supporting Evidence Regarding Accused Products		
	<p><code>public void <b>getMapAsync</b> (<b>OnMapReadyCallback</b> callback)</code></p> <p>Returns a non-null instance of the <code>GoogleMap</code>, ready to be used.</p> <p>Note that:</p> <ul style="list-style-type: none"> <li>• This method must be called from the main thread.</li> <li>• The callback will be executed in the main thread.</li> <li>• In the case where Google Play services is not installed on the user's device, the callback will not be triggered until the user installs it.</li> <li>• The <code>GoogleMap</code> object provided by the callback is non-null.</li> </ul> <p><b>Parameters</b></p> <table border="1" data-bbox="554 686 1761 753"> <tr> <td><code>callback</code></td> <td>The callback object that will be triggered when the map is ready to be used.</td> </tr> </table> <p><code>public final void <b>onCreate</b> (<b>Bundle</b> savedInstanceState)</code></p> <p>You must call this method from the parent Activity/Fragment's corresponding method.</p> <p><a href="https://developers.google.com/android/reference/com/google/android/gms/maps/MapView">https://developers.google.com/android/reference/com/google/android/gms/maps/MapView</a></p>	<code>callback</code>	The callback object that will be triggered when the map is ready to be used.
<code>callback</code>	The callback object that will be triggered when the map is ready to be used.		
<p>[1D] presenting, via an interactive display of the first device, a first interactive, georeferenced map and a first set of one or more user-selectable symbols corresponding to a first set of one or more of the second devices, wherein the first set of symbols are positioned on the</p>	<p>HTC infringes directly and/or indirectly by performing, inducing others to perform, and/or contributing to the performance of: presenting, via an interactive display of the first device, a first interactive, georeferenced map and a first set of one or more user-selectable symbols corresponding to a first set of one or more of the second devices, wherein the first set of symbols are positioned on the first georeferenced map at respective positions corresponding to the locations of the first set of second devices, and wherein first georeferenced map data relate positions on the first georeferenced map to spatial coordinates.</p> <p><b><u>Regarding Find My Device</u></b> and Android Device Manager, the Accused Products display, to the user on the display of the first device, a map with one or more symbols corresponding to one or more second devices. The map is interactive because the user may control the display of the map, e.g., pan, zoom, and/or effect change to the map in an otherwise interactive manner. The map is georeferenced for at least the reason that one or more symbols are associated with spatial locations, i.e., coordinates. The symbols</p>		

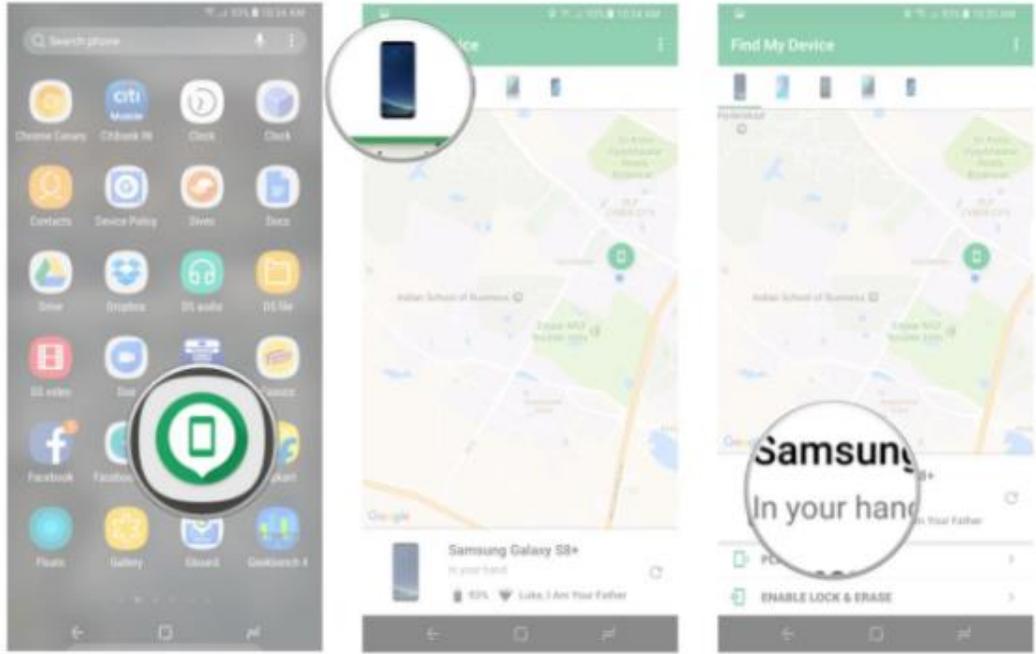
**Exhibit D - Claim Chart for U.S. Patent No. 9,467,838 Against HTC**

<b>US9467838</b>	<b>Exemplary Supporting Evidence Regarding Accused Products</b>
<p>first georeferenced map at respective positions corresponding to the locations of the first set of second devices, and wherein first georeferenced map data relate positions on the first georeferenced map to spatial coordinates;</p>	<p>are user-selectable because a user may touch the display to select the device associated with the symbol. For example, the user may input a touch selection directly on or near the portion of the display corresponding to the symbol's coordinates on the map to effect a selection of the device.</p> <p><b><u>Regarding Google Maps.</u></b> Google Latitude, Google Plus, Google Hangouts, Google Messages, Google Allo, Google Duo, Google Chrome, and Android Messenger, the Accused Products display, to the user on the display of the first device, a map with one or more symbols corresponding to one or more second users (or second devices corresponding to the second users). The map is interactive because the user may control the display of the map, e.g., pan, zoom, and/or effect change to the map in an otherwise interactive manner. The map is georeferenced for at least the reason that one or more symbols are associated with spatial locations, i.e., coordinates. The symbols are user-selectable because a user may touch the display to select the user or device associated with the symbol. For example, the user may input a touch selection directly on or near the portion of the display corresponding to the symbol's coordinates on the map to effect a selection of the user or device.</p> <p><b><u>Exemplary Support for Find My Device:</u></b></p>

**Exhibit D - Claim Chart for U.S. Patent No. 9,467,838 Against HTC**

US9467838	Exemplary Supporting Evidence Regarding Accused Products
	 <p><a href="https://play.google.com/store/apps/details?id=com.google.android.apps.adm&amp;hl=en">https://play.google.com/store/apps/details?id=com.google.android.apps.adm&amp;hl=en</a></p>

**Exhibit D - Claim Chart for U.S. Patent No. 9,467,838 Against HTC**

US9467838	Exemplary Supporting Evidence Regarding Accused Products
	<p>Once you're signed in to Find My Device, you'll see a map with your current location as well as the make and model of your phone, and two options — Play Sound, and Enable Lock &amp; Erase. Hitting the latter option will allow you to start using the Lock and Erase functions.</p> <p>If you've signed into more than one phone, you can select a particular device by browsing the list at the top of the screen.</p> <ol style="list-style-type: none"> <li>1. Open <b>Find My Device</b> from your home screen or app drawer.</li> <li>2. Select your phone from the list of devices at the top of the screen.</li> <li>3. See if your phone is discoverable.</li> </ol>  <p><a href="https://www.androidcentral.com/find-my-device">https://www.androidcentral.com/find-my-device</a></p>

**Exhibit D - Claim Chart for U.S. Patent No. 9,467,838 Against HTC**

US9467838	Exemplary Supporting Evidence Regarding Accused Products
	<h2 data-bbox="548 240 1686 289">How to locate your phone over the internet</h2> <p data-bbox="548 337 1692 475">If you've lost your phone, you can remotely locate it through the <a href="#">Find My Device website</a>. You'll need to sign in to the Google account that was used to set up Find My Device. It takes a few seconds, but the service should be able to track your phone. Alternatively, you can also do a Google search for "find my phone" to locate your handset.</p> <ol data-bbox="541 532 1031 686" style="list-style-type: none"><li>1. Head to the <a href="#">Find My Device website</a>.</li><li>2. Sign in to your Google account.</li><li>3. Check if your device is visible.</li></ol> <div data-bbox="585 727 1793 1182"></div> <p data-bbox="537 1214 1157 1247"><a href="https://www.androidcentral.com/find-my-device">https://www.androidcentral.com/find-my-device</a></p>

**Exhibit D - Claim Chart for U.S. Patent No. 9,467,838 Against HTC**

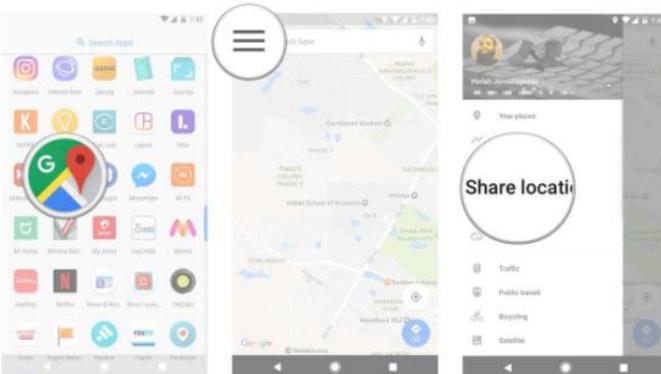
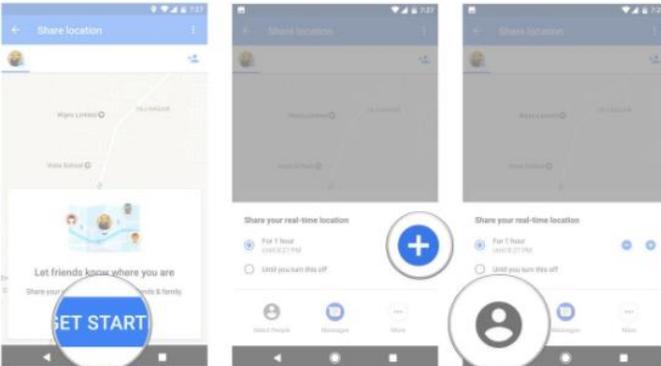
<b>US9467838</b>	<b>Exemplary Supporting Evidence Regarding Accused Products</b>
	<u><b>Exemplary Support for Google Maps:</b></u>

**Exhibit D - Claim Chart for U.S. Patent No. 9,467,838 Against HTC**

US9467838	Exemplary Supporting Evidence Regarding Accused Products
	<p data-bbox="556 253 863 297"><b>Share your E.T.A</b></p> <p data-bbox="556 321 1686 345">After you start your drive, you can share your destination, estimated arrival time, and where you are on the route.</p> <ol data-bbox="556 378 1381 618" style="list-style-type: none"> <li data-bbox="556 378 909 402">1. Open the Google Maps app .</li> <li data-bbox="556 418 1182 443">2. Set a driving destination. <a href="#">Learn how to navigate to a place.</a></li> <li data-bbox="556 459 1224 483">3. After you start navigation, tap More ^ &gt; <b>Share trip progress.</b></li> <li data-bbox="556 500 898 524">4. Choose a person from the list.</li> <li data-bbox="556 540 699 565">5. Tap <b>Share.</b></li> <li data-bbox="556 581 1381 605">6. Location Sharing will stop when you reach your destination or stop navigating.</li> </ol> <ul data-bbox="556 646 1224 670" style="list-style-type: none"> <li data-bbox="556 646 1224 670">• To stop sharing before you arrive, tap More ^ &gt; <b>Stop sharing.</b></li> </ul> <p data-bbox="556 740 978 784"><b>See where someone is</b></p> <p data-bbox="556 808 1287 833">If someone shares their location with you, you can see them on the map.</p> <ol data-bbox="556 865 940 971" style="list-style-type: none"> <li data-bbox="556 865 909 889">1. Open the Google Maps app .</li> <li data-bbox="556 906 940 930">2. Tap Menu ≡ &gt; <b>Location sharing.</b></li> <li data-bbox="556 946 772 971">3. Choose someone.</li> </ol> <ul data-bbox="556 1011 1325 1036" style="list-style-type: none"> <li data-bbox="556 1011 1325 1036">• To see an updated location, tap on a friend's icon &gt; More ⋮ &gt; <b>Refresh.</b></li> </ul> <p data-bbox="556 1084 1056 1128"><b>Stop seeing someone's location</b></p> <ol data-bbox="556 1144 1482 1295" style="list-style-type: none"> <li data-bbox="556 1144 909 1169">1. Open the Google Maps app .</li> <li data-bbox="556 1185 856 1209">2. On the map, tap their icon.</li> <li data-bbox="556 1226 877 1250">3. At the bottom, tap More ^ .</li> <li data-bbox="556 1266 1482 1295">4. To temporarily hide someone, tap <b>Hide from map.</b> You can stop hiding them at any time.</li> </ol> <p data-bbox="556 1328 1766 1352"><b>Note:</b> You can stop someone's location from ever appearing on your map. <a href="#">Learn how to block another person's account.</a></p> <p data-bbox="535 1377 1724 1406"><a href="https://support.google.com/maps/answer/7326816?co=GENIE.Platform%3DAndroid&amp;oco=1">https://support.google.com/maps/answer/7326816?co=GENIE.Platform%3DAndroid&amp;oco=1</a></p>



**Exhibit D - Claim Chart for U.S. Patent No. 9,467,838 Against HTC**

US9467838	Exemplary Supporting Evidence Regarding Accused Products
	<p data-bbox="541 272 1176 308"><b>How to share your location in Google Maps</b></p> <ol data-bbox="541 337 1155 422" style="list-style-type: none"> <li>1. Open Google Maps from the app drawer or the home screen.</li> <li>2. Tap the <b>hamburger menu</b> (the three horizontal lines) on the top left corner of the screen.</li> <li>3. Select <b>Share location</b>.</li> </ol>  <ol data-bbox="541 860 1186 966" style="list-style-type: none"> <li>4. Tap <b>Get Started</b>.</li> <li>5. Use the <b>+</b> icon to select a time period or select the <b>Until you turn this off</b> setting to share your location indefinitely.</li> <li>6. Tap <b>Select People</b>.</li> </ol>  <p data-bbox="541 1372 1375 1404"><a href="https://www.androidcentral.com/how-share-location-google-maps">https://www.androidcentral.com/how-share-location-google-maps</a></p>

**Exhibit D - Claim Chart for U.S. Patent No. 9,467,838 Against HTC**

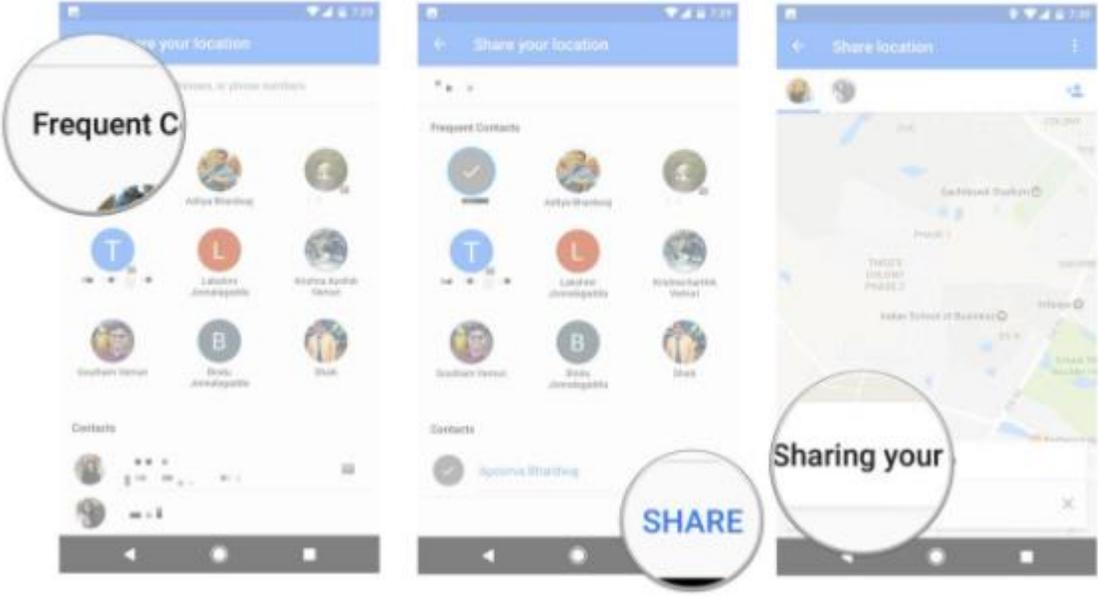
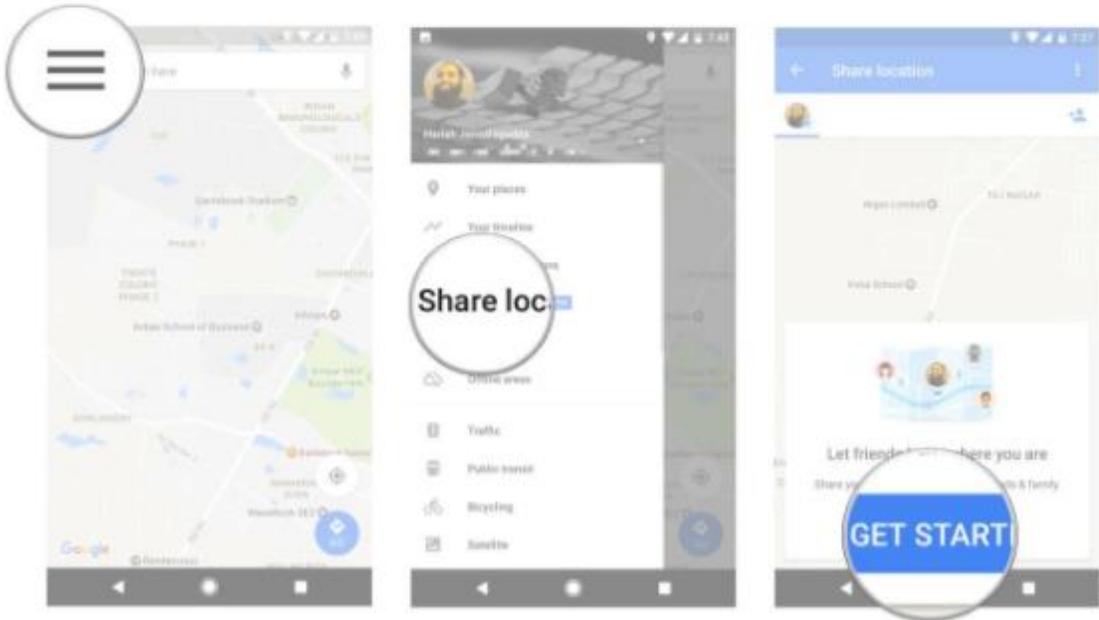
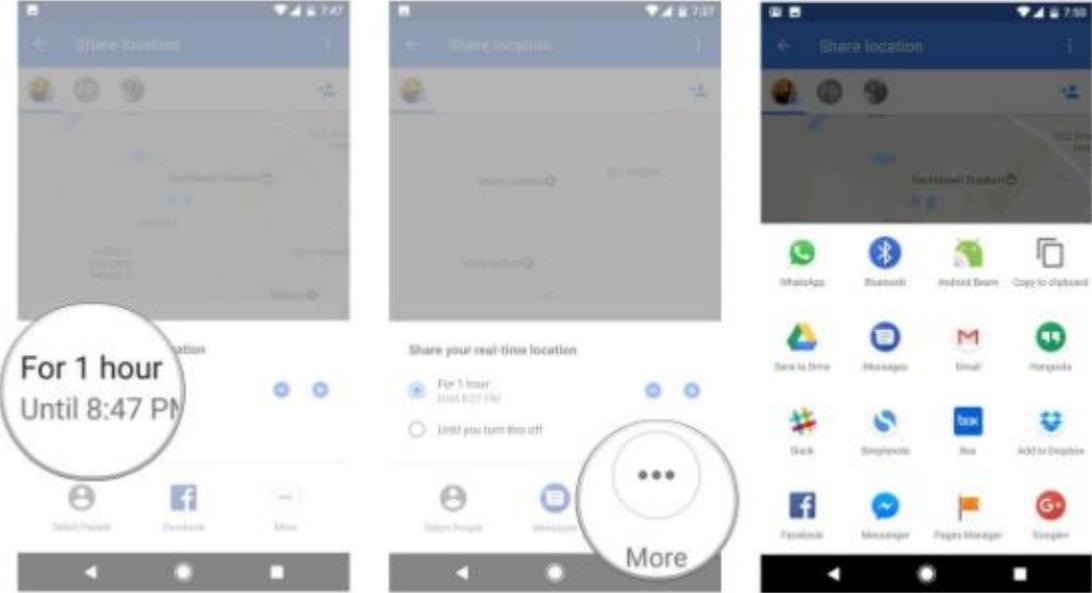
US9467838	Exemplary Supporting Evidence Regarding Accused Products
	<p data-bbox="552 289 1602 462">7. You'll see a list of your frequent contacts at the top, along with a full list of contacts. Pick the contacts by tapping their name.</p> <p data-bbox="552 375 1482 402">8. Once you've selected the contacts you want to share your location to, tap Share.</p> <p data-bbox="552 435 1442 462">9. You'll see a message saying that the selected contact can view your location.</p>  <p data-bbox="535 1138 1381 1170"><a href="https://www.androidcentral.com/how-share-location-google-maps">https://www.androidcentral.com/how-share-location-google-maps</a></p>

Exhibit D - Claim Chart for U.S. Patent No. 9,467,838 Against HTC

US9467838	Exemplary Supporting Evidence Regarding Accused Products
	<h2 data-bbox="548 240 1281 289">How to create a shareable link</h2> <p data-bbox="548 329 1486 358">You can also create a link and use it to share your location easily. Here's how to do it:</p> <ol data-bbox="541 407 1260 548" style="list-style-type: none"><li>1. Tap the hamburger menu on the top left corner of the screen.</li><li>2. Select Share location.</li><li>3. Tap Get Started.</li></ol>  <p data-bbox="533 1230 1381 1263"><a href="https://www.androidcentral.com/how-share-location-google-maps">https://www.androidcentral.com/how-share-location-google-maps</a></p>

**Exhibit D - Claim Chart for U.S. Patent No. 9,467,838 Against HTC**

US9467838	Exemplary Supporting Evidence Regarding Accused Products
	<p>4. Select the amount of time you want to share your location.</p> <p>5. Tap More.</p> <p>6. Select your app of choice to create and send a unique URL that broadcasts your current location. You can email it, send the link via Messenger, or even tweet it to the intended recipient.</p>  <p><a href="https://www.androidcentral.com/how-share-location-google-maps">https://www.androidcentral.com/how-share-location-google-maps</a></p>

**Exhibit D - Claim Chart for U.S. Patent No. 9,467,838 Against HTC**

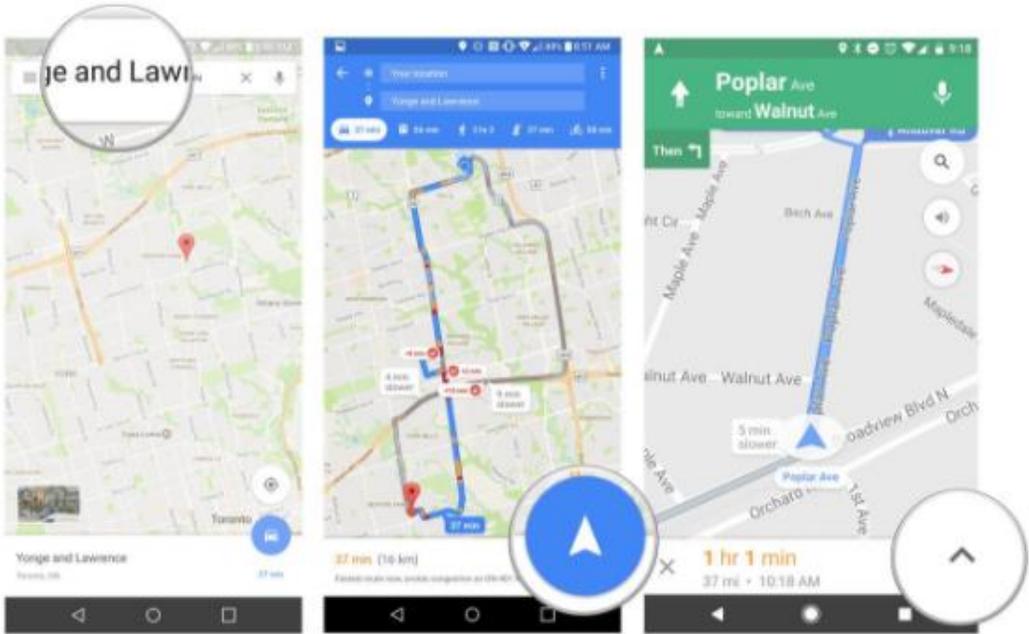
US9467838	Exemplary Supporting Evidence Regarding Accused Products
	<h2 data-bbox="550 240 1451 337">How to share your navigation directions while you walk, drive or transit</h2> <p data-bbox="550 375 1577 464">One of the best ways to share your location with someone is during a drive, walk or transit. If you're meeting a friend or family member somewhere, or navigating towards their home, Google Maps lets you share your location with them for the duration of the trip. It's magic!</p> <ol data-bbox="550 513 1419 646" style="list-style-type: none"><li>1. In the <b>search bar</b> enter your destination.</li><li>2. Pick your navigation type (drive, transit, walk) and press the <b>blue navigate button</b>.</li><li>3. Tap the arrow next to the <b>time-to-destination</b> number at the bottom of the screen.</li></ol>  <p data-bbox="535 1328 1381 1360"><a href="https://www.androidcentral.com/how-share-location-google-maps">https://www.androidcentral.com/how-share-location-google-maps</a></p>

Exhibit D - Claim Chart for U.S. Patent No. 9,467,838 Against HTC

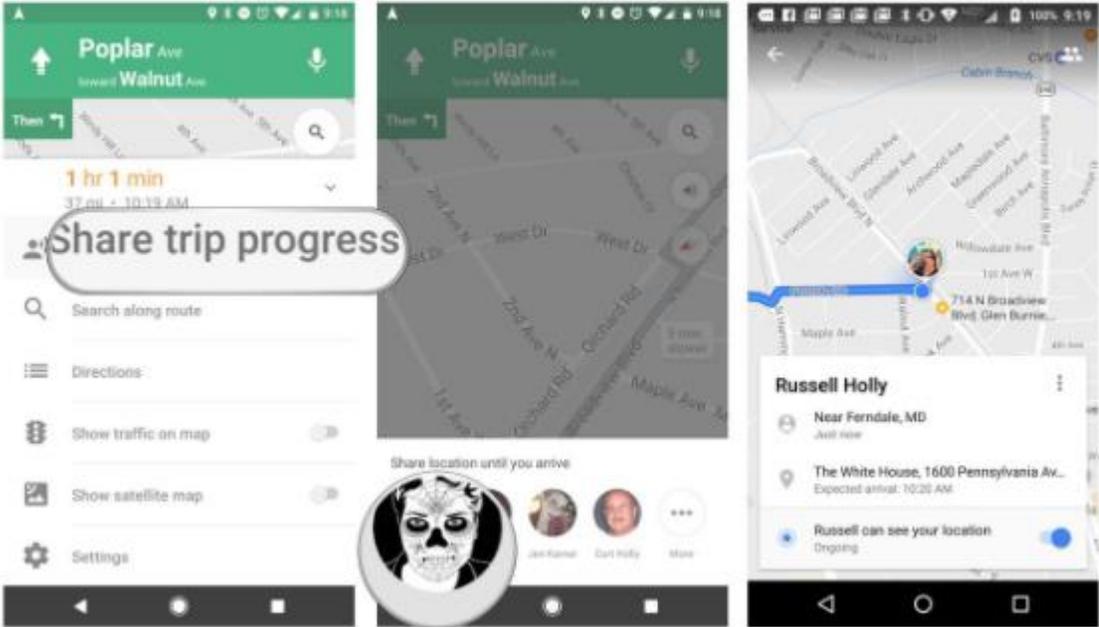
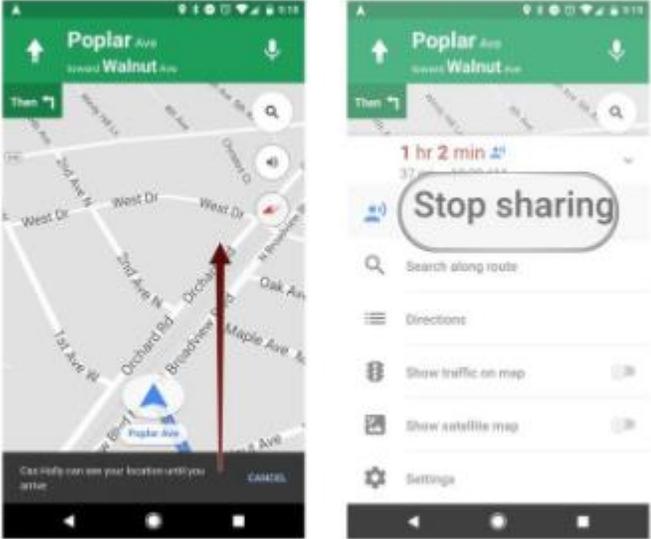
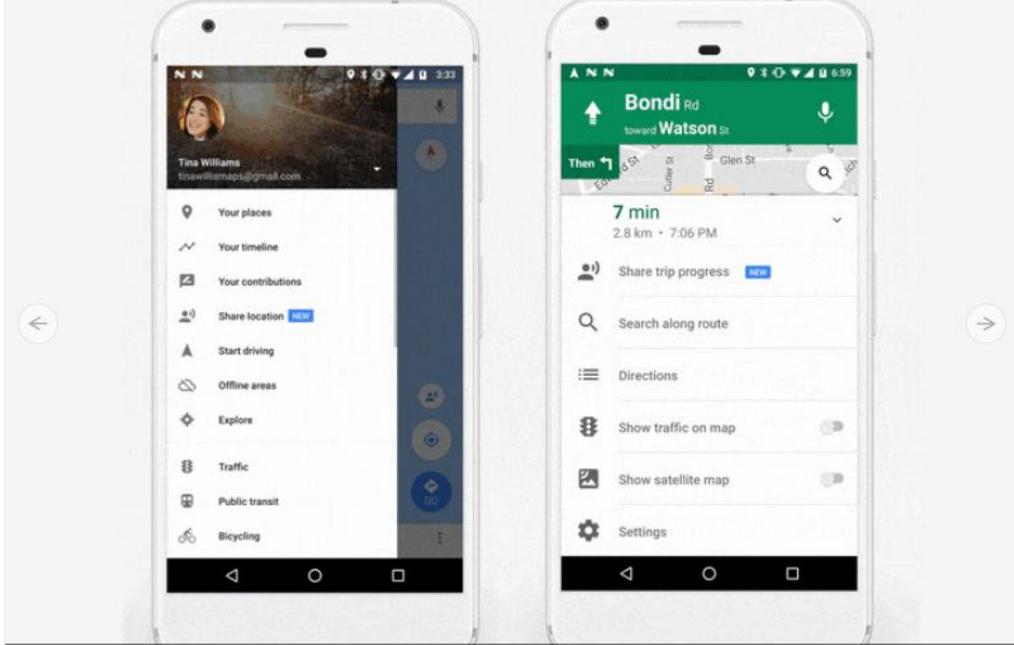
US9467838	Exemplary Supporting Evidence Regarding Accused Products
	<p data-bbox="552 277 861 305">4. Tap Share trip progress.</p> <p data-bbox="552 332 1171 360">5. Choose one or more contacts to share trip progress.</p>  <p data-bbox="552 1063 1365 1091">You can also stop sharing your location with someone before a trip ends.</p> <p data-bbox="535 1101 1381 1128"><a href="https://www.androidcentral.com/how-share-location-google-maps">https://www.androidcentral.com/how-share-location-google-maps</a></p>

Exhibit D - Claim Chart for U.S. Patent No. 9,467,838 Against HTC

US9467838	Exemplary Supporting Evidence Regarding Accused Products
	<ol style="list-style-type: none"><li data-bbox="562 245 1493 272">1. Tap the arrow next to the time-to-destination number at the bottom of the screen.</li><li data-bbox="562 302 793 329">2. Tap Stop sharing.</li></ol> <div data-bbox="772 383 1423 922"></div> <p data-bbox="569 976 659 1003">That's It!</p> <p data-bbox="569 1045 1633 1073">Are you excited that location sharing is back in Google Maps? How often do you use the feature?</p> <p data-bbox="537 1084 1381 1117"><a href="https://www.androidcentral.com/how-share-location-google-maps">https://www.androidcentral.com/how-share-location-google-maps</a></p>

**Exhibit D - Claim Chart for U.S. Patent No. 9,467,838 Against HTC**

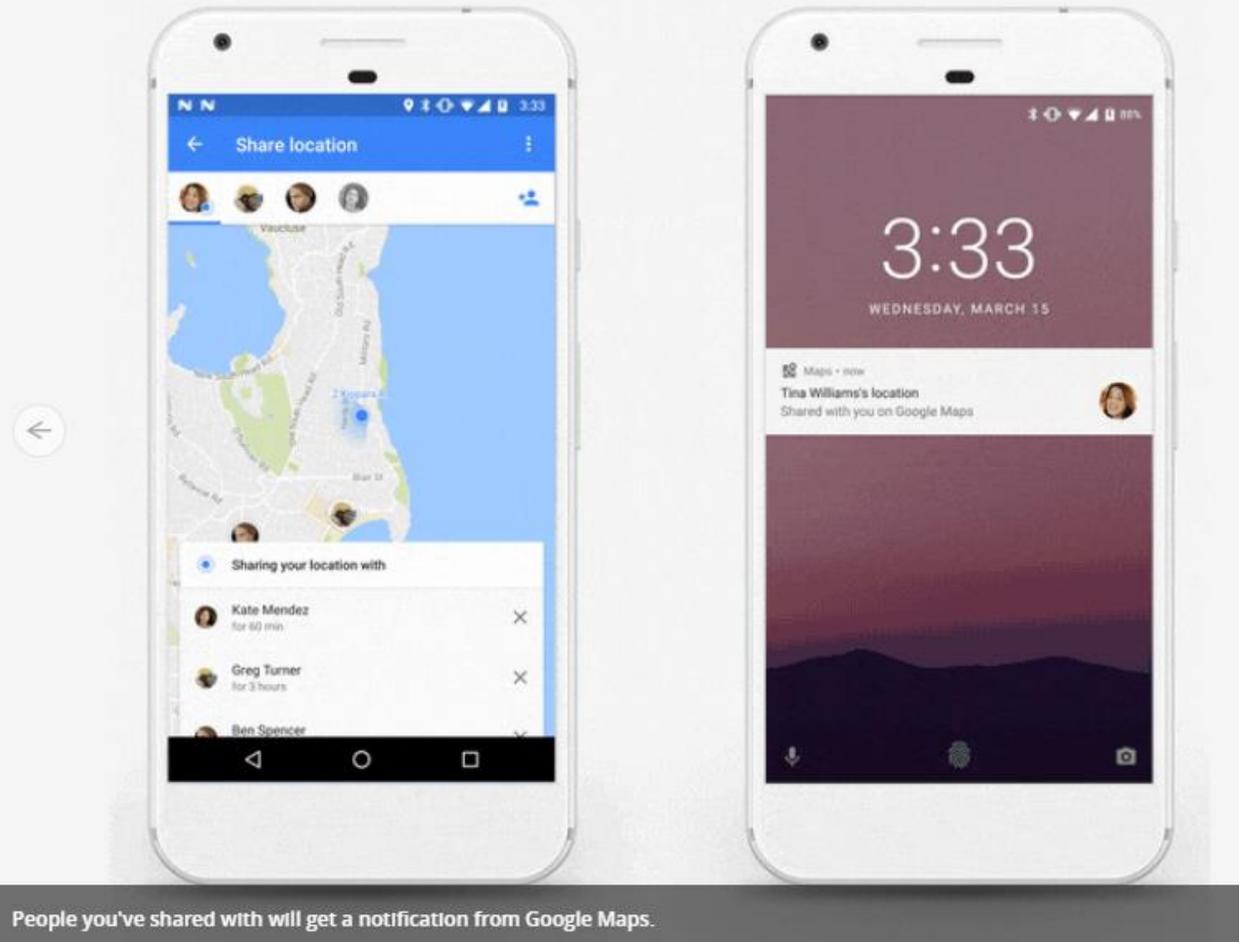
US9467838	Exemplary Supporting Evidence Regarding Accused Products
	 <p>The image displays two side-by-side smartphone screens from Google Maps. The left screen shows the navigation drawer with the 'Share location' option highlighted in blue. The right screen shows the navigation menu with the 'Share trip progress' option highlighted in blue. Both screens show a map in the background with a location marker. Below the screens is a caption: 'Ways to start location sharing: (left) from the app navigation drawer, and (right) from the menu, you can drag up while navigating somewhere.'</p> <p><a href="https://arstechnica.com/gadgets/2017/03/location-sharing-finally-returns-to-google-maps/">https://arstechnica.com/gadgets/2017/03/location-sharing-finally-returns-to-google-maps/</a></p>



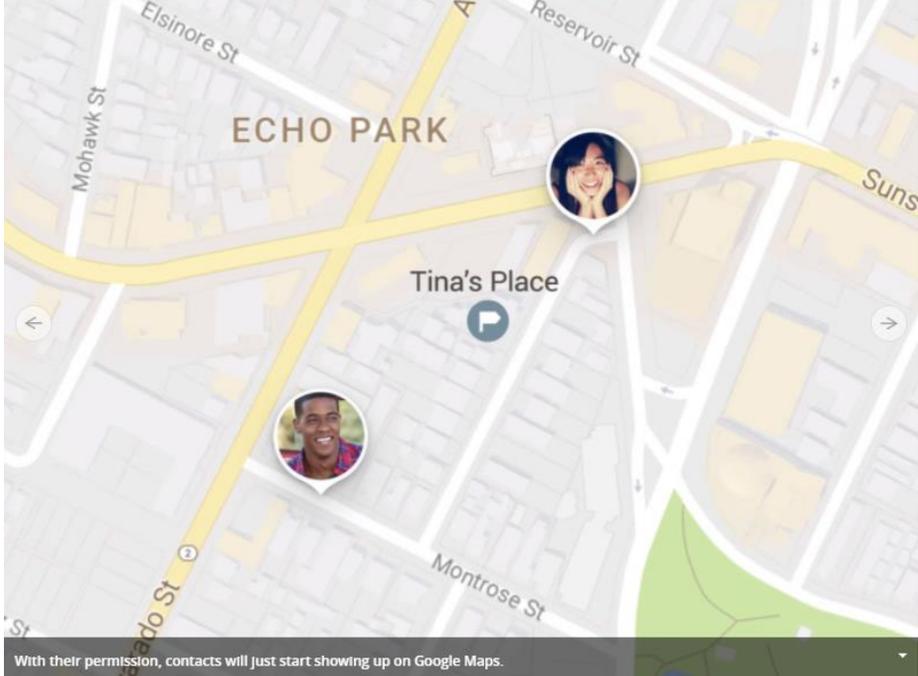
**Exhibit D - Claim Chart for U.S. Patent No. 9,467,838 Against HTC**

US9467838	Exemplary Supporting Evidence Regarding Accused Products
	<p data-bbox="535 1023 1680 1063">You can pick how long you want to share your location for, and with whom you want to share it.</p> <p data-bbox="535 1063 1680 1096"><a href="https://arstechnica.com/gadgets/2017/03/location-sharing-finally-returns-to-google-maps/">https://arstechnica.com/gadgets/2017/03/location-sharing-finally-returns-to-google-maps/</a></p>

**Exhibit D - Claim Chart for U.S. Patent No. 9,467,838 Against HTC**

US9467838	Exemplary Supporting Evidence Regarding Accused Products
	 <p data-bbox="541 1144 1197 1177">People you've shared with will get a notification from Google Maps.</p> <p data-bbox="541 1188 1680 1221"><a href="https://arstechnica.com/gadgets/2017/03/location-sharing-finally-returns-to-google-maps/">https://arstechnica.com/gadgets/2017/03/location-sharing-finally-returns-to-google-maps/</a></p>

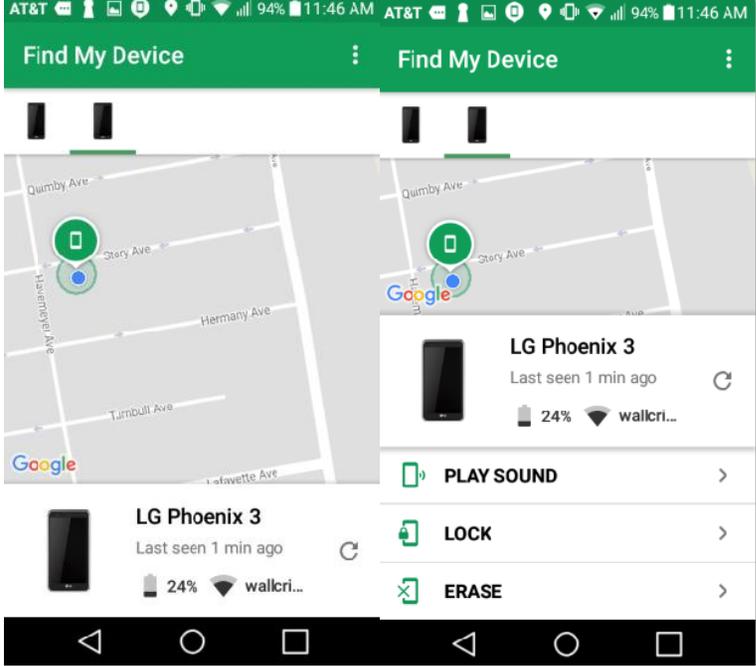
**Exhibit D - Claim Chart for U.S. Patent No. 9,467,838 Against HTC**

US9467838	Exemplary Supporting Evidence Regarding Accused Products
	 <p data-bbox="535 912 1453 917">With their permission, contacts will just start showing up on Google Maps.</p> <p data-bbox="535 919 1680 950"><a href="https://arstechnica.com/gadgets/2017/03/location-sharing-finally-returns-to-google-maps/">https://arstechnica.com/gadgets/2017/03/location-sharing-finally-returns-to-google-maps/</a></p> <p data-bbox="535 1023 1092 1055"><b><u>Exemplary Find My Device Screenshots:</u></b></p>

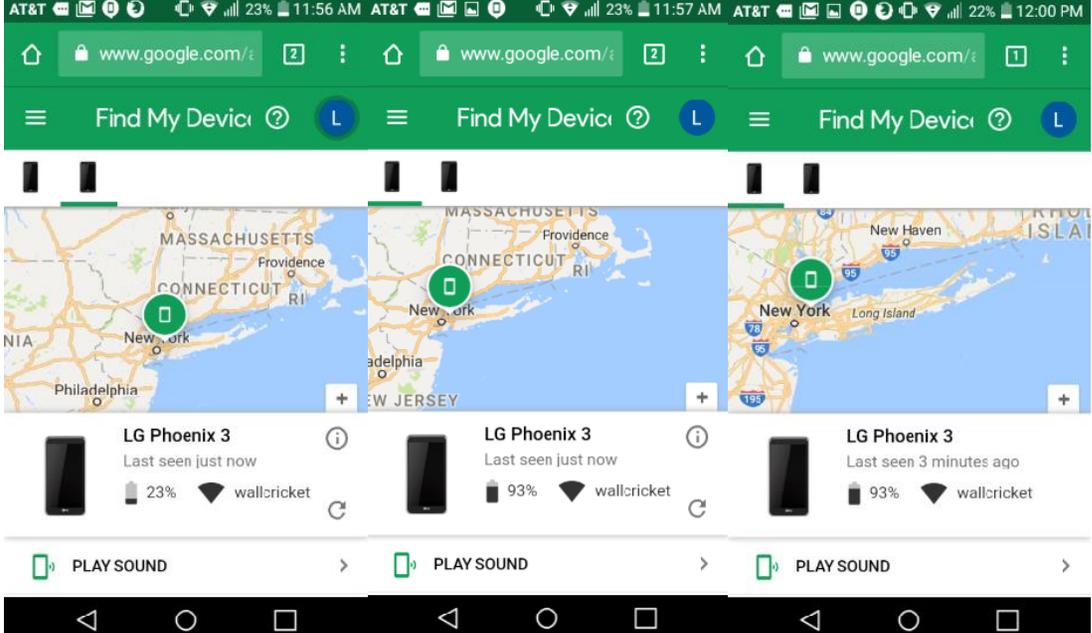
**Exhibit D - Claim Chart for U.S. Patent No. 9,467,838 Against HTC**

US9467838	Exemplary Supporting Evidence Regarding Accused Products
	<p>The evidence consists of three sequential screenshots of the 'Find My Device' application on an LG Phoenix 3. Each screenshot shows the top status bar with AT&amp;T service, 94% battery, and the time 11:45 AM or 11:46 AM. The app header is green with the text 'Find My Device' and a menu icon. Below the header are two black bars representing the phone's physical home and power buttons. The main content area features a Google Maps interface. The first screenshot shows a street-level view of a city street with a green location pin. The second screenshot shows a regional map of Connecticut and Rhode Island with a green location pin. The third screenshot shows a menu with three options: 'PLAY SOUND', 'LOCK', and 'ERASE', each with a right-pointing arrow. Below the menu, there are two device status cards for 'LG Phoenix 3', each showing 'In your hand', a battery icon at 94%, and a Wi-Fi icon labeled 'walkcri...'. The bottom of each screenshot shows the standard Android navigation bar with back, home, and recent apps buttons.</p>

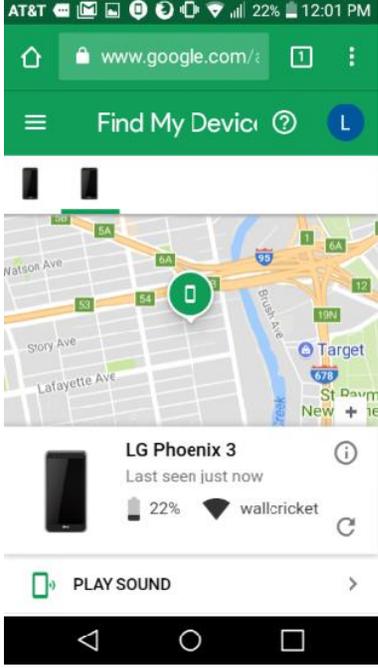
**Exhibit D - Claim Chart for U.S. Patent No. 9,467,838 Against HTC**

US9467838	Exemplary Supporting Evidence Regarding Accused Products
	 <p>The image displays two side-by-side screenshots of an Android phone's "Find My Device" application. The top status bar shows "AT&amp;T" and a battery level of 94% at 11:46 AM. The app's header is green with the text "Find My Device" and a menu icon. Below the header, there are two identical map views. The left map shows a green location pin on a street grid with labels like "Quimby Ave", "Story Ave", "Hermany Ave", "Turnbull Ave", and "Lafayette Ave". The right map shows the same location but with a detailed information card for an "LG Phoenix 3" device. The card includes a small phone icon, the text "Last seen 1 min ago", a refresh icon, a battery icon at 24%, and a Wi-Fi icon labeled "walkri...". Below the card are three action buttons: "PLAY SOUND", "LOCK", and "ERASE", each with a right-pointing arrow. The bottom of the screen shows the standard Android navigation bar with back, home, and recent apps buttons.</p>

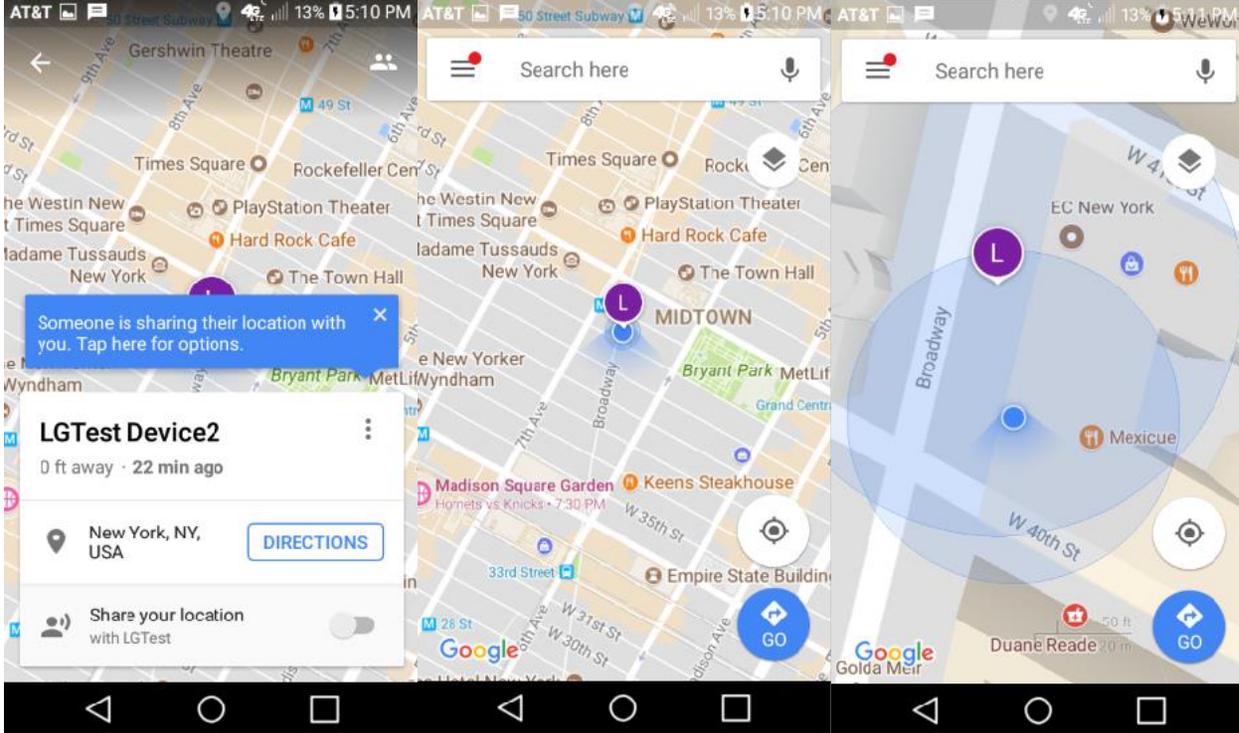
**Exhibit D - Claim Chart for U.S. Patent No. 9,467,838 Against HTC**

US9467838	Exemplary Supporting Evidence Regarding Accused Products
	 <p>The image displays three sequential screenshots of an Android phone's 'Find My Device' application. Each screenshot shows a map with a green location pin and a device information card below it. The device is identified as an 'LG Phoenix 3'. The first screenshot shows the device in New York, NY, with a battery level of 23% and the carrier 'wallcricnet'. The second screenshot shows the device in New York, NY, with a battery level of 93% and the carrier 'wallcricnet'. The third screenshot shows the device in New York, NY, with a battery level of 93% and the carrier 'wallcricnet'. The screenshots are taken at different times: 11:56 AM, 11:57 AM, and 12:00 PM. The status bar at the top of each screenshot shows the AT&amp;T carrier, signal strength, Wi-Fi, and battery icons. The bottom of each screenshot shows the Android navigation bar with back, home, and recent apps buttons.</p>

**Exhibit D - Claim Chart for U.S. Patent No. 9,467,838 Against HTC**

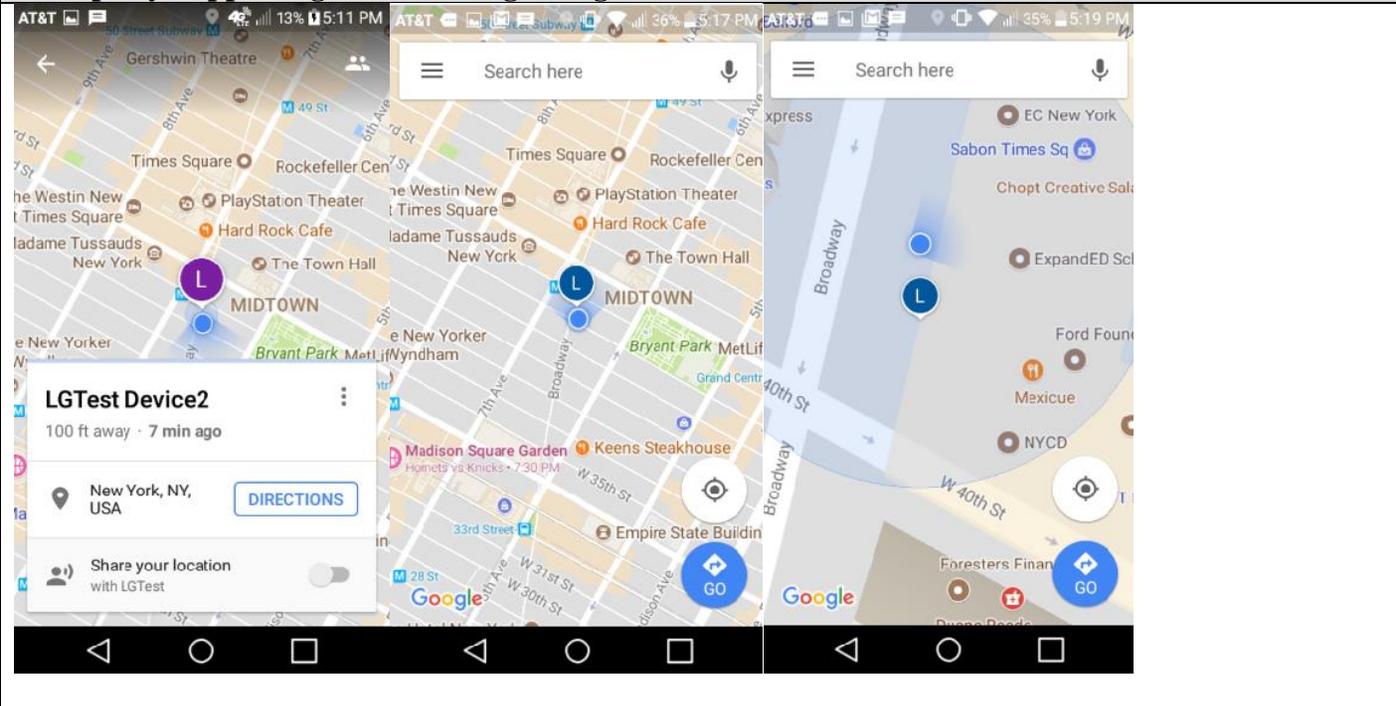
US9467838	Exemplary Supporting Evidence Regarding Accused Products
	 <p>The screenshot shows the Google Maps interface on a mobile device. At the top, the status bar displays 'AT&amp;T', signal strength, Wi-Fi, battery at 22%, and time 12:01 PM. Below the status bar is the address bar with 'www.google.com/'. The main heading is 'Find My Device' with a question mark and a user profile icon 'L'. The map shows a street grid with a green location pin on a street labeled 'Story Ave'. Below the map, a card for 'LG Phoenix 3' is displayed, indicating it was 'Last seen just now' with 22% battery and connected to 'wallcricket'. A 'PLAY SOUND' button is visible below the card. The bottom navigation bar shows the standard Android back, home, and recents buttons.</p> <p><u><b>Exemplary Google Maps Screenshots:</b></u></p>

**Exhibit D - Claim Chart for U.S. Patent No. 9,467,838 Against HTC**

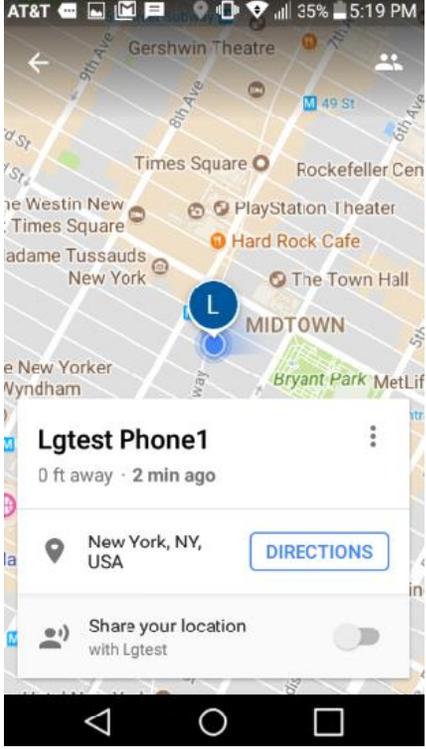
US9467838	Exemplary Supporting Evidence Regarding Accused Products
	 <p>The image displays two side-by-side screenshots of a mobile map application, likely Google Maps, showing a location sharing notification. The notification is for a device named "LGTest Device2" and indicates it is "0 ft away · 22 min ago" in "New York, NY, USA". The notification includes a "DIRECTIONS" button and a "Share your location with LGTest" toggle switch. The background of both screenshots is a map of Midtown Manhattan, New York City, with various landmarks and streets visible. The left screenshot shows the notification in the foreground, while the right screenshot shows the map with a blue location pin and a search bar at the top.</p>



**Exhibit D - Claim Chart for U.S. Patent No. 9,467,838 Against HTC**

US9467838	Exemplary Supporting Evidence Regarding Accused Products
	 <p>The image displays three sequential screenshots of a mobile map application, likely Google Maps, showing a location tracking interface. The screenshots are arranged horizontally, with the first on the left and the third on the right. Each screenshot shows a map of Times Square, New York, with a purple location pin and a blue circular icon labeled 'L' representing the tracked device. The first screenshot shows a notification card for 'LGTest Device2' with the text '100 ft away · 7 min ago', a location of 'New York, NY, USA', and a 'DIRECTIONS' button. The second and third screenshots show the device's position on the map relative to landmarks like Times Square, Rockefeller Center, and Bryant Park. The interface includes a search bar at the top, a compass, and a 'GO' button at the bottom right. The status bar at the top of each screenshot shows the time as 5:11 PM, 5:17 PM, and 5:19 PM respectively, along with battery and signal indicators.</p>

**Exhibit D - Claim Chart for U.S. Patent No. 9,467,838 Against HTC**

US9467838	Exemplary Supporting Evidence Regarding Accused Products
	 <p><b>Exemplary Source Code:</b></p> <p>The above functionality is performed at least in part by the following publicly available source code and/or source code that invokes or is invoked by the following source code (or a substantially similar copy compiled and loaded onto the Accused Products by HTC). AGIS reserves the right to supplement these contentions with additional source code as discovery progresses and as additional source code is made available.</p>

## Exhibit D - Claim Chart for U.S. Patent No. 9,467,838 Against HTC

US9467838	Exemplary Supporting Evidence Regarding Accused Products
	<p data-bbox="556 248 1001 280"><code>public static <a href="#">LocationRequest</a> create ()</code></p> <p data-bbox="546 315 1052 339">Create a location request with default parameters.</p> <p data-bbox="546 370 1661 430">Default parameters are for a block accuracy, slowly updated location. It can then be adjusted as required by the applications before passing to the <a href="#">FusedLocationProviderApi</a>.</p> <p data-bbox="569 454 653 477"><b>Returns</b></p> <ul data-bbox="575 500 835 524" style="list-style-type: none"> <li>• a new location request</li> </ul> <p data-bbox="535 539 1818 571"><a href="https://developers.google.com/android/reference/com/google/android/gms/location/LocationRequest">https://developers.google.com/android/reference/com/google/android/gms/location/LocationRequest</a></p> <p data-bbox="556 592 1297 625"><code>public static final int <b>PRIORITY_BALANCED_POWER_ACCURACY</b></code></p> <p data-bbox="546 656 1199 680">Used with <code>setPriority(int)</code> to request "block" level accuracy.</p> <p data-bbox="546 711 1665 771">Block level accuracy is considered to be about 100 meter accuracy. Using a coarse accuracy such as this often consumes less power.</p> <p data-bbox="546 797 753 821">Constant Value: 102</p> <p data-bbox="556 881 1129 914"><code>public static final int <b>PRIORITY_HIGH_ACCURACY</b></code></p> <p data-bbox="546 945 1358 969">Used with <code>setPriority(int)</code> to request the most accurate locations available.</p> <p data-bbox="546 1000 984 1024">This will return the finest location available.</p> <p data-bbox="546 1052 753 1076">Constant Value: 100</p> <p data-bbox="556 1136 1077 1169"><code>public static final int <b>PRIORITY_LOW_POWER</b></code></p> <p data-bbox="546 1200 1178 1224">Used with <code>setPriority(int)</code> to request "city" level accuracy.</p> <p data-bbox="546 1255 1759 1315">City level accuracy is considered to be about 10km accuracy. Using a coarse accuracy such as this often consumes less power.</p> <p data-bbox="546 1341 753 1365">Constant Value: 104</p> <p data-bbox="535 1385 1818 1417"><a href="https://developers.google.com/android/reference/com/google/android/gms/location/LocationRequest">https://developers.google.com/android/reference/com/google/android/gms/location/LocationRequest</a></p>

## Exhibit D - Claim Chart for U.S. Patent No. 9,467,838 Against HTC

US9467838	Exemplary Supporting Evidence Regarding Accused Products
	<p data-bbox="556 284 1774 321"><code>public Task&lt;Location&gt; getLastLocation ()</code></p> <p data-bbox="556 349 1123 373">Returns the best most recent location currently available.</p> <p data-bbox="556 406 1711 462">If a location is not available, which should happen very rarely, null will be returned. The best accuracy available while respecting the location permissions will be returned.</p> <p data-bbox="556 495 1753 552">This method provides a simplified way to get location. It is particularly well suited for applications that do not require an accurate location and that do not want to maintain extra logic for location updates.</p> <p data-bbox="556 609 1774 646"><code>public Task&lt;LocationAvailability&gt; getLocationAvailability ()</code></p> <p data-bbox="556 673 1711 738">Returns the availability of location data. When <code>isLocationAvailable()</code> returns true, then the location returned by <code>getLastLocation()</code> will be reasonably up to date within the hints specified by the active <code>LocationRequest</code> s.</p> <p data-bbox="556 771 1501 795">If the client isn't connected to Google Play services and the request times out, null is returned.</p> <p data-bbox="556 828 1690 885">Note it's always possible for <code>getLastLocation()</code> to return null even when this method returns true (e.g. location settings were disabled between calls).</p> <p data-bbox="556 901 1900 966"><a href="https://developers.google.com/android/reference/com/google/android/gms/location/FusedLocationProviderClient">https://developers.google.com/android/reference/com/google/android/gms/location/FusedLocationProviderClient</a></p>

**Exhibit D - Claim Chart for U.S. Patent No. 9,467,838 Against HTC**

US9467838	Exemplary Supporting Evidence Regarding Accused Products						
	<p data-bbox="546 240 1774 326"> <code>public Task&lt;Void&gt; requestLocationUpdates (LocationRequest request, LocationCallback callback, Looper looper)</code> </p> <p data-bbox="546 354 1291 380">Requests location updates with a callback on the specified Looper thread.</p> <p data-bbox="546 410 1711 472">This method is suited for the foreground use cases. For background use cases, the <code>PendingIntent</code> version of the method is recommended, see <code>requestLocationUpdates(LocationRequest, PendingIntent)</code>.</p> <p data-bbox="546 503 1396 529">Any previous LocationRequests registered on this LocationListener will be replaced.</p> <p data-bbox="546 560 1711 654">This call will keep the Google Play services connection active, so make sure to call <code>removeLocationUpdates(LocationCallback)</code> when you no longer need it, otherwise you lose the benefits of the automatic connection management.</p> <p data-bbox="546 685 1774 711">Callbacks for <code>LocationCallback</code> will be made on the specified thread, which must already be a prepared looper thread.</p> <p data-bbox="546 735 682 761"><b>Parameters</b></p> <table border="1" data-bbox="546 789 1774 1008"> <tbody> <tr> <td data-bbox="546 789 655 857"><b>request</b></td> <td data-bbox="655 789 1774 857">The location request for the updates.</td> </tr> <tr> <td data-bbox="546 857 655 925"><b>callback</b></td> <td data-bbox="655 857 1774 925">The callback for the location updates.</td> </tr> <tr> <td data-bbox="546 925 655 1008"><b>looper</b></td> <td data-bbox="655 925 1774 1008">The Looper object whose message queue will be used to implement the callback mechanism, or null to make callbacks on the calling thread.</td> </tr> </tbody> </table> <p data-bbox="546 1023 1911 1089"> <a href="https://developers.google.com/android/reference/com/google/android/gms/location/FusedLocationProviderClient">https://developers.google.com/android/reference/com/google/android/gms/location/FusedLocationProviderClient</a> </p>	<b>request</b>	The location request for the updates.	<b>callback</b>	The callback for the location updates.	<b>looper</b>	The Looper object whose message queue will be used to implement the callback mechanism, or null to make callbacks on the calling thread.
<b>request</b>	The location request for the updates.						
<b>callback</b>	The callback for the location updates.						
<b>looper</b>	The Looper object whose message queue will be used to implement the callback mechanism, or null to make callbacks on the calling thread.						

## Exhibit D - Claim Chart for U.S. Patent No. 9,467,838 Against HTC

US9467838	Exemplary Supporting Evidence Regarding Accused Products				
	<p data-bbox="556 240 1766 326">public Task&lt;Void&gt; requestLocationUpdates (LocationRequest request, PendingIntent callbackIntent)</p> <p data-bbox="556 354 1291 378">Requests location updates with a callback on the specified PendingIntent.</p> <p data-bbox="556 410 1755 537">This method is suited for the background use cases, more specifically for receiving location updates, even when the app has been killed by the system. In order to do so, use a PendingIntent for a started service. For foreground use cases, the LocationCallback version of the method is recommended, see requestLocationUpdates(LocationRequest, LocationCallback, Looper).</p> <p data-bbox="556 570 1749 630">Any previously registered requests that have the same PendingIntent (as defined by equals(Object)) will be replaced by this request.</p> <p data-bbox="556 662 1755 755">Both LocationResult and LocationAvailability are sent to the given PendingIntent. You can extract data from an Intent using hasResult(Intent), extractResult(Intent), hasLocationAvailability(Intent), and extractLocationAvailability(Intent).</p> <p data-bbox="569 781 693 805"><b>Parameters</b></p> <table border="1" data-bbox="556 833 1766 971"> <tbody> <tr> <td data-bbox="556 833 863 902">request</td> <td data-bbox="863 833 1766 902">The location request for the updates.</td> </tr> <tr> <td data-bbox="556 902 863 971">callbackIntent</td> <td data-bbox="863 902 1766 971">A pending intent to be sent for each location update.</td> </tr> </tbody> </table> <p data-bbox="569 997 653 1021"><b>Returns</b></p> <ul data-bbox="577 1044 1381 1068" style="list-style-type: none"> <li>• a Task for the call, check isSuccessful() to determine if it was successful.</li> </ul> <p data-bbox="537 1081 1902 1144"><a href="https://developers.google.com/android/reference/com/google/android/gms/location/FusedLocationProviderClient">https://developers.google.com/android/reference/com/google/android/gms/location/FusedLocationProviderClient</a></p>	request	The location request for the updates.	callbackIntent	A pending intent to be sent for each location update.
request	The location request for the updates.				
callbackIntent	A pending intent to be sent for each location update.				

## Exhibit D - Claim Chart for U.S. Patent No. 9,467,838 Against HTC

US9467838	Exemplary Supporting Evidence Regarding Accused Products						
	<p data-bbox="558 245 1766 280">public void <b>onLocationAvailability</b> (<a href="#">LocationAvailability</a> locationAvailability)</p> <p data-bbox="548 310 1192 334">Called when there is a change in the availability of location data.</p> <p data-bbox="548 367 1766 561">When <a href="#">isLocationAvailable()</a> returns <code>false</code> you can assume that location will not be returned in <a href="#">onLocationResult(LocationResult)</a> until something changes in the device's settings or environment. Even when <a href="#">isLocationAvailable()</a> returns <code>true</code> the <a href="#">onLocationResult(LocationResult)</a> may not always be called regularly, however the device location is known and both the most recently delivered location and <a href="#">getLastLocation(GoogleApiClient)</a> will be reasonably up to date given the hints specified by the active <a href="#">LocationRequest</a> s.</p> <p data-bbox="569 586 695 610"><b>Parameters</b></p> <table border="1" data-bbox="548 643 1766 708"> <tr> <td data-bbox="558 651 982 699"><b>locationAvailability</b></td> <td data-bbox="989 651 1755 699">The current status of location availability.</td> </tr> </table> <p data-bbox="558 756 1766 792">public void <b>onLocationResult</b> (<a href="#">LocationResult</a> result)</p> <p data-bbox="548 821 1077 846">Called when device location information is available.</p> <p data-bbox="548 878 1682 943">The most recent location returned by <a href="#">getLastLocation()</a> is not guaranteed to be immediately fresh, but will be reasonably up to date given the hints specified by the active <a href="#">LocationRequest</a> s.</p> <p data-bbox="569 967 695 992"><b>Parameters</b></p> <table border="1" data-bbox="548 1024 1766 1089"> <tr> <td data-bbox="558 1032 789 1081"><b>result</b></td> <td data-bbox="795 1032 1755 1081">The latest location result available.</td> </tr> </table> <p data-bbox="537 1097 1829 1130"><a href="https://developers.google.com/android/reference/com/google/android/gms/location/LocationCallback">https://developers.google.com/android/reference/com/google/android/gms/location/LocationCallback</a></p> <p data-bbox="558 1146 1766 1182">public abstract void <b>onLocationChanged</b> (<a href="#">Location</a> location)</p> <p data-bbox="548 1211 936 1235">Called when the location has changed.</p> <p data-bbox="569 1260 695 1284"><b>Parameters</b></p> <table border="1" data-bbox="548 1317 1766 1382"> <tr> <td data-bbox="558 1325 947 1373"><b>location</b></td> <td data-bbox="953 1325 1755 1373">The updated location.</td> </tr> </table>	<b>locationAvailability</b>	The current status of location availability.	<b>result</b>	The latest location result available.	<b>location</b>	The updated location.
<b>locationAvailability</b>	The current status of location availability.						
<b>result</b>	The latest location result available.						
<b>location</b>	The updated location.						

## Exhibit D - Claim Chart for U.S. Patent No. 9,467,838 Against HTC

US9467838	Exemplary Supporting Evidence Regarding Accused Products
	<p data-bbox="541 237 1822 264"><a href="https://developers.google.com/android/reference/com/google/android/gms/location/LocationListener">https://developers.google.com/android/reference/com/google/android/gms/location/LocationListener</a></p> <p data-bbox="541 318 835 345">Public Constructors</p> <hr data-bbox="541 358 1766 363"/> <p data-bbox="541 412 1766 451">public <b>MapView</b> (<b>Context</b> context)</p> <p data-bbox="541 500 1766 539">public <b>MapView</b> (<b>Context</b> context, <b>AttributeSet</b> attrs)</p> <p data-bbox="541 587 1766 626">public <b>MapView</b> (<b>Context</b> context, <b>AttributeSet</b> attrs, int defStyle)</p> <p data-bbox="541 675 1766 714">public <b>MapView</b> (<b>Context</b> context, <b>GoogleMapOptions</b> options)</p> <p data-bbox="541 747 1696 774"><a href="https://developers.google.com/android/reference/com/google/android/gms/maps/MapView">https://developers.google.com/android/reference/com/google/android/gms/maps/MapView</a></p>



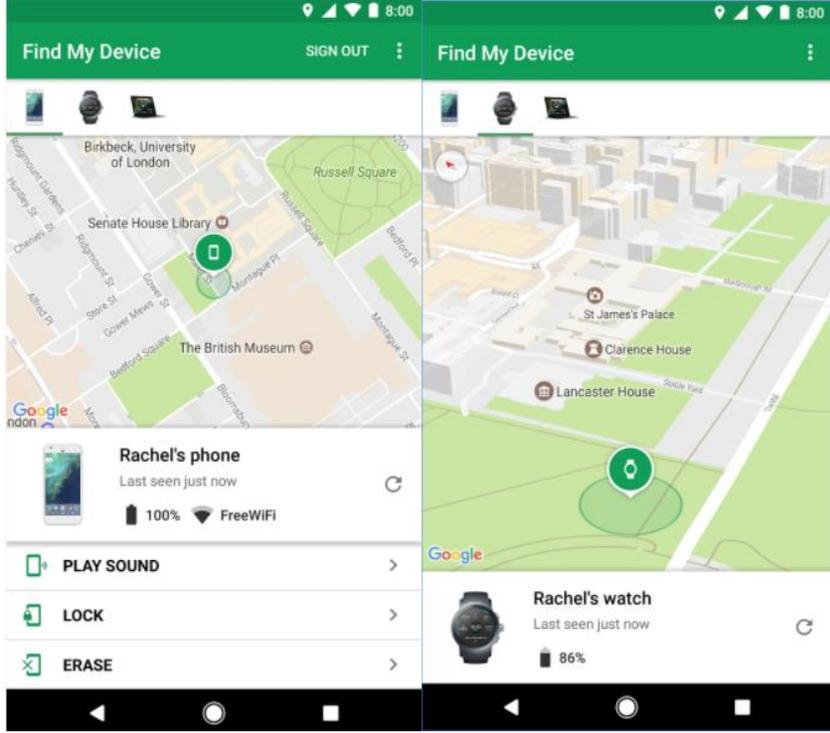
## Exhibit D - Claim Chart for U.S. Patent No. 9,467,838 Against HTC

US9467838	Exemplary Supporting Evidence Regarding Accused Products		
	<p><code>public void <b>getMapAsync</b> (<b>OnMapReadyCallback</b> callback)</code></p> <p>Returns a non-null instance of the <code>GoogleMap</code>, ready to be used.</p> <p>Note that:</p> <ul style="list-style-type: none"> <li>• This method must be called from the main thread.</li> <li>• The callback will be executed in the main thread.</li> <li>• In the case where Google Play services is not installed on the user's device, the callback will not be triggered until the user installs it.</li> <li>• The <code>GoogleMap</code> object provided by the callback is non-null.</li> </ul> <p>Parameters</p> <table border="1" data-bbox="554 688 1761 753"> <tr> <td><code>callback</code></td> <td>The callback object that will be triggered when the map is ready to be used.</td> </tr> </table> <p><code>public final void <b>onCreate</b> (<b>Bundle</b> savedInstanceState)</code></p> <p>You must call this method from the parent Activity/Fragment's corresponding method.</p> <p><a href="https://developers.google.com/android/reference/com/google/android/gms/maps/MapView">https://developers.google.com/android/reference/com/google/android/gms/maps/MapView</a></p>	<code>callback</code>	The callback object that will be triggered when the map is ready to be used.
<code>callback</code>	The callback object that will be triggered when the map is ready to be used.		
[1E] sending, to a second server, a request for second georeferenced map data different from the first georeferenced map data;	<p>HTC infringes directly and/or indirectly by performing, inducing others to perform, and/or contributing to the performance of: sending, to a second server, a request for second georeferenced map data different from the first georeferenced map data.</p> <p><b><u>Regarding Find My Device</u></b> and Android Device Manager, the Accused Products display, to the user on the display of the first device, a map with one or more symbols corresponding to one or more second devices. The user, via the first device, or the device itself requests different map data from a second server. The request occurs responsive to user input (e.g., zoom, drag, pan, change focus, change map type, symbol or device selection, another device or user selection, refresh or reload request, change in position of first device, change in position of a second device). Alternatively, the request occurs responsive to an</p>		

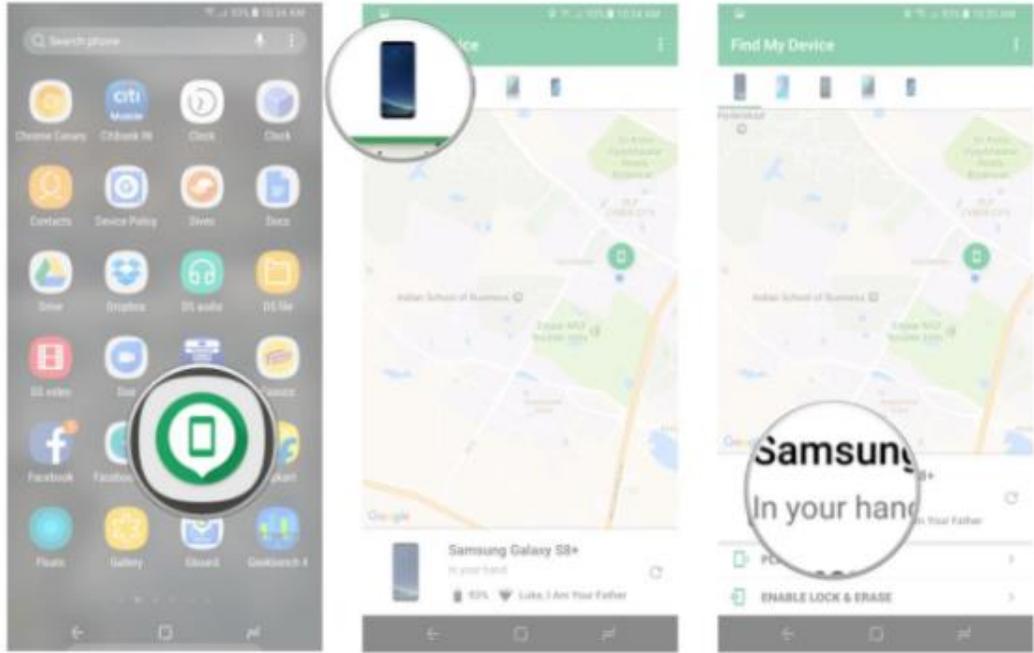
**Exhibit D - Claim Chart for U.S. Patent No. 9,467,838 Against HTC**

<b>US9467838</b>	<b>Exemplary Supporting Evidence Regarding Accused Products</b>
	<p>automatic and/or pre-determined control caused by an instruction from within the first device or from the one or more second devices, e.g. a refresh. The different map data includes an update to the first data or a replacement of the first data. Upon information and belief, the new map data may come from one of many sources and servers.</p> <p><b><u>Regarding Google Maps</u></b>, Google Latitude, Google Plus, Google Hangouts, Google Messages, Google Allo, Google Duo, Google Chrome, and Android Messenger, the Accused Products display, to the user on the display of the first device, a map with one or more symbols corresponding to one or more second users (or second devices corresponding to the second users). The user, via the first device, or the device itself requests different map data from a second server. The request occurs responsive to user input (e.g., zoom, drag, pan, change focus, change map type, refresh or reload request, symbol or device selection, another device or user selection, change in position of first device, change in position of a second device). Alternatively, the request occurs responsive to an automatic and/or pre-determined control caused by an instruction from within the first device or from the one or more second devices, e.g. a refresh. The different map data includes an update to the first data or a replacement of the first data. Upon information and belief, the new map data may come from one of many sources and servers.</p> <p><b><u>Exemplary Support for Find My Device:</u></b></p>

**Exhibit D - Claim Chart for U.S. Patent No. 9,467,838 Against HTC**

US9467838	Exemplary Supporting Evidence Regarding Accused Products
	 <p><a href="https://play.google.com/store/apps/details?id=com.google.android.apps.adm&amp;hl=en">https://play.google.com/store/apps/details?id=com.google.android.apps.adm&amp;hl=en</a></p>

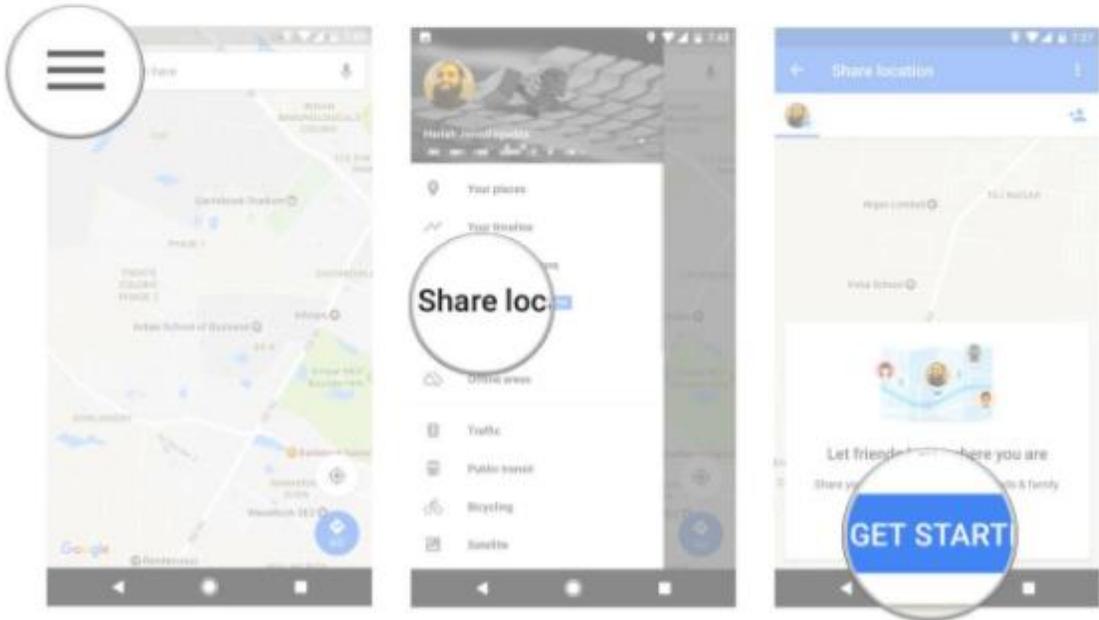
**Exhibit D - Claim Chart for U.S. Patent No. 9,467,838 Against HTC**

US9467838	Exemplary Supporting Evidence Regarding Accused Products
	<p>Once you're signed in to Find My Device, you'll see a map with your current location as well as the make and model of your phone, and two options — Play Sound, and Enable Lock &amp; Erase. Hitting the latter option will allow you to start using the Lock and Erase functions.</p> <p>If you've signed into more than one phone, you can select a particular device by browsing the list at the top of the screen.</p> <ol style="list-style-type: none"> <li>1. Open <b>Find My Device</b> from your home screen or app drawer.</li> <li>2. Select your phone from the list of devices at the top of the screen.</li> <li>3. See if your phone is discoverable.</li> </ol>  <p><a href="https://www.androidcentral.com/find-my-device">https://www.androidcentral.com/find-my-device</a></p>

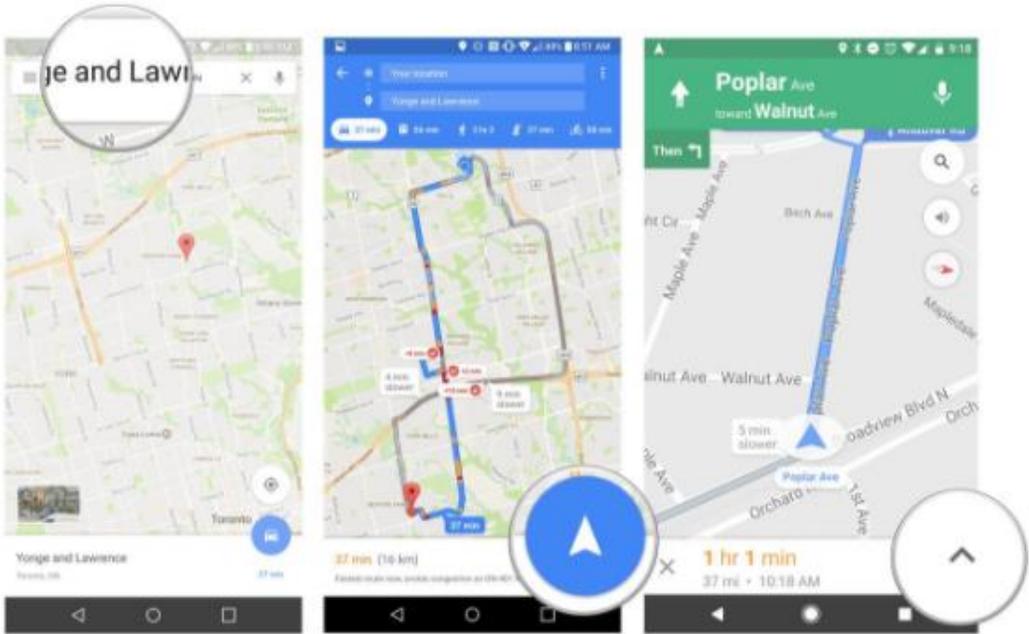
**Exhibit D - Claim Chart for U.S. Patent No. 9,467,838 Against HTC**

US9467838	Exemplary Supporting Evidence Regarding Accused Products
	<h2 data-bbox="548 240 1688 289">How to locate your phone over the internet</h2> <p data-bbox="548 337 1696 475">If you've lost your phone, you can remotely locate it through the <a href="#">Find My Device website</a>. You'll need to sign in to the Google account that was used to set up Find My Device. It takes a few seconds, but the service should be able to track your phone. Alternatively, you can also do a Google search for "find my phone" to locate your handset.</p> <ol data-bbox="548 532 1031 686" style="list-style-type: none"><li>1. Head to the <a href="#">Find My Device website</a>.</li><li>2. Sign in to your Google account.</li><li>3. Check if your device is visible.</li></ol> <div data-bbox="583 727 1793 1182"></div> <p data-bbox="537 1214 1157 1247"><a href="https://www.androidcentral.com/find-my-device">https://www.androidcentral.com/find-my-device</a></p> <p data-bbox="537 1287 1052 1320"><b><u>Exemplary Support for Google Maps:</u></b></p>

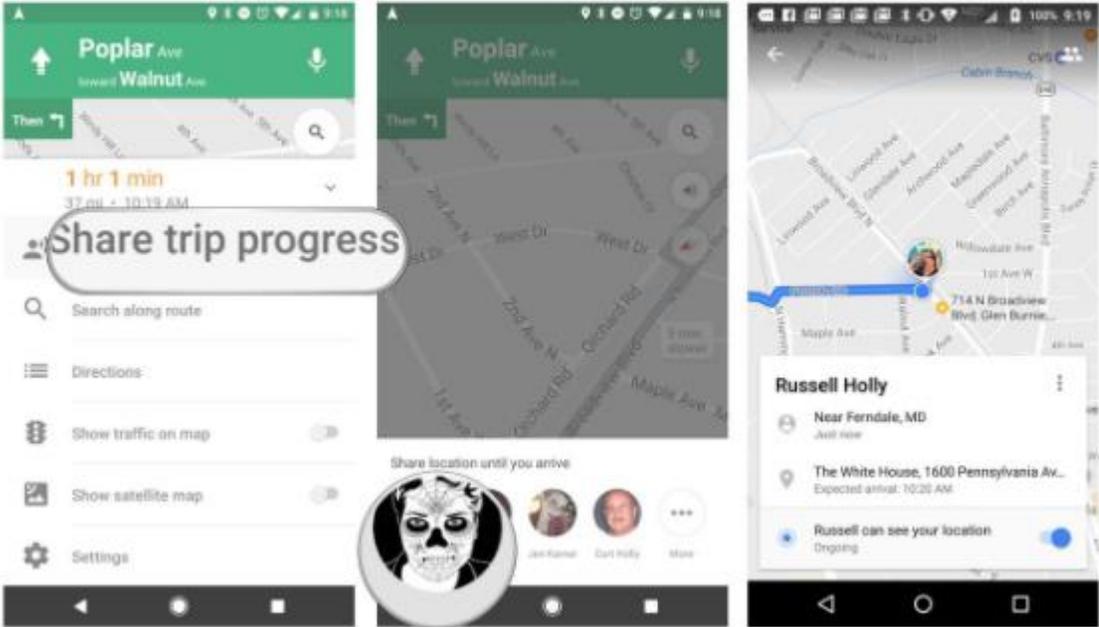
**Exhibit D - Claim Chart for U.S. Patent No. 9,467,838 Against HTC**

US9467838	Exemplary Supporting Evidence Regarding Accused Products
	<h2 data-bbox="548 240 1281 289">How to create a shareable link</h2> <p data-bbox="548 329 1486 358">You can also create a link and use it to share your location easily. Here's how to do it:</p> <ol data-bbox="541 407 1260 548" style="list-style-type: none"><li>1. Tap the hamburger menu on the top left corner of the screen.</li><li>2. Select Share location.</li><li>3. Tap Get Started.</li></ol>  <p data-bbox="533 1230 1381 1263"><a href="https://www.androidcentral.com/how-share-location-google-maps">https://www.androidcentral.com/how-share-location-google-maps</a></p>

**Exhibit D - Claim Chart for U.S. Patent No. 9,467,838 Against HTC**

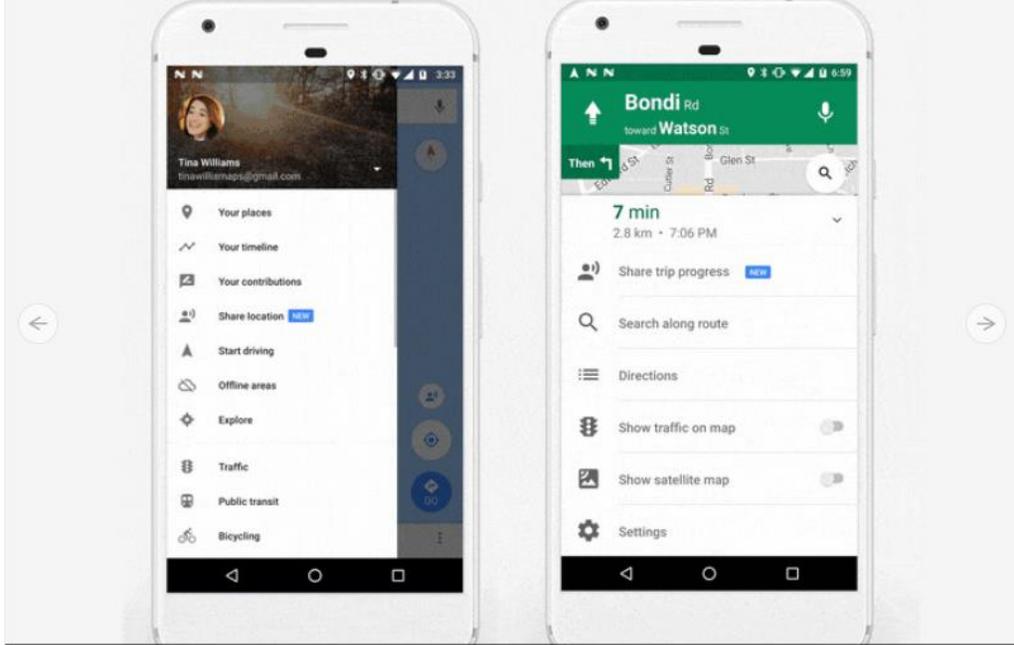
US9467838	Exemplary Supporting Evidence Regarding Accused Products
	<h2 data-bbox="550 240 1451 337">How to share your navigation directions while you walk, drive or transit</h2> <p data-bbox="550 375 1577 464">One of the best ways to share your location with someone is during a drive, walk or transit. If you're meeting a friend or family member somewhere, or navigating towards their home, Google Maps lets you share your location with them for the duration of the trip. It's magic!</p> <ol data-bbox="550 513 1419 643" style="list-style-type: none"><li>1. In the <b>search bar</b> enter your destination.</li><li>2. Pick your navigation type (drive, transit, walk) and press the <b>blue navigate button</b>.</li><li>3. Tap the arrow next to the <b>time-to-destination</b> number at the bottom of the screen.</li></ol>  <p data-bbox="535 1328 1381 1360"><a href="https://www.androidcentral.com/how-share-location-google-maps">https://www.androidcentral.com/how-share-location-google-maps</a></p>

**Exhibit D - Claim Chart for U.S. Patent No. 9,467,838 Against HTC**

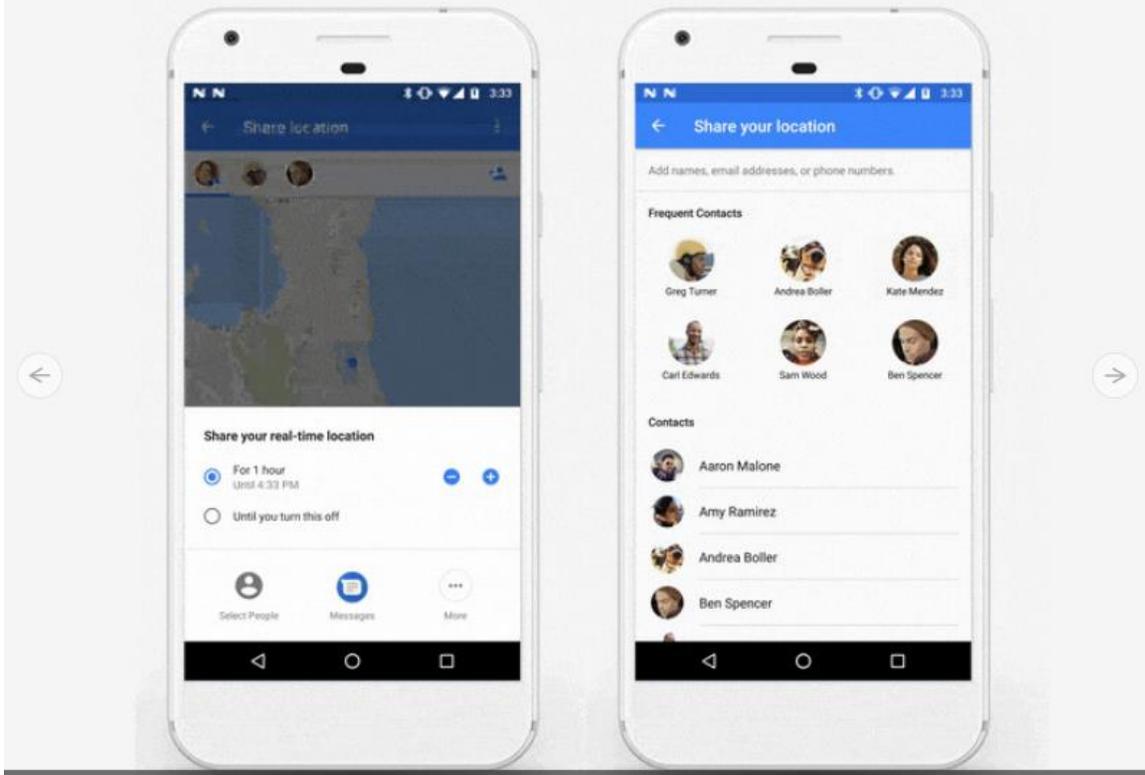
US9467838	Exemplary Supporting Evidence Regarding Accused Products
	<p>4. Tap Share trip progress.</p> <p>5. Choose one or more contacts to share trip progress.</p>  <p>You can also stop sharing your location with someone before a trip ends.  <a href="https://www.androidcentral.com/how-share-location-google-maps">https://www.androidcentral.com/how-share-location-google-maps</a></p>



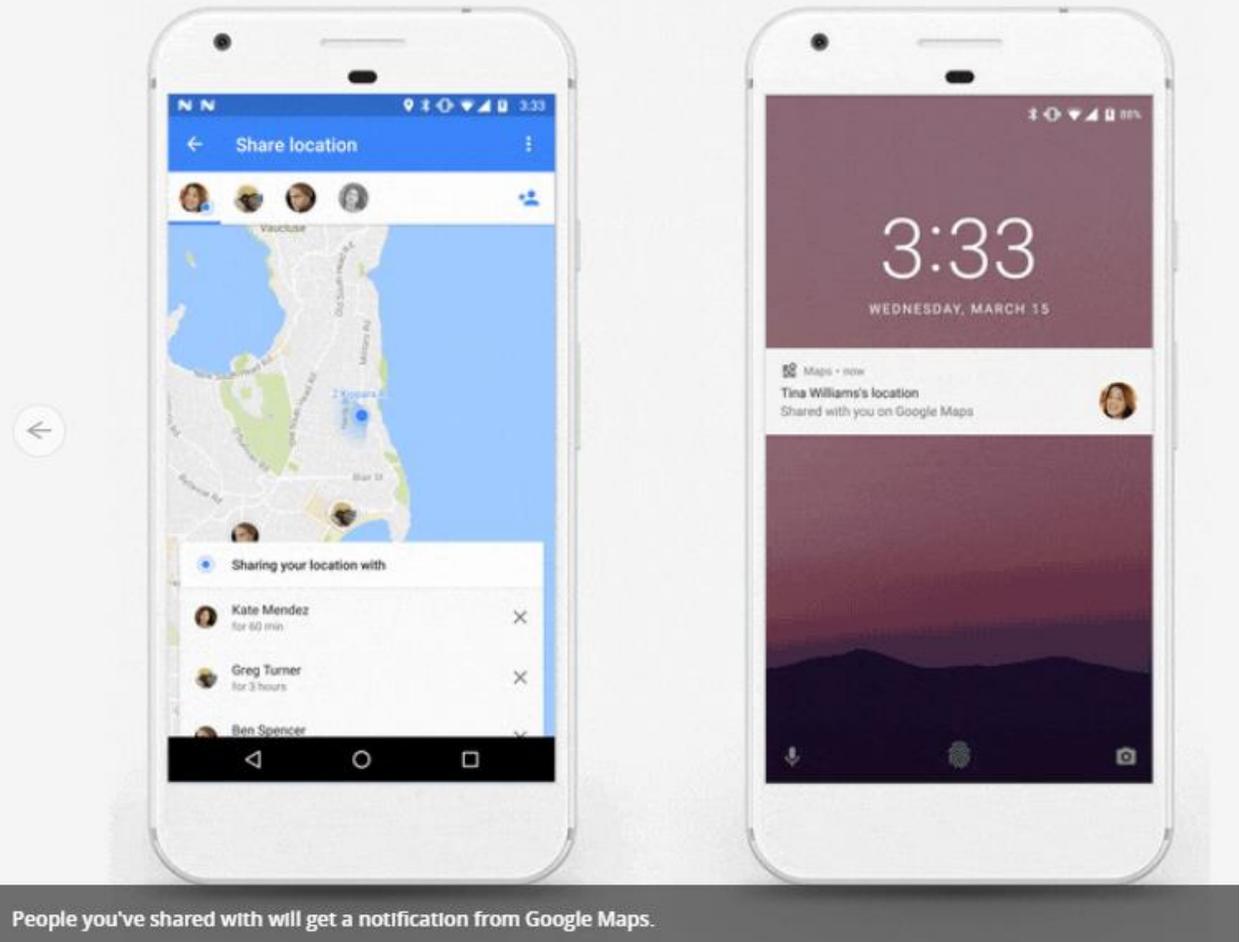
**Exhibit D - Claim Chart for U.S. Patent No. 9,467,838 Against HTC**

US9467838	Exemplary Supporting Evidence Regarding Accused Products
	 <p data-bbox="535 893 1549 950">Ways to start location sharing: (left) from the app navigation drawer, and (right) from the menu, you can drag up while navigating somewhere.</p> <p data-bbox="535 958 1549 990"><a href="https://arstechnica.com/gadgets/2017/03/location-sharing-finally-returns-to-google-maps/">https://arstechnica.com/gadgets/2017/03/location-sharing-finally-returns-to-google-maps/</a></p>

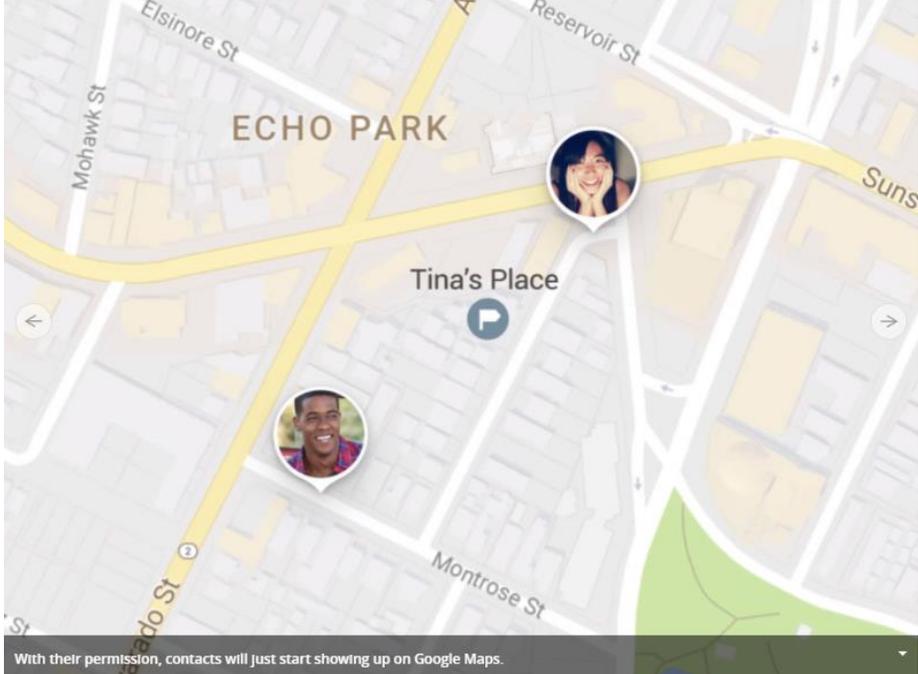
**Exhibit D - Claim Chart for U.S. Patent No. 9,467,838 Against HTC**

US9467838	Exemplary Supporting Evidence Regarding Accused Products
	 <p>The image displays two side-by-side smartphone screens showing the 'Share location' interface in Google Maps. The left screen shows a map with a location pin and options to share real-time location for '1 hour' (selected) or 'Until you turn this off'. The right screen shows a list of contacts to share with, including 'Greg Turner', 'Andrea Boller', 'Kate Mendez', 'Carl Edwards', 'Sam Wood', and 'Ben Spencer'. Below the contact list, there are more contacts listed: 'Aaron Malone', 'Amy Ramirez', 'Andrea Boller', and 'Ben Spencer'. A caption at the bottom of the image reads: 'You can pick how long you want to share your location for, and with whom you want to share it.' Below the caption is a URL: <a href="https://arstechnica.com/gadgets/2017/03/location-sharing-finally-returns-to-google-maps/">https://arstechnica.com/gadgets/2017/03/location-sharing-finally-returns-to-google-maps/</a></p>

**Exhibit D - Claim Chart for U.S. Patent No. 9,467,838 Against HTC**

US9467838	Exemplary Supporting Evidence Regarding Accused Products
	 <p data-bbox="541 1144 1197 1177">People you've shared with will get a notification from Google Maps.</p> <p data-bbox="541 1188 1680 1221"><a href="https://arstechnica.com/gadgets/2017/03/location-sharing-finally-returns-to-google-maps/">https://arstechnica.com/gadgets/2017/03/location-sharing-finally-returns-to-google-maps/</a></p>

**Exhibit D - Claim Chart for U.S. Patent No. 9,467,838 Against HTC**

US9467838	Exemplary Supporting Evidence Regarding Accused Products
	 <p><a href="https://arstechnica.com/gadgets/2017/03/location-sharing-finally-returns-to-google-maps/">https://arstechnica.com/gadgets/2017/03/location-sharing-finally-returns-to-google-maps/</a></p> <p>For example, Google requests and receives map data from many sources and servers, both internally and externally. In response to one of a number of user actions (e.g. zoom, drag or change focus, change map type, select another device or user), new map data is retrieved to complete the user's action and the displayed map is replaced or updated accordingly. The new map data may come from one of many sources and servers. The Accused Products request and receive the map data described above. Alternatively, any computer signed-in to the Google network services may request and receive the map data described above.</p> <p>Another example of this limitation in practice includes toggling between map types. The first-displayed map can be the standard interface map with geographical points of interest. In response to user action, a second map can be</p>

**Exhibit D - Claim Chart for U.S. Patent No. 9,467,838 Against HTC**

US9467838	Exemplary Supporting Evidence Regarding Accused Products
	<p>called with satellite imagery, which would be retrieved from a different server source having satellite information. Another known example is when a user zooms into a standard map and a transmit map is retrieved and overlaid or otherwise included to form a second map with transit data pulled from a transit data server.</p> <p>3.36 <b>United States.</b>Google maps of the United States include data provided from the following sources:</p> <ul style="list-style-type: none"> <li>a. U.S. Fish and Wildlife Service - <a href="http://www.fws.gov/">http://www.fws.gov/</a></li> <li>b. U.S. Census Bureau - <a href="http://www.census.gov/">http://www.census.gov/</a></li> <li>c. USDA Forest Service - <a href="http://www.fs.fed.us/">http://www.fs.fed.us/</a></li> <li>d. U.S. Geological Survey, Gap Analysis Program (GAP) - <a href="http://gapanalysis.usgs.gov/padus/">http://gapanalysis.usgs.gov/padus/</a></li> <li>e. U.S. Geological Survey, U.S. Geographic Names Information System (GNIS) - <a href="http://geonames.usgs.gov/">http://geonames.usgs.gov/</a></li> <li>f. U.S. Geological Survey, National Hydrography Dataset (NHD) - <a href="http://nhd.usgs.gov/">http://nhd.usgs.gov/</a></li> <li>g. U.S. Geological Survey, Topographic Maps - <a href="http://topomaps.usgs.gov/">http://topomaps.usgs.gov/</a></li> <li>h. U.S. Geological Survey - <a href="http://www.usgs.gov/">http://www.usgs.gov/</a></li> <li>i. U.S. Coast Guard - <a href="http://www.uscg.mil/">http://www.uscg.mil/</a></li> <li>j. University of New Hampshire - <a href="http://ccom.unh.edu/">http://ccom.unh.edu/</a></li> <li>k. U.S. National Parks Service - <a href="http://www.nps.gov/">http://www.nps.gov</a></li> <li>l. U.S. Department of Transportation, Research and Innovative Technology Administration - <a href="http://www.rita.dot.gov/">http://www.rita.dot.gov/</a></li> </ul> <p><a href="https://www.google.com/intl/en_us/help/legalnotices_maps.html">https://www.google.com/intl/en_us/help/legalnotices_maps.html</a></p>

**Exhibit D - Claim Chart for U.S. Patent No. 9,467,838 Against HTC**

US9467838	Exemplary Supporting Evidence Regarding Accused Products
	<p><b>Data center locations</b></p> <p>We own and operate data centers around the world to keep our products running 24 hours a day, 7 days a week. Find out more about our data center locations, community involvement, and job opportunities in our locations around the world.</p> <p><b>Americas</b></p> <ul style="list-style-type: none"> <li>Berkeley County, South Carolina</li> <li>Council Bluffs, Iowa</li> <li>Douglas County, Georgia</li> <li>Jackson County, Alabama</li> <li>Lenoir, North Carolina</li> <li>Mayes County, Oklahoma</li> <li>Montgomery County, Tennessee</li> <li>Quilicura, Chile</li> <li>The Dalles, Oregon</li> </ul> <p><b>Asia</b></p> <ul style="list-style-type: none"> <li>Changhua County, Taiwan</li> <li>Singapore</li> </ul> <p><b>Europe</b></p> <ul style="list-style-type: none"> <li>Dublin, Ireland</li> <li>Eemshaven, Netherlands</li> <li>Hamina, Finland</li> <li>St Ghislain, Belgium</li> </ul>  <p><a href="https://www.google.com/about/datacenters/inside/locations/index.html">https://www.google.com/about/datacenters/inside/locations/index.html</a></p>

**Exhibit D - Claim Chart for U.S. Patent No. 9,467,838 Against HTC**

US9467838	Exemplary Supporting Evidence Regarding Accused Products
	<h3 data-bbox="541 237 863 264">Request Location Updates</h3> <p data-bbox="541 289 1451 354">Before requesting location updates, your app must connect to location services and make a location request. The lesson on <a href="#">Changing Location Settings</a> shows you how to do this. Once a location request is in place you can start the regular updates by calling <code>requestLocationUpdates()</code>. Do this in the <code>onConnected()</code> callback provided by Google API Client, which is called when the client is ready.</p> <p data-bbox="541 370 1440 435">Depending on the form of the request, the fused location provider either invokes the <code>LocationListener.onLocationChanged()</code> callback method and passes it a <code>Location</code> object, or issues a <code>PendingIntent</code> that contains the location in its extended data. The accuracy and frequency of the updates are affected by the location permissions you've requested and the options you set in the location request object.</p> <p data-bbox="541 451 1451 516">This lesson shows you how to get the update using the <code>LocationListener</code> callback approach. Call <code>requestLocationUpdates()</code>, passing it your instance of the <code>GoogleApiClient</code>, the <code>LocationRequest</code> object, and a <code>LocationListener</code>. Define a <code>startLocationUpdates()</code> method, called from the <code>onConnected()</code> callback, as shown in the following code sample:</p> <pre data-bbox="552 532 1451 760">@Override public void onConnected(Bundle connectionHint) {     ...     if (mRequestingLocationUpdates) {         startLocationUpdates();     } }  protected void startLocationUpdates() {     LocationServices.FusedLocationApi.requestLocationUpdates(         mGoogleApiClient, mLocationRequest, this); }</pre> <p data-bbox="541 781 1440 821">Notice that the above code snippet refers to a boolean flag, <code>mRequestingLocationUpdates</code>, used to track whether the user has turned location updates on or off. For more about retaining the value of this flag across instances of the activity, see <a href="#">Save the State of the Activity</a>.</p> <p data-bbox="541 870 1486 898"><a href="https://developer.android.com/training/location/receive-location-updates.html">https://developer.android.com/training/location/receive-location-updates.html</a></p>

**Exhibit D - Claim Chart for U.S. Patent No. 9,467,838 Against HTC**

US9467838	Exemplary Supporting Evidence Regarding Accused Products
	<p>Configure a firewall to allow access to the Google Maps APIs Services</p> <p><i>Why it's important:</i> Google Maps APIs services use a variety of domains, some of which do not belong to the *google.com domain. If you are behind a restrictive firewall, it is important to allow access to the domains used by each Maps API service. If your firewall doesn't allow access to these domains, API requests will fail, which can break your applications. You can find a complete list of domains used by the Maps APIs in the Support Portal:</p> <ol style="list-style-type: none"> <li>1. Log in to the <a href="#">Google Cloud Support Portal</a>. The Support Portal is available only to customers with the Google Maps APIs Premium Plan or a previous Google Maps APIs for Work or Google Maps for Business license.</li> <li>2. Navigate to the <b>Resources</b> tab.</li> <li>3. Select the <b>list of domains used by the Google Maps APIs family</b>. (Here's the <a href="#">direct link</a>.)</li> <li>4. Allow your applications to access the listed domains.</li> </ol> <p>We do not recommend managing firewall restrictions by IP address, as the IPs associated with these domains are not static.</p> <p><b>Note:</b> Google Maps APIs services use port 80 (http) and 443 (https) for inbound and outbound traffic. These services also require GET, POST, PUT, DELETE, and HEAD requests. Configure your firewall to allow traffic over these ports and to allow requests, depending on API and use case.</p> <p><a href="https://developers.google.com/maps/premium/prelaunch-checklist#firewall">https://developers.google.com/maps/premium/prelaunch-checklist#firewall</a></p> <p><b><u>Exemplary Screenshots:</u></b></p> <p>See, e.g., 1D and 1E above.</p> <p><b><u>Exemplary Source Code:</u></b></p> <p>The above functionality is performed at least in part by the following publicly available source code and/or source code that invokes or is invoked by the following source code (or a substantially similar copy compiled and loaded onto the Accused Products by HTC). AGIS reserves the right to supplement these contentions with additional source code as discovery progresses and as additional source code is made available.</p>



## Exhibit D - Claim Chart for U.S. Patent No. 9,467,838 Against HTC

US9467838	Exemplary Supporting Evidence Regarding Accused Products
	<pre>44  * Class that sends chat message via SMS. 45  * 46  * The interface emulates a blocking sending similar to making an HTTP request. 47  * It calls the SmsManager to send a (potentially multipart) message and waits 48  * on the sent status on each part. The waiting has a timeout so it won't wait 49  * forever. Once the sent status of all parts received, the call returns. 50  * A successful sending requires success status for all parts. Otherwise, we 51  * pick the highest level of failure as the error for the whole message, which 52  * is used to determine if we need to retry the sending. 53  */ 54  public class SmsSender { 55      private static final String TAG = LogUtil.BUGLE_TAG; 56 57      public static final String EXTRA_PART_ID = "part_id"; 58 59      /* 60       * A map for pending sms messages. The key is the random request UUID. 61       */ 62      private static ConcurrentHashMap&lt;Uri, SendResult&gt; sPendingMessageMap = 63          new ConcurrentHashMap&lt;Uri, SendResult&gt;(); 64 65      private static final Random RANDOM = new Random(); 66 67      // Whether we should send multipart SMS as separate messages 68      private static Boolean sSendMultipartSmsAsSeparateMessages = null; 69  }</pre> <p><a href="https://android.googlesource.com/platform/packages/apps/Messaging/+nougat-mr1-release/src/com/android/messaging/sms/SmsSender.java">https://android.googlesource.com/platform/packages/apps/Messaging/+nougat-mr1-release/src/com/android/messaging/sms/SmsSender.java</a></p>

## Exhibit D - Claim Chart for U.S. Patent No. 9,467,838 Against HTC

US9467838	Exemplary Supporting Evidence Regarding Accused Products
	<pre> 253 // Actually sending the message using SmsManager 254 private static void sendInternal(final Context context, final int subId, String dest, 255     final ArrayList&lt;String&gt; messages, final String serviceCenter, 256     final boolean requireDeliveryReport, final Uri messageUri) throws SmsException { 257     Assert.notNull(context); 258     final SmsManager smsManager = PhoneUtils.get(subId).getSmsManager(); 259     final int messageCount = messages.size(); 260     final ArrayList&lt;PendingIntent&gt; deliveryIntents = new ArrayList&lt;PendingIntent&gt;(messageCount); 261     final ArrayList&lt;PendingIntent&gt; sentIntents = new ArrayList&lt;PendingIntent&gt;(messageCount); 262     for (int i = 0; i &lt; messageCount; i++) { 263         // Make pending intents different for each message part 264         final int partId = (messageCount &lt;= 1 ? 0 : i + 1); 265         if (requireDeliveryReport &amp;&amp; (i == (messageCount - 1))) { 266             // TODO we only care about the delivery status of the last part 267             // Shall we have better tracking of delivery status of all parts? 268             deliveryIntents.add(PendingIntent.getBroadcast( 269                 context, 270                 partId, 271                 getSendStatusIntent(context, SendStatusReceiver.MESSAGE_DELIVERED_ACTION, 272                     messageUri, partId, subId), 273                 0/*flag*/)); 274         } else { 275             deliveryIntents.add(null); 276         } 277         sentIntents.add(PendingIntent.getBroadcast( 278             context, 279             partId, 280             getSendStatusIntent(context, SendStatusReceiver.MESSAGE_SENT_ACTION, 281                 messageUri, partId, subId), 282             0/*flag*/)); 283     } 284     if (sSendMultipartSmsAsSeparateMessages == null) { 285         sSendMultipartSmsAsSeparateMessages = MmsConfig.get(subId) 286             .getSendMultipartSmsAsSeparateMessages(); 287     } </pre> <p><a href="https://android.googlesource.com/platform/packages/apps/Messaging/+nougat-mr1-release/src/com/android/messaging/sms/SmsSender.java">https://android.googlesource.com/platform/packages/apps/Messaging/+nougat-mr1-release/src/com/android/messaging/sms/SmsSender.java</a></p>

## Exhibit D - Claim Chart for U.S. Patent No. 9,467,838 Against HTC

US9467838	Exemplary Supporting Evidence Regarding Accused Products
	<pre> 288     try { 289         if (sSendMultipartSmsAsSeparateMessages) { 290             // If multipart sms is not supported, send them as separate messages 291             for (int i = 0; i &lt; messageCount; i++) { 292                 smsManager.sendTextMessage(dest, 293                     serviceCenter, 294                     messages.get(i), 295                     sentIntents.get(i), 296                     deliveryIntents.get(i)); 297             } 298         } else { 299             smsManager.sendMultipartTextMessage( 300                 dest, serviceCenter, messages, sentIntents, deliveryIntents); 301         } 302     } catch (final Exception e) { 303         throw new SmsException("SmsSender: caught exception in sending " + e); 304     } 305 } </pre> <p><a href="https://android.googlesource.com/platform/packages/apps/Messaging/+nougat-mr1-release/src/com/android/messaging/sms/SmsSender.java">https://android.googlesource.com/platform/packages/apps/Messaging/+nougat-mr1-release/src/com/android/messaging/sms/SmsSender.java</a></p> <pre> 56  * Class that receives incoming SMS messages through android.provider.Telephony.SMS_RECEIVED 57  * 58  * This class serves two purposes: 59  * - Process phone verification SMS messages 60  * - Handle SMS messages when the user has enabled us to be the default SMS app (Pre-KLP) 61  */ 62  public final class SmsReceiver extends BroadcastReceiver { 63      private static final String TAG = LogUtil.BUGLE_TAG; 64 65      private static ArrayList&lt;Pattern&gt; sIgnoreSmsPatterns; 66 </pre> <p><a href="https://android.googlesource.com/platform/packages/apps/Messaging/+nougat-mr1-release/src/com/android/messaging/receiver/SmsReceiver.java">https://android.googlesource.com/platform/packages/apps/Messaging/+nougat-mr1-release/src/com/android/messaging/receiver/SmsReceiver.java</a></p>

## Exhibit D - Claim Chart for U.S. Patent No. 9,467,838 Against HTC

US9467838	Exemplary Supporting Evidence Regarding Accused Products
	<pre>200     public static void deliverSmsMessages(final Context context, final int subId, 201         final int errorCode, final android.telephony.SmsMessage[] messages) { 202         final ContentValues messageValues = 203             MmsUtils.parseReceivedSmsMessage(context, messages, errorCode); 204 205         LogUtil.v(TAG, "SmsReceiver.deliverSmsMessages"); 206 207         final long nowInMillis = System.currentTimeMillis(); 208         final long receivedTimestampMs = MmsUtils.getMessageDate(messages[0], nowInMillis); 209 210         messageValues.put(Sms.Inbox.DATE, receivedTimestampMs); 211         // Default to unread and unseen for us but ReceiveSmsMessageAction will override 212         // seen for the telephony db. 213         messageValues.put(Sms.Inbox.READ, 0); 214         messageValues.put(Sms.Inbox.SEEN, 0); 215         if (OsUtil.isAtLeastL_MR1()) { 216             messageValues.put(Sms.SUBSCRIPTION_ID, subId); 217         } 218 219         if (messages[0].getMessageClass() == android.telephony.SmsMessage.MessageClass.CLASS_0    220             DebugUtils.debugClassZeroSmsEnabled()) { 221             Factory.get().getUIIntents().launchClassZeroActivity(context, messageValues); 222         } else { 223             final ReceiveSmsMessageAction action = new ReceiveSmsMessageAction(messageValues); 224             action.start(); 225         } 226     }</pre> <p><a href="https://android.googlesource.com/platform/packages/apps/Messaging/+nougat-mr1-release/src/com/android/messaging/receiver/SmsReceiver.java">https://android.googlesource.com/platform/packages/apps/Messaging/+nougat-mr1-release/src/com/android/messaging/receiver/SmsReceiver.java</a></p>

## Exhibit D - Claim Chart for U.S. Patent No. 9,467,838 Against HTC

US9467838	Exemplary Supporting Evidence Regarding Accused Products
	<pre>228     @Override 229     public void onReceive(final Context context, final Intent intent) { 230         LogUtil.v(TAG, "SmsReceiver.onReceive " + intent); 231         // On KLP+ we only take delivery of SMS messages in SmsDeliverReceiver. 232         if (PhoneUtils.getDefault().isSmsEnabled()) { 233             final String action = intent.getAction(); 234             if (OsUtil.isSecondaryUser() &amp;&amp; 235                 (Telephony.Sms.Intents.SMS_RECEIVED_ACTION.equals(action)    236                  // TODO: update this with the actual constant from Telephony 237                   "android.provider.Telephony.MMS_DOWNLOADED".equals(action))) { 238                 postNewMessageSecondaryUserNotification(); 239             } else if (!OsUtil.isAtLeastKLP()) { 240                 deliverSmsIntent(context, intent); 241             } 242         } 243     }</pre> <p><a href="https://android.googlesource.com/platform/packages/apps/Messaging/+nougat-mr1-release/src/com/android/messaging/receiver/SmsReceiver.java">https://android.googlesource.com/platform/packages/apps/Messaging/+nougat-mr1-release/src/com/android/messaging/receiver/SmsReceiver.java</a></p>

## Exhibit D - Claim Chart for U.S. Patent No. 9,467,838 Against HTC

US9467838	Exemplary Supporting Evidence Regarding Accused Products
	<pre>52  * Class that sends chat message via MMS. 53  * 54  * The interface emulates a blocking send similar to making an HTTP request. 55  */ 56  public class MmsSender { 57      private static final String TAG = LogUtil.BUGLE_TAG; 58 59      /** 60       * Send an MMS message. 61       * 62       * @param context Context 63       * @param messageUri The unique URI of the message for identifying it during sending 64       * @param sendReq The SendReq PDU of the message 65       * @throws MmsFailureException 66       */ 67      public static void sendMms(final Context context, final int subId, final Uri messageUri, 68                               final SendReq sendReq, final Bundle sentIntentExtras) throws MmsFailureException { 69          sendMms(context, 70                subId, 71                messageUri, 72                null /* locationUri */, 73                sendReq, 74                true /* responseImportant */, 75                sentIntentExtras); 76      }</pre> <p><a href="https://android.googlesource.com/platform/packages/apps/Messaging/+/nougat-mr1-release/src/com/android/messaging/sms/MmsSender.java">https://android.googlesource.com/platform/packages/apps/Messaging/+/nougat-mr1-release/src/com/android/messaging/sms/MmsSender.java</a></p>

## Exhibit D - Claim Chart for U.S. Patent No. 9,467,838 Against HTC

US9467838	Exemplary Supporting Evidence Regarding Accused Products
	<pre> 240      * Download an MMS message. 241      * 242      * @param context Context 243      * @param contentLocation The url of the MMS message 244      * @throws MmsFailureException 245      * @throws InvalidHeaderValueException 246      */ 247      public static void downloadMms(final Context context, final int subId, 248          final String contentLocation, Bundle extras) throws MmsFailureException, 249          InvalidHeaderValueException { 250          final Uri requestUri = Uri.parse(contentLocation); 251          final Uri contentUri = MmsFileProvider.buildRawMmsUri(); 252 253          final Intent downloadedIntent = new Intent(SendStatusReceiver.MMS_DOWNLOADED_ACTION, 254              requestUri, 255              context, 256              SendStatusReceiver.class); 257          downloadedIntent.putExtra(SendMessageAction.EXTRA_CONTENT_URI, contentUri); 258          if (extras != null) { 259              downloadedIntent.putExtras(extras); 260          } 261          final PendingIntent downloadedPendingIntent = PendingIntent.getBroadcast( 262              context, 263              0 /*request code*/, 264              downloadedIntent, 265              PendingIntent.FLAG_UPDATE_CURRENT); 266 267          MmsManager.downloadMultimediaMessage(subId, context, contentLocation, contentUri, 268              downloadedPendingIntent); 269      } </pre> <p><a href="https://android.googlesource.com/platform/packages/apps/Messaging/+nougat-mr1-release/src/com/android/messaging/sms/MmsSender.java">https://android.googlesource.com/platform/packages/apps/Messaging/+nougat-mr1-release/src/com/android/messaging/sms/MmsSender.java</a></p>

## Exhibit D - Claim Chart for U.S. Patent No. 9,467,838 Against HTC

US9467838	Exemplary Supporting Evidence Regarding Accused Products
	<pre> 97      * Execute an MMS HTTP request, either a POST (sending) or a GET (downloading) 98      * 99      * @param urlString The request URL, for sending it is usually the MMSC, and for downloading 100     *           it is the message URL 101     * @param pdu For POST (sending) only, the PDU to send 102     * @param method HTTP method, POST for sending and GET for downloading 103     * @param isProxySet Is there a proxy for the MMSC 104     * @param proxyHost The proxy host 105     * @param proxyPort The proxy port 106     * @param mmsConfig The MMS config to use 107     * @param userAgent The user agent header value 108     * @param uaProfUrl The UA Prof URL header value 109     * @return The HTTP response body 110     * @throws MmsHttpException For any failures 111     */ 112     public byte[] execute(String urlString, byte[] pdu, String method, boolean isProxySet, 113         String proxyHost, int proxyPort, Bundle mmsConfig, String userAgent, String uaProfUrl) 114         throws MmsHttpException { 115         Log.d(MmsService.TAG, "HTTP: " + method + " " + Utils.redactUrlForNonVerbose(urlString) 116             + (isProxySet ? (" proxy=" + proxyHost + ":" + proxyPort) : "") 117             + ", PDU size=" + (pdu != null ? pdu.length : 0)); 118         checkMethod(method); 119         HttpURLConnection connection = null; 120         try { 121             Proxy proxy = Proxy.NO_PROXY; 122             if (isProxySet) { 123                 proxy = new Proxy(Proxy.Type.HTTP, new InetSocketAddress(proxyHost, proxyPort)); 124             } 125             final URL url = new URL(urlString); 126             // Now get the connection 127             connection = (HttpURLConnection) url.openConnection(proxy); 128             connection.setDoInput(true); 129             connection.setConnectTimeout( 130                 mmsConfig.getInt(CarrierConfigValuesLoader.CONFIG_HTTP_SOCKET_TIMEOUT, 131                     CarrierConfigValuesLoader.CONFIG_HTTP_SOCKET_TIMEOUT_DEFAULT)); </pre> <p><a href="https://android.googlesource.com/platform/packages/apps/Messaging/+/nougat-mr1-release/src/android/support/v7/mms/MmsHttpClient.java">https://android.googlesource.com/platform/packages/apps/Messaging/+/nougat-mr1-release/src/android/support/v7/mms/MmsHttpClient.java</a></p>



## Exhibit D - Claim Chart for U.S. Patent No. 9,467,838 Against HTC

US9467838	Exemplary Supporting Evidence Regarding Accused Products
	<pre> 132 // ----- COMMON HEADERS ----- 133 // Header: Accept 134 connection.setRequestProperty(HEADER_ACCEPT, HEADER_VALUE_ACCEPT); 135 // Header: Accept-Language 136 connection.setRequestProperty( 137     HEADER_ACCEPT_LANGUAGE, getCurrentAcceptLanguage(Locale.getDefault())); 138 // Header: User-Agent 139 Log.i(MmsService.TAG, "HTTP: User-Agent=" + userAgent); 140 connection.setRequestProperty(HEADER_USER_AGENT, userAgent); 141 // Header: x-wap-profile 142 final String uaProfUrlTagName = mmsConfig.getString( 143     CarrierConfigValuesLoader.CONFIG_UA_PROF_TAG_NAME, 144     CarrierConfigValuesLoader.CONFIG_UA_PROF_TAG_NAME_DEFAULT); 145 if (uaProfUrl != null) { 146     Log.i(MmsService.TAG, "HTTP: UaProfUrl=" + uaProfUrl); 147     connection.setRequestProperty(uaProfUrlTagName, uaProfUrl); 148 } 149 // Add extra headers specified by mms_config.xml's httpparams 150 addExtraHeaders(connection, mmsConfig); 151 // Different stuff for GET and POST 152 if (METHOD_POST.equals(method)) { 153     if (pdu == null    pdu.length &lt; 1) { 154         Log.e(MmsService.TAG, "HTTP: empty pdu"); 155         throw new MmsHttpException(0/*statusCode*/, "Sending empty PDU"); 156     } 157     connection.setDoOutput(true); 158     connection.setRequestMethod(METHOD_POST); 159     if (mmsConfig.getBoolean( 160         CarrierConfigValuesLoader.CONFIG_SUPPORT_HTTP_CHARSET_HEADER, 161         CarrierConfigValuesLoader.CONFIG_SUPPORT_HTTP_CHARSET_HEADER_DEFAULT)) { 162         connection.setRequestProperty(HEADER_CONTENT_TYPE, 163             HEADER_VALUE_CONTENT_TYPE_WITH_CHARSET); 164     } else { 165         connection.setRequestProperty(HEADER_CONTENT_TYPE, 166             HEADER_VALUE_CONTENT_TYPE_WITHOUT_CHARSET); </pre>
	<p><a href="https://android.googlesource.com/platform/packages/apps/Messaging+/nougat-mr1-release/src/android/support/v7/mms/MmsHttpClient.java">https://android.googlesource.com/platform/packages/apps/Messaging+/nougat-mr1-release/src/android/support/v7/mms/MmsHttpClient.java</a></p>

## Exhibit D - Claim Chart for U.S. Patent No. 9,467,838 Against HTC

US9467838	Exemplary Supporting Evidence Regarding Accused Products
	<pre> 167         } 168         if (Log.isLoggable(MmsService.TAG, Log.VERBOSE)) { 169             logHttpHeaders(connection.getRequestProperties()); 170         } 171         connection.setFixedLengthStreamingMode(pdu.length); 172         // Sending request body 173         final OutputStream out = 174             new BufferedOutputStream(connection.getOutputStream()); 175         out.write(pdu); 176         out.flush(); 177         out.close(); 178     } else if (METHOD_GET.equals(method)) { 179         if (Log.isLoggable(MmsService.TAG, Log.VERBOSE)) { 180             logHttpHeaders(connection.getRequestProperties()); 181         } 182         connection.setRequestMethod(METHOD_GET); 183     } 184     // Get response 185     final int responseCode = connection.getResponseCode(); 186     final String responseMessage = connection.getResponseMessage(); 187     Log.d(MmsService.TAG, "HTTP: " + responseCode + " " + responseMessage); 188     if (Log.isLoggable(MmsService.TAG, Log.VERBOSE)) { 189         logHttpHeaders(connection.getHeaderFields()); 190     } 191     if (responseCode / 100 != 2) { 192         throw new MmsHttpException(responseCode, responseMessage); 193     } 194     final InputStream in = new BufferedInputStream(connection.getInputStream()); 195     final ByteArrayOutputStream byteOut = new ByteArrayOutputStream(); 196     final byte[] buf = new byte[4096]; 197     int count = 0; 198     while ((count = in.read(buf)) &gt; 0) { 199         byteOut.write(buf, 0, count); 200     } 201     in.close(); 202     final byte[] responseBody = byteOut.toByteArray(); 203     Log.d(MmsService.TAG, "HTTP: response size=" 204         + (responseBody != null ? responseBody.length : 0)); 205     return responseBody; </pre> <p><a href="https://android.googlesource.com/platform/packages/apps/Messaging/+nougat-mr1-release/src/android/support/v7/mms/MmsHttpClient.java">https://android.googlesource.com/platform/packages/apps/Messaging/+nougat-mr1-release/src/android/support/v7/mms/MmsHttpClient.java</a></p>

## Exhibit D - Claim Chart for U.S. Patent No. 9,467,838 Against HTC

US9467838	Exemplary Supporting Evidence Regarding Accused Products
	<pre> 38  * Request to send an MMS 39  */ 40  class SendRequest extends MmsRequest { 41      // Max send response PDU size in bytes (exceeding this may cause problem with 42      // system intent delivery). 43      private static final int MAX_SEND_RESPONSE_SIZE = 1000 * 1024; 44 45      private byte[] mPduData; 46 47      SendRequest(final String locationUri, final Uri pduUri, final PendingIntent sentIntent) { 48          super(locationUri, pduUri, sentIntent); 49      } 50 51      @Override 52      protected boolean loadRequest(final Context context, final Bundle mmsConfig) { 53          mPduData = readPduFromContentUri( 54              context, 55              mPduUri, 56              mmsConfig.getInt( 57                  CarrierConfigValuesLoader.CONFIG_MAX_MESSAGE_SIZE, 58                  CarrierConfigValuesLoader.CONFIG_MAX_MESSAGE_SIZE_DEFAULT)); 59          return (mPduData != null); 60      } 61 62      @Override 63      protected boolean transferResponse(final Context context, final Intent fillIn, 64          final byte[] response) { 65          // SendConf pdus are always small and can be included in the intent 66          if (response != null &amp;&amp; fillIn != null) { 67              if (response.length &gt; MAX_SEND_RESPONSE_SIZE) { 68                  // If the response PDU is too large, it won't be able to fit in 69                  // the PendingIntent to be transferred via system IPC. 70                  return false; 71              } 72              fillIn.putExtra(SmsManager.EXTRA_MMS_DATA, response); 73          } 74          return true; 75      } </pre>

**Exhibit D - Claim Chart for U.S. Patent No. 9,467,838 Against HTC**

US9467838	Exemplary Supporting Evidence Regarding Accused Products
	<p data-bbox="541 237 1619 302"><a href="https://android.googlesource.com/platform/packages/apps/Messaging+/nougat-mr1-release/src/android/support/v7/mms/SendRequest.java">https://android.googlesource.com/platform/packages/apps/Messaging+/nougat-mr1-release/src/android/support/v7/mms/SendRequest.java</a></p> <pre data-bbox="562 358 1003 391">public static LocationRequest create ()</pre> <p data-bbox="548 423 1052 448">Create a location request with default parameters.</p> <p data-bbox="548 480 1661 537">Default parameters are for a block accuracy, slowly updated location. It can then be adjusted as required by the applications before passing to the <code>FusedLocationProviderApi</code>.</p> <p data-bbox="573 561 657 586"><b>Returns</b></p> <ul data-bbox="579 610 835 634" style="list-style-type: none"><li>• a new location request</li></ul> <p data-bbox="541 651 1818 680"><a href="https://developers.google.com/android/reference/com/google/android/gms/location/LocationRequest">https://developers.google.com/android/reference/com/google/android/gms/location/LocationRequest</a></p>

**Exhibit D - Claim Chart for U.S. Patent No. 9,467,838 Against HTC**

US9467838	Exemplary Supporting Evidence Regarding Accused Products
	<p data-bbox="552 245 1770 282"><b>public static final int PRIORITY_BALANCED_POWER_ACCURACY</b></p> <p data-bbox="552 313 1199 337">Used with <code>setPriority(int)</code> to request "block" level accuracy.</p> <p data-bbox="552 370 1667 427">Block level accuracy is considered to be about 100 meter accuracy. Using a coarse accuracy such as this often consumes less power.</p> <p data-bbox="552 456 753 480">Constant Value: 102</p> <p data-bbox="552 537 1770 574"><b>public static final int PRIORITY_HIGH_ACCURACY</b></p> <p data-bbox="552 605 1360 630">Used with <code>setPriority(int)</code> to request the most accurate locations available.</p> <p data-bbox="552 662 984 686">This will return the finest location available.</p> <p data-bbox="552 716 753 740">Constant Value: 100</p> <p data-bbox="552 797 1770 834"><b>public static final int PRIORITY_LOW_POWER</b></p> <p data-bbox="552 865 1178 889">Used with <code>setPriority(int)</code> to request "city" level accuracy.</p> <p data-bbox="552 922 1759 979">City level accuracy is considered to be about 10km accuracy. Using a coarse accuracy such as this often consumes less power.</p> <p data-bbox="552 1008 753 1032">Constant Value: 104</p> <p data-bbox="537 1049 1818 1073"><a href="https://developers.google.com/android/reference/com/google/android/gms/location/LocationRequest">https://developers.google.com/android/reference/com/google/android/gms/location/LocationRequest</a></p>

## Exhibit D - Claim Chart for U.S. Patent No. 9,467,838 Against HTC

US9467838	Exemplary Supporting Evidence Regarding Accused Products
	<p data-bbox="556 248 1774 285"><code>public Task&lt;Location&gt; getLastLocation ()</code></p> <p data-bbox="556 313 1129 337">Returns the best most recent location currently available.</p> <p data-bbox="556 370 1719 428">If a location is not available, which should happen very rarely, null will be returned. The best accuracy available while respecting the location permissions will be returned.</p> <p data-bbox="556 461 1759 519">This method provides a simplified way to get location. It is particularly well suited for applications that do not require an accurate location and that do not want to maintain extra logic for location updates.</p> <p data-bbox="556 578 1774 615"><code>public Task&lt;LocationAvailability&gt; getLocationAvailability ()</code></p> <p data-bbox="556 643 1713 701">Returns the availability of location data. When <code>isLocationAvailable()</code> returns true, then the location returned by <code>getLastLocation()</code> will be reasonably up to date within the hints specified by the active <code>LocationRequest</code>s.</p> <p data-bbox="556 734 1495 758">If the client isn't connected to Google Play services and the request times out, null is returned.</p> <p data-bbox="556 790 1696 849">Note it's always possible for <code>getLastLocation()</code> to return null even when this method returns true (e.g. location settings were disabled between calls).</p> <p data-bbox="535 865 1902 928"><a href="https://developers.google.com/android/reference/com/google/android/gms/location/FusedLocationProviderClient">https://developers.google.com/android/reference/com/google/android/gms/location/FusedLocationProviderClient</a></p>

**Exhibit D - Claim Chart for U.S. Patent No. 9,467,838 Against HTC**

US9467838	Exemplary Supporting Evidence Regarding Accused Products						
	<p data-bbox="546 240 1774 326"> <code>public Task&lt;Void&gt; requestLocationUpdates (LocationRequest request, LocationCallback callback, Looper looper)</code> </p> <p data-bbox="546 354 1291 380">Requests location updates with a callback on the specified Looper thread.</p> <p data-bbox="546 410 1711 472">This method is suited for the foreground use cases. For background use cases, the <code>PendingIntent</code> version of the method is recommended, see <code>requestLocationUpdates(LocationRequest, PendingIntent)</code>.</p> <p data-bbox="546 503 1396 529">Any previous LocationRequests registered on this LocationListener will be replaced.</p> <p data-bbox="546 560 1711 654">This call will keep the Google Play services connection active, so make sure to call <code>removeLocationUpdates(LocationCallback)</code> when you no longer need it, otherwise you lose the benefits of the automatic connection management.</p> <p data-bbox="546 685 1774 711">Callbacks for <code>LocationCallback</code> will be made on the specified thread, which must already be a prepared looper thread.</p> <p data-bbox="546 735 682 761"><b>Parameters</b></p> <table border="1" data-bbox="546 789 1774 1008"> <tbody> <tr> <td data-bbox="546 789 655 857"><b>request</b></td> <td data-bbox="655 789 1774 857">The location request for the updates.</td> </tr> <tr> <td data-bbox="546 857 655 925"><b>callback</b></td> <td data-bbox="655 857 1774 925">The callback for the location updates.</td> </tr> <tr> <td data-bbox="546 925 655 1008"><b>looper</b></td> <td data-bbox="655 925 1774 1008">The Looper object whose message queue will be used to implement the callback mechanism, or null to make callbacks on the calling thread.</td> </tr> </tbody> </table> <p data-bbox="546 1023 1911 1089"> <a href="https://developers.google.com/android/reference/com/google/android/gms/location/FusedLocationProviderClient">https://developers.google.com/android/reference/com/google/android/gms/location/FusedLocationProviderClient</a> </p>	<b>request</b>	The location request for the updates.	<b>callback</b>	The callback for the location updates.	<b>looper</b>	The Looper object whose message queue will be used to implement the callback mechanism, or null to make callbacks on the calling thread.
<b>request</b>	The location request for the updates.						
<b>callback</b>	The callback for the location updates.						
<b>looper</b>	The Looper object whose message queue will be used to implement the callback mechanism, or null to make callbacks on the calling thread.						

Exhibit D - Claim Chart for U.S. Patent No. 9,467,838 Against HTC

US9467838	Exemplary Supporting Evidence Regarding Accused Products				
	<p data-bbox="556 240 1766 326">public Task&lt;Void&gt; requestLocationUpdates (LocationRequest request, PendingIntent callbackIntent)</p> <p data-bbox="556 354 1291 378">Requests location updates with a callback on the specified PendingIntent.</p> <p data-bbox="556 410 1755 537">This method is suited for the background use cases, more specifically for receiving location updates, even when the app has been killed by the system. In order to do so, use a PendingIntent for a started service. For foreground use cases, the LocationCallback version of the method is recommended, see requestLocationUpdates(LocationRequest, LocationCallback, Looper).</p> <p data-bbox="556 570 1749 630">Any previously registered requests that have the same PendingIntent (as defined by equals(Object)) will be replaced by this request.</p> <p data-bbox="556 662 1755 755">Both LocationResult and LocationAvailability are sent to the given PendingIntent. You can extract data from an Intent using hasResult(Intent), extractResult(Intent), hasLocationAvailability(Intent), and extractLocationAvailability(Intent).</p> <p data-bbox="569 781 693 805"><b>Parameters</b></p> <table border="1" data-bbox="556 833 1766 971"> <tr> <td data-bbox="556 833 863 902">request</td> <td data-bbox="863 833 1766 902">The location request for the updates.</td> </tr> <tr> <td data-bbox="556 902 863 971">callbackIntent</td> <td data-bbox="863 902 1766 971">A pending intent to be sent for each location update.</td> </tr> </table> <p data-bbox="569 997 653 1021"><b>Returns</b></p> <ul data-bbox="577 1044 1381 1068" style="list-style-type: none"> <li>• a Task for the call, check isSuccessful() to determine if it was successful.</li> </ul> <p data-bbox="537 1081 1902 1146"><a href="https://developers.google.com/android/reference/com/google/android/gms/location/FusedLocationProviderClient">https://developers.google.com/android/reference/com/google/android/gms/location/FusedLocationProviderClient</a></p>	request	The location request for the updates.	callbackIntent	A pending intent to be sent for each location update.
request	The location request for the updates.				
callbackIntent	A pending intent to be sent for each location update.				



**Exhibit D - Claim Chart for U.S. Patent No. 9,467,838 Against HTC**

US9467838	Exemplary Supporting Evidence Regarding Accused Products						
	<p data-bbox="558 245 1761 277"><code>public void onLocationAvailability (LocationAvailability locationAvailability)</code></p> <p data-bbox="548 310 1192 334">Called when there is a change in the availability of location data.</p> <p data-bbox="548 367 1761 561">When <code>isLocationAvailable()</code> returns <code>false</code> you can assume that location will not be returned in <code>onLocationResult(LocationResult)</code> until something changes in the device's settings or environment. Even when <code>isLocationAvailable()</code> returns <code>true</code> the <code>onLocationResult(LocationResult)</code> may not always be called regularly, however the device location is known and both the most recently delivered location and <code>getLastLocation(GoogleApiClient)</code> will be reasonably up to date given the hints specified by the active <code>LocationRequest</code> s.</p> <p data-bbox="569 586 695 610"><b>Parameters</b></p> <table border="1" data-bbox="548 643 1761 708"> <tr> <td data-bbox="558 651 984 699"><code>locationAvailability</code></td> <td data-bbox="995 651 1751 699">The current status of location availability.</td> </tr> </table> <p data-bbox="558 756 1761 789"><code>public void onLocationResult (LocationResult result)</code></p> <p data-bbox="548 821 1077 846">Called when device location information is available.</p> <p data-bbox="548 878 1682 943">The most recent location returned by <code>getLastLocation()</code> is not guaranteed to be immediately fresh, but will be reasonably up to date given the hints specified by the active <code>LocationRequest</code> s.</p> <p data-bbox="569 967 695 992"><b>Parameters</b></p> <table border="1" data-bbox="548 1024 1761 1089"> <tr> <td data-bbox="558 1032 789 1081"><code>result</code></td> <td data-bbox="800 1032 1751 1081">The latest location result available.</td> </tr> </table> <p data-bbox="537 1105 1829 1130"><a href="https://developers.google.com/android/reference/com/google/android/gms/location/LocationCallback">https://developers.google.com/android/reference/com/google/android/gms/location/LocationCallback</a></p> <p data-bbox="558 1146 1761 1179"><code>public abstract void onLocationChanged (Location location)</code></p> <p data-bbox="548 1211 936 1235">Called when the location has changed.</p> <p data-bbox="569 1260 695 1284"><b>Parameters</b></p> <table border="1" data-bbox="548 1317 1761 1382"> <tr> <td data-bbox="558 1325 947 1373"><code>location</code></td> <td data-bbox="957 1325 1751 1373">The updated location.</td> </tr> </table>	<code>locationAvailability</code>	The current status of location availability.	<code>result</code>	The latest location result available.	<code>location</code>	The updated location.
<code>locationAvailability</code>	The current status of location availability.						
<code>result</code>	The latest location result available.						
<code>location</code>	The updated location.						

**Exhibit D - Claim Chart for U.S. Patent No. 9,467,838 Against HTC**

US9467838	Exemplary Supporting Evidence Regarding Accused Products
	<p><a href="https://developers.google.com/android/reference/com/google/android/gms/location/LocationListener">https://developers.google.com/android/reference/com/google/android/gms/location/LocationListener</a></p> <p>Public Constructors</p> <hr/> <p>public <b>MapView</b> (<b>Context</b> context)</p> <p>public <b>MapView</b> (<b>Context</b> context, <b>AttributeSet</b> attrs)</p> <p>public <b>MapView</b> (<b>Context</b> context, <b>AttributeSet</b> attrs, int defStyle)</p> <p>public <b>MapView</b> (<b>Context</b> context, <b>GoogleMapOptions</b> options)</p> <p><a href="https://developers.google.com/android/reference/com/google/android/gms/maps/MapView">https://developers.google.com/android/reference/com/google/android/gms/maps/MapView</a></p>

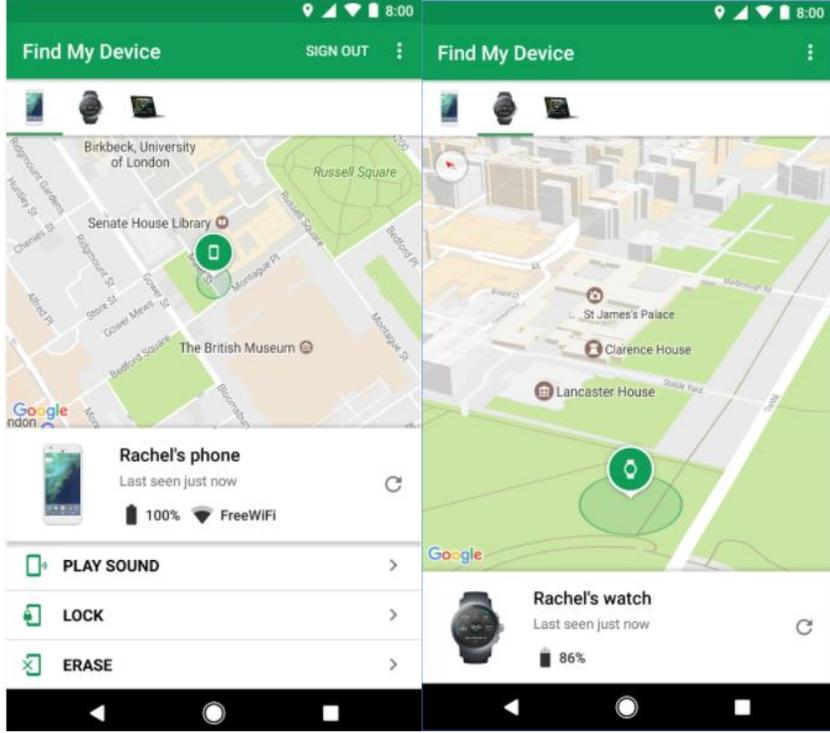
**Exhibit D - Claim Chart for U.S. Patent No. 9,467,838 Against HTC**

US9467838	Exemplary Supporting Evidence Regarding Accused Products		
	<p><code>public void <b>getMapAsync</b> (<b>OnMapReadyCallback</b> callback)</code></p> <p>Returns a non-null instance of the <code>GoogleMap</code>, ready to be used.</p> <p>Note that:</p> <ul style="list-style-type: none"> <li>• This method must be called from the main thread.</li> <li>• The callback will be executed in the main thread.</li> <li>• In the case where Google Play services is not installed on the user's device, the callback will not be triggered until the user installs it.</li> <li>• The <code>GoogleMap</code> object provided by the callback is non-null.</li> </ul> <p>Parameters</p> <table border="1" data-bbox="554 688 1761 753"> <tr> <td><code>callback</code></td> <td>The callback object that will be triggered when the map is ready to be used.</td> </tr> </table> <p><code>public final void <b>onCreate</b> (<b>Bundle</b> savedInstanceState)</code></p> <p>You must call this method from the parent Activity/Fragment's corresponding method.</p> <p><a href="https://developers.google.com/android/reference/com/google/android/gms/maps/MapView">https://developers.google.com/android/reference/com/google/android/gms/maps/MapView</a></p>	<code>callback</code>	The callback object that will be triggered when the map is ready to be used.
<code>callback</code>	The callback object that will be triggered when the map is ready to be used.		
<p>[1F] receiving, from the second server, the second georeferenced map data;</p>	<p>HTC infringes directly and/or indirectly by performing, inducing others to perform, and/or contributing to the performance of: receiving, from the second server, the second georeferenced map data.</p> <p><b><u>Regarding Find My Device</u></b> and Android Device Manager, the user, via the first device, or the device itself receives second map data from a second server. The received second map data occurs responsive to user input (e.g., zoom, drag, pan, change focus, change map type, refresh or reload request device or symbol selection, another device or user selection, change in position of first device, change in position of a second device). Alternatively, the second map data may be received responsive to an automatic and/or pre-determined control caused by an instruction from within the first device or from the one or more second devices, e.g. a refresh. The received second map data includes an update to the first data or a</p>		

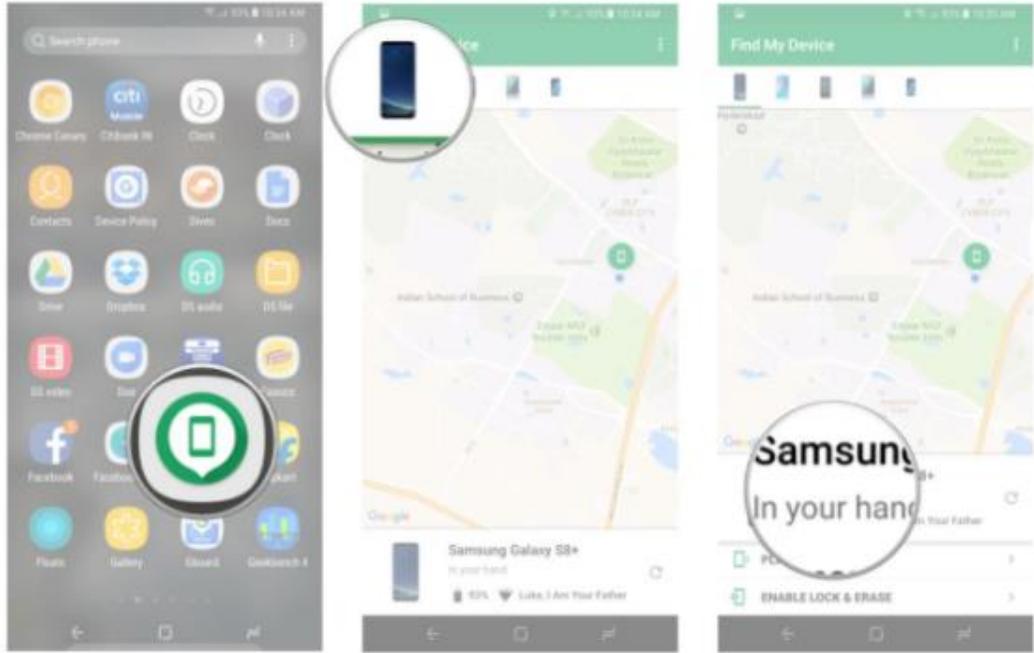
**Exhibit D - Claim Chart for U.S. Patent No. 9,467,838 Against HTC**

<b>US9467838</b>	<b>Exemplary Supporting Evidence Regarding Accused Products</b>
	<p>replacement of the first data. The second map data may come from one of many sources and servers. The second map data includes georeferenced data.</p> <p><b><u>Regarding Google Maps</u></b>, Google Latitude, Google Plus, Google Hangouts, Google Messages, Google Allo, Google Duo, Google Chrome, and Android Messenger, the user, via the first device, or the device itself receives second map data from a second server. The received second map data occurs responsive to user input (e.g., zoom, drag, pan, change focus, refresh or reload request, change map type, device or symbol selection, another device or user selection, change in position of first device, change in position of a second device). Alternatively, the second map data may be received responsive to an automatic and/or pre-determined control caused by an instruction from within the first device or from the one or more second devices, e.g. a refresh. The received second map data includes an update to the first data or a replacement of the first data. The second map data may come from one of many sources and servers. The second map data includes georeferenced data.</p> <p><b><u>Exemplary Support for Find My Device:</u></b></p>

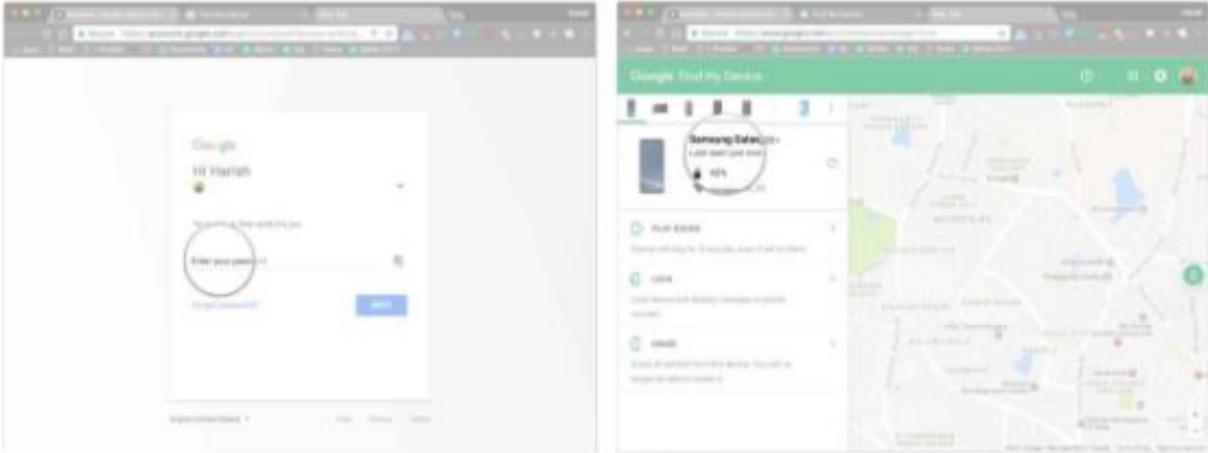
**Exhibit D - Claim Chart for U.S. Patent No. 9,467,838 Against HTC**

US9467838	Exemplary Supporting Evidence Regarding Accused Products
	 <p><a href="https://play.google.com/store/apps/details?id=com.google.android.apps.adm&amp;hl=en">https://play.google.com/store/apps/details?id=com.google.android.apps.adm&amp;hl=en</a></p>

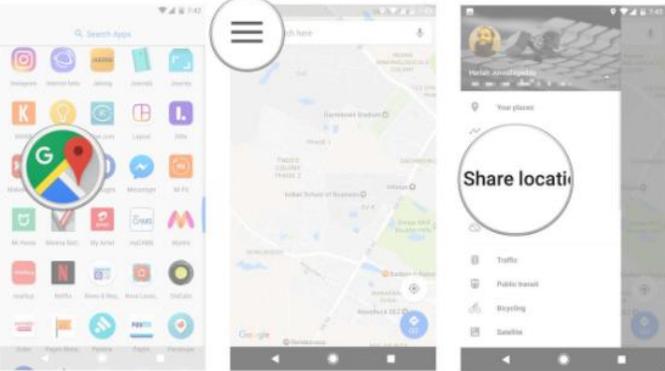
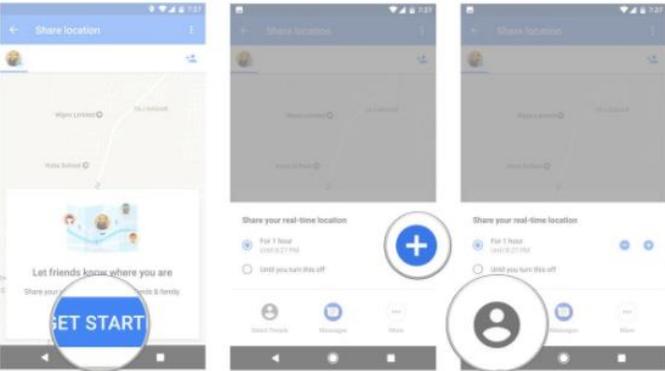
**Exhibit D - Claim Chart for U.S. Patent No. 9,467,838 Against HTC**

US9467838	Exemplary Supporting Evidence Regarding Accused Products
	<p>Once you're signed in to Find My Device, you'll see a map with your current location as well as the make and model of your phone, and two options — Play Sound, and Enable Lock &amp; Erase. Hitting the latter option will allow you to start using the Lock and Erase functions.</p> <p>If you've signed into more than one phone, you can select a particular device by browsing the list at the top of the screen.</p> <ol style="list-style-type: none"> <li>1. Open <b>Find My Device</b> from your home screen or app drawer.</li> <li>2. Select your phone from the list of devices at the top of the screen.</li> <li>3. See if your phone is discoverable.</li> </ol>  <p><a href="https://www.androidcentral.com/find-my-device">https://www.androidcentral.com/find-my-device</a></p>

**Exhibit D - Claim Chart for U.S. Patent No. 9,467,838 Against HTC**

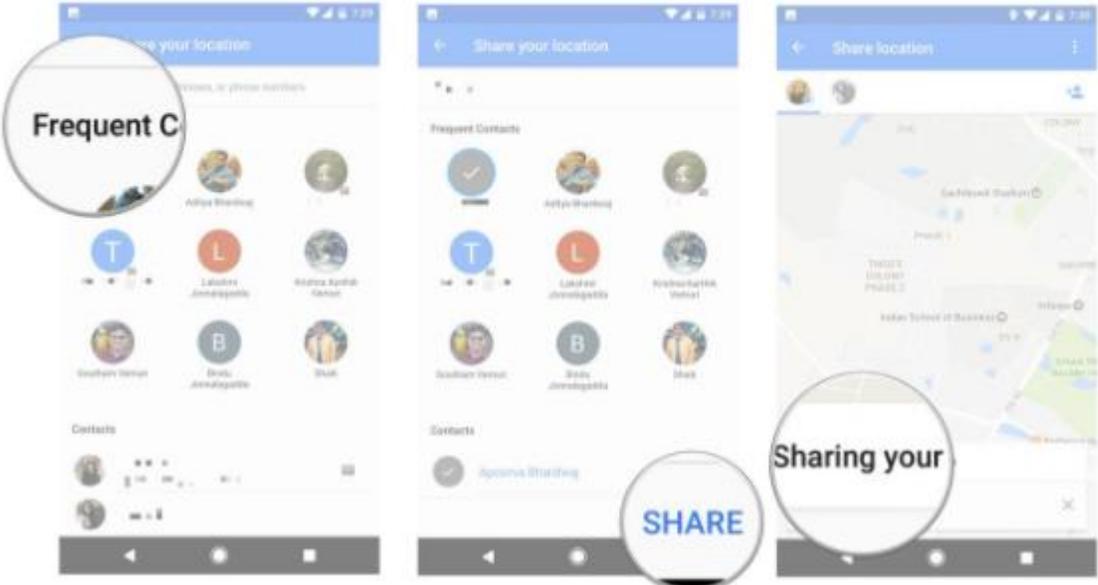
US9467838	Exemplary Supporting Evidence Regarding Accused Products
	<h2 data-bbox="548 313 1686 367">How to locate your phone over the internet</h2> <p data-bbox="548 407 1692 548">If you've lost your phone, you can remotely locate it through the <a href="#">Find My Device website</a>. You'll need to sign in to the Google account that was used to set up Find My Device. It takes a few seconds, but the service should be able to track your phone. Alternatively, you can also do a Google search for "find my phone" to locate your handset.</p> <ol data-bbox="539 607 1031 760" style="list-style-type: none"><li>1. Head to the <a href="#">Find My Device website</a>.</li><li>2. Sign in to your <a href="#">Google account</a>.</li><li>3. Check if your device is <b>visible</b>.</li></ol> <div data-bbox="585 802 1793 1255"></div> <p data-bbox="533 1284 1157 1317"><a href="https://www.androidcentral.com/find-my-device">https://www.androidcentral.com/find-my-device</a></p> <p data-bbox="533 1357 1052 1390"><b><u>Exemplary Support for Google Maps:</u></b></p>

**Exhibit D - Claim Chart for U.S. Patent No. 9,467,838 Against HTC**

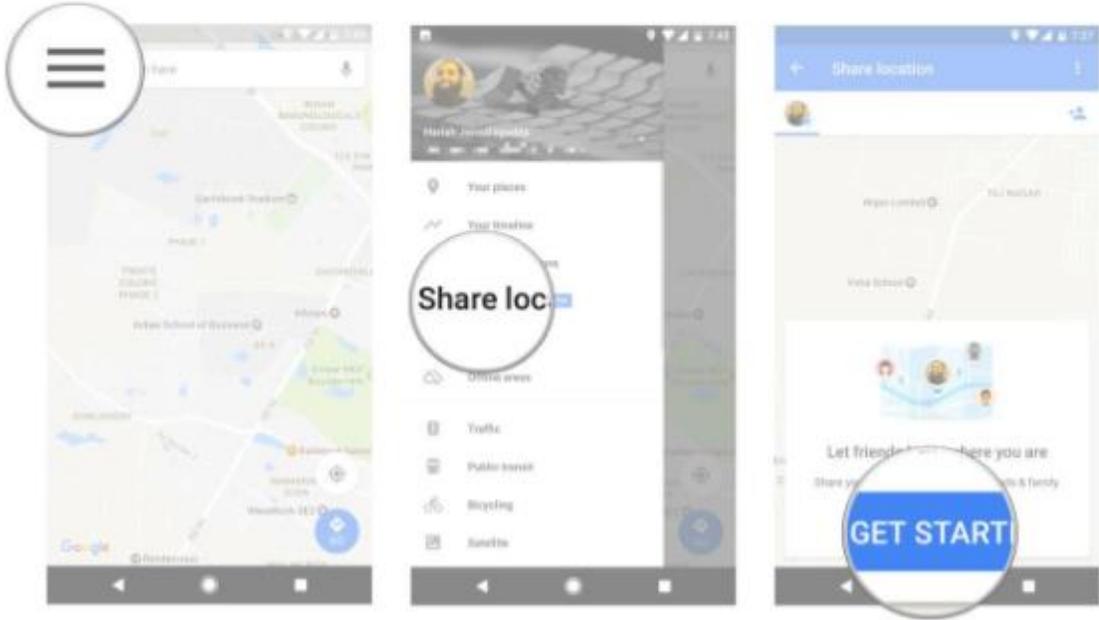
US9467838	Exemplary Supporting Evidence Regarding Accused Products
	<p data-bbox="541 238 1178 269"><b>How to share your location in Google Maps</b></p> <ol data-bbox="541 302 1157 386" style="list-style-type: none"> <li>1. Open <b>Google Maps</b> from the app drawer or the home screen.</li> <li>2. Tap the <b>hamburger menu</b> (the three horizontal lines) on the top left corner of the screen.</li> <li>3. Select <b>Share location</b>.</li> </ol>  <ol data-bbox="541 824 1188 932" style="list-style-type: none"> <li>4. Tap <b>Get Started</b>.</li> <li>5. Use the <b>+</b> icon to select a time period or select the <b>Until you turn this off</b> setting to share your location indefinitely.</li> <li>6. Tap <b>Select People</b>.</li> </ol>  <p data-bbox="541 1338 1381 1369"><a href="https://www.androidcentral.com/how-share-location-google-maps">https://www.androidcentral.com/how-share-location-google-maps</a></p>



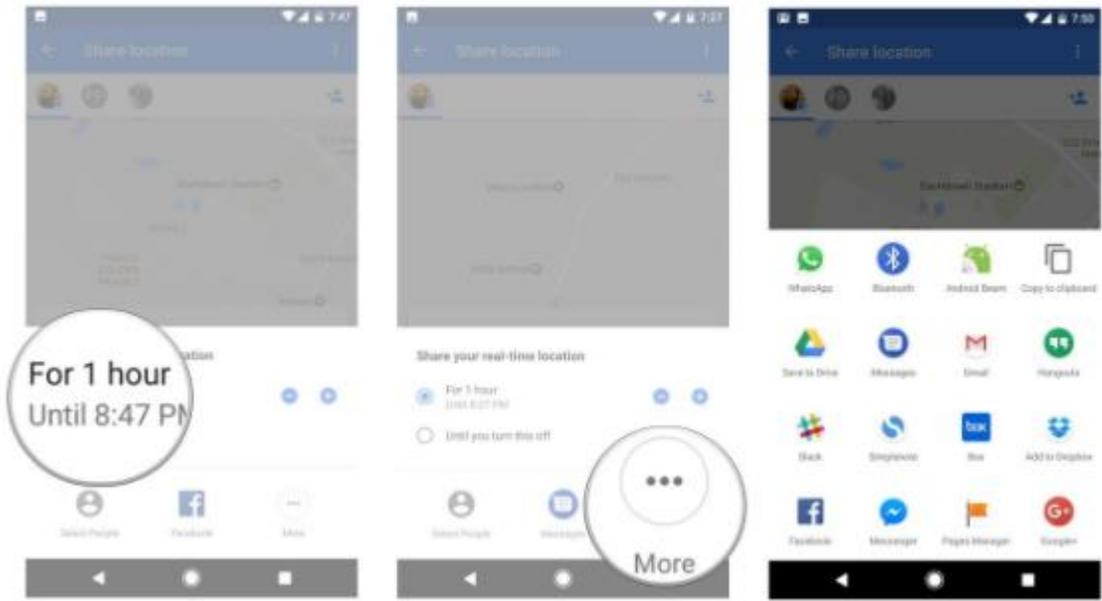
**Exhibit D - Claim Chart for U.S. Patent No. 9,467,838 Against HTC**

US9467838	Exemplary Supporting Evidence Regarding Accused Products
	<p>7. You'll see a list of your frequent contacts at the top, along with a full list of contacts. Pick the contacts by tapping their name.</p> <p>8. Once you've selected the contacts you want to share your location to, tap <b>Share</b>.</p> <p>9. You'll see a message saying that the selected contact can view your location.</p>  <p><a href="https://www.androidcentral.com/how-share-location-google-maps">https://www.androidcentral.com/how-share-location-google-maps</a></p>

**Exhibit D - Claim Chart for U.S. Patent No. 9,467,838 Against HTC**

US9467838	Exemplary Supporting Evidence Regarding Accused Products
	<h2 data-bbox="548 245 1276 289">How to create a shareable link</h2> <p data-bbox="548 334 1486 358">You can also create a link and use it to share your location easily. Here's how to do it:</p> <ol data-bbox="541 412 1262 548" style="list-style-type: none"><li>1. Tap the <b>hamburger menu</b> on the top left corner of the screen.</li><li>2. Select <b>Share location</b>.</li><li>3. Tap <b>Get Started</b>.</li></ol>  <p data-bbox="533 1230 1381 1263"><a href="https://www.androidcentral.com/how-share-location-google-maps">https://www.androidcentral.com/how-share-location-google-maps</a></p>

**Exhibit D - Claim Chart for U.S. Patent No. 9,467,838 Against HTC**

US9467838	Exemplary Supporting Evidence Regarding Accused Products
	<p data-bbox="541 245 1234 272">4. Select the amount of time you want to share your location.</p> <p data-bbox="541 302 709 329">5. Tap More.</p> <p data-bbox="541 358 1661 418">6. Select your app of choice to create and send a unique URL that broadcasts your current location. You can email it, send the link via Messenger, or even tweet it to the intended recipient.</p>  <p data-bbox="541 1084 1381 1112"><a href="https://www.androidcentral.com/how-share-location-google-maps">https://www.androidcentral.com/how-share-location-google-maps</a></p>

## Exhibit D - Claim Chart for U.S. Patent No. 9,467,838 Against HTC

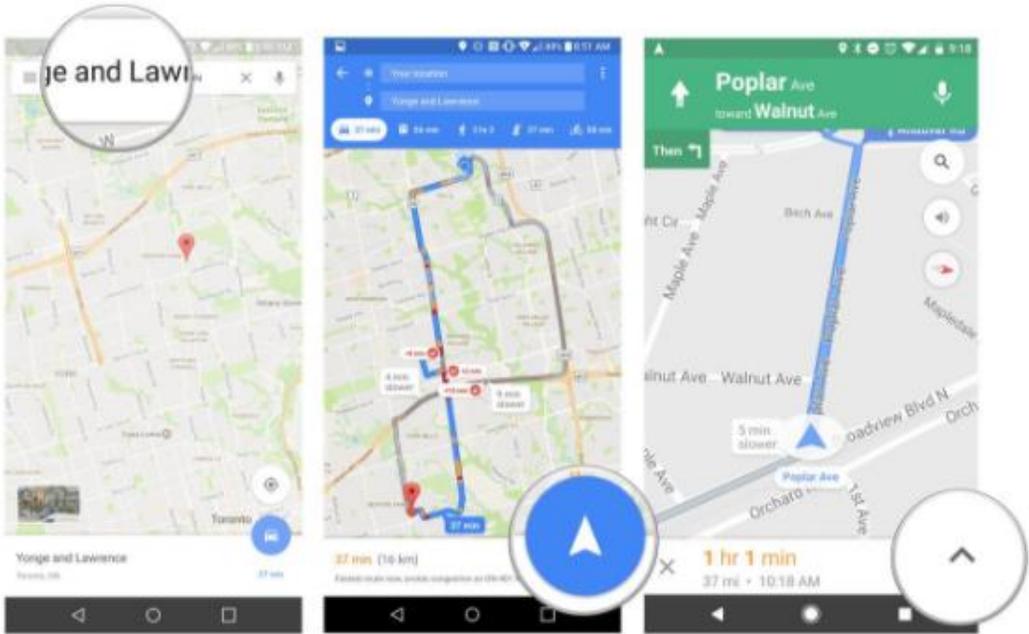
US9467838	Exemplary Supporting Evidence Regarding Accused Products
	<h2 data-bbox="548 240 1451 337">How to share your navigation directions while you walk, drive or transit</h2> <p data-bbox="548 375 1581 464">One of the best ways to share your location with someone is during a drive, walk or transit. If you're meeting a friend or family member somewhere, or navigating towards their home, Google Maps lets you share your location with them for the duration of the trip. It's magic!</p> <ol data-bbox="548 513 1419 646" style="list-style-type: none"><li>1. In the <b>search bar</b> enter your destination.</li><li>2. Pick your navigation type (drive, transit, walk) and press the <b>blue navigate button</b>.</li><li>3. Tap the arrow next to the <b>time-to-destination</b> number at the bottom of the screen.</li></ol>  <p data-bbox="533 1328 1381 1360"><a href="https://www.androidcentral.com/how-share-location-google-maps">https://www.androidcentral.com/how-share-location-google-maps</a></p>

Exhibit D - Claim Chart for U.S. Patent No. 9,467,838 Against HTC

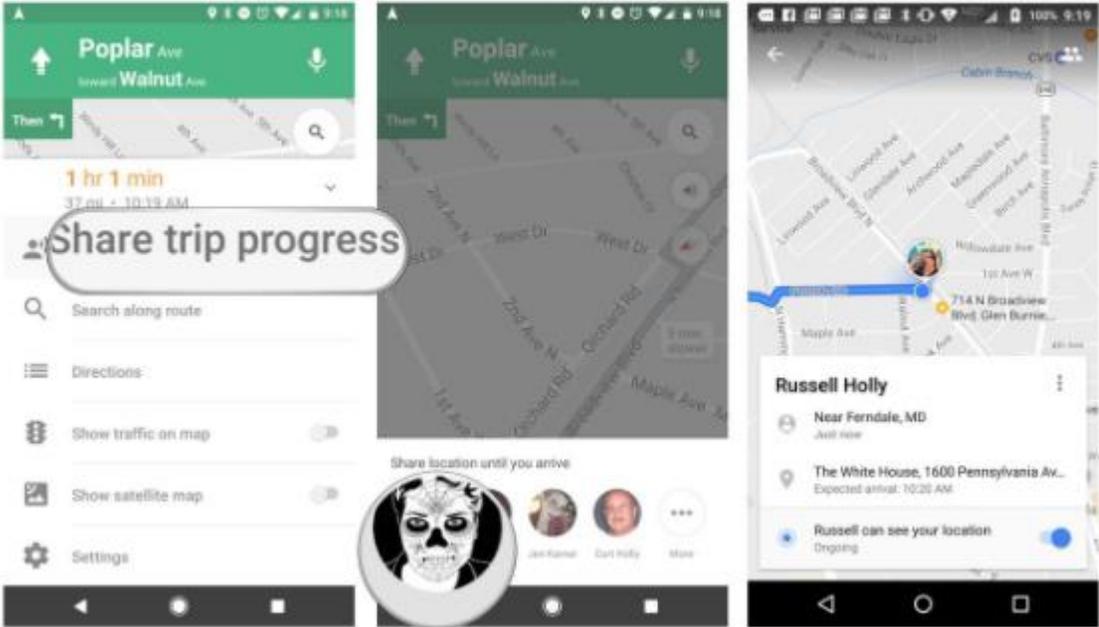
US9467838	Exemplary Supporting Evidence Regarding Accused Products
	<p data-bbox="552 277 861 305">4. Tap Share trip progress.</p> <p data-bbox="552 332 1171 360">5. Choose one or more contacts to share trip progress.</p>  <p data-bbox="552 1063 1365 1091">You can also stop sharing your location with someone before a trip ends.</p> <p data-bbox="535 1101 1381 1128"><a href="https://www.androidcentral.com/how-share-location-google-maps">https://www.androidcentral.com/how-share-location-google-maps</a></p>

Exhibit D - Claim Chart for U.S. Patent No. 9,467,838 Against HTC

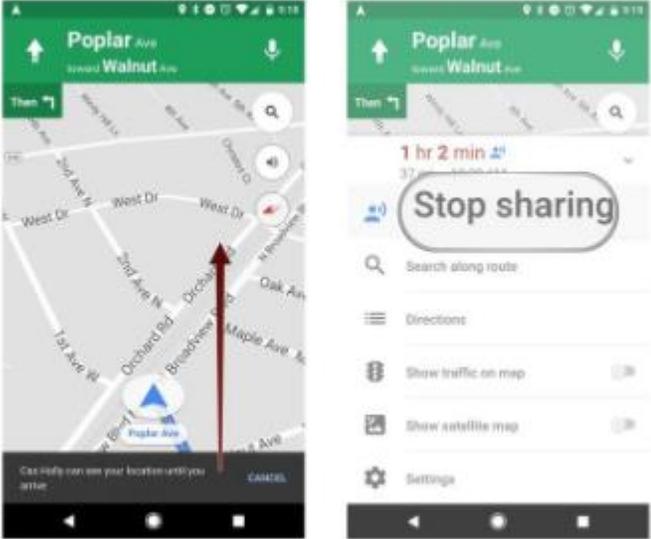
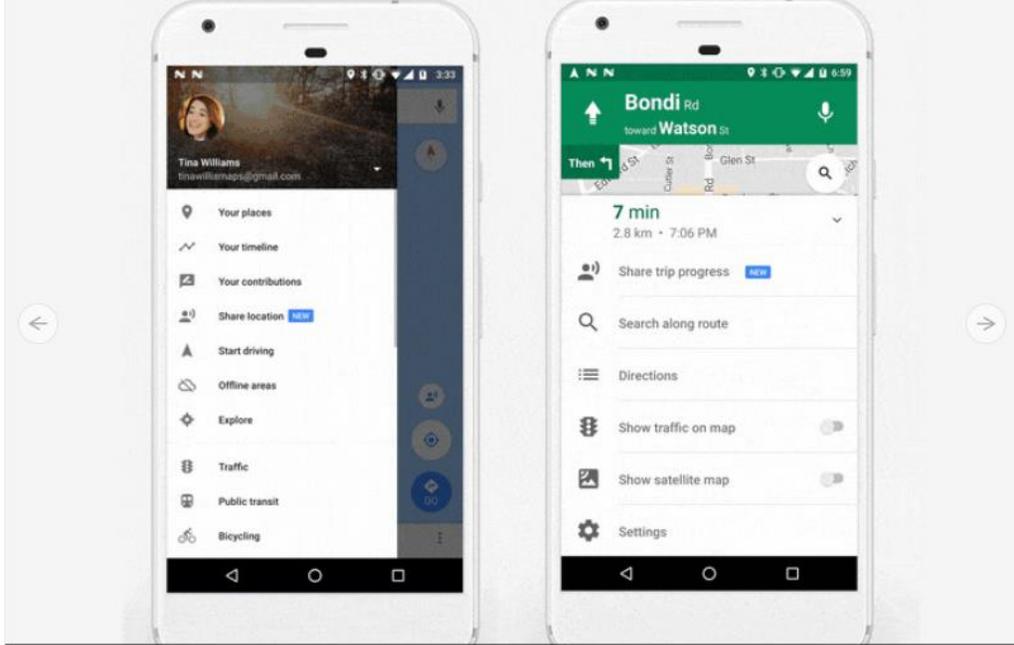
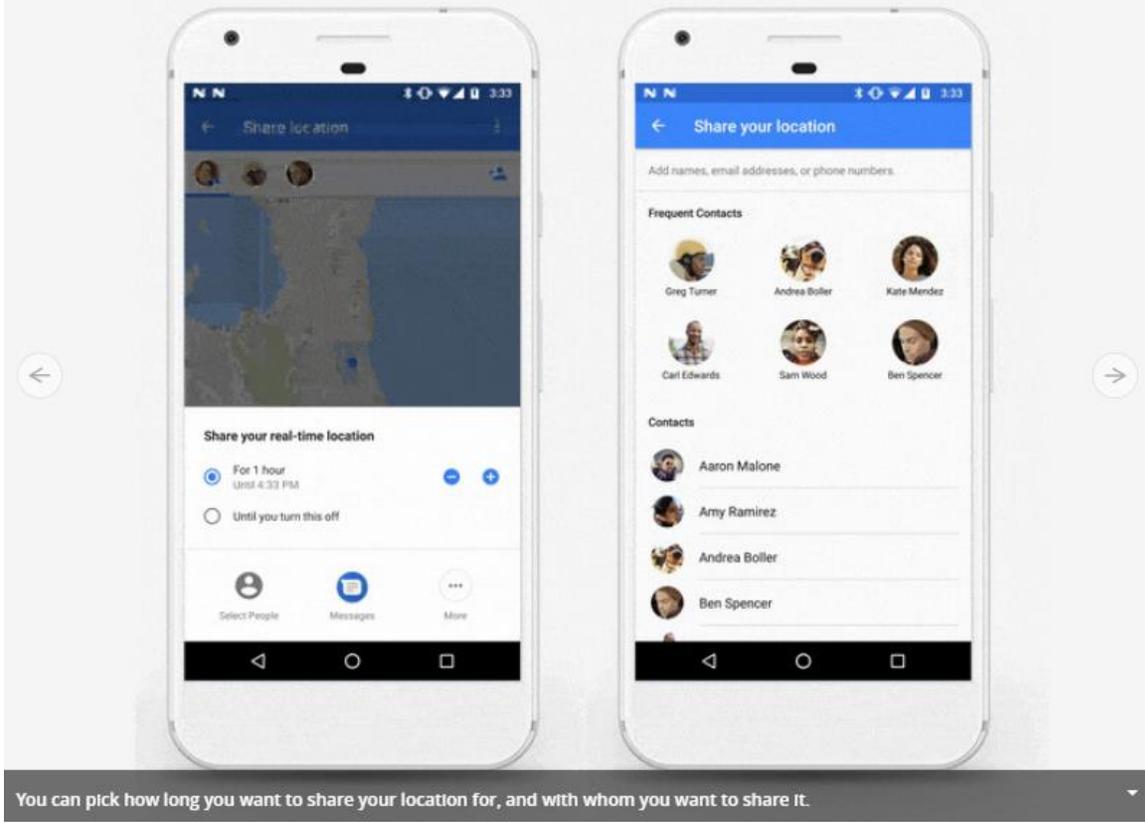
US9467838	Exemplary Supporting Evidence Regarding Accused Products
	<ol style="list-style-type: none"><li data-bbox="562 245 1493 272">1. Tap the arrow next to the time-to-destination number at the bottom of the screen.</li><li data-bbox="562 302 793 329">2. Tap Stop sharing.</li></ol> <div data-bbox="772 383 1423 922"></div> <p data-bbox="562 976 659 1003">That's It!</p> <p data-bbox="562 1045 1633 1073">Are you excited that location sharing is back in Google Maps? How often do you use the feature?</p> <p data-bbox="533 1084 1381 1117"><a href="https://www.androidcentral.com/how-share-location-google-maps">https://www.androidcentral.com/how-share-location-google-maps</a></p>

Exhibit D - Claim Chart for U.S. Patent No. 9,467,838 Against HTC

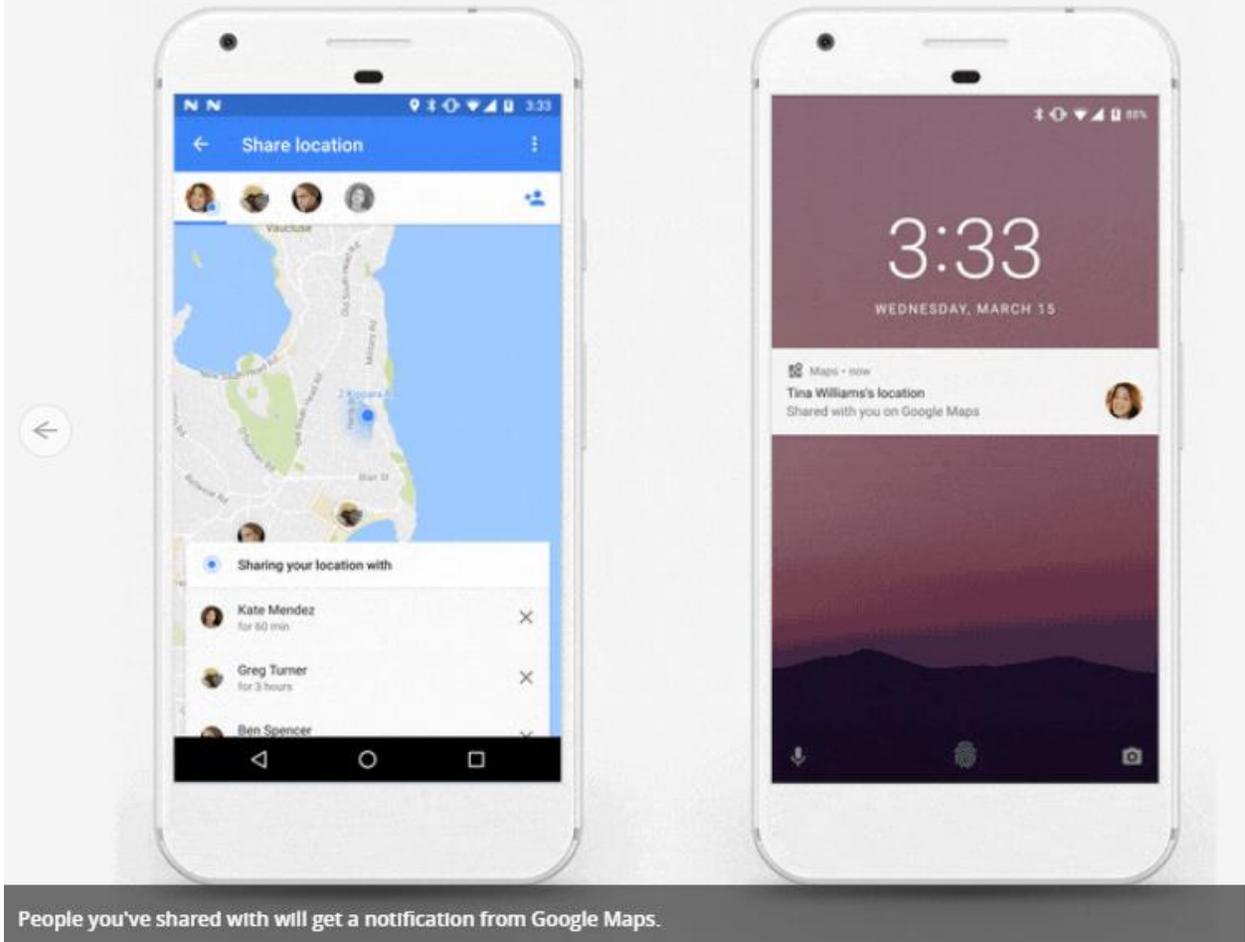
US9467838	Exemplary Supporting Evidence Regarding Accused Products
	 <p data-bbox="535 893 1549 950">Ways to start location sharing: (left) from the app navigation drawer, and (right) from the menu, you can drag up while navigating somewhere.</p> <p data-bbox="535 958 1549 982"><a href="https://arstechnica.com/gadgets/2017/03/location-sharing-finally-returns-to-google-maps/">https://arstechnica.com/gadgets/2017/03/location-sharing-finally-returns-to-google-maps/</a></p>

**Exhibit D - Claim Chart for U.S. Patent No. 9,467,838 Against HTC**

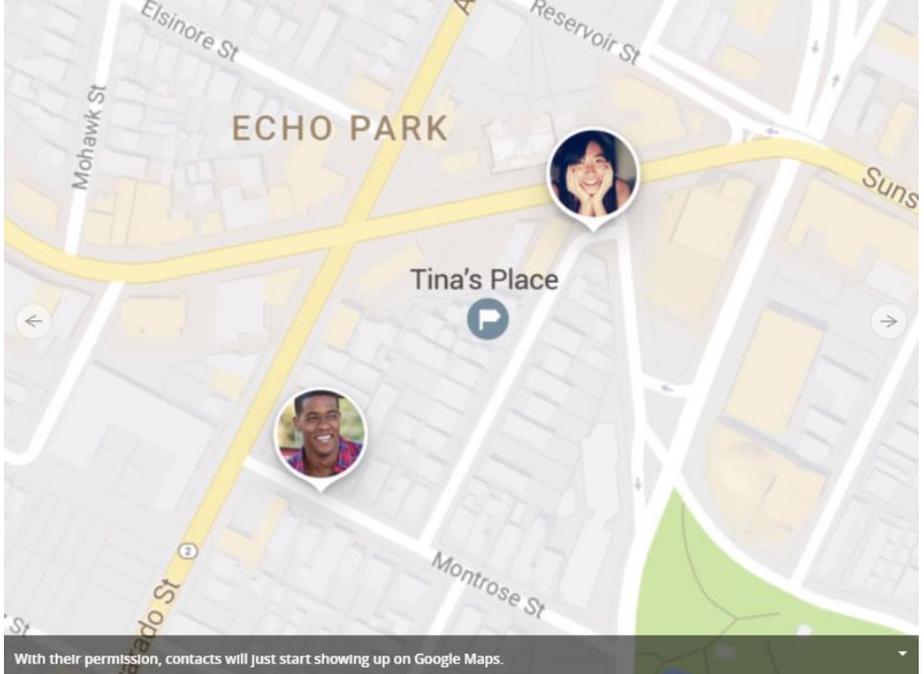
US9467838	Exemplary Supporting Evidence Regarding Accused Products
	 <p data-bbox="535 1023 1680 1063">You can pick how long you want to share your location for, and with whom you want to share it.</p> <p data-bbox="535 1063 1680 1096"><a href="https://arstechnica.com/gadgets/2017/03/location-sharing-finally-returns-to-google-maps/">https://arstechnica.com/gadgets/2017/03/location-sharing-finally-returns-to-google-maps/</a></p>



**Exhibit D - Claim Chart for U.S. Patent No. 9,467,838 Against HTC**

US9467838	Exemplary Supporting Evidence Regarding Accused Products
	 <p data-bbox="535 1144 1197 1177">People you've shared with will get a notification from Google Maps.</p> <p data-bbox="535 1185 1680 1226"><a href="https://arstechnica.com/gadgets/2017/03/location-sharing-finally-returns-to-google-maps/">https://arstechnica.com/gadgets/2017/03/location-sharing-finally-returns-to-google-maps/</a></p>

**Exhibit D - Claim Chart for U.S. Patent No. 9,467,838 Against HTC**

US9467838	Exemplary Supporting Evidence Regarding Accused Products
	 <p data-bbox="535 917 1680 950"><a href="https://arstechnica.com/gadgets/2017/03/location-sharing-finally-returns-to-google-maps/">https://arstechnica.com/gadgets/2017/03/location-sharing-finally-returns-to-google-maps/</a></p> <p data-bbox="535 982 871 1015"><b><u>Exemplary Screenshots:</u></b></p> <p data-bbox="535 1055 892 1088">See, e.g., 1D and 1E above.</p> <p data-bbox="535 1128 882 1161"><b><u>Exemplary Source Code:</u></b></p> <p data-bbox="535 1201 1911 1380">The above functionality is performed at least in part by the following publicly available source code and/or source code that invokes or is invoked by the following source code (or a substantially similar copy compiled and loaded onto the Accused Products by HTC). AGIS reserves the right to supplement these contentions with additional source code as discovery progresses and as additional source code is made available.</p>

## Exhibit D - Claim Chart for U.S. Patent No. 9,467,838 Against HTC

US9467838	Exemplary Supporting Evidence Regarding Accused Products
	<pre>44 * Class that sends chat message via SMS. 45 * 46 * The interface emulates a blocking sending similar to making an HTTP request. 47 * It calls the SmsManager to send a (potentially multipart) message and waits 48 * on the sent status on each part. The waiting has a timeout so it won't wait 49 * forever. Once the sent status of all parts received, the call returns. 50 * A successful sending requires success status for all parts. Otherwise, we 51 * pick the highest level of failure as the error for the whole message, which 52 * is used to determine if we need to retry the sending. 53 */ 54 public class SmsSender { 55     private static final String TAG = LogUtil.BUGLE_TAG; 56 57     public static final String EXTRA_PART_ID = "part_id"; 58 59     /* 60     * A map for pending sms messages. The key is the random request UUID. 61     */ 62     private static ConcurrentHashMap&lt;Uri, SendResult&gt; sPendingMessageMap = 63         new ConcurrentHashMap&lt;Uri, SendResult&gt;(); 64 65     private static final Random RANDOM = new Random(); 66 67     // Whether we should send multipart SMS as separate messages 68     private static Boolean sSendMultipartSmsAsSeparateMessages = null; 69</pre> <p><a href="https://android.googlesource.com/platform/packages/apps/Messaging/+nougat-mr1-release/src/com/android/messaging/sms/SmsSender.java">https://android.googlesource.com/platform/packages/apps/Messaging/+nougat-mr1-release/src/com/android/messaging/sms/SmsSender.java</a></p>

## Exhibit D - Claim Chart for U.S. Patent No. 9,467,838 Against HTC

US9467838	Exemplary Supporting Evidence Regarding Accused Products
	<pre> 253 // Actually sending the message using SmsManager 254 private static void sendInternal(final Context context, final int subId, String dest, 255     final ArrayList&lt;String&gt; messages, final String serviceCenter, 256     final boolean requireDeliveryReport, final Uri messageUri) throws SmsException { 257     Assert.notNull(context); 258     final SmsManager smsManager = PhoneUtils.get(subId).getSmsManager(); 259     final int messageCount = messages.size(); 260     final ArrayList&lt;PendingIntent&gt; deliveryIntents = new ArrayList&lt;PendingIntent&gt;(messageCount); 261     final ArrayList&lt;PendingIntent&gt; sentIntents = new ArrayList&lt;PendingIntent&gt;(messageCount); 262     for (int i = 0; i &lt; messageCount; i++) { 263         // Make pending intents different for each message part 264         final int partId = (messageCount &lt;= 1 ? 0 : i + 1); 265         if (requireDeliveryReport &amp;&amp; (i == (messageCount - 1))) { 266             // TODO we only care about the delivery status of the last part 267             // Shall we have better tracking of delivery status of all parts? 268             deliveryIntents.add(PendingIntent.getBroadcast( 269                 context, 270                 partId, 271                 getSendStatusIntent(context, SendStatusReceiver.MESSAGE_DELIVERED_ACTION, 272                     messageUri, partId, subId), 273                 0/*flag*/)); 274         } else { 275             deliveryIntents.add(null); 276         } 277         sentIntents.add(PendingIntent.getBroadcast( 278             context, 279             partId, 280             getSendStatusIntent(context, SendStatusReceiver.MESSAGE_SENT_ACTION, 281                 messageUri, partId, subId), 282             0/*flag*/)); 283     } 284     if (sSendMultipartSmsAsSeparateMessages == null) { 285         sSendMultipartSmsAsSeparateMessages = MmsConfig.get(subId) 286             .getSendMultipartSmsAsSeparateMessages(); 287     } </pre> <p><a href="https://android.googlesource.com/platform/packages/apps/Messaging/+nougat-mr1-release/src/com/android/messaging/sms/SmsSender.java">https://android.googlesource.com/platform/packages/apps/Messaging/+nougat-mr1-release/src/com/android/messaging/sms/SmsSender.java</a></p>

## Exhibit D - Claim Chart for U.S. Patent No. 9,467,838 Against HTC

US9467838	Exemplary Supporting Evidence Regarding Accused Products
	<pre>288     try { 289         if (sSendMultipartSmsAsSeparateMessages) { 290             // If multipart sms is not supported, send them as separate messages 291             for (int i = 0; i &lt; messageCount; i++) { 292                 smsManager.sendTextMessage(dest, 293                     serviceCenter, 294                     messages.get(i), 295                     sentIntents.get(i), 296                     deliveryIntents.get(i)); 297             } 298         } else { 299             smsManager.sendMultipartTextMessage( 300                 dest, serviceCenter, messages, sentIntents, deliveryIntents); 301         } 302     } catch (final Exception e) { 303         throw new SmsException("SmsSender: caught exception in sending " + e); 304     } 305 }</pre> <p><a href="https://android.googlesource.com/platform/packages/apps/Messaging+/nougat-mr1-release/src/com/android/messaging/sms/SmsSender.java">https://android.googlesource.com/platform/packages/apps/Messaging+/nougat-mr1-release/src/com/android/messaging/sms/SmsSender.java</a></p> <pre>56  * Class that receives incoming SMS messages through android.provider.Telephony.SMS_RECEIVED 57  * 58  * This class serves two purposes: 59  * - Process phone verification SMS messages 60  * - Handle SMS messages when the user has enabled us to be the default SMS app (Pre-KLP) 61  */ 62  public final class SmsReceiver extends BroadcastReceiver { 63      private static final String TAG = LogUtil.BUGLE_TAG; 64 65      private static ArrayList&lt;Pattern&gt; sIgnoreSmsPatterns; 66  }</pre> <p><a href="https://android.googlesource.com/platform/packages/apps/Messaging+/nougat-mr1-release/src/com/android/messaging/receiver/SmsReceiver.java">https://android.googlesource.com/platform/packages/apps/Messaging+/nougat-mr1-release/src/com/android/messaging/receiver/SmsReceiver.java</a></p>

2013

## Exhibit D - Claim Chart for U.S. Patent No. 9,467,838 Against HTC

US9467838	Exemplary Supporting Evidence Regarding Accused Products
	<pre> 200     public static void deliverSmsMessages(final Context context, final int subId, 201         final int errorCode, final android.telephony.SmsMessage[] messages) { 202         final ContentValues messageValues = 203             MmsUtils.parseReceivedSmsMessage(context, messages, errorCode); 204 205         LogUtil.v(TAG, "SmsReceiver.deliverSmsMessages"); 206 207         final long nowInMillis = System.currentTimeMillis(); 208         final long receivedTimestampMs = MmsUtils.getMessageDate(messages[0], nowInMillis); 209 210         messageValues.put(Sms.Inbox.DATE, receivedTimestampMs); 211         // Default to unread and unseen for us but ReceiveSmsMessageAction will override 212         // seen for the telephony db. 213         messageValues.put(Sms.Inbox.READ, 0); 214         messageValues.put(Sms.Inbox.SEEN, 0); 215         if (OsUtil.isAtLeastL_MR1()) { 216             messageValues.put(Sms.SUBSCRIPTION_ID, subId); 217         } 218 219         if (messages[0].getMessageClass() == android.telephony.SmsMessage.MessageClass.CLASS_0    220             DebugUtils.debugClassZeroSmsEnabled()) { 221             Factory.get().getUIIntents().launchClassZeroActivity(context, messageValues); 222         } else { 223             final ReceiveSmsMessageAction action = new ReceiveSmsMessageAction(messageValues); 224             action.start(); 225         } 226     } </pre> <p><a href="https://android.googlesource.com/platform/packages/apps/Messaging/+nougat-mr1-release/src/com/android/messaging/receiver/SmsReceiver.java">https://android.googlesource.com/platform/packages/apps/Messaging/+nougat-mr1-release/src/com/android/messaging/receiver/SmsReceiver.java</a></p>

## Exhibit D - Claim Chart for U.S. Patent No. 9,467,838 Against HTC

US9467838	Exemplary Supporting Evidence Regarding Accused Products
	<pre>228     @Override 229     public void onReceive(final Context context, final Intent intent) { 230         LogUtil.v(TAG, "SmsReceiver.onReceive " + intent); 231         // On KLP+ we only take delivery of SMS messages in SmsDeliverReceiver. 232         if (PhoneUtils.getDefault().isSmsEnabled()) { 233             final String action = intent.getAction(); 234             if (OsUtil.isSecondaryUser() &amp;&amp; 235                 (Telephony.Sms.Intents.SMS_RECEIVED_ACTION.equals(action)    236                  // TODO: update this with the actual constant from Telephony 237                   "android.provider.Telephony.MMS_DOWNLOADED".equals(action))) { 238                 postNewMessageSecondaryUserNotification(); 239             } else if (!OsUtil.isAtLeastKLP()) { 240                 deliverSmsIntent(context, intent); 241             } 242         } 243     }</pre> <p><a href="https://android.googlesource.com/platform/packages/apps/Messaging/+nougat-mr1-release/src/com/android/messaging/receiver/SmsReceiver.java">https://android.googlesource.com/platform/packages/apps/Messaging/+nougat-mr1-release/src/com/android/messaging/receiver/SmsReceiver.java</a></p>

2015

## Exhibit D - Claim Chart for U.S. Patent No. 9,467,838 Against HTC

US9467838	Exemplary Supporting Evidence Regarding Accused Products
	<pre>52  * Class that sends chat message via MMS. 53  * 54  * The interface emulates a blocking send similar to making an HTTP request. 55  */ 56  public class MmsSender { 57      private static final String TAG = LogUtil.BUGLE_TAG; 58 59      /** 60       * Send an MMS message. 61       * 62       * @param context Context 63       * @param messageUri The unique URI of the message for identifying it during sending 64       * @param sendReq The SendReq PDU of the message 65       * @throws MmsFailureException 66       */ 67      public static void sendMms(final Context context, final int subId, final Uri messageUri, 68                               final SendReq sendReq, final Bundle sentIntentExtras) throws MmsFailureException { 69          sendMms(context, 70                 subId, 71                 messageUri, 72                 null /* locationUri */, 73                 sendReq, 74                 true /* responseImportant */, 75                 sentIntentExtras); 76      }</pre> <p><a href="https://android.googlesource.com/platform/packages/apps/Messaging+/nougat-mr1-release/src/com/android/messaging/sms/MmsSender.java">https://android.googlesource.com/platform/packages/apps/Messaging+/nougat-mr1-release/src/com/android/messaging/sms/MmsSender.java</a></p>



## Exhibit D - Claim Chart for U.S. Patent No. 9,467,838 Against HTC

US9467838	Exemplary Supporting Evidence Regarding Accused Products
	<pre> 240      * Download an MMS message. 241      * 242      * @param context Context 243      * @param contentLocation The url of the MMS message 244      * @throws MmsFailureException 245      * @throws InvalidHeaderValueException 246      */ 247      public static void downloadMms(final Context context, final int subId, 248          final String contentLocation, Bundle extras) throws MmsFailureException, 249          InvalidHeaderValueException { 250          final Uri requestUri = Uri.parse(contentLocation); 251          final Uri contentUri = MmsFileProvider.buildRawMmsUri(); 252 253          final Intent downloadedIntent = new Intent(SendStatusReceiver.MMS_DOWNLOADED_ACTION, 254              requestUri, 255              context, 256              SendStatusReceiver.class); 257          downloadedIntent.putExtra(SendMessageAction.EXTRA_CONTENT_URI, contentUri); 258          if (extras != null) { 259              downloadedIntent.putExtras(extras); 260          } 261          final PendingIntent downloadedPendingIntent = PendingIntent.getBroadcast( 262              context, 263              0 /*request code*/, 264              downloadedIntent, 265              PendingIntent.FLAG_UPDATE_CURRENT); 266 267          MmsManager.downloadMultimediaMessage(subId, context, contentLocation, contentUri, 268              downloadedPendingIntent); 269      } </pre> <p><a href="https://android.googlesource.com/platform/packages/apps/Messaging/+nougat-mr1-release/src/com/android/messaging/sms/MmsSender.java">https://android.googlesource.com/platform/packages/apps/Messaging/+nougat-mr1-release/src/com/android/messaging/sms/MmsSender.java</a></p>

## Exhibit D - Claim Chart for U.S. Patent No. 9,467,838 Against HTC

US9467838	Exemplary Supporting Evidence Regarding Accused Products
	<pre> 97      * Execute an MMS HTTP request, either a POST (sending) or a GET (downloading) 98      * 99      * @param urlString The request URL, for sending it is usually the MMSC, and for downloading 100     *           it is the message URL 101     * @param pdu For POST (sending) only, the PDU to send 102     * @param method HTTP method, POST for sending and GET for downloading 103     * @param isProxySet Is there a proxy for the MMSC 104     * @param proxyHost The proxy host 105     * @param proxyPort The proxy port 106     * @param mmsConfig The MMS config to use 107     * @param userAgent The user agent header value 108     * @param uaProfUrl The UA Prof URL header value 109     * @return The HTTP response body 110     * @throws MmsHttpException For any failures 111     */ 112     public byte[] execute(String urlString, byte[] pdu, String method, boolean isProxySet, 113                          String proxyHost, int proxyPort, Bundle mmsConfig, String userAgent, String uaProfUrl) 114                          throws MmsHttpException { 115         Log.d(MmsService.TAG, "HTTP: " + method + " " + Utils.redactUrlForNonVerbose(urlString) 116             + (isProxySet ? (" proxy=" + proxyHost + ":" + proxyPort) : "") 117             + ", PDU size=" + (pdu != null ? pdu.length : 0)); 118         checkMethod(method); 119         HttpURLConnection connection = null; 120         try { 121             Proxy proxy = Proxy.NO_PROXY; 122             if (isProxySet) { 123                 proxy = new Proxy(Proxy.Type.HTTP, new InetSocketAddress(proxyHost, proxyPort)); 124             } 125             final URL url = new URL(urlString); 126             // Now get the connection 127             connection = (HttpURLConnection) url.openConnection(proxy); 128             connection.setDoInput(true); 129             connection.setConnectTimeout( 130                 mmsConfig.getInt(CarrierConfigValuesLoader.CONFIG_HTTP_SOCKET_TIMEOUT, 131                                 CarrierConfigValuesLoader.CONFIG_HTTP_SOCKET_TIMEOUT_DEFAULT)); </pre> <p><a href="https://android.googlesource.com/platform/packages/apps/Messaging/+/nougat-mr1-release/src/android/support/v7/mms/MmsHttpClient.java">https://android.googlesource.com/platform/packages/apps/Messaging/+/nougat-mr1-release/src/android/support/v7/mms/MmsHttpClient.java</a></p>

2018

## Exhibit D - Claim Chart for U.S. Patent No. 9,467,838 Against HTC

US9467838	Exemplary Supporting Evidence Regarding Accused Products
	<pre> 132 // ----- COMMON HEADERS ----- 133 // Header: Accept 134 connection.setRequestProperty(HEADER_ACCEPT, HEADER_VALUE_ACCEPT); 135 // Header: Accept-Language 136 connection.setRequestProperty( 137     HEADER_ACCEPT_LANGUAGE, getCurrentAcceptLanguage(Locale.getDefault())); 138 // Header: User-Agent 139 Log.i(MmsService.TAG, "HTTP: User-Agent=" + userAgent); 140 connection.setRequestProperty(HEADER_USER_AGENT, userAgent); 141 // Header: x-wap-profile 142 final String uaProfUrlTagName = mmsConfig.getString( 143     CarrierConfigValuesLoader.CONFIG_UA_PROF_TAG_NAME, 144     CarrierConfigValuesLoader.CONFIG_UA_PROF_TAG_NAME_DEFAULT); 145 if (uaProfUrl != null) { 146     Log.i(MmsService.TAG, "HTTP: UaProfUrl=" + uaProfUrl); 147     connection.setRequestProperty(uaProfUrlTagName, uaProfUrl); 148 } 149 // Add extra headers specified by mms_config.xml's httpparams 150 addExtraHeaders(connection, mmsConfig); 151 // Different stuff for GET and POST 152 if (METHOD_POST.equals(method)) { 153     if (pdu == null    pdu.length &lt; 1) { 154         Log.e(MmsService.TAG, "HTTP: empty pdu"); 155         throw new MmsHttpException(0/*statusCode*/, "Sending empty PDU"); 156     } 157     connection.setDoOutput(true); 158     connection.setRequestMethod(METHOD_POST); 159     if (mmsConfig.getBoolean( 160         CarrierConfigValuesLoader.CONFIG_SUPPORT_HTTP_CHARSET_HEADER, 161         CarrierConfigValuesLoader.CONFIG_SUPPORT_HTTP_CHARSET_HEADER_DEFAULT)) { 162         connection.setRequestProperty(HEADER_CONTENT_TYPE, 163             HEADER_VALUE_CONTENT_TYPE_WITH_CHARSET); 164     } else { 165         connection.setRequestProperty(HEADER_CONTENT_TYPE, 166             HEADER_VALUE_CONTENT_TYPE_WITHOUT_CHARSET); </pre> <p><a href="https://android.googlesource.com/platform/packages/apps/Messaging+/nougat-mr1-release/src/android/support/v7/mms/MmsHttpClient.java">https://android.googlesource.com/platform/packages/apps/Messaging+/nougat-mr1-release/src/android/support/v7/mms/MmsHttpClient.java</a></p>

2019

## Exhibit D - Claim Chart for U.S. Patent No. 9,467,838 Against HTC

US9467838	Exemplary Supporting Evidence Regarding Accused Products
	<pre> 167         } 168         if (Log.isLoggable(MmsService.TAG, Log.VERBOSE)) { 169             logHttpHeaders(connection.getRequestProperties()); 170         } 171         connection.setFixedLengthStreamingMode(pdu.length); 172         // Sending request body 173         final OutputStream out = 174             new BufferedOutputStream(connection.getOutputStream()); 175         out.write(pdu); 176         out.flush(); 177         out.close(); 178     } else if (METHOD_GET.equals(method)) { 179         if (Log.isLoggable(MmsService.TAG, Log.VERBOSE)) { 180             logHttpHeaders(connection.getRequestProperties()); 181         } 182         connection.setRequestMethod(METHOD_GET); 183     } 184     // Get response 185     final int responseCode = connection.getResponseCode(); 186     final String responseMessage = connection.getResponseMessage(); 187     Log.d(MmsService.TAG, "HTTP: " + responseCode + " " + responseMessage); 188     if (Log.isLoggable(MmsService.TAG, Log.VERBOSE)) { 189         logHttpHeaders(connection.getHeaderFields()); 190     } 191     if (responseCode / 100 != 2) { 192         throw new MmsHttpException(responseCode, responseMessage); 193     } 194     final InputStream in = new BufferedInputStream(connection.getInputStream()); 195     final ByteArrayOutputStream byteOut = new ByteArrayOutputStream(); 196     final byte[] buf = new byte[4096]; 197     int count = 0; 198     while ((count = in.read(buf)) &gt; 0) { 199         byteOut.write(buf, 0, count); 200     } 201     in.close(); 202     final byte[] responseBody = byteOut.toByteArray(); 203     Log.d(MmsService.TAG, "HTTP: response size=" 204         + (responseBody != null ? responseBody.length : 0)); 205     return responseBody; </pre> <p><a href="https://android.googlesource.com/platform/packages/apps/Messaging/+nougat-mr1-release/src/android/support/v7/mms/MmsHttpClient.java">https://android.googlesource.com/platform/packages/apps/Messaging/+nougat-mr1-release/src/android/support/v7/mms/MmsHttpClient.java</a></p>

## Exhibit D - Claim Chart for U.S. Patent No. 9,467,838 Against HTC

US9467838	Exemplary Supporting Evidence Regarding Accused Products
	<pre> 38  * Request to send an MMS 39  */ 40  class SendRequest extends MmsRequest { 41      // Max send response PDU size in bytes (exceeding this may cause problem with 42      // system intent delivery). 43      private static final int MAX_SEND_RESPONSE_SIZE = 1000 * 1024; 44 45      private byte[] mPduData; 46 47      SendRequest(final String locationUri, final Uri pduUri, final PendingIntent sentIntent) { 48          super(locationUri, pduUri, sentIntent); 49      } 50 51      @Override 52      protected boolean loadRequest(final Context context, final Bundle mmsConfig) { 53          mPduData = readPduFromContentUri( 54              context, 55              mPduUri, 56              mmsConfig.getInt( 57                  CarrierConfigValuesLoader.CONFIG_MAX_MESSAGE_SIZE, 58                  CarrierConfigValuesLoader.CONFIG_MAX_MESSAGE_SIZE_DEFAULT)); 59          return (mPduData != null); 60      } 61 62      @Override 63      protected boolean transferResponse(final Context context, final Intent fillIn, 64          final byte[] response) { 65          // SendConf pdus are always small and can be included in the intent 66          if (response != null &amp;&amp; fillIn != null) { 67              if (response.length &gt; MAX_SEND_RESPONSE_SIZE) { 68                  // If the response PDU is too large, it won't be able to fit in 69                  // the PendingIntent to be transferred via system IPC. 70                  return false; 71              } 72              fillIn.putExtra(SmsManager.EXTRA_MMS_DATA, response); 73          } 74          return true; 75      } </pre>

**Exhibit D - Claim Chart for U.S. Patent No. 9,467,838 Against HTC**

US9467838	Exemplary Supporting Evidence Regarding Accused Products
	<p data-bbox="537 235 1619 302"><a href="https://android.googlesource.com/platform/packages/apps/Messaging+/nougat-mr1-release/src/android/support/v7/mms/SendRequest.java">https://android.googlesource.com/platform/packages/apps/Messaging+/nougat-mr1-release/src/android/support/v7/mms/SendRequest.java</a></p> <pre data-bbox="562 358 999 391">public static LocationRequest create ()</pre> <p data-bbox="548 423 1052 448">Create a location request with default parameters.</p> <p data-bbox="548 480 1661 540">Default parameters are for a block accuracy, slowly updated location. It can then be adjusted as required by the applications before passing to the <code>FusedLocationProviderApi</code>.</p> <p data-bbox="573 565 653 589"><b>Returns</b></p> <ul data-bbox="579 610 835 634" style="list-style-type: none"><li>• a new location request</li></ul> <p data-bbox="537 651 1818 680"><a href="https://developers.google.com/android/reference/com/google/android/gms/location/LocationRequest">https://developers.google.com/android/reference/com/google/android/gms/location/LocationRequest</a></p>

**Exhibit D - Claim Chart for U.S. Patent No. 9,467,838 Against HTC**

US9467838	Exemplary Supporting Evidence Regarding Accused Products
	<p data-bbox="552 245 1770 282"><b>public static final int PRIORITY_BALANCED_POWER_ACCURACY</b></p> <p data-bbox="552 313 1199 337">Used with <code>setPriority(int)</code> to request "block" level accuracy.</p> <p data-bbox="552 370 1667 427">Block level accuracy is considered to be about 100 meter accuracy. Using a coarse accuracy such as this often consumes less power.</p> <p data-bbox="552 456 753 480">Constant Value: 102</p> <p data-bbox="552 532 1770 570"><b>public static final int PRIORITY_HIGH_ACCURACY</b></p> <p data-bbox="552 600 1360 625">Used with <code>setPriority(int)</code> to request the most accurate locations available.</p> <p data-bbox="552 657 984 682">This will return the finest location available.</p> <p data-bbox="552 711 753 735">Constant Value: 100</p> <p data-bbox="552 787 1770 824"><b>public static final int PRIORITY_LOW_POWER</b></p> <p data-bbox="552 855 1178 880">Used with <code>setPriority(int)</code> to request "city" level accuracy.</p> <p data-bbox="552 912 1759 969">City level accuracy is considered to be about 10km accuracy. Using a coarse accuracy such as this often consumes less power.</p> <p data-bbox="552 998 753 1023">Constant Value: 104</p> <p data-bbox="537 1044 1818 1073"><a href="https://developers.google.com/android/reference/com/google/android/gms/location/LocationRequest">https://developers.google.com/android/reference/com/google/android/gms/location/LocationRequest</a></p>

## Exhibit D - Claim Chart for U.S. Patent No. 9,467,838 Against HTC

US9467838	Exemplary Supporting Evidence Regarding Accused Products
	<p data-bbox="562 248 1772 285"><code>public Task&lt;Location&gt; getLastLocation ()</code></p> <p data-bbox="548 313 1125 337">Returns the best most recent location currently available.</p> <p data-bbox="548 370 1717 427">If a location is not available, which should happen very rarely, null will be returned. The best accuracy available while respecting the location permissions will be returned.</p> <p data-bbox="548 459 1759 516">This method provides a simplified way to get location. It is particularly well suited for applications that do not require an accurate location and that do not want to maintain extra logic for location updates.</p> <p data-bbox="562 578 1772 615"><code>public Task&lt;LocationAvailability&gt; getLocationAvailability ()</code></p> <p data-bbox="548 643 1713 699">Returns the availability of location data. When <code>isLocationAvailable()</code> returns true, then the location returned by <code>getLastLocation()</code> will be reasonably up to date within the hints specified by the active <code>LocationRequest</code>s.</p> <p data-bbox="548 732 1493 756">If the client isn't connected to Google Play services and the request times out, null is returned.</p> <p data-bbox="548 789 1696 846">Note it's always possible for <code>getLastLocation()</code> to return null even when this method returns true (e.g. location settings were disabled between calls).</p> <p data-bbox="537 862 1902 928"><a href="https://developers.google.com/android/reference/com/google/android/gms/location/FusedLocationProviderClient">https://developers.google.com/android/reference/com/google/android/gms/location/FusedLocationProviderClient</a></p>



## Exhibit D - Claim Chart for U.S. Patent No. 9,467,838 Against HTC

US9467838	Exemplary Supporting Evidence Regarding Accused Products						
	<p data-bbox="546 240 1772 326">public Task&lt;Void&gt; requestLocationUpdates (LocationRequest request, LocationCallback callback, Looper looper)</p> <p data-bbox="546 354 1297 380">Requests location updates with a callback on the specified Looper thread.</p> <p data-bbox="546 410 1709 472">This method is suited for the foreground use cases. For background use cases, the PendingIntent version of the method is recommended, see requestLocationUpdates(LocationRequest, PendingIntent).</p> <p data-bbox="546 503 1394 529">Any previous LocationRequests registered on this LocationListener will be replaced.</p> <p data-bbox="546 560 1709 654">This call will keep the Google Play services connection active, so make sure to call removeLocationUpdates(LocationCallback) when you no longer need it, otherwise you lose the benefits of the automatic connection management.</p> <p data-bbox="546 685 1766 711">Callbacks for LocationCallback will be made on the specified thread, which must already be a prepared looper thread.</p> <p data-bbox="546 735 688 761"><b>Parameters</b></p> <table border="1" data-bbox="546 789 1772 1008"> <tbody> <tr> <td data-bbox="546 789 655 857">request</td> <td data-bbox="655 789 1772 857">The location request for the updates.</td> </tr> <tr> <td data-bbox="546 857 655 925">callback</td> <td data-bbox="655 857 1772 925">The callback for the location updates.</td> </tr> <tr> <td data-bbox="546 925 655 1008">looper</td> <td data-bbox="655 925 1772 1008">The Looper object whose message queue will be used to implement the callback mechanism, or null to make callbacks on the calling thread.</td> </tr> </tbody> </table> <p data-bbox="546 1023 1902 1089"><a href="https://developers.google.com/android/reference/com/google/android/gms/location/FusedLocationProviderClient">https://developers.google.com/android/reference/com/google/android/gms/location/FusedLocationProviderClient</a></p>	request	The location request for the updates.	callback	The callback for the location updates.	looper	The Looper object whose message queue will be used to implement the callback mechanism, or null to make callbacks on the calling thread.
request	The location request for the updates.						
callback	The callback for the location updates.						
looper	The Looper object whose message queue will be used to implement the callback mechanism, or null to make callbacks on the calling thread.						

## Exhibit D - Claim Chart for U.S. Patent No. 9,467,838 Against HTC

US9467838	Exemplary Supporting Evidence Regarding Accused Products				
	<p data-bbox="556 240 1766 321">public <code>Task&lt;Void&gt; requestLocationUpdates</code> (<code>LocationRequest</code> request, <code>PendingIntent</code> callbackIntent)</p> <p data-bbox="556 354 1291 378">Requests location updates with a callback on the specified <code>PendingIntent</code>.</p> <p data-bbox="556 410 1755 540">This method is suited for the background use cases, more specifically for receiving location updates, even when the app has been killed by the system. In order to do so, use a <code>PendingIntent</code> for a started service. For foreground use cases, the <code>LocationCallback</code> version of the method is recommended, see <code>requestLocationUpdates(LocationRequest, LocationCallback, Looper)</code>.</p> <p data-bbox="556 573 1749 630">Any previously registered requests that have the same <code>PendingIntent</code> (as defined by <code>equals(Object)</code>) will be replaced by this request.</p> <p data-bbox="556 662 1755 751">Both <code>LocationResult</code> and <code>LocationAvailability</code> are sent to the given <code>PendingIntent</code>. You can extract data from an <code>Intent</code> using <code>hasResult(Intent)</code>, <code>extractResult(Intent)</code>, <code>hasLocationAvailability(Intent)</code>, and <code>extractLocationAvailability(Intent)</code>.</p> <p data-bbox="573 784 695 808"><b>Parameters</b></p> <table border="1" data-bbox="556 833 1766 971"> <tbody> <tr> <td data-bbox="556 833 863 898"><code>request</code></td> <td data-bbox="863 833 1766 898">The location request for the updates.</td> </tr> <tr> <td data-bbox="556 898 863 971"><code>callbackIntent</code></td> <td data-bbox="863 898 1766 971">A pending intent to be sent for each location update.</td> </tr> </tbody> </table> <p data-bbox="573 995 653 1019"><b>Returns</b></p> <ul data-bbox="573 1044 1381 1068" style="list-style-type: none"> <li>• a <code>Task</code> for the call, check <code>isSuccessful()</code> to determine if it was successful.</li> </ul> <p data-bbox="537 1076 1902 1141"><a href="https://developers.google.com/android/reference/com/google/android/gms/location/FusedLocationProviderClient">https://developers.google.com/android/reference/com/google/android/gms/location/FusedLocationProviderClient</a></p>	<code>request</code>	The location request for the updates.	<code>callbackIntent</code>	A pending intent to be sent for each location update.
<code>request</code>	The location request for the updates.				
<code>callbackIntent</code>	A pending intent to be sent for each location update.				

## Exhibit D - Claim Chart for U.S. Patent No. 9,467,838 Against HTC

US9467838	Exemplary Supporting Evidence Regarding Accused Products						
	<p data-bbox="558 245 1766 277">public void <b>onLocationAvailability</b> (<a href="#">LocationAvailability</a> locationAvailability)</p> <p data-bbox="548 310 1192 334">Called when there is a change in the availability of location data.</p> <p data-bbox="548 367 1766 561">When <a href="#">isLocationAvailable()</a> returns <code>false</code> you can assume that location will not be returned in <a href="#">onLocationResult(LocationResult)</a> until something changes in the device's settings or environment. Even when <a href="#">isLocationAvailable()</a> returns <code>true</code> the <a href="#">onLocationResult(LocationResult)</a> may not always be called regularly, however the device location is known and both the most recently delivered location and <a href="#">getLastLocation(GoogleApiClient)</a> will be reasonably up to date given the hints specified by the active <a href="#">LocationRequest</a> s.</p> <p data-bbox="569 586 695 610"><b>Parameters</b></p> <table border="1" data-bbox="548 643 1766 708"> <tr> <td data-bbox="558 651 982 699"><b>locationAvailability</b></td> <td data-bbox="989 651 1755 699">The current status of location availability.</td> </tr> </table> <p data-bbox="558 756 1766 789">public void <b>onLocationResult</b> (<a href="#">LocationResult</a> result)</p> <p data-bbox="548 821 1077 846">Called when device location information is available.</p> <p data-bbox="548 878 1682 943">The most recent location returned by <a href="#">getLastLocation()</a> is not guaranteed to be immediately fresh, but will be reasonably up to date given the hints specified by the active <a href="#">LocationRequest</a> s.</p> <p data-bbox="569 967 695 992"><b>Parameters</b></p> <table border="1" data-bbox="548 1024 1766 1089"> <tr> <td data-bbox="558 1032 789 1081"><b>result</b></td> <td data-bbox="795 1032 1755 1081">The latest location result available.</td> </tr> </table> <p data-bbox="537 1105 1829 1130"><a href="https://developers.google.com/android/reference/com/google/android/gms/location/LocationCallback">https://developers.google.com/android/reference/com/google/android/gms/location/LocationCallback</a></p> <p data-bbox="558 1146 1766 1179">public abstract void <b>onLocationChanged</b> (<a href="#">Location</a> location)</p> <p data-bbox="548 1211 936 1235">Called when the location has changed.</p> <p data-bbox="569 1260 695 1284"><b>Parameters</b></p> <table border="1" data-bbox="548 1317 1766 1382"> <tr> <td data-bbox="558 1325 947 1373"><b>location</b></td> <td data-bbox="953 1325 1755 1373">The updated location.</td> </tr> </table>	<b>locationAvailability</b>	The current status of location availability.	<b>result</b>	The latest location result available.	<b>location</b>	The updated location.
<b>locationAvailability</b>	The current status of location availability.						
<b>result</b>	The latest location result available.						
<b>location</b>	The updated location.						

## Exhibit D - Claim Chart for U.S. Patent No. 9,467,838 Against HTC

US9467838	Exemplary Supporting Evidence Regarding Accused Products
	<p data-bbox="535 237 1822 264"><a href="https://developers.google.com/android/reference/com/google/android/gms/location/LocationListener">https://developers.google.com/android/reference/com/google/android/gms/location/LocationListener</a></p> <p data-bbox="535 318 840 345">Public Constructors</p> <hr data-bbox="535 358 1766 363"/> <p data-bbox="535 407 1766 451">public <b>MapView</b> (<a href="#">Context</a> context)</p> <p data-bbox="535 500 1766 544">public <b>MapView</b> (<a href="#">Context</a> context, <a href="#">AttributeSet</a> attrs)</p> <p data-bbox="535 592 1766 636">public <b>MapView</b> (<a href="#">Context</a> context, <a href="#">AttributeSet</a> attrs, int defStyleAttr)</p> <p data-bbox="535 685 1766 729">public <b>MapView</b> (<a href="#">Context</a> context, <a href="#">GoogleMapOptions</a> options)</p> <p data-bbox="535 748 1696 776"><a href="https://developers.google.com/android/reference/com/google/android/gms/maps/MapView">https://developers.google.com/android/reference/com/google/android/gms/maps/MapView</a></p>

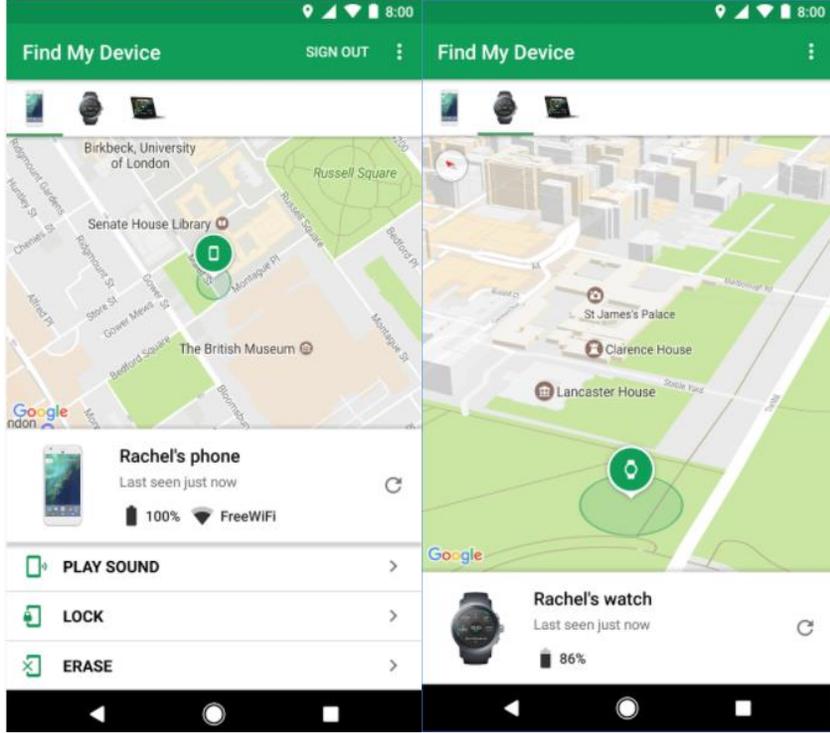
**Exhibit D - Claim Chart for U.S. Patent No. 9,467,838 Against HTC**

US9467838	Exemplary Supporting Evidence Regarding Accused Products		
	<p><code>public void <b>getMapAsync</b> (<b>OnMapReadyCallback</b> callback)</code></p> <p>Returns a non-null instance of the <code>GoogleMap</code>, ready to be used.</p> <p>Note that:</p> <ul style="list-style-type: none"> <li>• This method must be called from the main thread.</li> <li>• The callback will be executed in the main thread.</li> <li>• In the case where Google Play services is not installed on the user's device, the callback will not be triggered until the user installs it.</li> <li>• The <code>GoogleMap</code> object provided by the callback is non-null.</li> </ul> <p><b>Parameters</b></p> <table border="1" data-bbox="554 688 1761 753"> <tr> <td><code>callback</code></td> <td>The callback object that will be triggered when the map is ready to be used.</td> </tr> </table> <p><code>public final void <b>onCreate</b> (<b>Bundle</b> savedInstanceState)</code></p> <p>You must call this method from the parent Activity/Fragment's corresponding method.</p> <p><a href="https://developers.google.com/android/reference/com/google/android/gms/maps/MapView">https://developers.google.com/android/reference/com/google/android/gms/maps/MapView</a></p>	<code>callback</code>	The callback object that will be triggered when the map is ready to be used.
<code>callback</code>	The callback object that will be triggered when the map is ready to be used.		
<p>[1G] presenting, via the interactive display of the first device, a second georeferenced map and a second set of one or more user-selectable symbols corresponding to a second set of one or more of the second devices, wherein the second set of symbols are positioned on the second georeferenced map at respective positions corresponding to the locations of the second set of second devices, and wherein the second georeferenced map data relate positions on the second georeferenced map to spatial coordinates.</p>	<p>HTC infringes directly and/or indirectly by performing, inducing others to perform, and/or contributing to the performance of: presenting, via the interactive display of the first device, a second georeferenced map and a second set of one or more user-selectable symbols corresponding to a second set of one or more of the second devices, wherein the second set of symbols are positioned on the second georeferenced map at respective positions corresponding to the locations of the second set of second devices, and wherein the second georeferenced map data relate positions on the second georeferenced map to spatial coordinates.</p> <p><b>Regarding Find My Device</b> and Android Device Manager, the Accused Products present the user with a second and/or updated map display on the display. The second and/or updated map comprises a first symbol positioned on the map and corresponding to the first device. The second and/or updated map comprises one or more second symbols positioned on the map and corresponding to one or more second</p>		

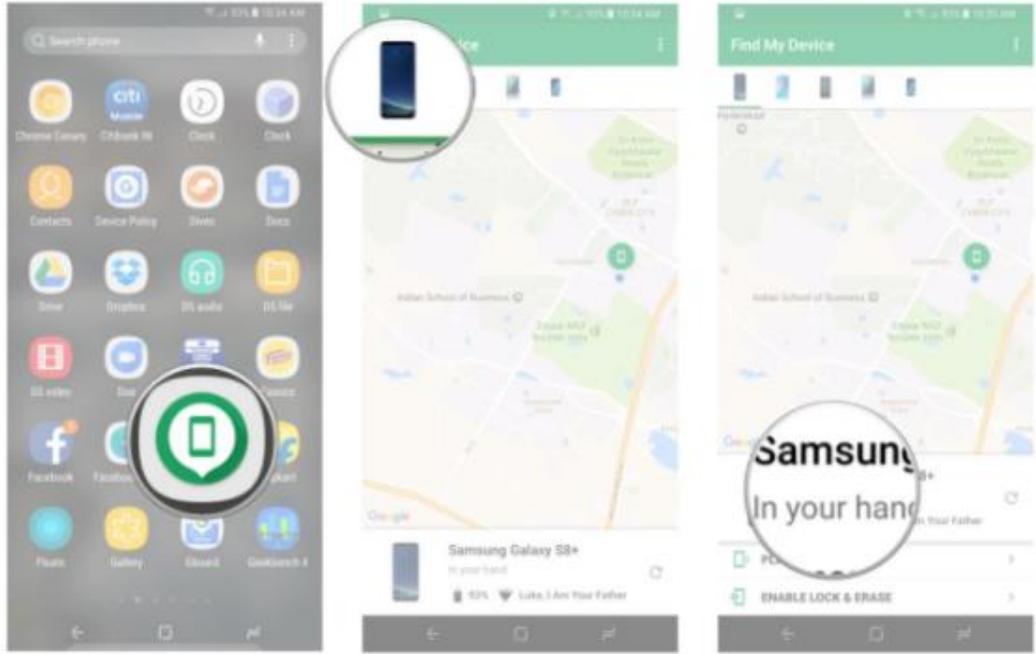
**Exhibit D - Claim Chart for U.S. Patent No. 9,467,838 Against HTC**

<b>US9467838</b>	<b>Exemplary Supporting Evidence Regarding Accused Products</b>
<p>are positioned on the second georeferenced map at respective positions corresponding to the locations of the second set of second devices, and wherein the second georeferenced map data relate positions on the second georeferenced map to spatial coordinates;</p>	<p>devices. The map is interactive because the user may control the display of the map, e.g., pan, zoom, and/or effect change to the map in an otherwise interactive manner. The map is georeferenced for at least the reason that one or more symbols are associated with spatial locations, i.e., coordinates. The symbols are user-selectable because a user may touch the display to select the device associated with the symbol. For example, the user may input a touch selection directly on or near the portion of the display corresponding to the symbol's coordinates on the map to effect a selection of the device. In an alternative example, the user may input a selection event on the display that does not directly correspond to the symbol's coordinates, but which effects a selection of the device because the selection event is otherwise associated with the device corresponding to the symbol.</p> <p><b><u>Regarding Google Maps.</u></b> Google Latitude, Google Plus, Google Hangouts, Google Messages, Google Allo, Google Duo, Google Chrome, and Android Messenger, the Accused Products present the user of a first device with a second and/or updated map display on the display. The second and/or updated map comprises a first symbol positioned on the map and corresponding to the first user or corresponding first device. The second and/or updated map comprises one or more second symbols positioned on the map and corresponding to one or more second users, contacts and/or corresponding second devices. The map is interactive because the user may control the display of the map, e.g., pan, zoom, and/or effect change to the map in an otherwise interactive manner. The map is georeferenced for at least the reason that one or more symbols are associated with spatial locations, i.e., coordinates. The symbols are user-selectable because a user may touch the display to select the user or device associated with the symbol. For example, the user may input a touch selection directly on or near the portion of the display corresponding to the symbol's coordinates on the map to effect a selection of the user or device. In an alternative example, the user may input a selection event on the display that does not directly correspond to the symbol's coordinates, but which effects a selection of the user or device because the selection event is otherwise associated with the user or device corresponding to the symbol.</p> <p><b><u>Exemplary Support for Find My Device:</u></b></p>

**Exhibit D - Claim Chart for U.S. Patent No. 9,467,838 Against HTC**

US9467838	Exemplary Supporting Evidence Regarding Accused Products
	 <p><a href="https://play.google.com/store/apps/details?id=com.google.android.apps.adm&amp;hl=en">https://play.google.com/store/apps/details?id=com.google.android.apps.adm&amp;hl=en</a></p>

**Exhibit D - Claim Chart for U.S. Patent No. 9,467,838 Against HTC**

US9467838	Exemplary Supporting Evidence Regarding Accused Products
	<p>Once you're signed in to Find My Device, you'll see a map with your current location as well as the make and model of your phone, and two options — Play Sound, and Enable Lock &amp; Erase. Hitting the latter option will allow you to start using the Lock and Erase functions.</p> <p>If you've signed into more than one phone, you can select a particular device by browsing the list at the top of the screen.</p> <ol style="list-style-type: none"> <li>1. Open <b>Find My Device</b> from your home screen or app drawer.</li> <li>2. Select your phone from the list of devices at the top of the screen.</li> <li>3. See if your phone is discoverable.</li> </ol>  <p><a href="https://www.androidcentral.com/find-my-device">https://www.androidcentral.com/find-my-device</a></p>



**Exhibit D - Claim Chart for U.S. Patent No. 9,467,838 Against HTC**

US9467838	Exemplary Supporting Evidence Regarding Accused Products
	<h2 data-bbox="548 272 1686 326">How to locate your phone over the internet</h2> <p data-bbox="548 370 1692 509">If you've lost your phone, you can remotely locate it through the <a href="#">Find My Device website</a>. You'll need to sign in to the Google account that was used to set up Find My Device. It takes a few seconds, but the service should be able to track your phone. Alternatively, you can also do a Google search for "find my phone" to locate your handset.</p> <ol data-bbox="539 565 1031 721" style="list-style-type: none"><li>1. Head to the <a href="#">Find My Device website</a>.</li><li>2. Sign in to your <a href="#">Google account</a>.</li><li>3. Check if your device is visible.</li></ol> <div data-bbox="583 764 1793 1219"></div> <p data-bbox="533 1247 1157 1279"><a href="https://www.androidcentral.com/find-my-device">https://www.androidcentral.com/find-my-device</a></p> <p data-bbox="533 1320 1052 1352"><b><u>Exemplary Support for Google Maps:</u></b></p>

**Exhibit D - Claim Chart for U.S. Patent No. 9,467,838 Against HTC**

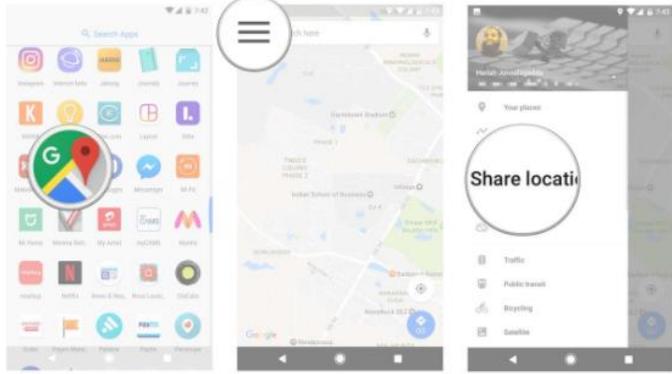
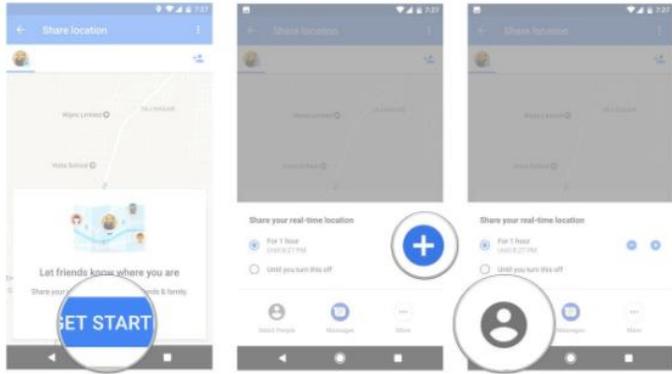
US9467838	Exemplary Supporting Evidence Regarding Accused Products
	<p><b>How to share your location in Google Maps</b></p> <ol style="list-style-type: none"> <li>1. Open <b>Google Maps</b> from the app drawer or the home screen.</li> <li>2. Tap the <b>hamburger menu</b> (the three horizontal lines) on the top left corner of the screen.</li> <li>3. Select <b>Share location</b>.</li> </ol>  <ol style="list-style-type: none"> <li>4. Tap <b>Get Started</b>.</li> <li>5. Use the <b>+</b> icon to select a time period or select the <b>Until you turn this off</b> setting to share your location indefinitely.</li> <li>6. Tap <b>Select People</b>.</li> </ol>  <p><a href="https://www.androidcentral.com/how-share-location-google-maps">https://www.androidcentral.com/how-share-location-google-maps</a></p>

Exhibit D - Claim Chart for U.S. Patent No. 9,467,838 Against HTC

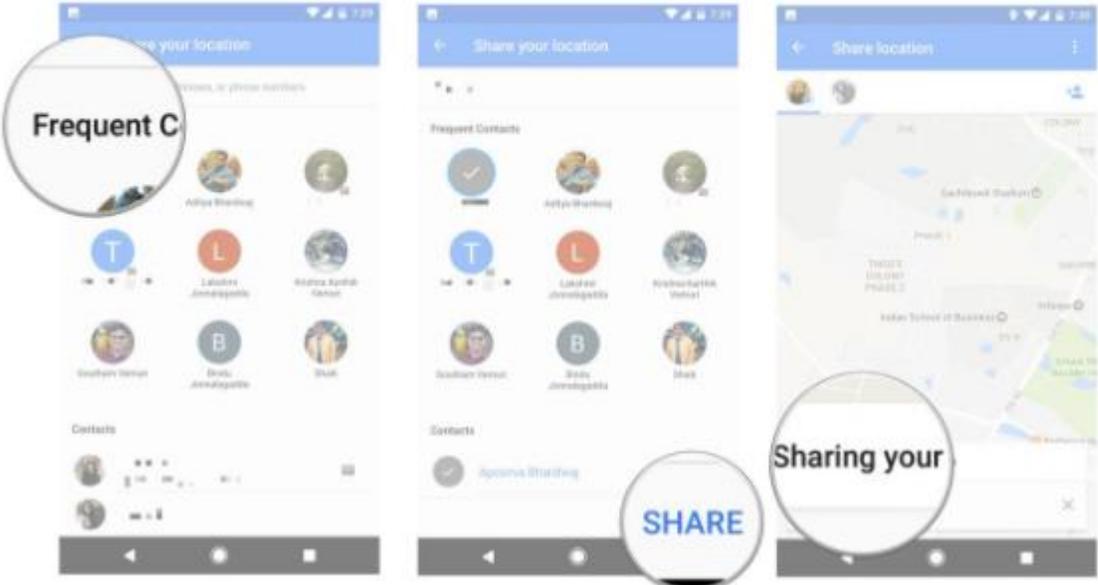
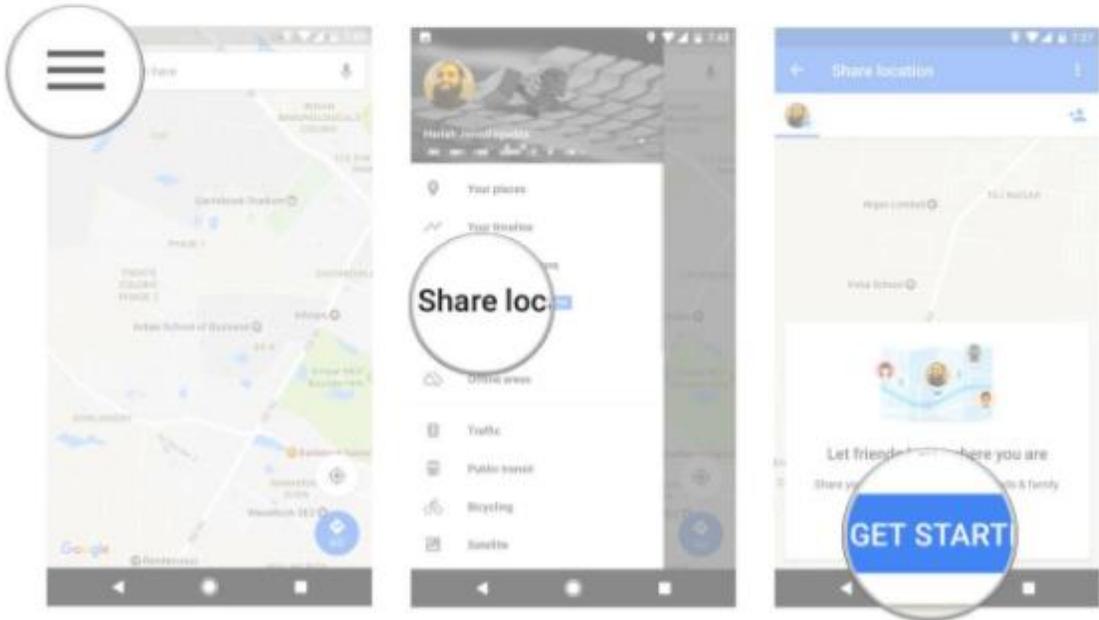
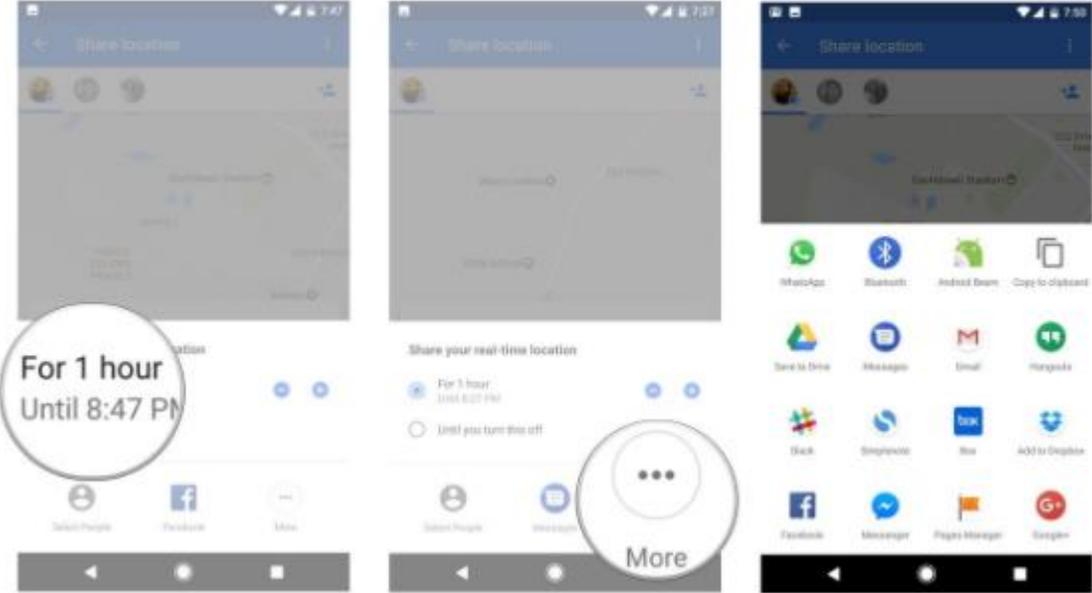
US9467838	Exemplary Supporting Evidence Regarding Accused Products
	<p data-bbox="552 253 1602 310">7. You'll see a list of your frequent contacts at the top, along with a full list of contacts. Pick the contacts by tapping their name.</p> <p data-bbox="552 337 1482 367">8. Once you've selected the contacts you want to share your location to, tap <b>Share</b>.</p> <p data-bbox="552 394 1442 423">9. You'll see a message saying that the selected contact can view your location.</p>  <p data-bbox="533 1101 1381 1130"><a href="https://www.androidcentral.com/how-share-location-google-maps">https://www.androidcentral.com/how-share-location-google-maps</a></p>

Exhibit D - Claim Chart for U.S. Patent No. 9,467,838 Against HTC

US9467838	Exemplary Supporting Evidence Regarding Accused Products
	<h2 data-bbox="548 240 1281 289">How to create a shareable link</h2> <p data-bbox="548 329 1486 358">You can also create a link and use it to share your location easily. Here's how to do it:</p> <ol data-bbox="541 407 1260 548" style="list-style-type: none"><li>1. Tap the hamburger menu on the top left corner of the screen.</li><li>2. Select Share location.</li><li>3. Tap Get Started.</li></ol>  <p data-bbox="533 1230 1381 1263"><a href="https://www.androidcentral.com/how-share-location-google-maps">https://www.androidcentral.com/how-share-location-google-maps</a></p>

**Exhibit D - Claim Chart for U.S. Patent No. 9,467,838 Against HTC**

US9467838	Exemplary Supporting Evidence Regarding Accused Products
	<p>4. Select the amount of time you want to share your location.</p> <p>5. Tap More.</p> <p>6. Select your app of choice to create and send a unique URL that broadcasts your current location. You can email it, send the link via Messenger, or even tweet it to the intended recipient.</p>  <p><a href="https://www.androidcentral.com/how-share-location-google-maps">https://www.androidcentral.com/how-share-location-google-maps</a></p>

**Exhibit D - Claim Chart for U.S. Patent No. 9,467,838 Against HTC**

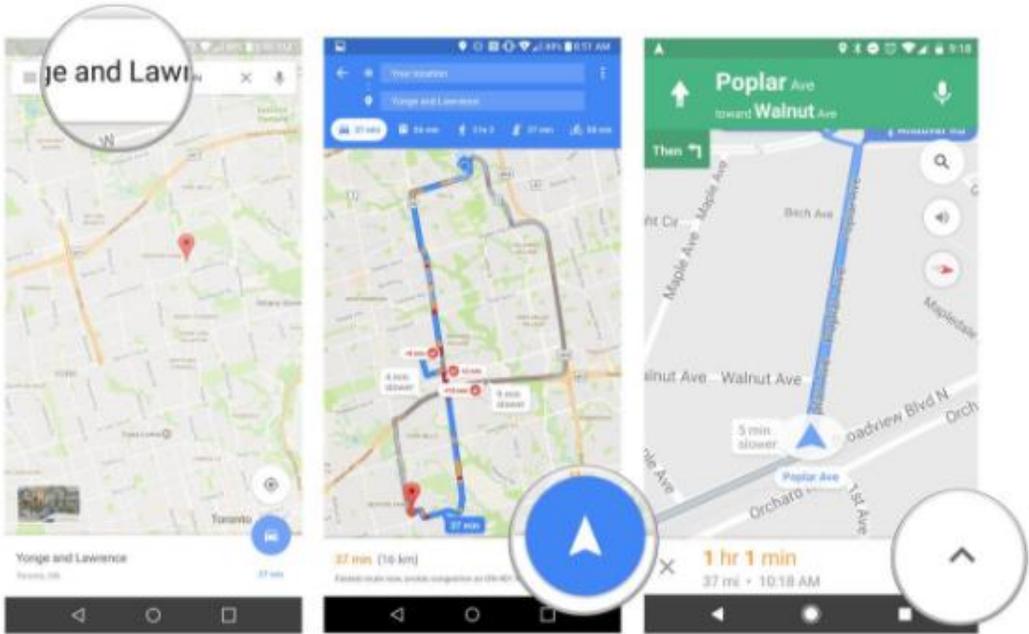
US9467838	Exemplary Supporting Evidence Regarding Accused Products
	<h2 data-bbox="550 240 1451 337">How to share your navigation directions while you walk, drive or transit</h2> <p data-bbox="550 375 1577 464">One of the best ways to share your location with someone is during a drive, walk or transit. If you're meeting a friend or family member somewhere, or navigating towards their home, Google Maps lets you share your location with them for the duration of the trip. It's magic!</p> <ol data-bbox="550 513 1419 643" style="list-style-type: none"><li>1. In the <b>search bar</b> enter your destination.</li><li>2. Pick your navigation type (drive, transit, walk) and press the <b>blue navigate button</b>.</li><li>3. Tap the arrow next to the <b>time-to-destination</b> number at the bottom of the screen.</li></ol>  <p data-bbox="535 1328 1381 1360"><a href="https://www.androidcentral.com/how-share-location-google-maps">https://www.androidcentral.com/how-share-location-google-maps</a></p>

Exhibit D - Claim Chart for U.S. Patent No. 9,467,838 Against HTC

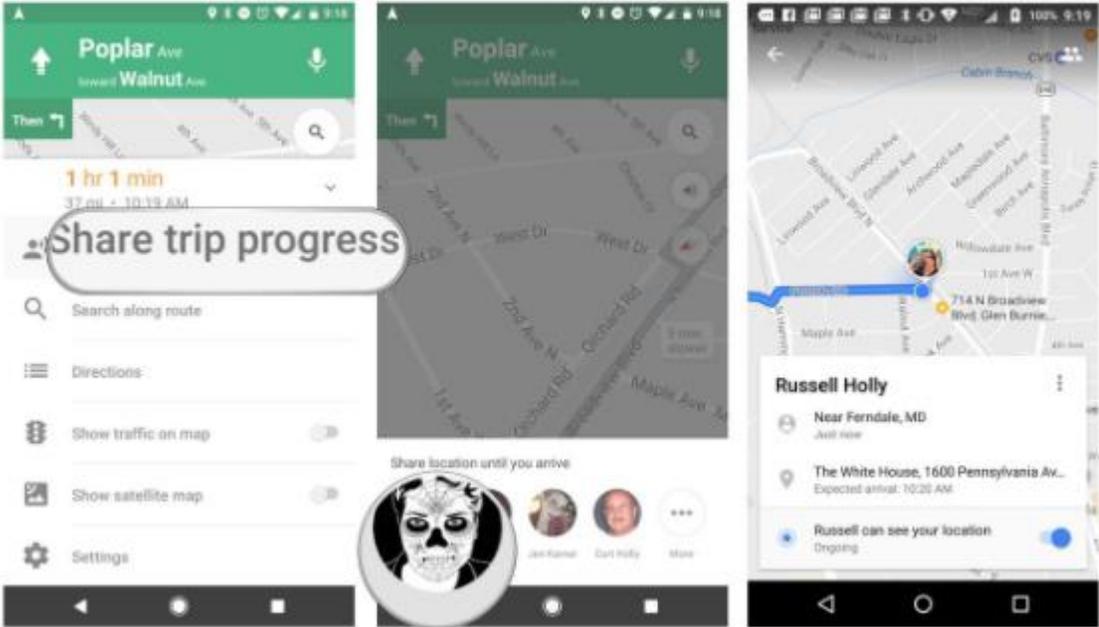
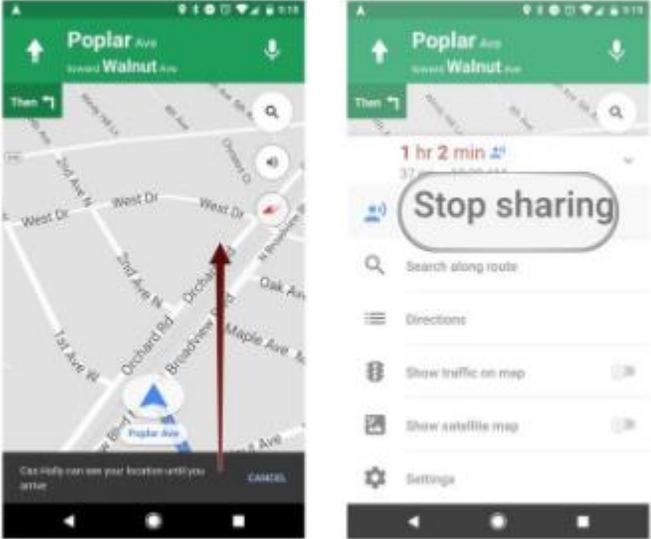
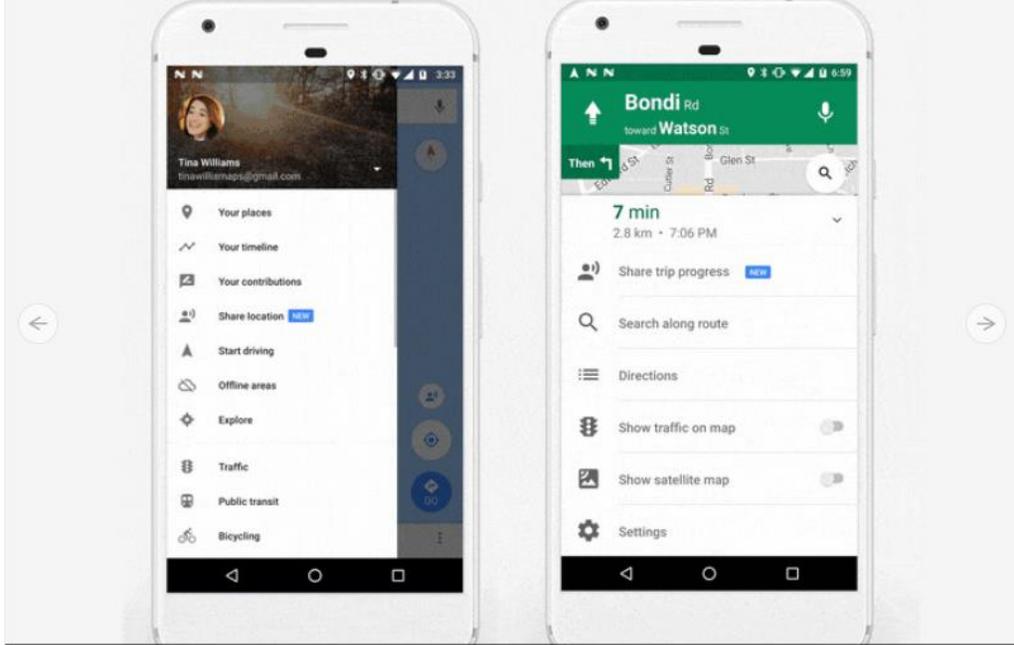
US9467838	Exemplary Supporting Evidence Regarding Accused Products
	<p data-bbox="552 277 861 305">4. Tap Share trip progress.</p> <p data-bbox="552 332 1171 360">5. Choose one or more contacts to share trip progress.</p>  <p data-bbox="552 1063 1381 1091">You can also stop sharing your location with someone before a trip ends.</p> <p data-bbox="535 1101 1381 1128"><a href="https://www.androidcentral.com/how-share-location-google-maps">https://www.androidcentral.com/how-share-location-google-maps</a></p>

Exhibit D - Claim Chart for U.S. Patent No. 9,467,838 Against HTC

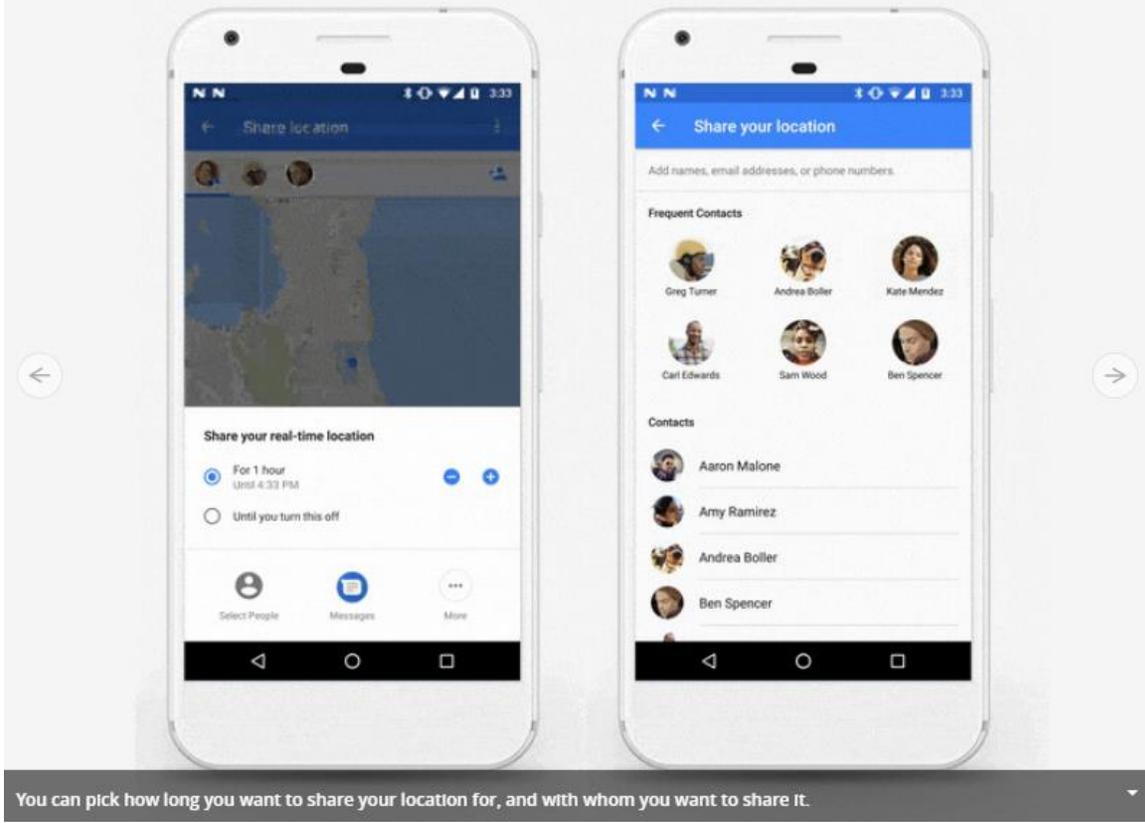
US9467838	Exemplary Supporting Evidence Regarding Accused Products
	<ol style="list-style-type: none"><li data-bbox="562 245 1493 272">1. Tap the arrow next to the time-to-destination number at the bottom of the screen.</li><li data-bbox="562 302 793 329">2. Tap Stop sharing.</li></ol> <div data-bbox="772 383 1423 922"></div> <p data-bbox="569 976 659 1003">That's It!</p> <p data-bbox="569 1045 1633 1073">Are you excited that location sharing is back in Google Maps? How often do you use the feature?</p> <p data-bbox="537 1084 1381 1117"><a href="https://www.androidcentral.com/how-share-location-google-maps">https://www.androidcentral.com/how-share-location-google-maps</a></p>



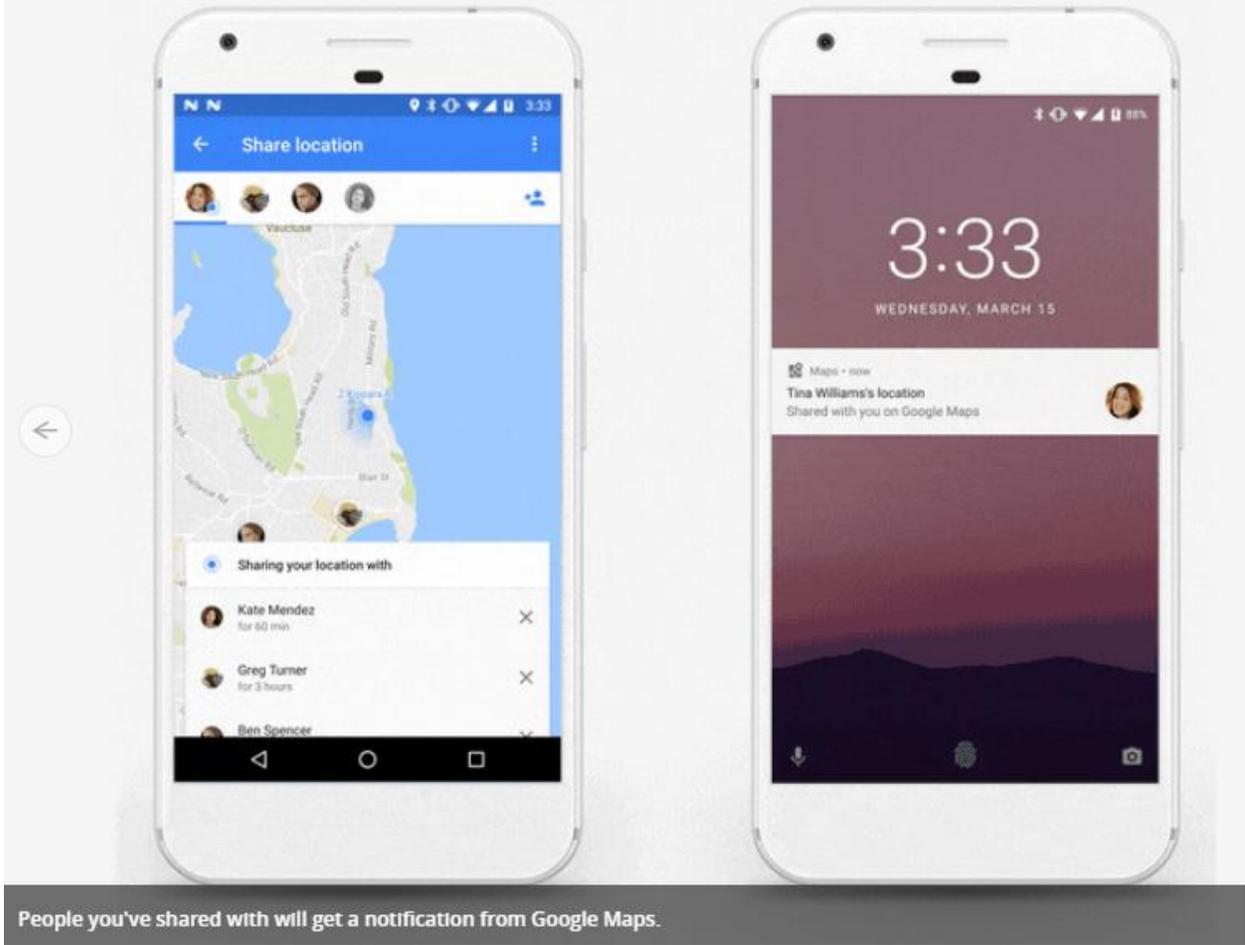
**Exhibit D - Claim Chart for U.S. Patent No. 9,467,838 Against HTC**

US9467838	Exemplary Supporting Evidence Regarding Accused Products
	 <p data-bbox="535 893 1549 950">Ways to start location sharing: (left) from the app navigation drawer, and (right) from the menu, you can drag up while navigating somewhere.</p> <p data-bbox="535 958 1549 982"><a href="https://arstechnica.com/gadgets/2017/03/location-sharing-finally-returns-to-google-maps/">https://arstechnica.com/gadgets/2017/03/location-sharing-finally-returns-to-google-maps/</a></p>

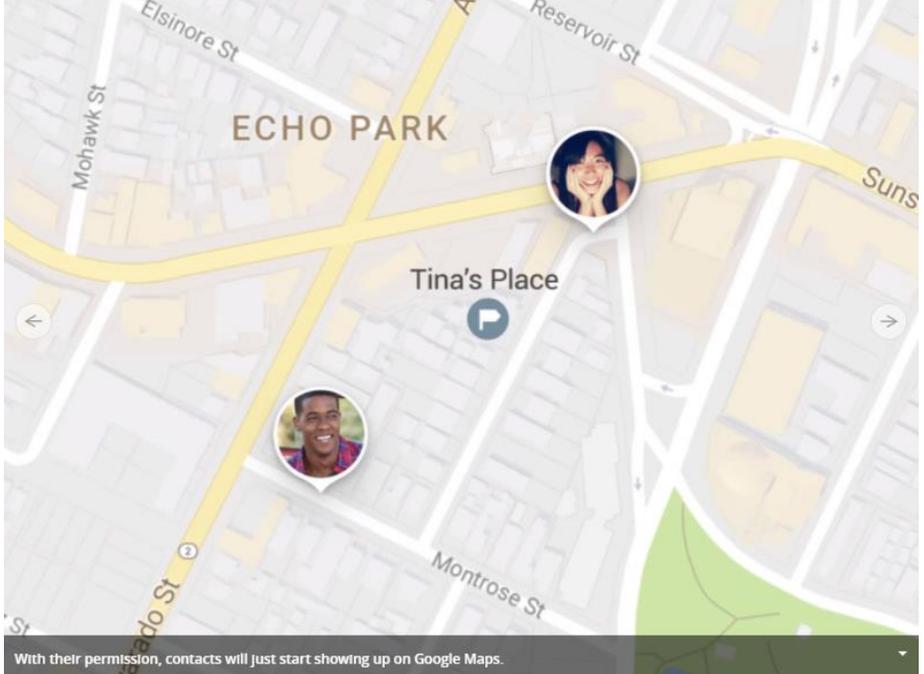
**Exhibit D - Claim Chart for U.S. Patent No. 9,467,838 Against HTC**

US9467838	Exemplary Supporting Evidence Regarding Accused Products
	 <p data-bbox="535 1023 1680 1063">You can pick how long you want to share your location for, and with whom you want to share it.</p> <p data-bbox="535 1063 1680 1096"><a href="https://arstechnica.com/gadgets/2017/03/location-sharing-finally-returns-to-google-maps/">https://arstechnica.com/gadgets/2017/03/location-sharing-finally-returns-to-google-maps/</a></p>

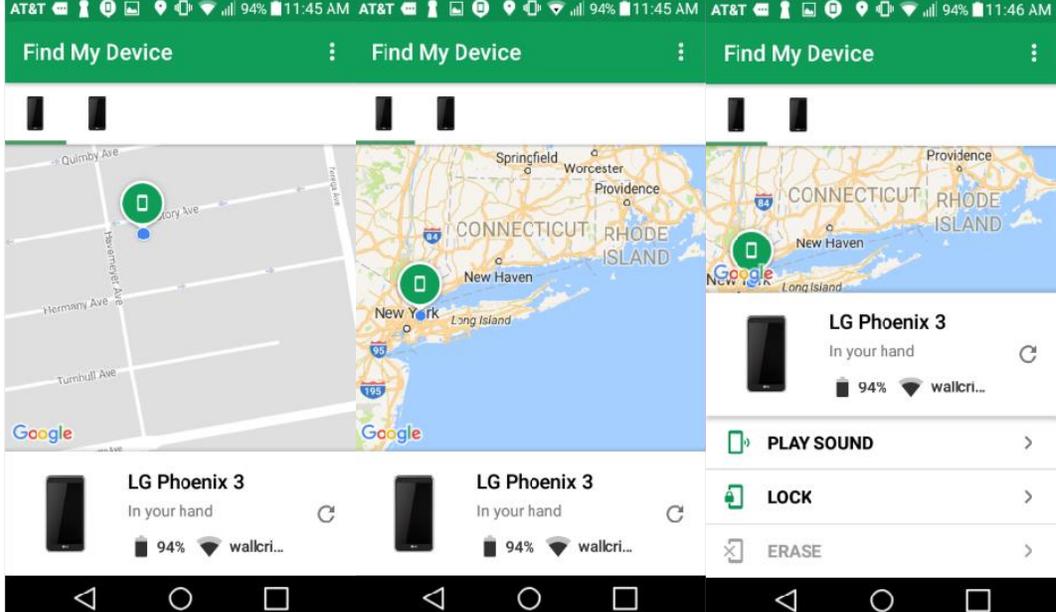
**Exhibit D - Claim Chart for U.S. Patent No. 9,467,838 Against HTC**

US9467838	Exemplary Supporting Evidence Regarding Accused Products
	 <p data-bbox="535 1144 1197 1177">People you've shared with will get a notification from Google Maps.</p> <p data-bbox="535 1188 1680 1221"><a href="https://arstechnica.com/gadgets/2017/03/location-sharing-finally-returns-to-google-maps/">https://arstechnica.com/gadgets/2017/03/location-sharing-finally-returns-to-google-maps/</a></p>

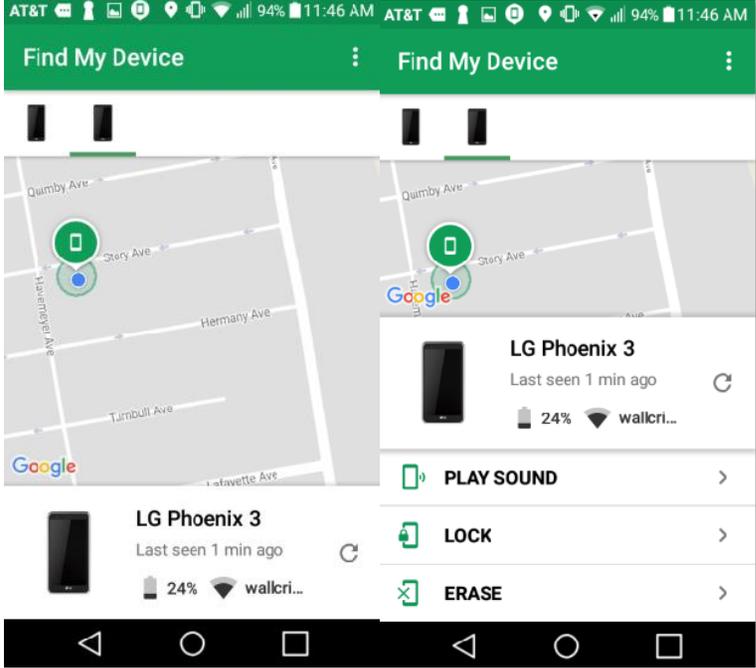
**Exhibit D - Claim Chart for U.S. Patent No. 9,467,838 Against HTC**

US9467838	Exemplary Supporting Evidence Regarding Accused Products
	 <p data-bbox="535 909 1459 950">With their permission, contacts will just start showing up on Google Maps.</p> <p data-bbox="535 950 1459 982"><a href="https://arstechnica.com/gadgets/2017/03/location-sharing-finally-returns-to-google-maps/">https://arstechnica.com/gadgets/2017/03/location-sharing-finally-returns-to-google-maps/</a></p> <p data-bbox="535 982 1459 1023"><b><u>Exemplary Find My Device Screenshots:</u></b></p>

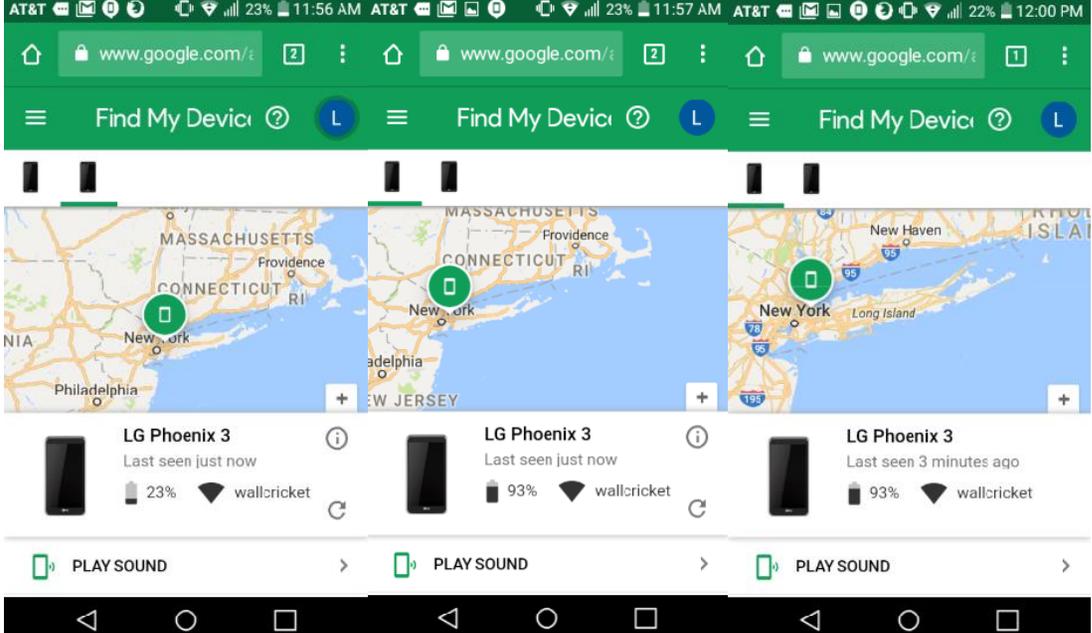
**Exhibit D - Claim Chart for U.S. Patent No. 9,467,838 Against HTC**

US9467838	Exemplary Supporting Evidence Regarding Accused Products
	 <p>The image displays three sequential screenshots of an Android mobile application interface, specifically the 'Find My Device' feature. Each screenshot shows the top status bar with 'AT&amp;T' carrier, signal strength, battery at 94%, and time around 11:45 AM. The app's header is a green bar with 'Find My Device' and a menu icon. The first screenshot shows a street-level map with a green location pin. The second screenshot shows a regional map of Connecticut and Rhode Island with a green location pin. The third screenshot shows the device status card for 'LG Phoenix 3', indicating it is 'In your hand' with 94% battery and 'walkcri...' connectivity. Below the status card are three action buttons: 'PLAY SOUND', 'LOCK', and 'ERASE', each with a right-pointing arrow.</p>

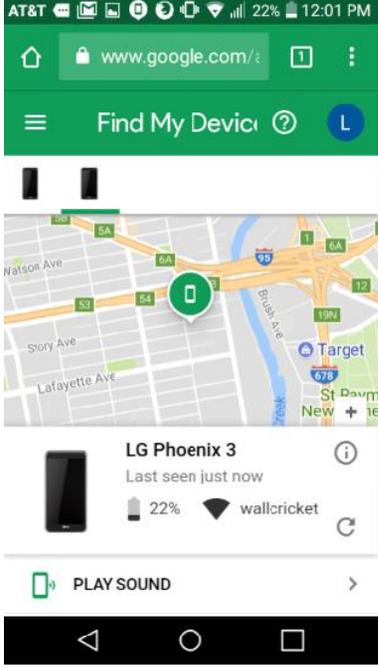
**Exhibit D - Claim Chart for U.S. Patent No. 9,467,838 Against HTC**

US9467838	Exemplary Supporting Evidence Regarding Accused Products
	

**Exhibit D - Claim Chart for U.S. Patent No. 9,467,838 Against HTC**

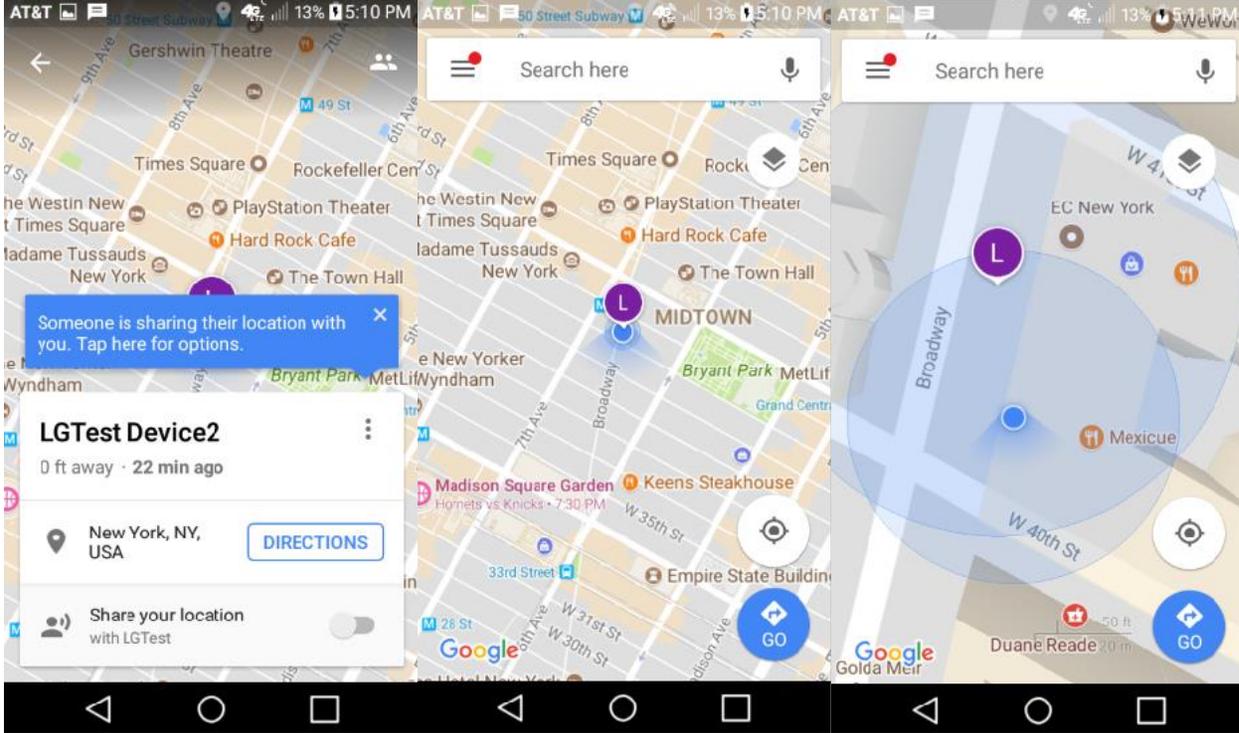
US9467838	Exemplary Supporting Evidence Regarding Accused Products
	

**Exhibit D - Claim Chart for U.S. Patent No. 9,467,838 Against HTC**

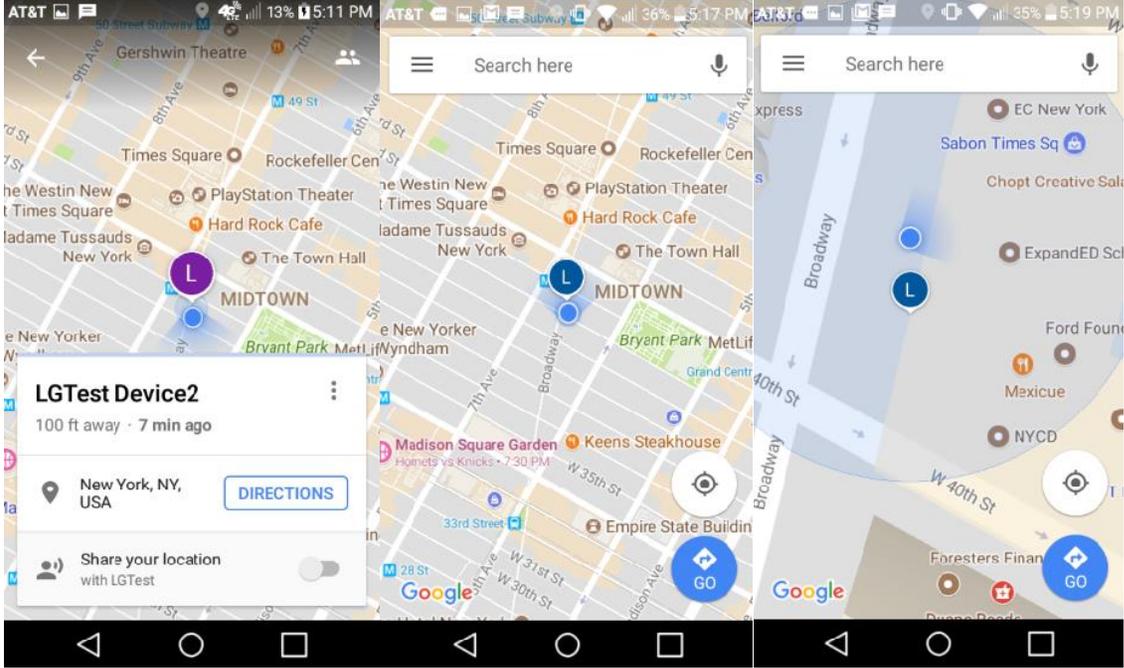
US9467838	Exemplary Supporting Evidence Regarding Accused Products
	 <p>The screenshot shows a mobile phone interface for Google Maps. At the top, the status bar displays 'AT&amp;T', signal strength, Wi-Fi, battery at 22%, and time 12:01 PM. Below the status bar is a green navigation bar with a home icon, a search bar containing 'www.google.com/...', and a profile icon. A secondary green bar contains a menu icon, the text 'Find My Device', a help icon, and a profile icon. The main area shows a map with a green location pin. Below the map, a card displays 'LG Phoenix 3' with a small phone icon, the text 'Last seen just now', a battery icon at 22%, and a Wi-Fi icon labeled 'wallcricket'. At the bottom of the card is a 'PLAY SOUND' button with a speaker icon and a right-pointing arrow. The Android navigation bar is visible at the very bottom.</p> <p><b><u>Exemplary Google Maps Screenshots:</u></b></p>



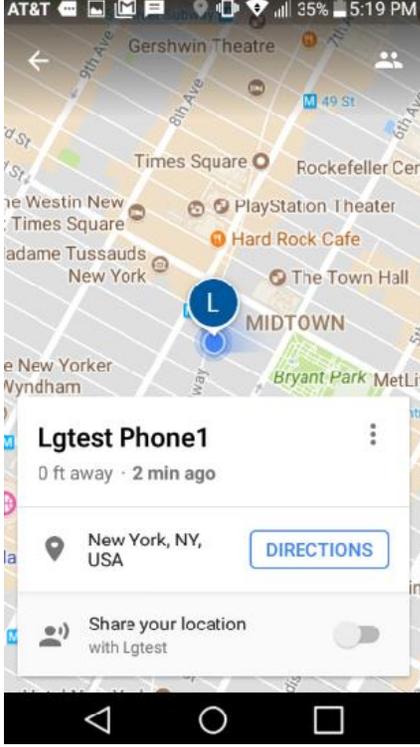
**Exhibit D - Claim Chart for U.S. Patent No. 9,467,838 Against HTC**

US9467838	Exemplary Supporting Evidence Regarding Accused Products
	 <p>The image displays two side-by-side screenshots of a mobile map application, likely Google Maps, showing a location sharing notification. The notification is a white card with a blue header that reads "Someone is sharing their location with you. Tap here for options." Below the header, the card identifies the device as "LGTest Device2" and shows it is "0 ft away · 22 min ago". The location is listed as "New York, NY, USA" with a "DIRECTIONS" button. At the bottom of the card, there is a toggle switch for "Share your location with LGTest" which is currently turned off. The background of both screenshots is a map of Midtown Manhattan, New York City, with various landmarks like Times Square, Bryant Park, and the Empire State Building visible. The top status bar shows "AT&amp;T", signal strength, 13% battery, and 5:10 PM. The bottom navigation bar shows the standard Android icons.</p>

**Exhibit D - Claim Chart for U.S. Patent No. 9,467,838 Against HTC**

US9467838	Exemplary Supporting Evidence Regarding Accused Products
	 <p>The image displays three sequential screenshots of a mobile map application, likely Google Maps, showing a location tracking interface. The screenshots are arranged horizontally, with the first on the left and the third on the right. Each screenshot shows a map of Times Square, New York, with a purple location pin and a corresponding information card.</p> <p>The information card in the first screenshot (left) is titled "LGTest Device2" and shows the device is "100 ft away · 7 min ago". It includes a location pin for "New York, NY, USA" and a "DIRECTIONS" button. Below this, there is a "Share your location with LGTest" option with a toggle switch.</p> <p>The second screenshot (middle) shows the same map area, but the information card is partially obscured or faded. The third screenshot (right) shows the map area with the information card still visible, though the details are less clear.</p> <p>The screenshots also show the mobile OS interface, including the status bar at the top (AT&amp;T, signal strength, battery, and time) and the navigation bar at the bottom (back, home, and recent apps buttons).</p>

**Exhibit D - Claim Chart for U.S. Patent No. 9,467,838 Against HTC**

US9467838	Exemplary Supporting Evidence Regarding Accused Products
	 <p><b><u>Exemplary Source Code:</u></b></p> <p>The above functionality is performed at least in part by the following publicly available source code and/or source code that invokes or is invoked by the following source code (or a substantially similar copy compiled and loaded onto the Accused Products by HTC):</p>

## Exhibit D - Claim Chart for U.S. Patent No. 9,467,838 Against HTC

US9467838	Exemplary Supporting Evidence Regarding Accused Products
	<p data-bbox="556 248 1003 280"><code>public static <a href="#">LocationRequest</a> create ()</code></p> <p data-bbox="546 313 1054 337">Create a location request with default parameters.</p> <p data-bbox="546 370 1663 427">Default parameters are for a block accuracy, slowly updated location. It can then be adjusted as required by the applications before passing to the <a href="#">FusedLocationProviderApi</a>.</p> <p data-bbox="569 451 655 475"><b>Returns</b></p> <ul data-bbox="577 500 835 524" style="list-style-type: none"> <li>• a new location request</li> </ul> <p data-bbox="535 540 1820 573"><a href="https://developers.google.com/android/reference/com/google/android/gms/location/LocationRequest">https://developers.google.com/android/reference/com/google/android/gms/location/LocationRequest</a></p> <p data-bbox="556 589 1297 621"><code>public static final int <b>PRIORITY_BALANCED_POWER_ACCURACY</b></code></p> <p data-bbox="546 654 1201 678">Used with <code>setPriority(int)</code> to request "block" level accuracy.</p> <p data-bbox="546 711 1667 768">Block level accuracy is considered to be about 100 meter accuracy. Using a coarse accuracy such as this often consumes less power.</p> <p data-bbox="546 800 751 824">Constant Value: 102</p> <p data-bbox="556 881 1129 914"><code>public static final int <b>PRIORITY_HIGH_ACCURACY</b></code></p> <p data-bbox="546 946 1360 971">Used with <code>setPriority(int)</code> to request the most accurate locations available.</p> <p data-bbox="546 1003 987 1027">This will return the finest location available.</p> <p data-bbox="546 1060 751 1084">Constant Value: 100</p> <p data-bbox="556 1141 1075 1174"><code>public static final int <b>PRIORITY_LOW_POWER</b></code></p> <p data-bbox="546 1206 1180 1230">Used with <code>setPriority(int)</code> to request "city" level accuracy.</p> <p data-bbox="546 1263 1759 1320">City level accuracy is considered to be about 10km accuracy. Using a coarse accuracy such as this often consumes less power.</p> <p data-bbox="546 1352 751 1377">Constant Value: 104</p> <p data-bbox="535 1385 1820 1417"><a href="https://developers.google.com/android/reference/com/google/android/gms/location/LocationRequest">https://developers.google.com/android/reference/com/google/android/gms/location/LocationRequest</a></p>

## Exhibit D - Claim Chart for U.S. Patent No. 9,467,838 Against HTC

US9467838	Exemplary Supporting Evidence Regarding Accused Products
	<p data-bbox="556 284 1774 321"><code>public Task&lt;Location&gt; getLastLocation ()</code></p> <p data-bbox="556 349 1123 373">Returns the best most recent location currently available.</p> <p data-bbox="556 406 1711 462">If a location is not available, which should happen very rarely, null will be returned. The best accuracy available while respecting the location permissions will be returned.</p> <p data-bbox="556 495 1753 552">This method provides a simplified way to get location. It is particularly well suited for applications that do not require an accurate location and that do not want to maintain extra logic for location updates.</p> <p data-bbox="556 609 1774 646"><code>public Task&lt;LocationAvailability&gt; getLocationAvailability ()</code></p> <p data-bbox="556 673 1711 738">Returns the availability of location data. When <code>isLocationAvailable()</code> returns true, then the location returned by <code>getLastLocation()</code> will be reasonably up to date within the hints specified by the active <code>LocationRequest</code> s.</p> <p data-bbox="556 771 1501 795">If the client isn't connected to Google Play services and the request times out, null is returned.</p> <p data-bbox="556 828 1690 885">Note it's always possible for <code>getLastLocation()</code> to return null even when this method returns true (e.g. location settings were disabled between calls).</p> <p data-bbox="556 901 1900 966"><a href="https://developers.google.com/android/reference/com/google/android/gms/location/FusedLocationProviderClient">https://developers.google.com/android/reference/com/google/android/gms/location/FusedLocationProviderClient</a></p>

**Exhibit D - Claim Chart for U.S. Patent No. 9,467,838 Against HTC**

US9467838	Exemplary Supporting Evidence Regarding Accused Products						
	<p data-bbox="546 240 1774 326"> <code>public Task&lt;Void&gt; requestLocationUpdates (LocationRequest request, LocationCallback callback, Looper looper)</code> </p> <p data-bbox="546 354 1291 380">Requests location updates with a callback on the specified Looper thread.</p> <p data-bbox="546 410 1711 472">This method is suited for the foreground use cases. For background use cases, the <code>PendingIntent</code> version of the method is recommended, see <code>requestLocationUpdates(LocationRequest, PendingIntent)</code>.</p> <p data-bbox="546 503 1396 529">Any previous LocationRequests registered on this LocationListener will be replaced.</p> <p data-bbox="546 560 1711 654">This call will keep the Google Play services connection active, so make sure to call <code>removeLocationUpdates(LocationCallback)</code> when you no longer need it, otherwise you lose the benefits of the automatic connection management.</p> <p data-bbox="546 685 1774 711">Callbacks for <code>LocationCallback</code> will be made on the specified thread, which must already be a prepared looper thread.</p> <p data-bbox="546 735 682 761"><b>Parameters</b></p> <table border="1" data-bbox="546 792 1774 1008"> <tbody> <tr> <td data-bbox="546 792 655 857"><b>request</b></td> <td data-bbox="655 792 1774 857">The location request for the updates.</td> </tr> <tr> <td data-bbox="546 857 655 922"><b>callback</b></td> <td data-bbox="655 857 1774 922">The callback for the location updates.</td> </tr> <tr> <td data-bbox="546 922 655 1008"><b>looper</b></td> <td data-bbox="655 922 1774 1008">The Looper object whose message queue will be used to implement the callback mechanism, or null to make callbacks on the calling thread.</td> </tr> </tbody> </table> <p data-bbox="546 1023 1911 1089"> <a href="https://developers.google.com/android/reference/com/google/android/gms/location/FusedLocationProviderClient">https://developers.google.com/android/reference/com/google/android/gms/location/FusedLocationProviderClient</a> </p>	<b>request</b>	The location request for the updates.	<b>callback</b>	The callback for the location updates.	<b>looper</b>	The Looper object whose message queue will be used to implement the callback mechanism, or null to make callbacks on the calling thread.
<b>request</b>	The location request for the updates.						
<b>callback</b>	The callback for the location updates.						
<b>looper</b>	The Looper object whose message queue will be used to implement the callback mechanism, or null to make callbacks on the calling thread.						

## Exhibit D - Claim Chart for U.S. Patent No. 9,467,838 Against HTC

US9467838	Exemplary Supporting Evidence Regarding Accused Products				
	<p data-bbox="556 240 1766 321">public <code>Task&lt;Void&gt; requestLocationUpdates</code> (<code>LocationRequest</code> request, <code>PendingIntent</code> callbackIntent)</p> <p data-bbox="556 354 1291 378">Requests location updates with a callback on the specified <code>PendingIntent</code>.</p> <p data-bbox="556 410 1759 540">This method is suited for the background use cases, more specifically for receiving location updates, even when the app has been killed by the system. In order to do so, use a <code>PendingIntent</code> for a started service. For foreground use cases, the <code>LocationCallback</code> version of the method is recommended, see <code>requestLocationUpdates(LocationRequest, LocationCallback, Looper)</code>.</p> <p data-bbox="556 573 1749 630">Any previously registered requests that have the same <code>PendingIntent</code> (as defined by <code>equals(Object)</code>) will be replaced by this request.</p> <p data-bbox="556 662 1753 751">Both <code>LocationResult</code> and <code>LocationAvailability</code> are sent to the given <code>PendingIntent</code>. You can extract data from an <code>Intent</code> using <code>hasResult(Intent)</code>, <code>extractResult(Intent)</code>, <code>hasLocationAvailability(Intent)</code>, and <code>extractLocationAvailability(Intent)</code>.</p> <p data-bbox="573 784 695 808"><b>Parameters</b></p> <table border="1" data-bbox="556 833 1766 971"> <tbody> <tr> <td data-bbox="556 833 863 906"><code>request</code></td> <td data-bbox="863 833 1766 906">The location request for the updates.</td> </tr> <tr> <td data-bbox="556 906 863 971"><code>callbackIntent</code></td> <td data-bbox="863 906 1766 971">A pending intent to be sent for each location update.</td> </tr> </tbody> </table> <p data-bbox="573 1003 653 1027"><b>Returns</b></p> <ul data-bbox="573 1044 1381 1068" style="list-style-type: none"> <li>• a <code>Task</code> for the call, check <code>isSuccessful()</code> to determine if it was successful.</li> </ul> <p data-bbox="535 1084 1906 1141"><a href="https://developers.google.com/android/reference/com/google/android/gms/location/FusedLocationProviderClient">https://developers.google.com/android/reference/com/google/android/gms/location/FusedLocationProviderClient</a></p>	<code>request</code>	The location request for the updates.	<code>callbackIntent</code>	A pending intent to be sent for each location update.
<code>request</code>	The location request for the updates.				
<code>callbackIntent</code>	A pending intent to be sent for each location update.				

## Exhibit D - Claim Chart for U.S. Patent No. 9,467,838 Against HTC

US9467838	Exemplary Supporting Evidence Regarding Accused Products						
	<p data-bbox="558 245 1761 277"><code>public void onLocationAvailability (LocationAvailability locationAvailability)</code></p> <p data-bbox="548 310 1192 334">Called when there is a change in the availability of location data.</p> <p data-bbox="548 367 1761 561">When <code>isLocationAvailable()</code> returns <code>false</code> you can assume that location will not be returned in <code>onLocationResult(LocationResult)</code> until something changes in the device's settings or environment. Even when <code>isLocationAvailable()</code> returns <code>true</code> the <code>onLocationResult(LocationResult)</code> may not always be called regularly, however the device location is known and both the most recently delivered location and <code>getLastLocation(GoogleApiClient)</code> will be reasonably up to date given the hints specified by the active <code>LocationRequest</code> s.</p> <p data-bbox="569 586 695 610"><b>Parameters</b></p> <table border="1" data-bbox="548 643 1761 708"> <tr> <td data-bbox="558 651 984 699"><code>locationAvailability</code></td> <td data-bbox="995 651 1761 699">The current status of location availability.</td> </tr> </table> <p data-bbox="558 756 1761 789"><code>public void onLocationResult (LocationResult result)</code></p> <p data-bbox="548 821 1077 846">Called when device location information is available.</p> <p data-bbox="548 878 1682 943">The most recent location returned by <code>getLastLocation()</code> is not guaranteed to be immediately fresh, but will be reasonably up to date given the hints specified by the active <code>LocationRequest</code> s.</p> <p data-bbox="569 967 695 992"><b>Parameters</b></p> <table border="1" data-bbox="548 1024 1761 1089"> <tr> <td data-bbox="558 1032 791 1081"><code>result</code></td> <td data-bbox="802 1032 1761 1089">The latest location result available.</td> </tr> </table> <p data-bbox="537 1105 1829 1130"><a href="https://developers.google.com/android/reference/com/google/android/gms/location/LocationCallback">https://developers.google.com/android/reference/com/google/android/gms/location/LocationCallback</a></p> <p data-bbox="558 1146 1761 1179"><code>public abstract void onLocationChanged (Location location)</code></p> <p data-bbox="548 1211 936 1235">Called when the location has changed.</p> <p data-bbox="569 1260 695 1284"><b>Parameters</b></p> <table border="1" data-bbox="548 1317 1761 1382"> <tr> <td data-bbox="558 1325 947 1373"><code>location</code></td> <td data-bbox="957 1325 1761 1382">The updated location.</td> </tr> </table>	<code>locationAvailability</code>	The current status of location availability.	<code>result</code>	The latest location result available.	<code>location</code>	The updated location.
<code>locationAvailability</code>	The current status of location availability.						
<code>result</code>	The latest location result available.						
<code>location</code>	The updated location.						



## Exhibit D - Claim Chart for U.S. Patent No. 9,467,838 Against HTC

US9467838	Exemplary Supporting Evidence Regarding Accused Products
	<p data-bbox="531 237 1919 264"><a href="https://developers.google.com/android/reference/com/google/android/gms/location/LocationListener">https://developers.google.com/android/reference/com/google/android/gms/location/LocationListener</a></p> <p data-bbox="531 318 1919 345">Public Constructors</p> <hr data-bbox="531 358 1919 363"/> <p data-bbox="531 410 1919 438">public <b>MapView</b> (<b>Context</b> context)</p> <p data-bbox="531 508 1919 535">public <b>MapView</b> (<b>Context</b> context, <b>AttributeSet</b> attrs)</p> <p data-bbox="531 605 1919 633">public <b>MapView</b> (<b>Context</b> context, <b>AttributeSet</b> attrs, int defStyle)</p> <p data-bbox="531 703 1919 730">public <b>MapView</b> (<b>Context</b> context, <b>GoogleMapOptions</b> options)</p> <p data-bbox="531 751 1919 779"><a href="https://developers.google.com/android/reference/com/google/android/gms/maps/MapView">https://developers.google.com/android/reference/com/google/android/gms/maps/MapView</a></p>

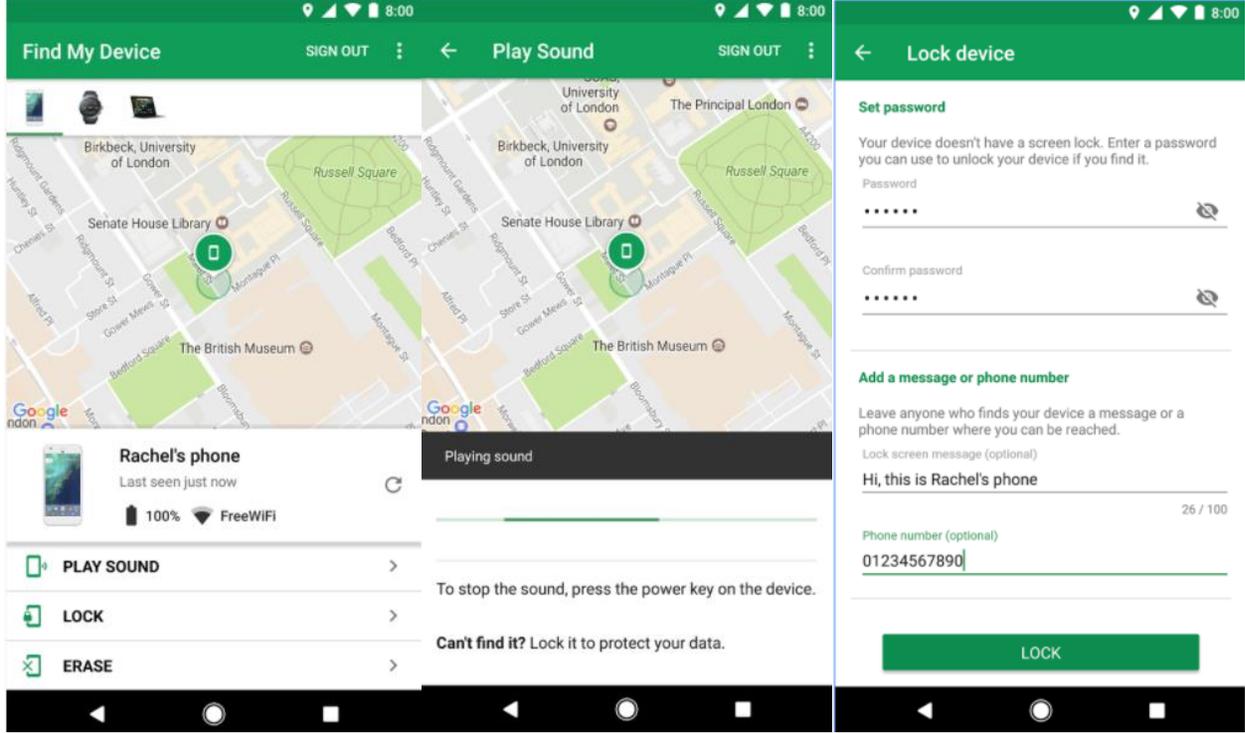
**Exhibit D - Claim Chart for U.S. Patent No. 9,467,838 Against HTC**

US9467838	Exemplary Supporting Evidence Regarding Accused Products		
	<p><code>public void <b>getMapAsync</b> (<b>OnMapReadyCallback</b> callback)</code></p> <p>Returns a non-null instance of the <code>GoogleMap</code>, ready to be used.</p> <p>Note that:</p> <ul style="list-style-type: none"> <li>• This method must be called from the main thread.</li> <li>• The callback will be executed in the main thread.</li> <li>• In the case where Google Play services is not installed on the user's device, the callback will not be triggered until the user installs it.</li> <li>• The <code>GoogleMap</code> object provided by the callback is non-null.</li> </ul> <p>Parameters</p> <table border="1" data-bbox="554 688 1761 753"> <tr> <td><code>callback</code></td> <td>The callback object that will be triggered when the map is ready to be used.</td> </tr> </table> <p><code>public final void <b>onCreate</b> (<b>Bundle</b> savedInstanceState)</code></p> <p>You must call this method from the parent Activity/Fragment's corresponding method.</p> <p><a href="https://developers.google.com/android/reference/com/google/android/gms/maps/MapView">https://developers.google.com/android/reference/com/google/android/gms/maps/MapView</a></p>	<code>callback</code>	The callback object that will be triggered when the map is ready to be used.
<code>callback</code>	The callback object that will be triggered when the map is ready to be used.		
<p>[1H] and identifying user interaction with the interactive display selecting one or more of the second set of user-selectable symbols corresponding to one or more of the second devices and positioned on the second georeferenced map and</p>	<p>HTC infringes directly and/or indirectly by performing, inducing others to perform, and/or contributing to the performance of: identifying user interaction with the interactive display selecting one or more of the second set of user-selectable symbols corresponding to one or more of the second devices and positioned on the second georeferenced map and user interaction with the display specifying an action and, based thereon, sending third data to the selected one or more second devices via the first server.</p> <p><b><u>Regarding Find My Device</u></b> and Android Device Manager, the Accused Products are configured to allow a user of a first device to interact with the display, to select a device corresponding to a symbol, and to select an action to be performed, such as: play a sound on the second device, put the second device into a lost mode, and erase the second device. Selection of one of the aforementioned actions results in sending data from the first device to a server and then sending data from the server to the second device.</p>		

**Exhibit D - Claim Chart for U.S. Patent No. 9,467,838 Against HTC**

US9467838	Exemplary Supporting Evidence Regarding Accused Products
<p>user interaction with the display specifying an action and, based thereon, sending third data to the selected one or more second devices via the first server.</p>	<p><b><u>Regarding Google Maps</u></b>, Google Latitude, Google Plus, Google Hangouts, Google Messages, and Android Messenger, the Accused Products are configured to allow a user of the first device to interact with the display, to select a user, contact, or device, and to select an action to be performed, such as: sending a message, initiating a call, initiating a data conference, sharing a location, stop sharing a location, block a user from location sharing, sending a location, requesting a location, or sending other data. When an action is specified, data is sent from the first device to the second device via a server.</p> <p><b><u>Exemplary Support for Find My Device:</u></b></p> <p>Introducing Find My Device - the new and improved Android Device Manager. Find My Device helps you easily locate a lost Android device, and keeps your information safe and sound while you look.</p> <p><b>Locate your phone, tablet or watch.</b> Misplaced your Android Wear device? No problem.</p> <p><b>Play a sound.</b> Find My Device helps you track down your device when it's close by.</p> <p><b>Lock, erase or show a message.</b> With Find My Device you can secure your device remotely and help someone get in touch.</p> <p>Permissions Notice</p> <ul style="list-style-type: none"> <li>• Location: Needed to show your device's current location on the map.</li> <li>• Contacts: Needed to access the email address associated with your Google account.</li> </ul> <p><a href="https://play.google.com/store/apps/details?id=com.google.android.apps.adm&amp;hl=en">https://play.google.com/store/apps/details?id=com.google.android.apps.adm&amp;hl=en</a></p>

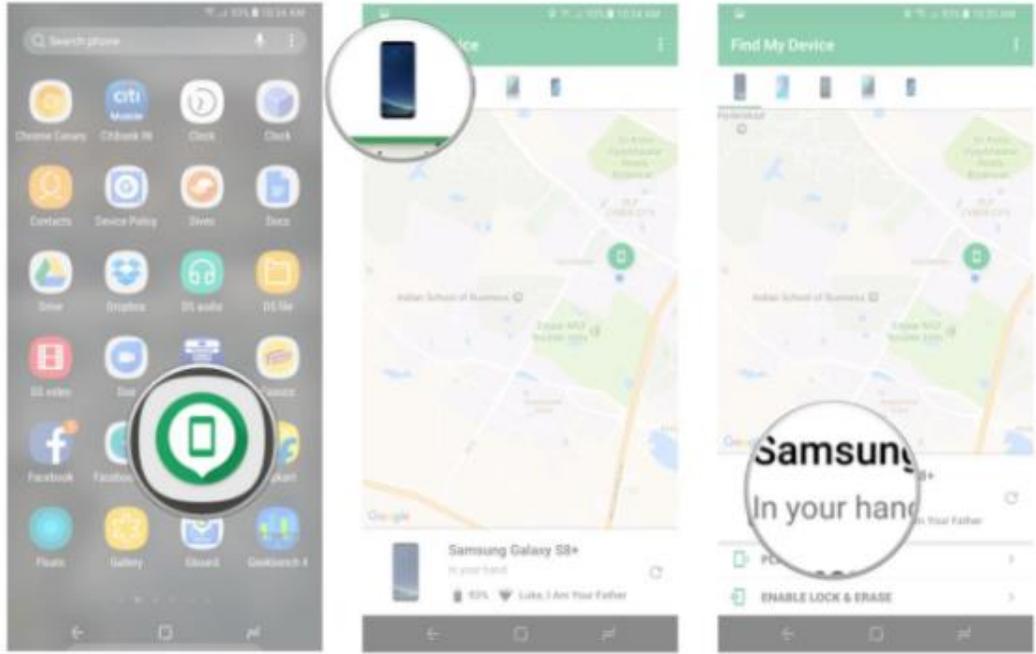
Exhibit D - Claim Chart for U.S. Patent No. 9,467,838 Against HTC

US9467838	Exemplary Supporting Evidence Regarding Accused Products
	 <p><a href="https://play.google.com/store/apps/details?id=com.google.android.apps.adm&amp;hl=en">https://play.google.com/store/apps/details?id=com.google.android.apps.adm&amp;hl=en</a></p>

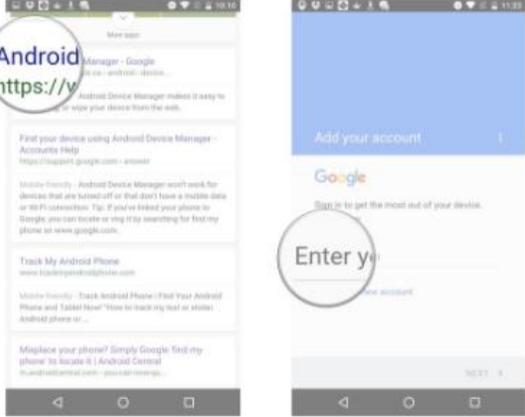
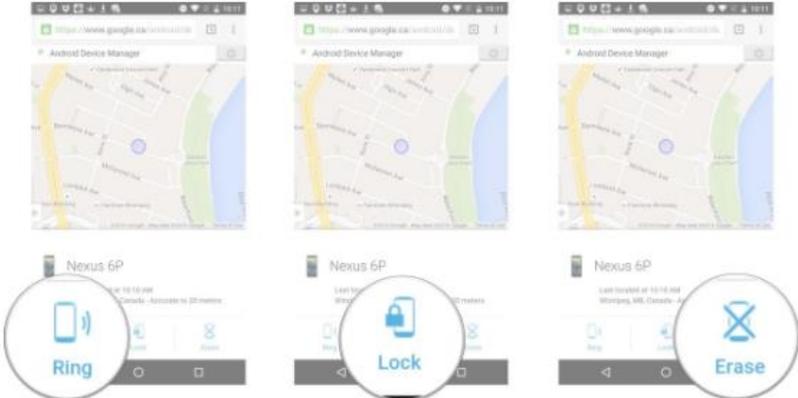
**Exhibit D - Claim Chart for U.S. Patent No. 9,467,838 Against HTC**

US9467838	Exemplary Supporting Evidence Regarding Accused Products
	<p data-bbox="556 240 1205 277"><b>How to locate your phone with Google</b></p> <p data-bbox="556 310 1520 363">Should you happen to lose your phone, you can locate its whereabouts by logging into your Google account from any computer or even from another phone.</p> <ol data-bbox="556 407 1276 532" style="list-style-type: none"> <li data-bbox="556 407 1136 431">1. Launch a web browser from a phone, tablet, or computer.</li> <li data-bbox="556 459 1276 483">2. Navigate to Google if it is not your default search engine or home page.</li> <li data-bbox="556 511 1108 535">3. Type find my phone android in the Google search bar.</li> </ol> <div data-bbox="585 565 1549 1133"> <p>The image contains three sequential screenshots from an Android phone. The first screenshot shows the home screen with various app icons; the Chrome browser icon is circled in red. The second screenshot shows the Google search page with the search bar containing the word 'google' and the 'ALL' filter selected. The third screenshot shows the search results for 'find my phone android', with the first result, 'find my phone android app', circled in red.</p> </div> <ol data-bbox="556 1182 1549 1317" style="list-style-type: none"> <li data-bbox="556 1182 1171 1206">4. Tap on Find My Device (usually the first option in the search).</li> <li data-bbox="556 1230 1549 1317">5. Enter your email address and password just as though you were checking your email. If you have 2-step verification set up on your Google account (and you most certainly should), you'll need to complete that process as well.</li> </ol> <p data-bbox="535 1323 1285 1356"><a href="https://www.androidcentral.com/how-track-android-phone">https://www.androidcentral.com/how-track-android-phone</a></p>

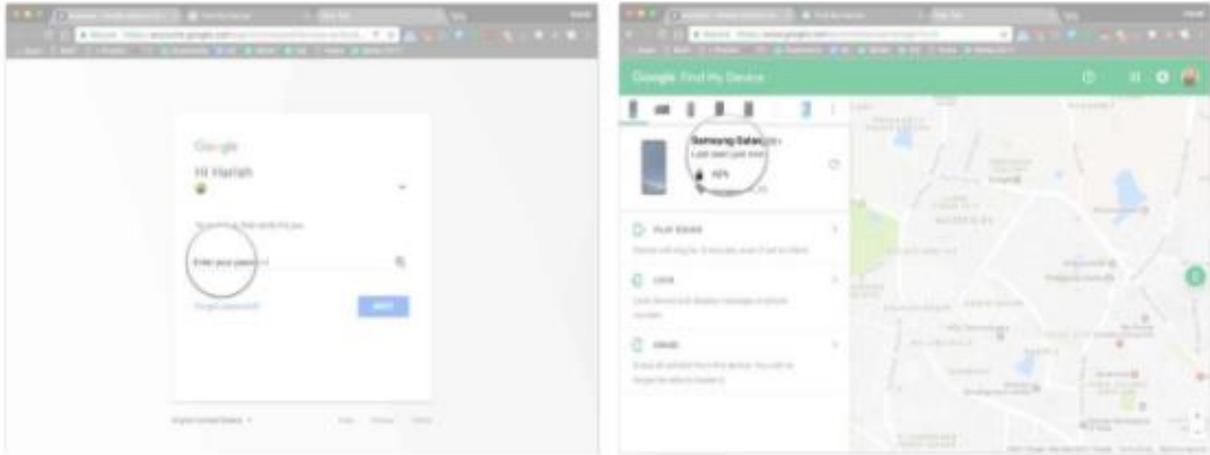
**Exhibit D - Claim Chart for U.S. Patent No. 9,467,838 Against HTC**

US9467838	Exemplary Supporting Evidence Regarding Accused Products
	<p>Once you're signed in to Find My Device, you'll see a map with your current location as well as the make and model of your phone, and two options — Play Sound, and Enable Lock &amp; Erase. Hitting the latter option will allow you to start using the Lock and Erase functions.</p> <p>If you've signed into more than one phone, you can select a particular device by browsing the list at the top of the screen.</p> <ol style="list-style-type: none"> <li>1. Open <b>Find My Device</b> from your home screen or app drawer.</li> <li>2. Select your phone from the list of devices at the top of the screen.</li> <li>3. See if your phone is discoverable.</li> </ol>  <p><a href="https://www.androidcentral.com/find-my-device">https://www.androidcentral.com/find-my-device</a></p>

**Exhibit D - Claim Chart for U.S. Patent No. 9,467,838 Against HTC**

US9467838	Exemplary Supporting Evidence Regarding Accused Products
	 <p>When your phone is located, you have three options to choose from:</p> <ul style="list-style-type: none"> <li>• You can <b>Ring</b> your phone so that it makes noise (even if you had it on silent). This feature is helpful if the map indicates that the phone is within earshot and you simply can't see it.</li> <li>• You can <b>Lock</b> your phone so that the finder can't access your home screen. This feature is most helpful if your phone wasn't previously secured with a passcode or a fingerprint sensor.</li> <li>• You can <b>Erase</b> your phone. This is the best option if you know for certain that you aren't likely to retrieve your phone.</li> </ul>  <p><a href="https://www.androidcentral.com/how-track-android-phone">https://www.androidcentral.com/how-track-android-phone</a></p>

**Exhibit D - Claim Chart for U.S. Patent No. 9,467,838 Against HTC**

US9467838	Exemplary Supporting Evidence Regarding Accused Products
	<h2 data-bbox="548 272 1688 326">How to locate your phone over the internet</h2> <p data-bbox="548 370 1696 513">If you've lost your phone, you can remotely locate it through the <a href="#">Find My Device website</a>. You'll need to sign in to the Google account that was used to set up Find My Device. It takes a few seconds, but the service should be able to track your phone. Alternatively, you can also do a Google search for "find my phone" to locate your handset.</p> <ol data-bbox="541 565 1031 724" style="list-style-type: none"><li>1. Head to the <a href="#">Find My Device website</a>.</li><li>2. Sign in to your <a href="#">Google account</a>.</li><li>3. Check if your device is visible.</li></ol> <div data-bbox="583 764 1793 1219"></div> <p data-bbox="533 1247 1157 1279"><a href="https://www.androidcentral.com/find-my-device">https://www.androidcentral.com/find-my-device</a></p>



**Exhibit D - Claim Chart for U.S. Patent No. 9,467,838 Against HTC**

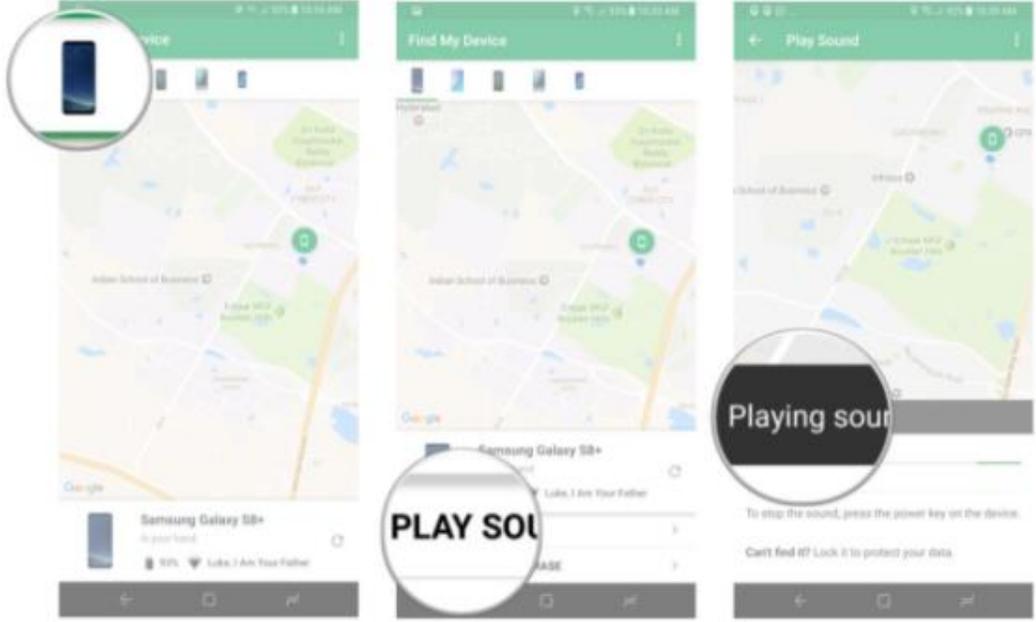
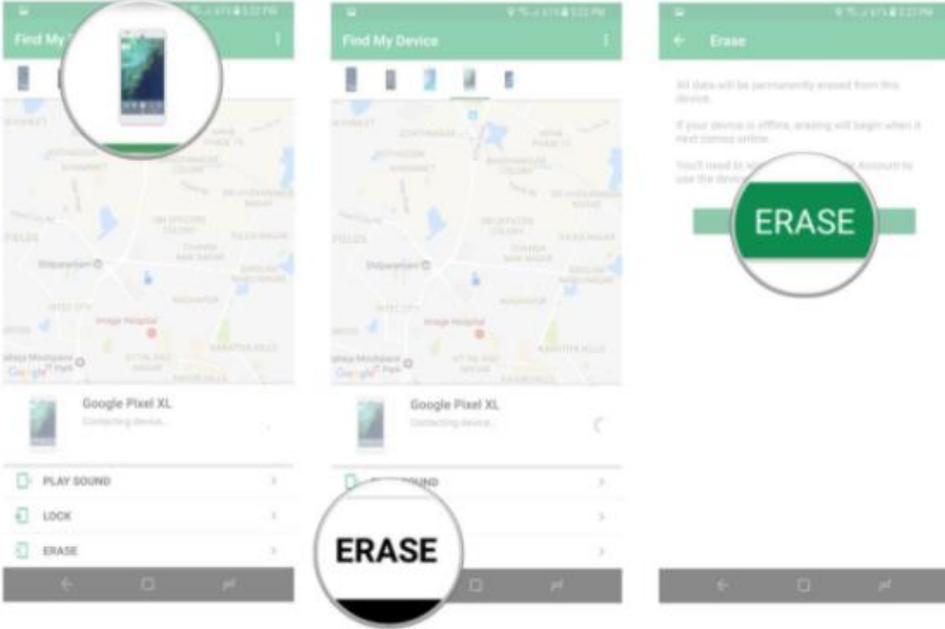
US9467838	Exemplary Supporting Evidence Regarding Accused Products
	<p data-bbox="552 240 1556 289"><b>How to ring your phone with Find My Device</b></p> <p data-bbox="552 323 1581 477">The best part about Find My Device is that it is easily accessible. If you need to locate your phone, just head to the website or log in to the service from another phone. Once you sign in to Find My Device and locate your device, you can use the <b>Play Sound</b> option, which plays a loud tone on your phone continuously at full volume for five minutes even if you turned the ringer off. Once you find your phone, you can hit the power button to stop the ringing.</p> <ol data-bbox="546 524 1377 656" style="list-style-type: none"> <li>1. Locate your phone on Find My Device.</li> <li>2. Tap Play Sound.</li> <li>3. Your device will start ringing. You can hit the power button to stop the sound.</li> </ol>  <p data-bbox="533 1344 1157 1377"><a href="https://www.androidcentral.com/find-my-device">https://www.androidcentral.com/find-my-device</a></p>

Exhibit D - Claim Chart for U.S. Patent No. 9,467,838 Against HTC

US9467838	Exemplary Supporting Evidence Regarding Accused Products
	<div data-bbox="541 272 1600 326" data-label="Section-Header"> <h2>How to lock your phone with Find My Device</h2> </div> <div data-bbox="541 358 1610 456" data-label="Text"> <p>There's also a <b>Lock</b> option that lets you set a new password to unlock the phone. You can also display a message over the lock screen and add a button to call back your number so that anyone that comes across your phone can easily get in touch with you.</p> </div> <div data-bbox="533 505 1444 646" data-label="List-Group"> <ol style="list-style-type: none"> <li>1. Locate your phone on Find My Device.</li> <li>2. Tap <b>Lock</b>.</li> <li>3. Enter a message and phone number to display on the lock screen and tap <b>Lock</b>.</li> </ol> </div> <div data-bbox="577 678 1654 1352" data-label="Image"> <p>The image consists of three sequential screenshots from the Find My Device app. The first screenshot shows the 'Find My Device' map interface with a green location pin and a 'LOCK' callout bubble. The second screenshot shows the 'Lock device' screen with fields for 'Add a message or phone number', 'Lock screen message (optional)', and 'Phone number (optional)', with a 'LOCK' callout bubble. The third screenshot shows the lock screen with a 'LOCK' callout bubble.</p> </div> <div data-bbox="527 1369 1159 1404" data-label="Text"> <p><a href="https://www.androidcentral.com/find-my-device">https://www.androidcentral.com/find-my-device</a></p> </div>

**Exhibit D - Claim Chart for U.S. Patent No. 9,467,838 Against HTC**

US9467838	Exemplary Supporting Evidence Regarding Accused Products
	<p data-bbox="550 240 1495 282"><b>How to erase your lost phone's data remotely</b></p> <p data-bbox="550 316 1503 456">If you're certain that you're not going to see your phone again, there is the nuclear option of erasing the data remotely. Selecting the <b>Erase</b> option deletes all the data on your phone. The service also deletes data from a connected SD card, but there is a chance that it may not be able to, based on the manufacturer and Android platform version. Even if your phone is switched off when you send the Erase command, the factory reset process will be initiated as soon as it goes online.</p> <ol data-bbox="550 500 1075 623" style="list-style-type: none"> <li>1. Locate your phone on <b>Find My Device</b>.</li> <li>2. Tap <b>Erase</b>.</li> <li>3. Confirm deletion of data by hitting the <b>Erase</b> button.</li> </ol>  <p data-bbox="535 1317 1157 1349"><a href="https://www.androidcentral.com/find-my-device">https://www.androidcentral.com/find-my-device</a></p> <p data-bbox="535 1388 1052 1421"><b>Exemplary Support for Google Maps:</b></p>

**Exhibit D - Claim Chart for U.S. Patent No. 9,467,838 Against HTC**

US9467838	Exemplary Supporting Evidence Regarding Accused Products
	<p style="text-align: center;"> <a href="#">COMPUTER</a>   <a href="#">ANDROID</a>   <a href="#">IPHONE &amp; IPAD</a> </p> <hr/> <h3>If they have a Google Account</h3> <ol style="list-style-type: none"> <li>1. If you haven't already, add their Gmail address to your <a href="#">Google Contacts</a> .</li> <li>2. On your Android phone or tablet, open the Google Maps app  and sign in. <a href="#">Learn how to sign in.</a></li> <li>3. Tap Menu ≡ &gt; <b>Location sharing</b> &gt; Add People .</li> <li>4. Choose how long you want to share your location.</li> <li>5. Tap <b>Select People</b>.             <ul style="list-style-type: none"> <li>• If you're asked about your contacts, give Google Maps access.</li> </ul> </li> <li>6. Choose who you want to share with.</li> <li>7. Tap <b>Share</b>.</li> </ol> <h3>If they don't have a Google Account</h3> <ol style="list-style-type: none"> <li>1. On your Android phone or tablet, open the Google Maps app  and sign in. <a href="#">Learn how to sign in.</a></li> <li>2. Tap Menu ≡ &gt; <b>Location sharing</b> &gt; Add People .</li> <li>3. Tap More ... &gt; <b>Copy to clipboard</b>. People with this link can see your location for as long as you choose, up to 72 hours.</li> </ol> <h3>Share using another app</h3> <p>You can also share through messaging apps. Tap More ... &gt; select an app.</p> <h3>Stop sharing</h3> <ol style="list-style-type: none"> <li>1. Open the Google Maps app .</li> <li>2. Tap Menu ≡ &gt; <b>Location sharing</b>.</li> <li>3. Next to the person with whom you want to stop sharing, tap Remove ✕ .</li> </ol> <p><a href="https://support.google.com/maps/answer/7326816?co=GENIE.Platform%3DAndroid&amp;oco=1">https://support.google.com/maps/answer/7326816?co=GENIE.Platform%3DAndroid&amp;oco=1</a></p>

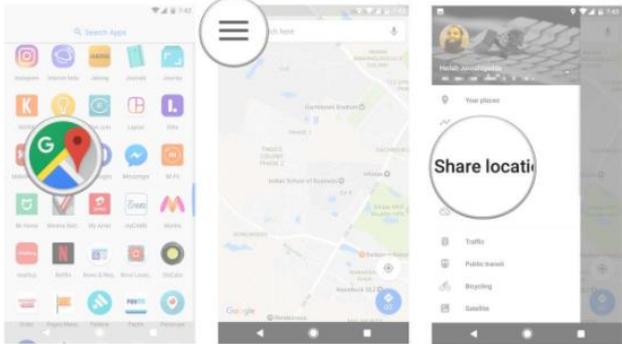
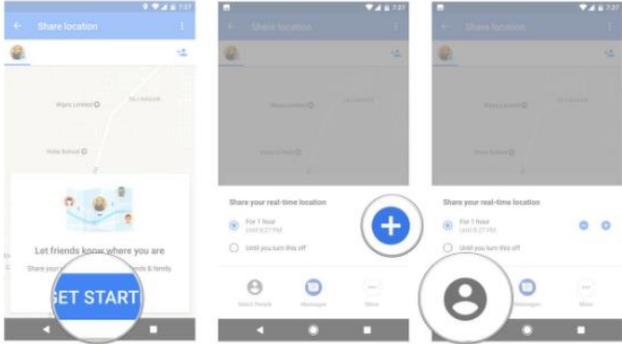
**Exhibit D - Claim Chart for U.S. Patent No. 9,467,838 Against HTC**

US9467838	Exemplary Supporting Evidence Regarding Accused Products
	<p data-bbox="556 253 863 297"><b>Share your E.T.A</b></p> <p data-bbox="556 321 1686 347">After you start your drive, you can share your destination, estimated arrival time, and where you are on the route.</p> <ol data-bbox="556 375 1381 618" style="list-style-type: none"> <li data-bbox="556 375 911 401">1. Open the Google Maps app .</li> <li data-bbox="556 418 1184 444">2. Set a driving destination. <a href="#">Learn how to navigate to a place.</a></li> <li data-bbox="556 462 1224 488">3. After you start navigation, tap More ^ &gt; <b>Share trip progress.</b></li> <li data-bbox="556 506 898 532">4. Choose a person from the list.</li> <li data-bbox="556 550 701 576">5. Tap <b>Share.</b></li> <li data-bbox="556 594 1381 620">6. Location Sharing will stop when you reach your destination or stop navigating.</li> </ol> <ul data-bbox="556 646 1226 672" style="list-style-type: none"> <li data-bbox="556 646 1226 672">• To stop sharing before you arrive, tap More ^ &gt; <b>Stop sharing.</b></li> </ul> <p data-bbox="556 740 978 784"><b>See where someone is</b></p> <p data-bbox="556 808 1289 834">If someone shares their location with you, you can see them on the map.</p> <ol data-bbox="556 862 940 976" style="list-style-type: none"> <li data-bbox="556 862 911 888">1. Open the Google Maps app .</li> <li data-bbox="556 906 940 932">2. Tap Menu ≡ &gt; <b>Location sharing.</b></li> <li data-bbox="556 950 772 976">3. Choose someone.</li> </ol> <ul data-bbox="556 1002 1325 1027" style="list-style-type: none"> <li data-bbox="556 1002 1325 1027">• To see an updated location, tap on a friend's icon &gt; More ⋮ &gt; <b>Refresh.</b></li> </ul> <p data-bbox="556 1084 1056 1128"><b>Stop seeing someone's location</b></p> <ol data-bbox="556 1146 1482 1300" style="list-style-type: none"> <li data-bbox="556 1146 911 1172">1. Open the Google Maps app .</li> <li data-bbox="556 1190 856 1216">2. On the map, tap their icon.</li> <li data-bbox="556 1234 877 1260">3. At the bottom, tap More ^ .</li> <li data-bbox="556 1278 1482 1304">4. To temporarily hide someone, tap <b>Hide from map.</b> You can stop hiding them at any time.</li> </ol> <p data-bbox="556 1330 1766 1356"><b>Note:</b> You can stop someone's location from ever appearing on your map. <a href="#">Learn how to block another person's account.</a></p> <p data-bbox="533 1377 1724 1403"><a href="https://support.google.com/maps/answer/7326816?co=GENIE.Platform%3DAndroid&amp;oco=1">https://support.google.com/maps/answer/7326816?co=GENIE.Platform%3DAndroid&amp;oco=1</a></p>

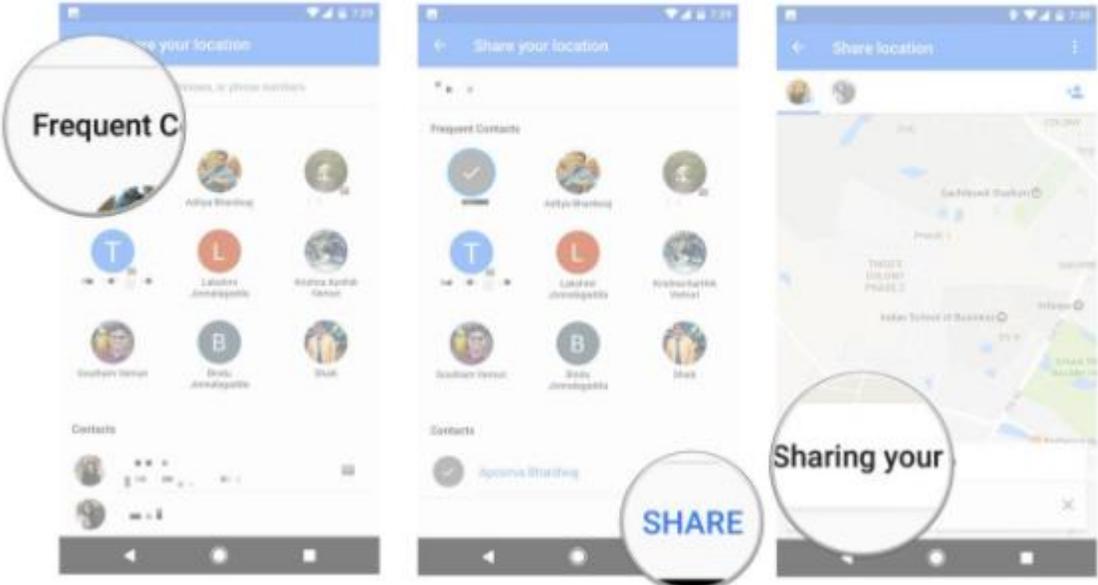
**Exhibit D - Claim Chart for U.S. Patent No. 9,467,838 Against HTC**

US9467838	Exemplary Supporting Evidence Regarding Accused Products
	<p data-bbox="569 282 898 321"><b>Hide or share lists</b></p> <p data-bbox="569 349 930 373"><b>Note:</b> You can't share starred places.</p> <ol data-bbox="579 402 1703 667" style="list-style-type: none"> <li data-bbox="579 402 911 427">1. Open the Google Maps app .</li> <li data-bbox="579 444 989 469">2. Tap Menu  &gt; <b>Your places</b> &gt; <b>Saved</b>.</li> <li data-bbox="579 487 1703 667">3. Next to the list you want to share, tap More  &gt; choose an option: <ul data-bbox="604 529 1703 667" style="list-style-type: none"> <li data-bbox="604 529 1465 553">• <b>Hide/Show on your map:</b> Display or hide your saved places when looking at the map.</li> <li data-bbox="604 571 1079 596">• <b>Share list:</b> Allow others to see your saved list.</li> <li data-bbox="604 613 1703 667">• <b>Sharing options:</b> You can make your list public, private, or shared. To let anyone with the link see your list, tap <b>Shared</b>. To let anyone find and follow your list, tap <b>Public</b>.</li> </ul> </li> </ol> <p data-bbox="569 737 789 776"><b>Follow a list</b></p> <p data-bbox="569 803 1749 862">If you follow a list made by someone else, their saved places will show up in Your Places. The places will also appear as suggested locations in Google Maps.</p> <p data-bbox="569 915 936 954"><b>Follow a list using a link</b></p> <ol data-bbox="579 976 1377 1081" style="list-style-type: none"> <li data-bbox="579 976 982 1000">1. Tap on the link you received to open it.</li> <li data-bbox="579 1018 1293 1042">2. Tap <b>Follow</b>. This list will now be added to the group of lists you follow.</li> <li data-bbox="579 1060 1377 1081">3. <b>Optional:</b> To unfollow a list someone shared with you, tap the list &gt; <b>Following</b>.</li> </ol> <p data-bbox="569 1136 947 1175"><b>See lists made by others</b></p> <p data-bbox="569 1196 1356 1221">If a user has any Google Maps lists that were made public, you can follow them.</p> <ol data-bbox="579 1252 1157 1357" style="list-style-type: none"> <li data-bbox="579 1252 1157 1276">1. Tap on the name of a user whose list you want to follow.</li> <li data-bbox="579 1294 699 1318">2. Tap <b>Lists</b>.</li> <li data-bbox="579 1336 1157 1357">3. Tap on the list you want to follow &gt; More  &gt; <b>Follow</b>.</li> </ol> <p data-bbox="537 1375 1906 1403"><a href="https://support.google.com/maps/answer/7280933?hl=en&amp;ref_topic=7301134&amp;co=GENIE.Platform%3DA">https://support.google.com/maps/answer/7280933?hl=en&amp;ref_topic=7301134&amp;co=GENIE.Platform%3DA</a></p>

**Exhibit D - Claim Chart for U.S. Patent No. 9,467,838 Against HTC**

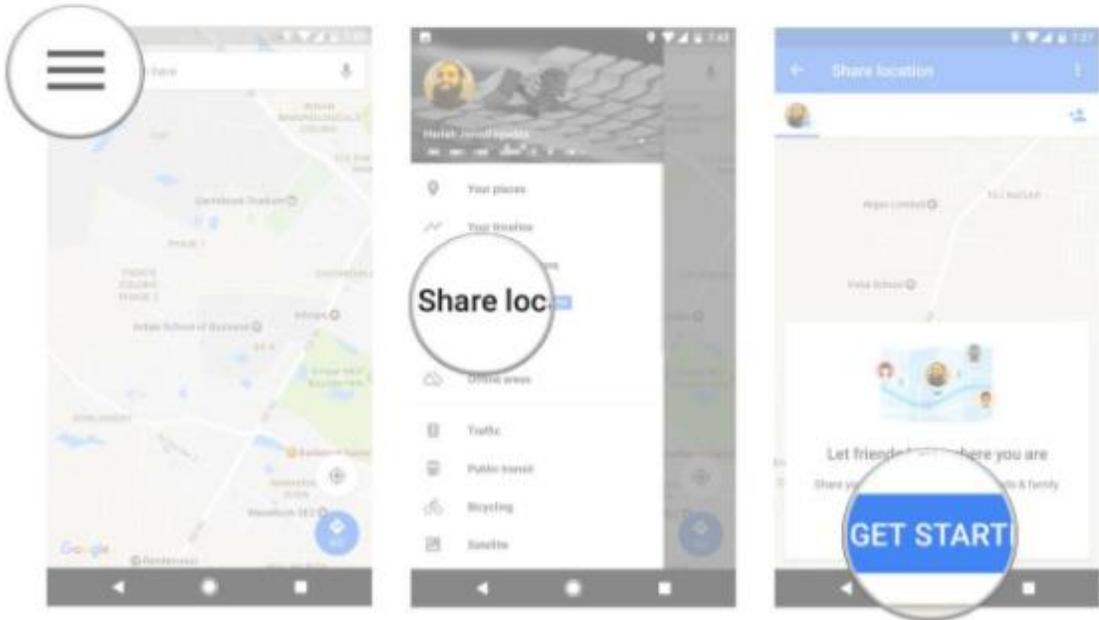
US9467838	Exemplary Supporting Evidence Regarding Accused Products
	<p>ndroid&amp;oco=1</p> <p><b>How to share your location in Google Maps</b></p> <ol style="list-style-type: none"> <li>1. Open <b>Google Maps</b> from the app drawer or the home screen.</li> <li>2. Tap the <b>hamburger menu</b> (the three horizontal lines) on the top left corner of the screen.</li> <li>3. Select <b>Share location</b>.</li> </ol>  <ol style="list-style-type: none"> <li>4. Tap <b>Get Started</b>.</li> <li>5. Use the <b>+</b> icon to select a time period or select the <b>Until you turn this off</b> setting to share your location indefinitely.</li> <li>6. Tap <b>Select People</b>.</li> </ol>  <p><a href="https://www.androidcentral.com/how-share-location-google-maps">https://www.androidcentral.com/how-share-location-google-maps</a></p>

**Exhibit D - Claim Chart for U.S. Patent No. 9,467,838 Against HTC**

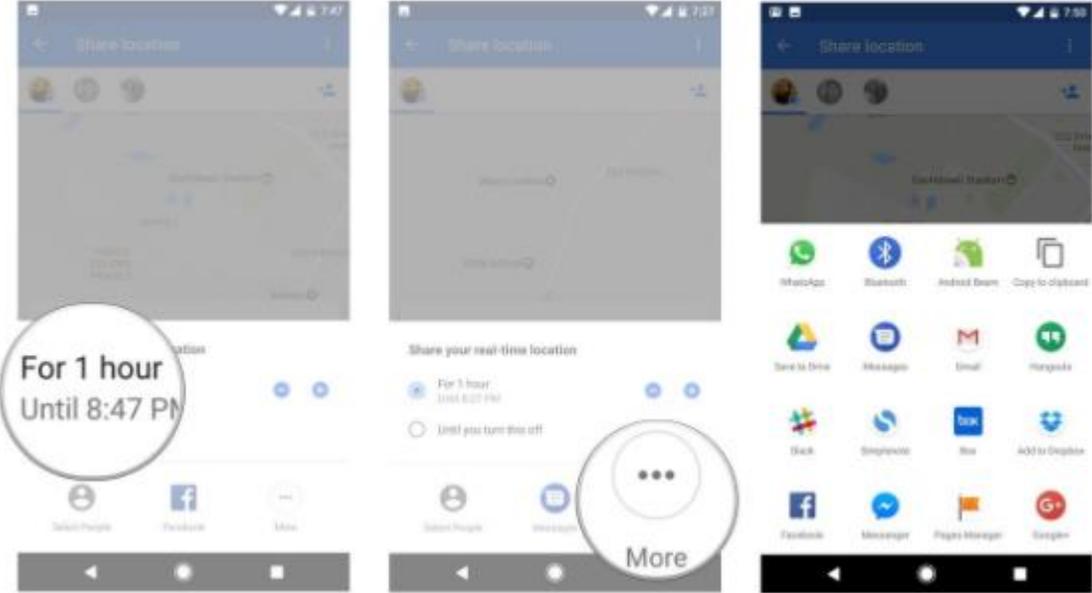
US9467838	Exemplary Supporting Evidence Regarding Accused Products
	<p>7. You'll see a list of your frequent contacts at the top, along with a full list of contacts. Pick the contacts by tapping their name.</p> <p>8. Once you've selected the contacts you want to share your location to, tap <b>Share</b>.</p> <p>9. You'll see a message saying that the selected contact can view your location.</p>  <p><a href="https://www.androidcentral.com/how-share-location-google-maps">https://www.androidcentral.com/how-share-location-google-maps</a></p>



**Exhibit D - Claim Chart for U.S. Patent No. 9,467,838 Against HTC**

US9467838	Exemplary Supporting Evidence Regarding Accused Products
	<h2 data-bbox="548 240 1281 289">How to create a shareable link</h2> <p data-bbox="548 329 1486 358">You can also create a link and use it to share your location easily. Here's how to do it:</p> <ol data-bbox="541 407 1260 548" style="list-style-type: none"><li>1. Tap the hamburger menu on the top left corner of the screen.</li><li>2. Select Share location.</li><li>3. Tap Get Started.</li></ol>  <p data-bbox="533 1230 1381 1263"><a href="https://www.androidcentral.com/how-share-location-google-maps">https://www.androidcentral.com/how-share-location-google-maps</a></p>

**Exhibit D - Claim Chart for U.S. Patent No. 9,467,838 Against HTC**

US9467838	Exemplary Supporting Evidence Regarding Accused Products
	<p>4. Select the amount of time you want to share your location.</p> <p>5. Tap More.</p> <p>6. Select your app of choice to create and send a unique URL that broadcasts your current location. You can email it, send the link via Messenger, or even tweet it to the intended recipient.</p>  <p><a href="https://www.androidcentral.com/how-share-location-google-maps">https://www.androidcentral.com/how-share-location-google-maps</a></p>

**Exhibit D - Claim Chart for U.S. Patent No. 9,467,838 Against HTC**

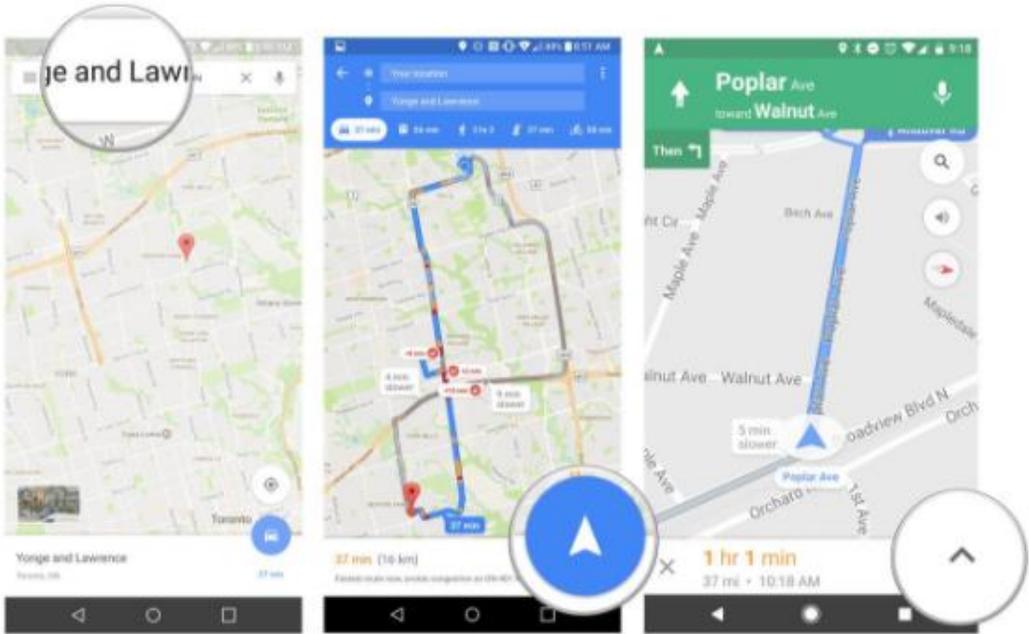
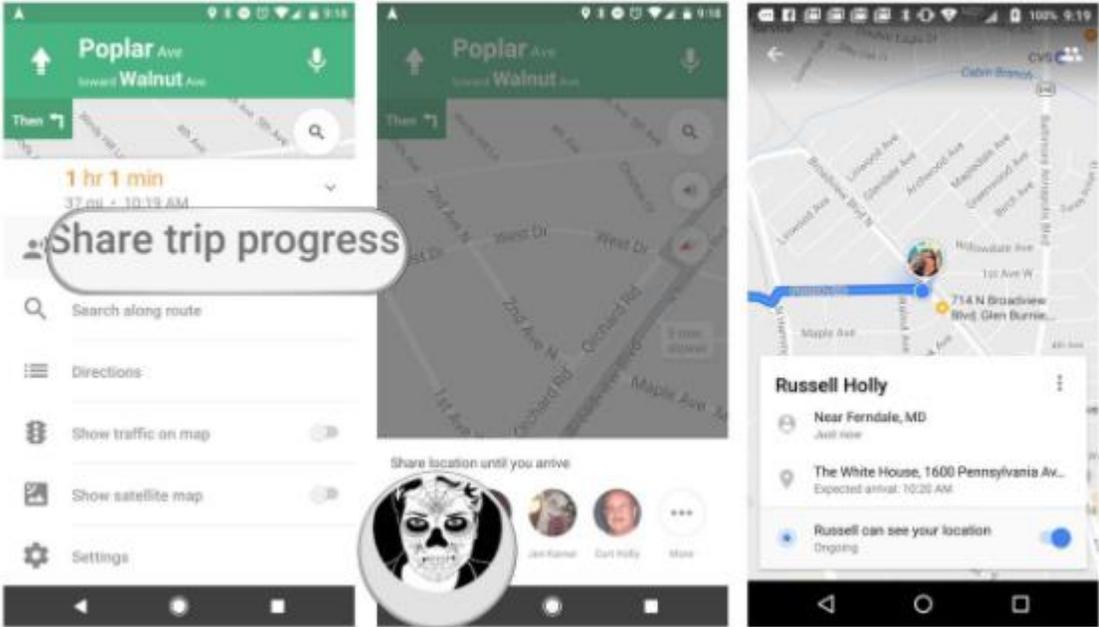
US9467838	Exemplary Supporting Evidence Regarding Accused Products
	<h2 data-bbox="548 240 1451 337">How to share your navigation directions while you walk, drive or transit</h2> <p data-bbox="548 375 1581 464">One of the best ways to share your location with someone is during a drive, walk or transit. If you're meeting a friend or family member somewhere, or navigating towards their home, Google Maps lets you share your location with them for the duration of the trip. It's magic!</p> <ol data-bbox="548 513 1419 643" style="list-style-type: none"><li>1. In the <b>search bar</b> enter your destination.</li><li>2. Pick your navigation type (drive, transit, walk) and press the <b>blue navigate button</b>.</li><li>3. Tap the arrow next to the <b>time-to-destination</b> number at the bottom of the screen.</li></ol>  <p data-bbox="533 1328 1381 1360"><a href="https://www.androidcentral.com/how-share-location-google-maps">https://www.androidcentral.com/how-share-location-google-maps</a></p>

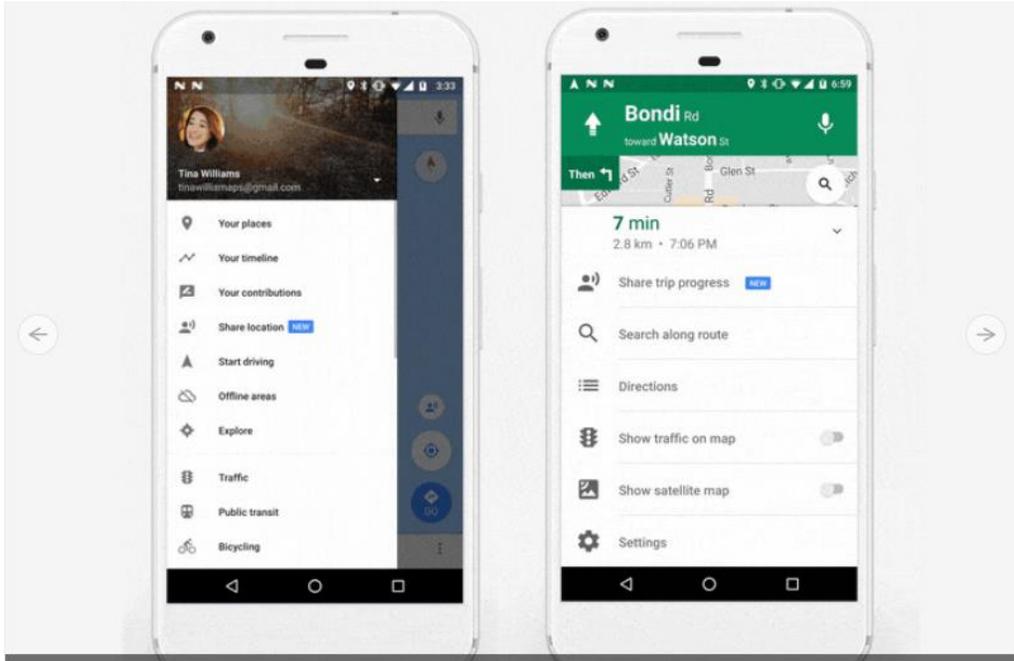
Exhibit D - Claim Chart for U.S. Patent No. 9,467,838 Against HTC

US9467838	Exemplary Supporting Evidence Regarding Accused Products
	<p data-bbox="552 277 861 305">4. Tap Share trip progress.</p> <p data-bbox="552 332 1171 360">5. Choose one or more contacts to share trip progress.</p>  <p data-bbox="552 1063 1381 1091">You can also stop sharing your location with someone before a trip ends.</p> <p data-bbox="535 1101 1381 1128"><a href="https://www.androidcentral.com/how-share-location-google-maps">https://www.androidcentral.com/how-share-location-google-maps</a></p>

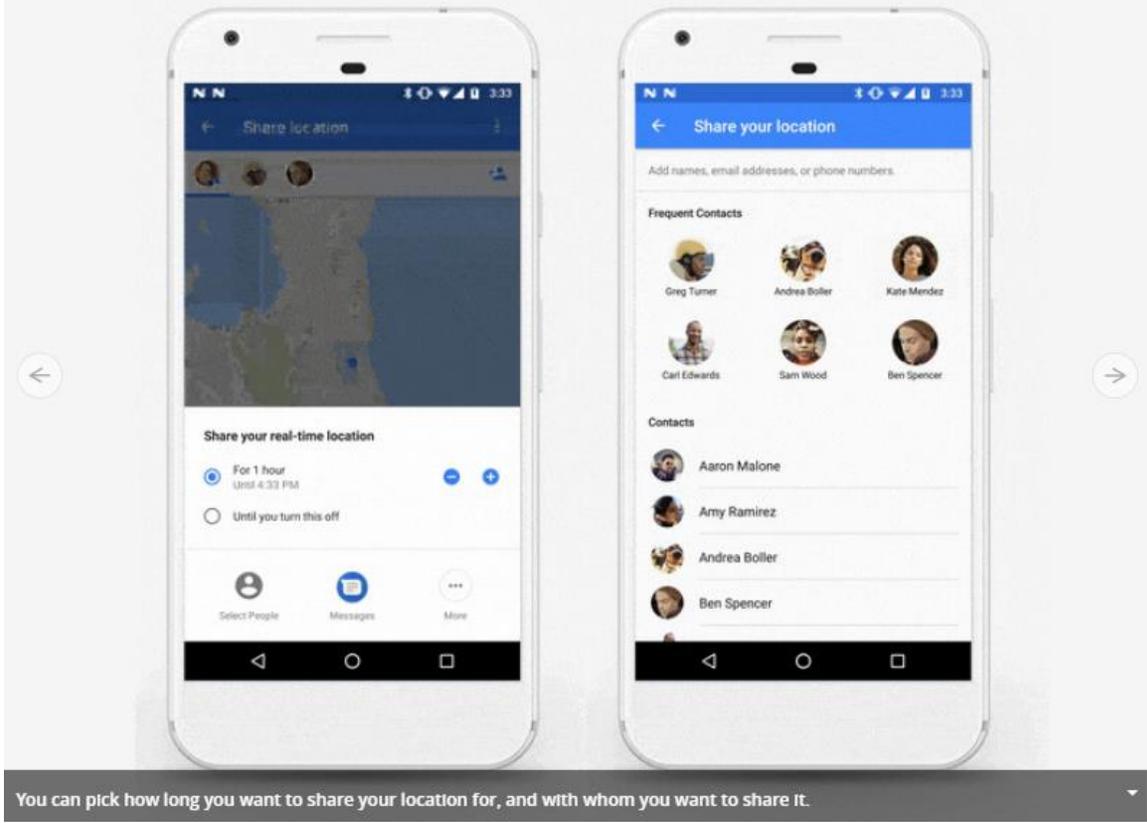
**Exhibit D - Claim Chart for U.S. Patent No. 9,467,838 Against HTC**

US9467838	Exemplary Supporting Evidence Regarding Accused Products
	<p>1. Tap the arrow next to the time-to-destination number at the bottom of the screen.</p> <p>2. Tap Stop sharing.</p> <div data-bbox="772 381 1423 922" data-label="Image"> </div> <p>That's it!</p> <p>Are you excited that location sharing is back in Google Maps? How often do you use the feature?  <a href="https://www.androidcentral.com/how-share-location-google-maps">https://www.androidcentral.com/how-share-location-google-maps</a></p> <p>As shown below, a group may also be defined within Google Contacts.</p>

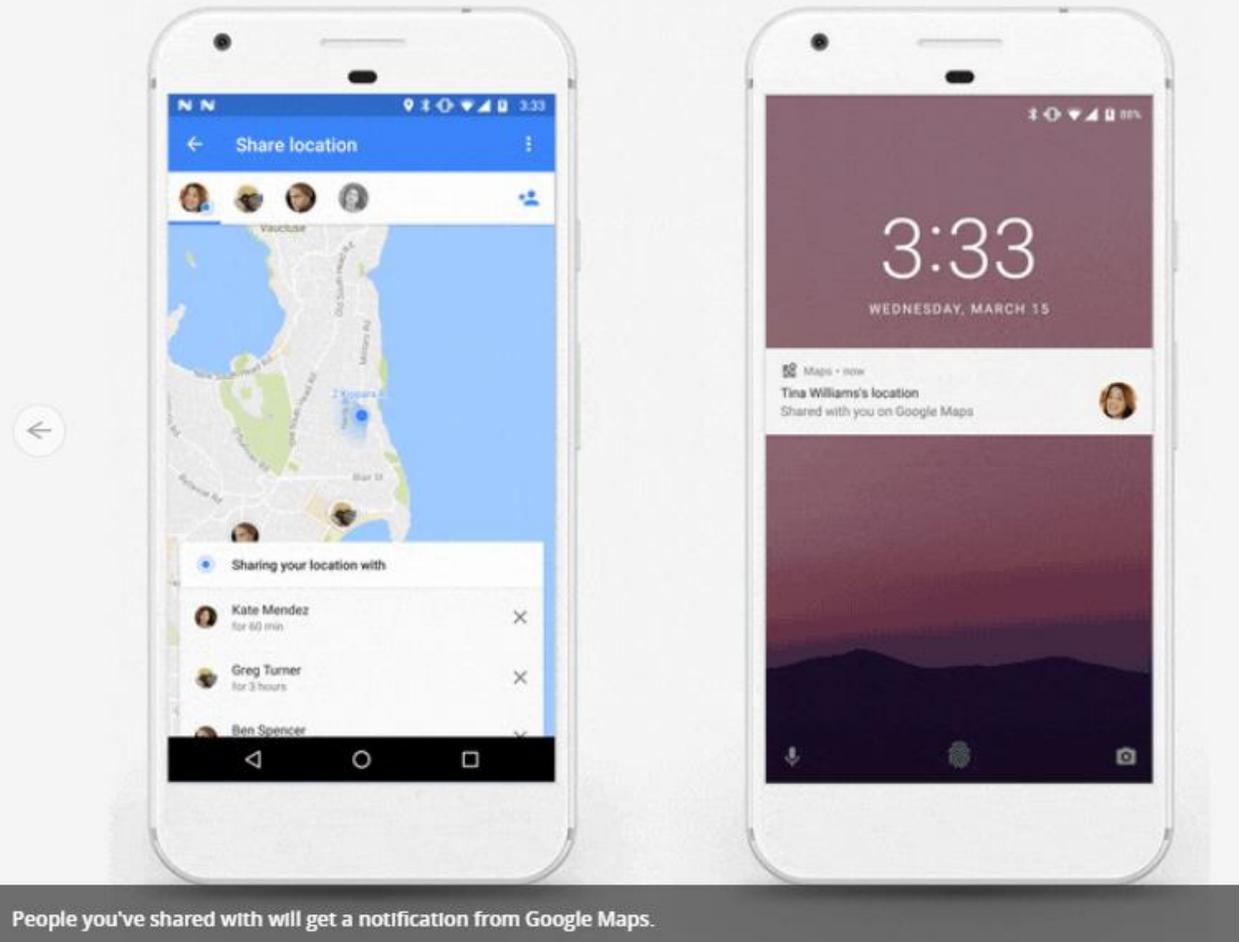
**Exhibit D - Claim Chart for U.S. Patent No. 9,467,838 Against HTC**

US9467838	Exemplary Supporting Evidence Regarding Accused Products
	<h3 data-bbox="573 245 966 289">Share your contacts</h3> <ol data-bbox="583 316 1071 479" style="list-style-type: none"><li>1. Open your device's Contacts app .</li><li>2. Tap a contact in the list.</li><li>3. Tap More  &gt; <b>Share</b>.</li><li>4. Choose how you want to share the contact.</li></ol> <p data-bbox="535 495 1560 527"><a href="https://support.google.com/android/answer/6118731?hl=en&amp;ref_topic=6118711">https://support.google.com/android/answer/6118731?hl=en&amp;ref_topic=6118711</a></p>  <p data-bbox="541 1230 1543 1291">Ways to start location sharing: (left) from the app navigation drawer, and (right) from the menu, you can drag up while navigating somewhere.</p> <p data-bbox="535 1295 1680 1323"><a href="https://arstechnica.com/gadgets/2017/03/location-sharing-finally-returns-to-google-maps/">https://arstechnica.com/gadgets/2017/03/location-sharing-finally-returns-to-google-maps/</a></p>

**Exhibit D - Claim Chart for U.S. Patent No. 9,467,838 Against HTC**

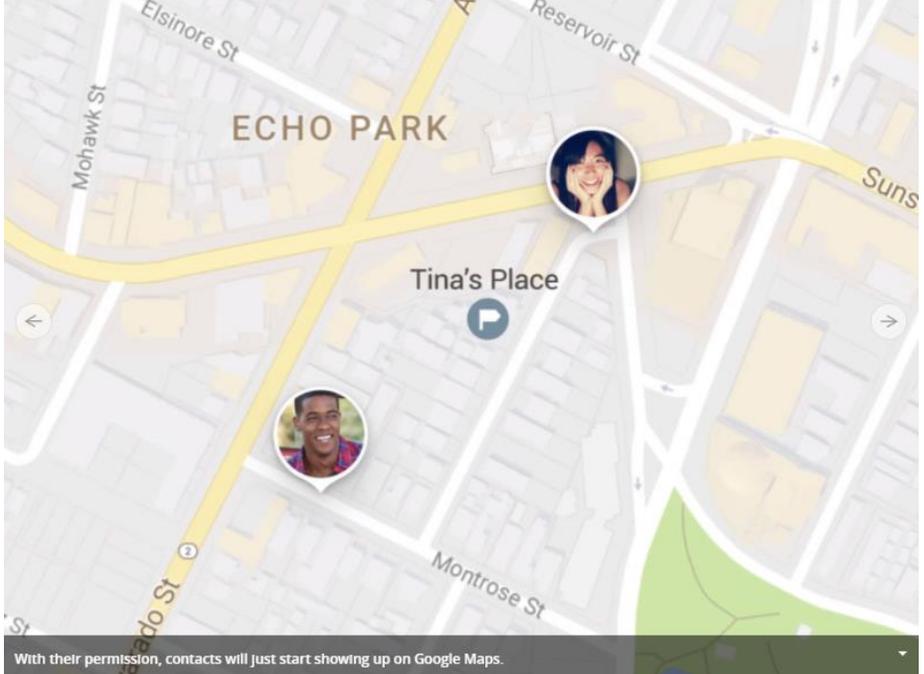
US9467838	Exemplary Supporting Evidence Regarding Accused Products
	 <p data-bbox="535 1027 1346 1052">You can pick how long you want to share your location for, and with whom you want to share it.</p> <p data-bbox="535 1065 1682 1097"><a href="https://arstechnica.com/gadgets/2017/03/location-sharing-finally-returns-to-google-maps/">https://arstechnica.com/gadgets/2017/03/location-sharing-finally-returns-to-google-maps/</a></p>

**Exhibit D - Claim Chart for U.S. Patent No. 9,467,838 Against HTC**

US9467838	Exemplary Supporting Evidence Regarding Accused Products
	 <p data-bbox="541 1144 1197 1177">People you've shared with will get a notification from Google Maps.</p> <p data-bbox="541 1188 1680 1221"><a href="https://arstechnica.com/gadgets/2017/03/location-sharing-finally-returns-to-google-maps/">https://arstechnica.com/gadgets/2017/03/location-sharing-finally-returns-to-google-maps/</a></p>



**Exhibit D - Claim Chart for U.S. Patent No. 9,467,838 Against HTC**

US9467838	Exemplary Supporting Evidence Regarding Accused Products
	 <p>The screenshot shows a Google Maps view of the Echo Park neighborhood in Los Angeles. Two circular location pins are overlaid on the map. The first pin, located near the intersection of Reservoir St and Sun St, contains a photo of a woman with dark hair. The second pin, located near the intersection of Montrose St and a street labeled 'ado St', contains a photo of a man with short dark hair. The map also shows streets like Elsinore St, Mohawk St, and Tina's Place. A blue parking 'P' icon is visible near Tina's Place. At the bottom of the map, a grey notification bar reads: "With their permission, contacts will just start showing up on Google Maps."</p> <p><a href="https://arstechnica.com/gadgets/2017/03/location-sharing-finally-returns-to-google-maps/">https://arstechnica.com/gadgets/2017/03/location-sharing-finally-returns-to-google-maps/</a></p> <p><b><u>Exemplary Find My Device Screenshots:</u></b></p>

**Exhibit D - Claim Chart for U.S. Patent No. 9,467,838 Against HTC**

US9467838	Exemplary Supporting Evidence Regarding Accused Products	
	<p>AT&amp;T 94% 11:46 AM</p> <p>Find My Device</p> <p>CONNECTICUT RHODE ISLAND</p> <p>Providence New Haven</p> <p>LG Phoenix 3 In your hand 94% wallcri...</p> <p>PLAY SOUND</p> <p>LOCK</p> <p>ERASE</p>	<p>AT&amp;T 94% 11:47 AM</p> <p>Find My Device</p> <p>← Lock device</p> <p>Add a message or phone number</p> <p>Leave anyone who finds your device a message or a phone number where you can be reached.</p> <p>Lock screen message (optional) 0 / 100</p> <p>Phone number (optional)</p> <p>LOCK</p>

**Exhibit D - Claim Chart for U.S. Patent No. 9,467,838 Against HTC**

US9467838	Exemplary Supporting Evidence Regarding Accused Products		
	<p>AT&amp;T 93% 11:48 AM</p> <p>← Lock device</p> <p><b>Add a message or phone number</b></p> <p>Leave anyone who finds your device a message or a phone number where you can be reached.</p> <p>Lock screen message (optional)</p> <p><b>This is a force message alert!</b></p> <p>32 / 100</p> <p>Phone number (optional)</p> <p>123456789</p> <p><b>LOCK</b></p>	<p>AT&amp;T 93% 11:48 AM</p> <p>Find My Device</p> <p>Lock requested</p> <p><b>LG Phoenix 3</b></p> <p>Last seen 3 mins ago</p> <p>24% walkcri...</p> <p>PLAY SOUND &gt;</p> <p>LOCK &gt;</p> <p>ERASE &gt;</p>	<p>AT&amp;T 93% 11:48 AM</p> <p>Find My Device</p> <p>Locked</p> <p><b>LG Phoenix 3</b></p> <p>Last seen 3 mins ago</p> <p>24% walkcri...</p> <p>PLAY SOUND &gt;</p> <p>LOCK &gt;</p> <p>ERASE &gt;</p>

**Exhibit D - Claim Chart for U.S. Patent No. 9,467,838 Against HTC**

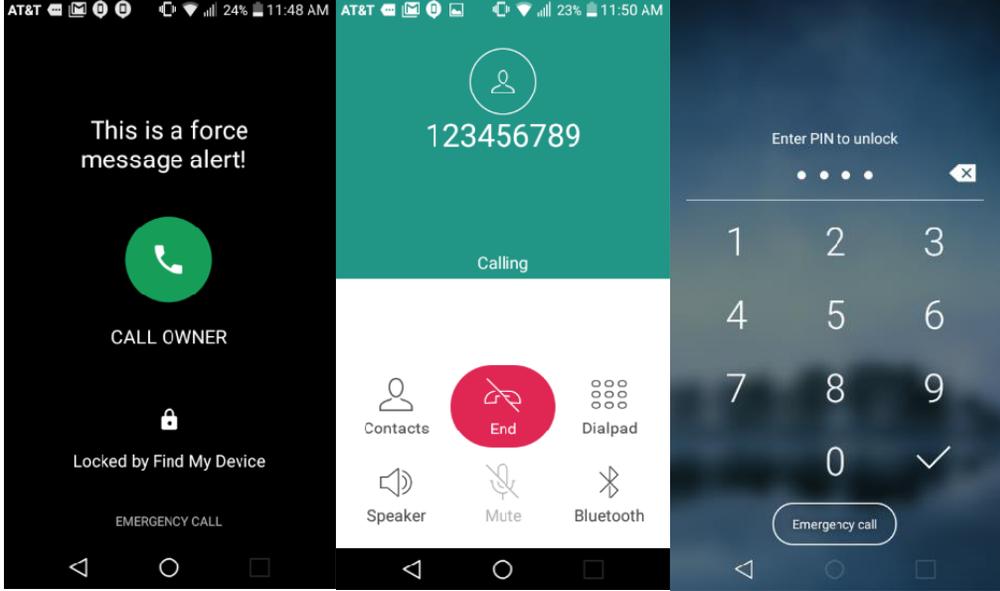
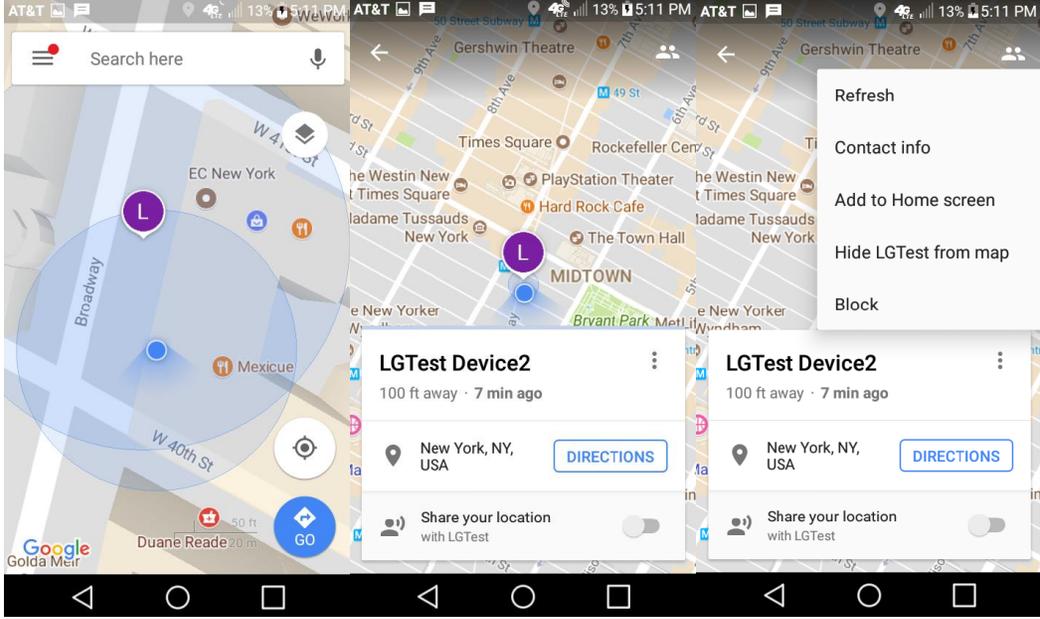
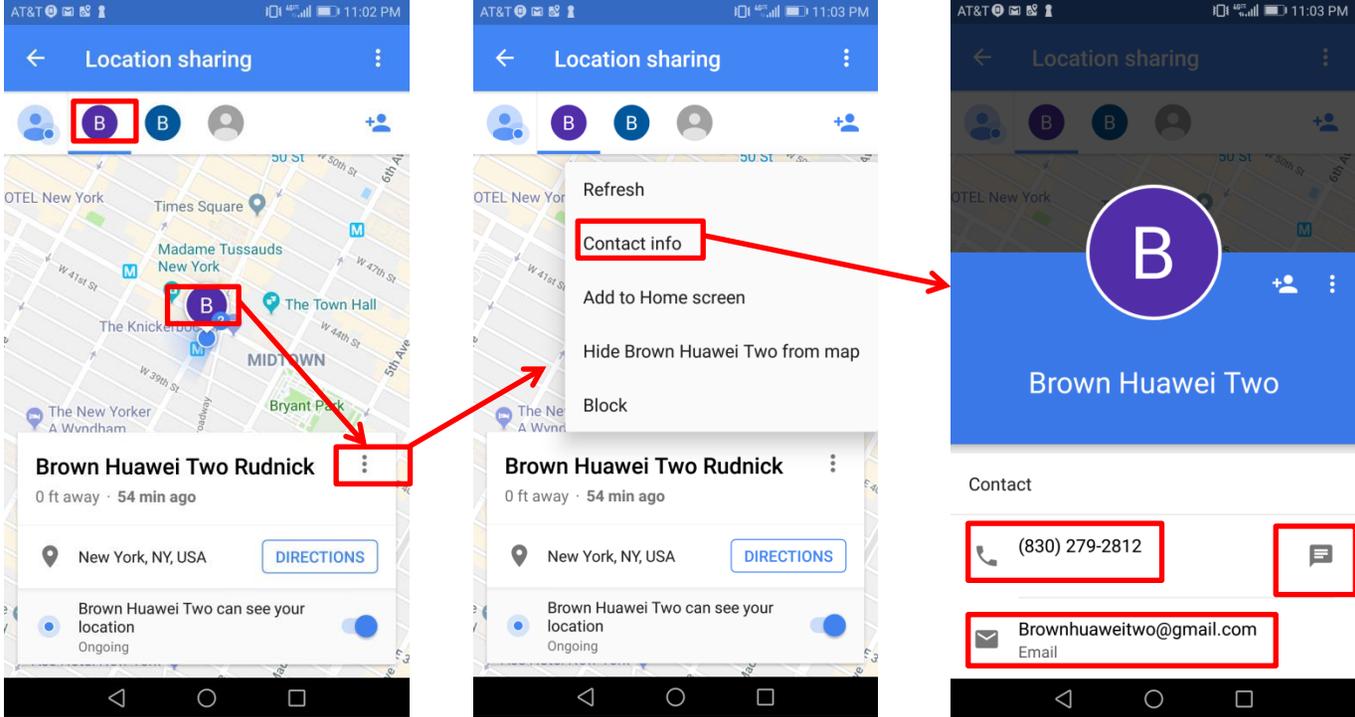
US9467838	Exemplary Supporting Evidence Regarding Accused Products
	 <p>The image displays three sequential screenshots of an Android mobile phone interface. The first screenshot, taken at 11:48 AM with 24% battery, shows a black screen with a green phone icon and the text "This is a force message alert!" and "CALL OWNER". Below this, it says "Locked by Find My Device" and "EMERGENCY CALL". The second screenshot, taken at 11:50 AM with 23% battery, shows a teal calling screen for the number 123456789. It features a red "End" button and icons for "Contacts", "Dialpad", "Speaker", "Mute", and "Bluetooth". The third screenshot shows a dark blue PIN unlock screen with the text "Enter PIN to unlock" and a numeric keypad with an "Emergency call" button at the bottom.</p> <p><b><u>Exemplary Find My Device Screenshots:</u></b></p>

Exhibit D - Claim Chart for U.S. Patent No. 9,467,838 Against HTC

US9467838	Exemplary Supporting Evidence Regarding Accused Products
	 <p>Upon information and belief, selection of contact info results in the presentation of actions to be chosen. See, e.g.,</p>

**Exhibit D - Claim Chart for U.S. Patent No. 9,467,838 Against HTC**

US9467838	Exemplary Supporting Evidence Regarding Accused Products
	 <p><b>Exemplary Source Code:</b></p> <p>The above functionality is performed at least in part by the following publicly available source code and/or source code that invokes or is invoked by the following source code (or a substantially similar copy compiled and loaded onto the Accused Products by HTC). AGIS reserves the right to supplement these contentions with additional source code as discovery progresses and as additional source code is made available. AGIS reserves the right to supplement these contentions with additional source code as discovery progresses and as additional source code is made available.</p>

## Exhibit D - Claim Chart for U.S. Patent No. 9,467,838 Against HTC

US9467838	Exemplary Supporting Evidence Regarding Accused Products
	<pre>44  * Class that sends chat message via SMS. 45  * 46  * The interface emulates a blocking sending similar to making an HTTP request. 47  * It calls the SmsManager to send a (potentially multipart) message and waits 48  * on the sent status on each part. The waiting has a timeout so it won't wait 49  * forever. Once the sent status of all parts received, the call returns. 50  * A successful sending requires success status for all parts. Otherwise, we 51  * pick the highest level of failure as the error for the whole message, which 52  * is used to determine if we need to retry the sending. 53  */ 54  public class SmsSender { 55      private static final String TAG = LogUtil.BUGLE_TAG; 56 57      public static final String EXTRA_PART_ID = "part_id"; 58 59      /* 60       * A map for pending sms messages. The key is the random request UUID. 61       */ 62      private static ConcurrentHashMap&lt;Uri, SendResult&gt; sPendingMessageMap = 63          new ConcurrentHashMap&lt;Uri, SendResult&gt;(); 64 65      private static final Random RANDOM = new Random(); 66 67      // Whether we should send multipart SMS as separate messages 68      private static Boolean sSendMultipartSmsAsSeparateMessages = null; 69  }</pre> <p><a href="https://android.googlesource.com/platform/packages/apps/Messaging+/nougat-mr1-release/src/com/android/messaging/sms/SmsSender.java">https://android.googlesource.com/platform/packages/apps/Messaging+/nougat-mr1-release/src/com/android/messaging/sms/SmsSender.java</a></p>

## Exhibit D - Claim Chart for U.S. Patent No. 9,467,838 Against HTC

US9467838	Exemplary Supporting Evidence Regarding Accused Products
	<pre> 253 // Actually sending the message using SmsManager 254 private static void sendInternal(final Context context, final int subId, String dest, 255     final ArrayList&lt;String&gt; messages, final String serviceCenter, 256     final boolean requireDeliveryReport, final Uri messageUri) throws SmsException { 257     Assert.notNull(context); 258     final SmsManager smsManager = PhoneUtils.get(subId).getSmsManager(); 259     final int messageCount = messages.size(); 260     final ArrayList&lt;PendingIntent&gt; deliveryIntents = new ArrayList&lt;PendingIntent&gt;(messageCount); 261     final ArrayList&lt;PendingIntent&gt; sentIntents = new ArrayList&lt;PendingIntent&gt;(messageCount); 262     for (int i = 0; i &lt; messageCount; i++) { 263         // Make pending intents different for each message part 264         final int partId = (messageCount &lt;= 1 ? 0 : i + 1); 265         if (requireDeliveryReport &amp;&amp; (i == (messageCount - 1))) { 266             // TODO we only care about the delivery status of the last part 267             // Shall we have better tracking of delivery status of all parts? 268             deliveryIntents.add(PendingIntent.getBroadcast( 269                 context, 270                 partId, 271                 getSendStatusIntent(context, SendStatusReceiver.MESSAGE_DELIVERED_ACTION, 272                     messageUri, partId, subId), 273                 0/*flag*/)); 274         } else { 275             deliveryIntents.add(null); 276         } 277         sentIntents.add(PendingIntent.getBroadcast( 278             context, 279             partId, 280             getSendStatusIntent(context, SendStatusReceiver.MESSAGE_SENT_ACTION, 281                 messageUri, partId, subId), 282             0/*flag*/)); 283     } 284     if (sSendMultipartSmsAsSeparateMessages == null) { 285         sSendMultipartSmsAsSeparateMessages = MmsConfig.get(subId) 286             .getSendMultipartSmsAsSeparateMessages(); 287     } </pre> <p><a href="https://android.googlesource.com/platform/packages/apps/Messaging/+nougat-mr1-release/src/com/android/messaging/sms/SmsSender.java">https://android.googlesource.com/platform/packages/apps/Messaging/+nougat-mr1-release/src/com/android/messaging/sms/SmsSender.java</a></p>



## Exhibit D - Claim Chart for U.S. Patent No. 9,467,838 Against HTC

US9467838	Exemplary Supporting Evidence Regarding Accused Products
	<pre> 288     try { 289         if (sSendMultipartSmsAsSeparateMessages) { 290             // If multipart sms is not supported, send them as separate messages 291             for (int i = 0; i &lt; messageCount; i++) { 292                 smsManager.sendTextMessage(dest, 293                     serviceCenter, 294                     messages.get(i), 295                     sentIntents.get(i), 296                     deliveryIntents.get(i)); 297             } 298         } else { 299             smsManager.sendMultipartTextMessage( 300                 dest, serviceCenter, messages, sentIntents, deliveryIntents); 301         } 302     } catch (final Exception e) { 303         throw new SmsException("SmsSender: caught exception in sending " + e); 304     } 305 } </pre> <p><a href="https://android.googlesource.com/platform/packages/apps/Messaging+/nougat-mr1-release/src/com/android/messaging/sms/SmsSender.java">https://android.googlesource.com/platform/packages/apps/Messaging+/nougat-mr1-release/src/com/android/messaging/sms/SmsSender.java</a></p> <pre> 56  * Class that receives incoming SMS messages through android.provider.Telephony.SMS_RECEIVED 57  * 58  * This class serves two purposes: 59  * - Process phone verification SMS messages 60  * - Handle SMS messages when the user has enabled us to be the default SMS app (Pre-KLP) 61  */ 62  public final class SmsReceiver extends BroadcastReceiver { 63      private static final String TAG = LogUtil.BUGLE_TAG; 64 65      private static ArrayList&lt;Pattern&gt; sIgnoreSmsPatterns; 66 </pre> <p><a href="https://android.googlesource.com/platform/packages/apps/Messaging+/nougat-mr1-release/src/com/android/messaging/receiver/SmsReceiver.java">https://android.googlesource.com/platform/packages/apps/Messaging+/nougat-mr1-release/src/com/android/messaging/receiver/SmsReceiver.java</a></p>

## Exhibit D - Claim Chart for U.S. Patent No. 9,467,838 Against HTC

US9467838	Exemplary Supporting Evidence Regarding Accused Products
	<pre>200     public static void deliverSmsMessages(final Context context, final int subId, 201         final int errorCode, final android.telephony.SmsMessage[] messages) { 202         final ContentValues messageValues = 203             MmsUtils.parseReceivedSmsMessage(context, messages, errorCode); 204 205         LogUtil.v(TAG, "SmsReceiver.deliverSmsMessages"); 206 207         final long nowInMillis = System.currentTimeMillis(); 208         final long receivedTimestampMs = MmsUtils.getMessageDate(messages[0], nowInMillis); 209 210         messageValues.put(Sms.Inbox.DATE, receivedTimestampMs); 211         // Default to unread and unseen for us but ReceiveSmsMessageAction will override 212         // seen for the telephony db. 213         messageValues.put(Sms.Inbox.READ, 0); 214         messageValues.put(Sms.Inbox.SEEN, 0); 215         if (OsUtil.isAtLeastL_MR1()) { 216             messageValues.put(Sms.SUBSCRIPTION_ID, subId); 217         } 218 219         if (messages[0].getMessageClass() == android.telephony.SmsMessage.MessageClass.CLASS_0    220             DebugUtils.debugClassZeroSmsEnabled()) { 221             Factory.get().getUIIntents().launchClassZeroActivity(context, messageValues); 222         } else { 223             final ReceiveSmsMessageAction action = new ReceiveSmsMessageAction(messageValues); 224             action.start(); 225         } 226     }</pre> <p><a href="https://android.googlesource.com/platform/packages/apps/Messaging/+nougat-mr1-release/src/com/android/messaging/receiver/SmsReceiver.java">https://android.googlesource.com/platform/packages/apps/Messaging/+nougat-mr1-release/src/com/android/messaging/receiver/SmsReceiver.java</a></p>

## Exhibit D - Claim Chart for U.S. Patent No. 9,467,838 Against HTC

US9467838	Exemplary Supporting Evidence Regarding Accused Products
	<pre>228     @Override 229     public void onReceive(final Context context, final Intent intent) { 230         LogUtil.v(TAG, "SmsReceiver.onReceive " + intent); 231         // On KLP+ we only take delivery of SMS messages in SmsDeliverReceiver. 232         if (PhoneUtils.getDefault().isSmsEnabled()) { 233             final String action = intent.getAction(); 234             if (OsUtil.isSecondaryUser() &amp;&amp; 235                 (Telephony.Sms.Intents.SMS_RECEIVED_ACTION.equals(action)    236                  // TODO: update this with the actual constant from Telephony 237                  "android.provider.Telephony.MMS_DOWNLOADED".equals(action))) { 238                 postNewMessageSecondaryUserNotification(); 239             } else if (!OsUtil.isAtLeastKLP()) { 240                 deliverSmsIntent(context, intent); 241             } 242         } 243     }</pre> <p><a href="https://android.googlesource.com/platform/packages/apps/Messaging/+/nougat-mr1-release/src/com/android/messaging/receiver/SmsReceiver.java">https://android.googlesource.com/platform/packages/apps/Messaging/+/nougat-mr1-release/src/com/android/messaging/receiver/SmsReceiver.java</a></p>

## Exhibit D - Claim Chart for U.S. Patent No. 9,467,838 Against HTC

US9467838	Exemplary Supporting Evidence Regarding Accused Products
	<pre>52  * Class that sends chat message via MMS. 53  * 54  * The interface emulates a blocking send similar to making an HTTP request. 55  */ 56  public class MmsSender { 57      private static final String TAG = LogUtil.BUGLE_TAG; 58 59      /** 60       * Send an MMS message. 61       * 62       * @param context Context 63       * @param messageUri The unique URI of the message for identifying it during sending 64       * @param sendReq The SendReq PDU of the message 65       * @throws MmsFailureException 66       */ 67      public static void sendMms(final Context context, final int subId, final Uri messageUri, 68                               final SendReq sendReq, final Bundle sentIntentExtras) throws MmsFailureException { 69          sendMms(context, 70                subId, 71                messageUri, 72                null /* locationUri */, 73                sendReq, 74                true /* responseImportant */, 75                sentIntentExtras); 76      }</pre> <p><a href="https://android.googlesource.com/platform/packages/apps/Messaging/+/nougat-mr1-release/src/com/android/messaging/sms/MmsSender.java">https://android.googlesource.com/platform/packages/apps/Messaging/+/nougat-mr1-release/src/com/android/messaging/sms/MmsSender.java</a></p>

## Exhibit D - Claim Chart for U.S. Patent No. 9,467,838 Against HTC

US9467838	Exemplary Supporting Evidence Regarding Accused Products
	<pre> 240      * Download an MMS message. 241      * 242      * @param context Context 243      * @param contentLocation The url of the MMS message 244      * @throws MmsFailureException 245      * @throws InvalidHeaderValueException 246      */ 247      public static void downloadMms(final Context context, final int subId, 248          final String contentLocation, Bundle extras) throws MmsFailureException, 249          InvalidHeaderValueException { 250          final Uri requestUri = Uri.parse(contentLocation); 251          final Uri contentUri = MmsFileProvider.buildRawMmsUri(); 252 253          final Intent downloadedIntent = new Intent(SendStatusReceiver.MMS_DOWNLOADED_ACTION, 254              requestUri, 255              context, 256              SendStatusReceiver.class); 257          downloadedIntent.putExtra(SendMessageAction.EXTRA_CONTENT_URI, contentUri); 258          if (extras != null) { 259              downloadedIntent.putExtras(extras); 260          } 261          final PendingIntent downloadedPendingIntent = PendingIntent.getBroadcast( 262              context, 263              0 /*request code*/, 264              downloadedIntent, 265              PendingIntent.FLAG_UPDATE_CURRENT); 266 267          MmsManager.downloadMultimediaMessage(subId, context, contentLocation, contentUri, 268              downloadedPendingIntent); 269      } </pre> <p><a href="https://android.googlesource.com/platform/packages/apps/Messaging/+nougat-mr1-release/src/com/android/messaging/sms/MmsSender.java">https://android.googlesource.com/platform/packages/apps/Messaging/+nougat-mr1-release/src/com/android/messaging/sms/MmsSender.java</a></p>

## Exhibit D - Claim Chart for U.S. Patent No. 9,467,838 Against HTC

US9467838	Exemplary Supporting Evidence Regarding Accused Products
	<pre> 97      * Execute an MMS HTTP request, either a POST (sending) or a GET (downloading) 98      * 99      * @param urlString The request URL, for sending it is usually the MMSC, and for downloading 100     *           it is the message URL 101     * @param pdu For POST (sending) only, the PDU to send 102     * @param method HTTP method, POST for sending and GET for downloading 103     * @param isProxySet Is there a proxy for the MMSC 104     * @param proxyHost The proxy host 105     * @param proxyPort The proxy port 106     * @param mmsConfig The MMS config to use 107     * @param userAgent The user agent header value 108     * @param uaProfUrl The UA Prof URL header value 109     * @return The HTTP response body 110     * @throws MmsHttpException For any failures 111     */ 112     public byte[] execute(String urlString, byte[] pdu, String method, boolean isProxySet, 113                          String proxyHost, int proxyPort, Bundle mmsConfig, String userAgent, String uaProfUrl) 114                          throws MmsHttpException { 115         Log.d(MmsService.TAG, "HTTP: " + method + " " + Utils.redactUrlForNonVerbose(urlString) 116             + (isProxySet ? (" proxy=" + proxyHost + ":" + proxyPort) : "") 117             + ", PDU size=" + (pdu != null ? pdu.length : 0)); 118         checkMethod(method); 119         HttpURLConnection connection = null; 120         try { 121             Proxy proxy = Proxy.NO_PROXY; 122             if (isProxySet) { 123                 proxy = new Proxy(Proxy.Type.HTTP, new InetSocketAddress(proxyHost, proxyPort)); 124             } 125             final URL url = new URL(urlString); 126             // Now get the connection 127             connection = (HttpURLConnection) url.openConnection(proxy); 128             connection.setDoInput(true); 129             connection.setConnectTimeout( 130                 mmsConfig.getInt(CarrierConfigValuesLoader.CONFIG_HTTP_SOCKET_TIMEOUT, 131                                 CarrierConfigValuesLoader.CONFIG_HTTP_SOCKET_TIMEOUT_DEFAULT)); </pre> <p><a href="https://android.googlesource.com/platform/packages/apps/Messaging/+/nougat-mr1-release/src/android/support/v7/mms/MmsHttpClient.java">https://android.googlesource.com/platform/packages/apps/Messaging/+/nougat-mr1-release/src/android/support/v7/mms/MmsHttpClient.java</a></p>

## Exhibit D - Claim Chart for U.S. Patent No. 9,467,838 Against HTC

US9467838	Exemplary Supporting Evidence Regarding Accused Products
	<pre> 132 // ----- COMMON HEADERS ----- 133 // Header: Accept 134 connection.setRequestProperty(HEADER_ACCEPT, HEADER_VALUE_ACCEPT); 135 // Header: Accept-Language 136 connection.setRequestProperty( 137     HEADER_ACCEPT_LANGUAGE, getCurrentAcceptLanguage(Locale.getDefault())); 138 // Header: User-Agent 139 Log.i(MmsService.TAG, "HTTP: User-Agent=" + userAgent); 140 connection.setRequestProperty(HEADER_USER_AGENT, userAgent); 141 // Header: x-wap-profile 142 final String uaProfUrlTagName = mmsConfig.getString( 143     CarrierConfigValuesLoader.CONFIG_UA_PROF_TAG_NAME, 144     CarrierConfigValuesLoader.CONFIG_UA_PROF_TAG_NAME_DEFAULT); 145 if (uaProfUrl != null) { 146     Log.i(MmsService.TAG, "HTTP: UaProfUrl=" + uaProfUrl); 147     connection.setRequestProperty(uaProfUrlTagName, uaProfUrl); 148 } 149 // Add extra headers specified by mms_config.xml's httpparams 150 addExtraHeaders(connection, mmsConfig); 151 // Different stuff for GET and POST 152 if (METHOD_POST.equals(method)) { 153     if (pdu == null    pdu.length &lt; 1) { 154         Log.e(MmsService.TAG, "HTTP: empty pdu"); 155         throw new MmsHttpException(0/*statusCode*/, "Sending empty PDU"); 156     } 157     connection.setDoOutput(true); 158     connection.setRequestMethod(METHOD_POST); 159     if (mmsConfig.getBoolean( 160         CarrierConfigValuesLoader.CONFIG_SUPPORT_HTTP_CHARSET_HEADER, 161         CarrierConfigValuesLoader.CONFIG_SUPPORT_HTTP_CHARSET_HEADER_DEFAULT)) { 162         connection.setRequestProperty(HEADER_CONTENT_TYPE, 163             HEADER_VALUE_CONTENT_TYPE_WITH_CHARSET); 164     } else { 165         connection.setRequestProperty(HEADER_CONTENT_TYPE, 166             HEADER_VALUE_CONTENT_TYPE_WITHOUT_CHARSET); </pre>
	<p><a href="https://android.googlesource.com/platform/packages/apps/Messaging+/nougat-mr1-release/src/android/support/v7/mms/MmsHttpClient.java">https://android.googlesource.com/platform/packages/apps/Messaging+/nougat-mr1-release/src/android/support/v7/mms/MmsHttpClient.java</a></p>

## Exhibit D - Claim Chart for U.S. Patent No. 9,467,838 Against HTC

US9467838	Exemplary Supporting Evidence Regarding Accused Products
	<pre>167         } 168         if (Log.isLoggable(MmsService.TAG, Log.VERBOSE)) { 169             logHttpHeaders(connection.getRequestProperties()); 170         } 171         connection.setFixedLengthStreamingMode(pdu.length); 172         // Sending request body 173         final OutputStream out = 174             new BufferedOutputStream(connection.getOutputStream()); 175         out.write(pdu); 176         out.flush(); 177         out.close(); 178     } else if (METHOD_GET.equals(method)) { 179         if (Log.isLoggable(MmsService.TAG, Log.VERBOSE)) { 180             logHttpHeaders(connection.getRequestProperties()); 181         } 182         connection.setRequestMethod(METHOD_GET); 183     } 184     // Get response 185     final int responseCode = connection.getResponseCode(); 186     final String responseMessage = connection.getResponseMessage(); 187     Log.d(MmsService.TAG, "HTTP: " + responseCode + " " + responseMessage); 188     if (Log.isLoggable(MmsService.TAG, Log.VERBOSE)) { 189         logHttpHeaders(connection.getHeaderFields()); 190     } 191     if (responseCode / 100 != 2) { 192         throw new MmsHttpException(responseCode, responseMessage); 193     } 194     final InputStream in = new BufferedInputStream(connection.getInputStream()); 195     final ByteArrayOutputStream byteOut = new ByteArrayOutputStream(); 196     final byte[] buf = new byte[4096]; 197     int count = 0; 198     while ((count = in.read(buf)) &gt; 0) { 199         byteOut.write(buf, 0, count); 200     } 201     in.close(); 202     final byte[] responseBody = byteOut.toByteArray(); 203     Log.d(MmsService.TAG, "HTTP: response size=" 204         + (responseBody != null ? responseBody.length : 0)); 205     return responseBody;</pre> <p><a href="https://android.googlesource.com/platform/packages/apps/Messaging/+nougat-mr1-release/src/android/support/v7/mms/MmsHttpClient.java">https://android.googlesource.com/platform/packages/apps/Messaging/+nougat-mr1-release/src/android/support/v7/mms/MmsHttpClient.java</a></p>



## Exhibit D - Claim Chart for U.S. Patent No. 9,467,838 Against HTC

US9467838	Exemplary Supporting Evidence Regarding Accused Products
	<pre>38  * Request to send an MMS 39  */ 40  class SendRequest extends MmsRequest { 41      // Max send response PDU size in bytes (exceeding this may cause problem with 42      // system intent delivery). 43      private static final int MAX_SEND_RESPONSE_SIZE = 1000 * 1024; 44 45      private byte[] mPduData; 46 47      SendRequest(final String locationUri, final Uri pduUri, final PendingIntent sentIntent) { 48          super(locationUri, pduUri, sentIntent); 49      } 50 51      @Override 52      protected boolean loadRequest(final Context context, final Bundle mmsConfig) { 53          mPduData = readPduFromContentUri( 54              context, 55              mPduUri, 56              mmsConfig.getInt( 57                  CarrierConfigValuesLoader.CONFIG_MAX_MESSAGE_SIZE, 58                  CarrierConfigValuesLoader.CONFIG_MAX_MESSAGE_SIZE_DEFAULT)); 59          return (mPduData != null); 60      } 61 62      @Override 63      protected boolean transferResponse(final Context context, final Intent fillIn, 64          final byte[] response) { 65          // SendConf pdus are always small and can be included in the intent 66          if (response != null &amp;&amp; fillIn != null) { 67              if (response.length &gt; MAX_SEND_RESPONSE_SIZE) { 68                  // If the response PDU is too large, it won't be able to fit in 69                  // the PendingIntent to be transferred via system IPC. 70                  return false; 71              } 72              fillIn.putExtra(SmsManager.EXTRA_MMS_DATA, response); 73          } 74          return true; 75      } }</pre>

## Exhibit D - Claim Chart for U.S. Patent No. 9,467,838 Against HTC

US9467838	Exemplary Supporting Evidence Regarding Accused Products
	<p data-bbox="537 235 1619 302"><a href="https://android.googlesource.com/platform/packages/apps/Messaging+/nougat-mr1-release/src/android/support/v7/mms/SendRequest.java">https://android.googlesource.com/platform/packages/apps/Messaging+/nougat-mr1-release/src/android/support/v7/mms/SendRequest.java</a></p> <pre data-bbox="562 358 1003 391">public static LocationRequest create ()</pre> <p data-bbox="548 423 1052 448">Create a location request with default parameters.</p> <p data-bbox="548 480 1661 540">Default parameters are for a block accuracy, slowly updated location. It can then be adjusted as required by the applications before passing to the <code>FusedLocationProviderApi</code>.</p> <p data-bbox="573 565 657 586"><b>Returns</b></p> <ul data-bbox="579 610 835 634" style="list-style-type: none"><li>• a new location request</li></ul> <p data-bbox="537 651 1818 680"><a href="https://developers.google.com/android/reference/com/google/android/gms/location/LocationRequest">https://developers.google.com/android/reference/com/google/android/gms/location/LocationRequest</a></p>

**Exhibit D - Claim Chart for U.S. Patent No. 9,467,838 Against HTC**

US9467838	Exemplary Supporting Evidence Regarding Accused Products
	<p data-bbox="552 248 1766 285"><b>public static final int PRIORITY_BALANCED_POWER_ACCURACY</b></p> <p data-bbox="552 313 1199 341">Used with <code>setPriority(int)</code> to request "block" level accuracy.</p> <p data-bbox="552 370 1667 430">Block level accuracy is considered to be about 100 meter accuracy. Using a coarse accuracy such as this often consumes less power.</p> <p data-bbox="552 456 753 483">Constant Value: 102</p> <p data-bbox="552 537 1766 574"><b>public static final int PRIORITY_HIGH_ACCURACY</b></p> <p data-bbox="552 602 1360 630">Used with <code>setPriority(int)</code> to request the most accurate locations available.</p> <p data-bbox="552 659 984 686">This will return the finest location available.</p> <p data-bbox="552 712 753 740">Constant Value: 100</p> <p data-bbox="552 794 1766 831"><b>public static final int PRIORITY_LOW_POWER</b></p> <p data-bbox="552 859 1178 886">Used with <code>setPriority(int)</code> to request "city" level accuracy.</p> <p data-bbox="552 915 1759 976">City level accuracy is considered to be about 10km accuracy. Using a coarse accuracy such as this often consumes less power.</p> <p data-bbox="552 1002 753 1029">Constant Value: 104</p> <p data-bbox="537 1045 1818 1073"><a href="https://developers.google.com/android/reference/com/google/android/gms/location/LocationRequest">https://developers.google.com/android/reference/com/google/android/gms/location/LocationRequest</a></p>

## Exhibit D - Claim Chart for U.S. Patent No. 9,467,838 Against HTC

US9467838	Exemplary Supporting Evidence Regarding Accused Products
	<p data-bbox="556 248 1774 285"><code>public Task&lt;Location&gt; getLastLocation ()</code></p> <p data-bbox="556 313 1129 337">Returns the best most recent location currently available.</p> <p data-bbox="556 370 1719 428">If a location is not available, which should happen very rarely, null will be returned. The best accuracy available while respecting the location permissions will be returned.</p> <p data-bbox="556 461 1759 519">This method provides a simplified way to get location. It is particularly well suited for applications that do not require an accurate location and that do not want to maintain extra logic for location updates.</p> <p data-bbox="556 578 1774 615"><code>public Task&lt;LocationAvailability&gt; getLocationAvailability ()</code></p> <p data-bbox="556 643 1713 701">Returns the availability of location data. When <code>isLocationAvailable()</code> returns true, then the location returned by <code>getLastLocation()</code> will be reasonably up to date within the hints specified by the active <code>LocationRequest</code>s.</p> <p data-bbox="556 734 1495 758">If the client isn't connected to Google Play services and the request times out, null is returned.</p> <p data-bbox="556 790 1696 849">Note it's always possible for <code>getLastLocation()</code> to return null even when this method returns true (e.g. location settings were disabled between calls).</p> <p data-bbox="535 865 1902 928"><a href="https://developers.google.com/android/reference/com/google/android/gms/location/FusedLocationProviderClient">https://developers.google.com/android/reference/com/google/android/gms/location/FusedLocationProviderClient</a></p>

**Exhibit D - Claim Chart for U.S. Patent No. 9,467,838 Against HTC**

US9467838	Exemplary Supporting Evidence Regarding Accused Products						
	<p data-bbox="546 240 1774 326"> <code>public Task&lt;Void&gt; requestLocationUpdates (LocationRequest request, LocationCallback callback, Looper looper)</code> </p> <p data-bbox="546 354 1291 378">Requests location updates with a callback on the specified Looper thread.</p> <p data-bbox="546 410 1711 472">This method is suited for the foreground use cases. For background use cases, the <code>PendingIntent</code> version of the method is recommended, see <code>requestLocationUpdates(LocationRequest, PendingIntent)</code>.</p> <p data-bbox="546 505 1396 529">Any previous LocationRequests registered on this LocationListener will be replaced.</p> <p data-bbox="546 561 1711 656">This call will keep the Google Play services connection active, so make sure to call <code>removeLocationUpdates(LocationCallback)</code> when you no longer need it, otherwise you lose the benefits of the automatic connection management.</p> <p data-bbox="546 688 1774 712">Callbacks for <code>LocationCallback</code> will be made on the specified thread, which must already be a prepared looper thread.</p> <p data-bbox="546 737 682 761"><b>Parameters</b></p> <table border="1" data-bbox="546 789 1774 1008"> <tbody> <tr> <td data-bbox="546 789 655 857"><b>request</b></td> <td data-bbox="655 789 1774 857">The location request for the updates.</td> </tr> <tr> <td data-bbox="546 857 655 925"><b>callback</b></td> <td data-bbox="655 857 1774 925">The callback for the location updates.</td> </tr> <tr> <td data-bbox="546 925 655 1008"><b>looper</b></td> <td data-bbox="655 925 1774 1008">The Looper object whose message queue will be used to implement the callback mechanism, or null to make callbacks on the calling thread.</td> </tr> </tbody> </table> <p data-bbox="546 1024 1900 1089"> <a href="https://developers.google.com/android/reference/com/google/android/gms/location/FusedLocationProviderClient">https://developers.google.com/android/reference/com/google/android/gms/location/FusedLocationProviderClient</a> </p>	<b>request</b>	The location request for the updates.	<b>callback</b>	The callback for the location updates.	<b>looper</b>	The Looper object whose message queue will be used to implement the callback mechanism, or null to make callbacks on the calling thread.
<b>request</b>	The location request for the updates.						
<b>callback</b>	The callback for the location updates.						
<b>looper</b>	The Looper object whose message queue will be used to implement the callback mechanism, or null to make callbacks on the calling thread.						

## Exhibit D - Claim Chart for U.S. Patent No. 9,467,838 Against HTC

US9467838	Exemplary Supporting Evidence Regarding Accused Products				
	<p data-bbox="556 240 1766 326">public Task&lt;Void&gt; requestLocationUpdates (LocationRequest request, PendingIntent callbackIntent)</p> <p data-bbox="556 354 1293 378">Requests location updates with a callback on the specified PendingIntent.</p> <p data-bbox="556 410 1755 537">This method is suited for the background use cases, more specifically for receiving location updates, even when the app has been killed by the system. In order to do so, use a PendingIntent for a started service. For foreground use cases, the LocationCallback version of the method is recommended, see requestLocationUpdates(LocationRequest, LocationCallback, Looper).</p> <p data-bbox="556 570 1749 630">Any previously registered requests that have the same PendingIntent (as defined by equals(Object)) will be replaced by this request.</p> <p data-bbox="556 662 1755 755">Both LocationResult and LocationAvailability are sent to the given PendingIntent. You can extract data from an Intent using hasResult(Intent), extractResult(Intent), hasLocationAvailability(Intent), and extractLocationAvailability(Intent).</p> <p data-bbox="569 781 695 805"><b>Parameters</b></p> <table border="1" data-bbox="556 833 1766 971"> <tbody> <tr> <td data-bbox="556 833 863 902">request</td> <td data-bbox="863 833 1766 902">The location request for the updates.</td> </tr> <tr> <td data-bbox="556 902 863 971">callbackIntent</td> <td data-bbox="863 902 1766 971">A pending intent to be sent for each location update.</td> </tr> </tbody> </table> <p data-bbox="569 997 653 1021"><b>Returns</b></p> <ul data-bbox="577 1044 1381 1068" style="list-style-type: none"> <li>• a Task for the call, check isSuccessful() to determine if it was successful.</li> </ul> <p data-bbox="537 1081 1902 1144"><a href="https://developers.google.com/android/reference/com/google/android/gms/location/FusedLocationProviderClient">https://developers.google.com/android/reference/com/google/android/gms/location/FusedLocationProviderClient</a></p>	request	The location request for the updates.	callbackIntent	A pending intent to be sent for each location update.
request	The location request for the updates.				
callbackIntent	A pending intent to be sent for each location update.				

## Exhibit D - Claim Chart for U.S. Patent No. 9,467,838 Against HTC

US9467838	Exemplary Supporting Evidence Regarding Accused Products						
	<p data-bbox="558 245 1761 277"><code>public void onLocationAvailability (LocationAvailability locationAvailability)</code></p> <p data-bbox="548 310 1192 334">Called when there is a change in the availability of location data.</p> <p data-bbox="548 367 1761 561">When <code>isLocationAvailable()</code> returns <code>false</code> you can assume that location will not be returned in <code>onLocationResult(LocationResult)</code> until something changes in the device's settings or environment. Even when <code>isLocationAvailable()</code> returns <code>true</code> the <code>onLocationResult(LocationResult)</code> may not always be called regularly, however the device location is known and both the most recently delivered location and <code>getLastLocation(GoogleApiClient)</code> will be reasonably up to date given the hints specified by the active <code>LocationRequest</code>s.</p> <p data-bbox="569 586 695 610"><b>Parameters</b></p> <table border="1" data-bbox="548 643 1761 708"> <tr> <td data-bbox="558 651 984 699"><code>locationAvailability</code></td> <td data-bbox="995 651 1761 699">The current status of location availability.</td> </tr> </table> <p data-bbox="558 756 1761 789"><code>public void onLocationResult (LocationResult result)</code></p> <p data-bbox="548 821 1077 846">Called when device location information is available.</p> <p data-bbox="548 878 1682 943">The most recent location returned by <code>getLastLocation()</code> is not guaranteed to be immediately fresh, but will be reasonably up to date given the hints specified by the active <code>LocationRequest</code>s.</p> <p data-bbox="569 967 695 992"><b>Parameters</b></p> <table border="1" data-bbox="548 1024 1761 1089"> <tr> <td data-bbox="558 1032 789 1081"><code>result</code></td> <td data-bbox="800 1032 1761 1089">The latest location result available.</td> </tr> </table> <p data-bbox="537 1105 1829 1130"><a href="https://developers.google.com/android/reference/com/google/android/gms/location/LocationCallback">https://developers.google.com/android/reference/com/google/android/gms/location/LocationCallback</a></p> <p data-bbox="558 1146 1761 1179"><code>public abstract void onLocationChanged (Location location)</code></p> <p data-bbox="548 1211 936 1235">Called when the location has changed.</p> <p data-bbox="569 1260 695 1284"><b>Parameters</b></p> <table border="1" data-bbox="548 1317 1761 1382"> <tr> <td data-bbox="558 1325 947 1373"><code>location</code></td> <td data-bbox="957 1325 1761 1382">The updated location.</td> </tr> </table>	<code>locationAvailability</code>	The current status of location availability.	<code>result</code>	The latest location result available.	<code>location</code>	The updated location.
<code>locationAvailability</code>	The current status of location availability.						
<code>result</code>	The latest location result available.						
<code>location</code>	The updated location.						

## Exhibit D - Claim Chart for U.S. Patent No. 9,467,838 Against HTC

US9467838	Exemplary Supporting Evidence Regarding Accused Products
	<p data-bbox="541 237 1822 264"><a href="https://developers.google.com/android/reference/com/google/android/gms/location/LocationListener">https://developers.google.com/android/reference/com/google/android/gms/location/LocationListener</a></p> <p data-bbox="541 318 835 345">Public Constructors</p> <hr data-bbox="541 358 1766 362"/> <p data-bbox="552 410 947 438">public <b>MapView</b> (<b>Context</b> context)</p> <p data-bbox="552 508 1150 535">public <b>MapView</b> (<b>Context</b> context, <b>AttributeSet</b> attrs)</p> <p data-bbox="552 605 1289 633">public <b>MapView</b> (<b>Context</b> context, <b>AttributeSet</b> attrs, int defStyle)</p> <p data-bbox="552 695 1268 722">public <b>MapView</b> (<b>Context</b> context, <b>GoogleMapOptions</b> options)</p> <p data-bbox="541 751 1696 779"><a href="https://developers.google.com/android/reference/com/google/android/gms/maps/MapView">https://developers.google.com/android/reference/com/google/android/gms/maps/MapView</a></p>



**Exhibit D - Claim Chart for U.S. Patent No. 9,467,838 Against HTC**

US9467838	Exemplary Supporting Evidence Regarding Accused Products		
	<p><code>public void <b>getMapAsync</b> (<b>OnMapReadyCallback</b> callback)</code></p> <p>Returns a non-null instance of the <code>GoogleMap</code>, ready to be used.</p> <p>Note that:</p> <ul style="list-style-type: none"> <li>• This method must be called from the main thread.</li> <li>• The callback will be executed in the main thread.</li> <li>• In the case where Google Play services is not installed on the user's device, the callback will not be triggered until the user installs it.</li> <li>• The <code>GoogleMap</code> object provided by the callback is non-null.</li> </ul> <p><b>Parameters</b></p> <table border="1" data-bbox="554 686 1761 753"> <tr> <td><code>callback</code></td> <td>The callback object that will be triggered when the map is ready to be used.</td> </tr> </table> <p><code>public final void <b>onCreate</b> (<b>Bundle</b> savedInstanceState)</code></p> <p>You must call this method from the parent Activity/Fragment's corresponding method.</p> <p><a href="https://developers.google.com/android/reference/com/google/android/gms/maps/MapView">https://developers.google.com/android/reference/com/google/android/gms/maps/MapView</a></p>	<code>callback</code>	The callback object that will be triggered when the map is ready to be used.
<code>callback</code>	The callback object that will be triggered when the map is ready to be used.		
<p>2. The method of claim 1, wherein the message is transmitted to the first server.</p>	<p>HTC infringes directly and/or indirectly by performing, inducing others to perform, and/or contributing to the performance of the method of claim 1, wherein the message is transmitted to the first server. See, e.g. 1B and 1C above.</p>		
<p>3. The method of claim 1, wherein the group comprises a plurality of group members permitted to communicate with each</p>	<p>HTC infringes directly and/or indirectly by performing, inducing others to perform, and/or contributing to the performance of the method of claim 1, wherein the group comprises a plurality of group members permitted to communicate with each other via the communication network. See, e.g. 1B above.</p>		

**Exhibit D - Claim Chart for U.S. Patent No. 9,467,838 Against HTC**

<b>US9467838</b>	<b>Exemplary Supporting Evidence Regarding Accused Products</b>
other via the communication network.	
4. The method of claim 1, wherein sending the third data to the selected one or more second devices via the first server comprises using an Internet Protocol to send the third data to the first server.	HTC infringes directly and/or indirectly by performing, inducing others to perform, and/or contributing to the performance of the method of claim 1, wherein sending the third data to the selected one or more second devices via the first server comprises using an Internet Protocol to send the third data to the first server. See, e.g. 1H above. For example, contacting via chat, messaging, email, voice, and/or over applications such as Android Device Manager, Find My Phone, Find My Device, Google Latitude, Google Plus, Google Hangouts, Google Maps, Google Assistant, Google Search, Google Messages, Android Messenger, Google Allo, Google Duo, Gmail, and Google Chrome each include sending data over IP. For example, playing a sound and sending a message includes sending data over IP.
5. The method of claim 4, wherein the first device does not have access to respective Internet Protocol addresses of the one or more second devices included in the group.	HTC infringes directly and/or indirectly by performing, inducing others to perform, and/or contributing to the performance of the method of claim 4, wherein the first device does not have access to respective Internet Protocol addresses of the one or more second devices included in the group. See, e.g. 1H above. Upon information and belief, sending data from a first device contacting via chat, messaging, email, voice, and/or over applications such as Android Device Manager, Find My Phone, Find My Device, Google Latitude, Google Plus, Google Hangouts, Google Maps, Google Assistant, Google Search, Google Messages, Android Messenger, Google Allo, Google Duo, Gmail, and Google Chrome does not require the IP address of the recipient device.
6. The method of claim 1, wherein the third data include a short message service message, a text message, an image, or a video.	HTC infringes directly and/or indirectly by performing, inducing others to perform, and/or contributing to the performance of the method of claim 1, wherein the third data include a short message service message, a text message, an image, or a video. See, e.g. 1H above. For example, playing a sound and sending a message includes sending the third. For example, contacting via chat, messaging, email, voice, and/or over applications such as Android Device Manager, Find My Phone, Find My Device, Google Latitude, Google Plus, Google Hangouts, Google Maps, Google Assistant, Google Search, Google Messages, Android Messenger, Google Allo, Google Duo, Gmail, and Google Chrome each include sending data over IP. Upon information and belief, contacting via message includes SMS and text messages. Upon information and belief, contacting via Android Device Manager, Find My Phone, Find My Device, Google Latitude, Google Plus, Google Hangouts, Google Maps, Google Assistant, Google Search, Google Messages, Android Messenger, Google Allo, Google Duo, Gmail, and Google Chrome includes messaging text, images, and video.
7. The method of claim	HTC infringes directly and/or indirectly by performing, inducing others to perform, and/or contributing to

**Exhibit D - Claim Chart for U.S. Patent No. 9,467,838 Against HTC**

<b>US9467838</b>	<b>Exemplary Supporting Evidence Regarding Accused Products</b>
6, wherein the video comprises a video clip.	the performance of the method of claim 6, wherein the video comprises a video clip. See, e.g. 1H above. For example, contact over Android Device Manager, Find My Phone, Find My Device, Google Latitude, Google Plus, Google Hangouts, Google Maps, Google Assistant, Google Search, Google Messages, Android Messenger, Google Allo, Google Duo, Gmail, and Google Chrome includes sending a video clip.
8. The method of claim 1, wherein the third data include a voice recording.	<p>HTC infringes directly and/or indirectly by performing, inducing others to perform, and/or contributing to the performance of the method of claim 1, wherein the third data include a voice recording. See, e.g. 1H above. For example, contacting via Android Device Manager, Find My Phone, Find My Device, Google Latitude, Google Plus, Google Hangouts, Google Maps, Google Assistant, Google Search, Google Messages, Android Messenger, Google Allo, Google Duo, Gmail, and Google Chrome, include sending a voice recording or audio recording. For example, playing a sound a second device including sending and playing a voice file. For example, Facetime includes sending a recording of a voice. For example, each Accused Product can send a recording.</p> <p>For example, each Accused Product can send a recording.</p> <p><a href="https://www.techrepublic.com/article/how-to-send-audio-clips-via-sms-in-android/">https://www.techrepublic.com/article/how-to-send-audio-clips-via-sms-in-android/</a></p>
9. The method of claim 1, wherein sending the third data to the selected one or more second devices comprises transmitting a text message to at least one of the selected one or more second devices using an Internet Protocol (IP).	HTC infringes directly and/or indirectly by performing, inducing others to perform, and/or contributing to the performance of the method of claim 1, wherein sending the third data to the selected one or more second devices comprises transmitting a text message to at least one of the selected one or more second devices using an Internet Protocol (IP). See, e.g. 1H above. For example, contacting Android Device Manager, Find My Phone, Find My Device, Google Latitude, Google Plus, Google Hangouts, Google Maps, Google Assistant, Google Search, Google Messages, Android Messenger, Google Allo, Google Duo, Gmail, and Google Chrome includes sending a text message over IP. For example, entering and sending a message to a device includes sending the text message over IP.
10. The method of claim 1, further comprising performing by the first device: identifying user interaction with the	HTC infringes directly and/or indirectly by performing, inducing others to perform, and/or contributing to the performance of the method of claim 1, further comprising performing by the first device: identifying user interaction with the interactive display selecting at least one of the second set of user-selectable symbols corresponding to at least one of the second devices and user interaction with the display specifying an action and, based thereon, initiating a phone call or phone conference with the at least one

**Exhibit D - Claim Chart for U.S. Patent No. 9,467,838 Against HTC**

<b>US9467838</b>	<b>Exemplary Supporting Evidence Regarding Accused Products</b>
<p>interactive display selecting at least one of the second set of user-selectable symbols corresponding to at least one of the second devices and user interaction with the display specifying an action and, based thereon, initiating a phone call or phone conference with the at least one second device.</p>	<p>second device. See, e.g. 1H above. For example, Find My Device and Android Device Manager allow a first device to initiate a phone call or phone conference by sending a phone number to the second device. For example, Google Maps and its predecessors allow a user to call another user by presenting the user of the first device with the option to select the friend's phone number to initiate call and the option to initiate communication. For example, Android Device Manager, Find My Phone, Find My Device, Google Latitude, Google Plus, Google Hangouts, Google Maps, Google Assistant, Google Search, Google Messages, Android Messenger, Google Allo, Google Duo, Gmail, and Google Chrome each provide for initiating a call and/or conference.</p>
<p>11. The method of claim 1, further comprising performing by the first device: identifying user interaction with the interactive display selecting a particular user-selectable symbol corresponding to a particular second device and user interaction with the display specifying an action and, based thereon, initiating voice-over-IP (VOIP) communication with the particular second device.</p>	<p>HTC infringes directly and/or indirectly by performing, inducing others to perform, and/or contributing to the performance of the method of claim 1, further comprising performing by the first device: identifying user interaction with the interactive display selecting a particular user-selectable symbol corresponding to a particular second device and user interaction with the display specifying an action and, based thereon, initiating voice-over-IP (VOIP) communication with the particular second device. See, e.g. 1H above. For example, Find My Device allows a first device to initiate a phone call or phone conference by sending a phone number to the second device. For example, Google Maps allows a user to call the friend by presenting the user of the first device with the option to select the friend's phone number to initiate call and the option to initiate communication. For example, Android Device Manager, Find My Phone, Find My Device, Google Latitude, Google Plus, Google Hangouts, Google Maps, Google Assistant, Google Search, Google Messages, Android Messenger, Google Allo, Google Duo, Gmail, and Google Chrome each provide for initiating VOIP communication with another device.</p>
<p>12. The method of claim</p>	<p>HTC infringes directly and/or indirectly by performing, inducing others to perform, and/or contributing to</p>

**Exhibit D - Claim Chart for U.S. Patent No. 9,467,838 Against HTC**

<b>US9467838</b>	<b>Exemplary Supporting Evidence Regarding Accused Products</b>
1, further comprising performing by the first device: identifying user interaction with the interactive display selecting a particular user-selectable symbol corresponding to a particular second device and user interaction with the display specifying an action and, based thereon, initiating a data call with the particular second device.	the performance of the method of claim 1, further comprising performing by the first device: identifying user interaction with the interactive display selecting a particular user-selectable symbol corresponding to a particular second device and user interaction with the display specifying an action and, based thereon, initiating a data call with the particular second device. See, e.g. 1H above. For example, Find My Device allows a first device to initiate a phone call or phone conference by sending a phone number to the second device. For example, Google Maps allows a user to call the friend by presenting the user of the first device with the option to select the friend's phone number to initiate call and the option to initiate a Facetime session. For example, Android Device Manager, Find My Phone, Find My Device, Google Latitude, Google Plus, Google Hangouts, Google Maps, Google Assistant, Google Search, Google Messages, Android Messenger, Google Allo, Google Duo, GMail, and Google Chrome each provide for initiating communication upon selection of a contact.
13. The method of claim 1, wherein the first device is a personal digital assistant (PDA) or a personal computer (PC).	HTC infringes directly and/or indirectly by performing, inducing others to perform, and/or contributing to the performance of the method of claim 1, wherein the first device is a personal digital assistant (PDA) or a personal computer (PC). See, e.g., 1P and 1A above. Upon information and belief, the Accused Products are forms of PDAs in that the functionality of a PDA has been subsumed into smartphones, tablets, and portable media players having the functionalities of a PDA. Upon information and belief, the Accused products are forms of personal computers in that the Accused Products are mobile or portable forms of personal computers. To the extent that it is necessary, AGIS submits that the Accused Products meet the claim limitation "the first device is a personal digital assistant (PDA) or a personal computer (PC)" under the doctrine of equivalents.
14. The method of claim 1, wherein the first device is a smart phone.	HTC infringes directly and/or indirectly by performing, inducing others to perform, and/or contributing to the performance of the method of claim 1, wherein the first device is a smart phone. See, e.g., 1P and 1A above. For example, the Accused Products include a smartphone. To the extent that it is necessary, AGIS submits that tablets within the Accused Products meet the claim limitation "the first device is smartphone" under the doctrine of equivalents.
15. The method of claim 1, wherein the display of the first device is a touch	HTC infringes directly and/or indirectly by performing, inducing others to perform, and/or contributing to the performance of the method of claim 1, wherein the display of the first device is a touch screen display, and wherein the user interaction with the display selecting the one or more user-selectable symbols in the

**Exhibit D - Claim Chart for U.S. Patent No. 9,467,838 Against HTC**

<b>US9467838</b>	<b>Exemplary Supporting Evidence Regarding Accused Products</b>
<p>screen display, and wherein the user interaction with the display selecting the one or more user-selectable symbols in the second set of symbols comprises touching the one or more user-selectable symbols in the second set of symbols.</p>	<p>second set of symbols comprises touching the one or more user-selectable symbols in the second set of symbols. See, e.g., 1P, 1A, and 1H above. For example, the Accused Devices are controlled by touch displays and selection of the symbol occurs on the touch display.</p>
<p>16. The method of claim 1, further comprising performing by the first device: sending updated location information comprising an updated location of the first device, the updated location information being sent based on passage of a predetermined time interval since sending previous location information comprising a previous location of the first device, displacement of the first device by a predetermined distance relative to a previous location of the first device, the updated location information being sent based on passage of a predetermined time interval since sending previous location information comprising a previous location of the first device, displacement of the first device by a predetermined distance relative to a previous</p>	<p>HTC infringes directly and/or indirectly by performing, inducing others to perform, and/or contributing to the performance of the method of claim 1, further comprising performing by the first device: sending updated location information comprising an updated location of the first device, the updated location information being sent based on passage of a predetermined time interval since sending previous location information comprising a previous location of the first device, displacement of the first device by a predetermined distance relative to a previous location of the first device, or both. See, e.g., 1G above. For example, when a first device moves from a first to a second position, the symbol's position on the map is updated to reflect the change. Upon information and belief, the transmission of updated location information is based on a predetermined time passage or a predetermined distance change.</p> <p>For example the Accused Products utilize a location manager service such as LocationServices that monitors, among other things, periods of time that are predetermined, or displacement from a predetermined distance.</p> <p>This method is thus performed at least when a user passes a geographic threshold that results in a location update (e.g., "leaves" notification). For example, Android utilizes the following framework that meets this limitation as implemented on the Accused Products:</p> <p><a href="https://developers.google.com/android/reference/com/google/android/gms/location/LocationServices">https://developers.google.com/android/reference/com/google/android/gms/location/LocationServices</a>  <a href="https://developer.android.com/reference/android/location/LocationManager.html">https://developer.android.com/reference/android/location/LocationManager.html</a></p>

**Exhibit D - Claim Chart for U.S. Patent No. 9,467,838 Against HTC**

<b>US9467838</b>	<b>Exemplary Supporting Evidence Regarding Accused Products</b>
location of the first device, or both.	e.g. requestLocationUpdates(String provider, long minTime, float minDistance, LocationListener listener) Register for location updates using the named provider, and a pending intent.  Android also makes use of “geotagging” that invokes time-based updating.
17. The method of claim 1, further comprising performing by the first device: using a Global Positioning Satellite (GPS) receiver of the first device to obtain data indicative of the location of the first device, wherein sending the first location information to the first server comprises using an Internet Protocol (IP) to send the first location information to the first server.	HTC infringes directly and/or indirectly by performing, inducing others to perform, and/or contributing to the performance of the method of claim 1, further comprising performing by the first device: using a Global Positioning Satellite (GPS) receiver of the first device to obtain data indicative of the location of the first device, wherein sending the first location information to the first server comprises using an Internet Protocol (IP) to send the first location information to the first server. See, e.g., 1P and 1A above regarding the hardware and software including GPS and A-GPS. See, e.g., 1C and 1D regarding the employment of location services and location sharing to obtain and send locations. For example, Android Device Manager, Find My Phone, Find My Device, Google Latitude, Google Plus, Google Hangouts, Google Maps, Google Assistant, Google Search, Google Messages, Android Messenger, Google Allo, Google Duo, Gmail, and Google Chrome use location services and location sharing to obtain locations and send those locations and transmit locations to a server.
18. The method of claim 17, wherein sending the first location information to the first server further comprises sending the first location information via the Internet.	HTC infringes directly and/or indirectly by performing, inducing others to perform, and/or contributing to the performance of the method of claim 17, wherein sending the first location information to the first server further comprises sending the first location information via the Internet. See, e.g., 1P and 1A above regarding the hardware and software for transmitting data over the Internet. See, e.g., 1C and 1D regarding the employment of location services and location sharing to obtain and send locations over the Internet. For example, Android Device Manager, Find My Phone, Find My Device, Google Latitude, Google Plus, Google Hangouts, Google Maps, Google Assistant, Google Search, Google Messages, Android Messenger, Google Allo, Google Duo, Gmail, and Google Chrome use location services and location sharing to obtain locations and send those locations over the Internet for transmission to recipient devices/users.
19. The method of claim 1, wherein participating	HTC infringes directly and/or indirectly by performing, inducing others to perform, and/or contributing to the performance of the method of claim 1, wherein participating in the group further includes sending first

**Exhibit D - Claim Chart for U.S. Patent No. 9,467,838 Against HTC**

<b>US9467838</b>	<b>Exemplary Supporting Evidence Regarding Accused Products</b>
<p>in the group further includes sending first status information to the first server and receiving second status information from the first server, wherein the first status information comprises data indicative of a battery level of the first device, a signal strength of a wireless signal of the first device, a status of a Global Positioning Satellite (GPS) receiver of the first device, or a combination thereof, and wherein the second status information comprises data indicative of one or more battery levels of the respective one or more second devices included in the group, one or more signal strengths of wireless signals of the respective one or more second devices included in the group, one or more statuses of GPS receivers of the respective one or more second devices included in the group, or a combination thereof.</p>	<p>status information to the first server and receiving second status information from the first server, wherein the first status information comprises data indicative of a battery level of the first device, a signal strength of a wireless signal of the first device, a status of a Global Positioning Satellite (GPS) receiver of the first device, or a combination thereof, and wherein the second status information comprises data indicative of one or more battery levels of the respective one or more second devices included in the group, one or more signal strengths of wireless signals of the respective one or more second devices included in the group, one or more statuses of GPS receivers of the respective one or more second devices included in the group, or a combination thereof. See, e.g., 1A, 1C, and 1D. For example, Android Device Manager, Find My Phone, Find My Device, Google Latitude, Google Plus, Google Hangouts, Google Maps, Google Assistant, Google Search, Google Messages, Android Messenger, Google Allo, Google Duo, Gmail, and Google Chrome show status indicators for each member of the group, including an indicator of connectivity status of each member (see below). Upon information and belief, status metadata regarding the GPS receiver, wireless signals, device battery, and other status metadata are sent and received between the Accused Products.</p> <p>For example, the Accused Products each the ability to communicate battery level and wireless networks to servers, where that battery level and wireless networks are communicated to the first device.</p> <p>Additionally, the status of the GPS receivers is also communicated within the apps. For example, the second devices communicate the accuracy of their GPS signal, as well as status so that the first device can view both the accuracy of the signal (represented by a circle around the user) as well as the status (online vs. offline).</p> <p>Additionally, status of a GPS receiver may be communicated through notifications that communicate when a device has passed a location threshold.</p>



**Exhibit D - Claim Chart for U.S. Patent No. 9,467,838 Against HTC**

<b>US9467838</b>	<b>Exemplary Supporting Evidence Regarding Accused Products</b>
<p>statuses of GPS receivers of the respective one or more second devices included in the group, or a combination thereof.</p>	
<p>20. The method of claim 1, wherein the second georeferenced map data comprise a satellite image or aerial photograph.</p>	<p>HTC infringes directly and/or indirectly by performing, inducing others to perform, and/or contributing to the performance of the method of claim 1, wherein the second georeferenced map data comprise a satellite image or aerial photograph. See, e.g., 1D and 1G. For example and shown below, Android Device Manager, Find My Phone, Find My Device, Google Latitude, Google Plus, Google Hangouts, Google Maps, Google Assistant, Google Search, Google Messages, Android Messenger, Google Allo, Google Duo, Gmail, and Google Chrome show maps in satellite and/or aerial formats.</p>
<p>21. The method of claim 1, wherein the spatial coordinates comprise latitude and longitude coordinates.</p>	<p>HTC infringes directly and/or indirectly by performing, inducing others to perform, and/or contributing to the performance of the method of claim 1, wherein the spatial coordinates comprise latitude and longitude coordinates. See, e.g., 1C, 1D, and 1G. For example, the Accused Products present maps with, at least, latitude and longitude coordinates.</p>
<p>22[A]. The method of claim 1, further comprising identifying, by the first device, user interaction with the display selecting a particular user-selectable symbol positioned on the second georeferenced map and corresponding to a particular second device, wherein identifying the</p>	<p>HTC infringes directly and/or indirectly by performing, inducing others to perform, and/or contributing to the performance of the method of claim 1, further comprising identifying, by the first device, user interaction with the display selecting a particular user-selectable symbol positioned on the second georeferenced map and corresponding to a particular second device, wherein identifying the user interaction selecting the particular user-selectable symbol comprises: detecting user selection of a portion of the interactive display corresponding to a position on the second georeferenced map. See, e.g., 1G and 1H. For example, the user of the first device selects a symbol on the map by touching the display. The selection occurs on a touch display of the first device which includes logic for detecting the touch position and corresponding the touch position to a position on the georeferenced map. See, e.g., 1P and 1A.</p>

## Exhibit D - Claim Chart for U.S. Patent No. 9,467,838 Against HTC

US9467838	Exemplary Supporting Evidence Regarding Accused Products
<p>user interaction selecting the particular user-selectable symbol comprises: detecting user selection of a portion of the interactive display corresponding to a position on the second georeferenced map;</p>	
<p>[22B] based at least in part on coordinates of the selected position on the second georeferenced map and on the second georeferenced map data relating positions on the second georeferenced map to spatial coordinates, determining spatial coordinates of a location represented by the selected position on the second georeferenced map;</p>	<p>HTC infringes directly and/or indirectly by performing, inducing others to perform, and/or contributing to the performance of, based at least in part on coordinates of the selected position on the second georeferenced map and on the second georeferenced map data relating positions on the second georeferenced map to spatial coordinates, determining spatial coordinates of a location represented by the selected position on the second georeferenced map. See, e.g., 1P, 1A, 1G and 1H. Upon information and belief, the first device determines spatial coordinates representing selected positions on the map.</p> <p>For example, when markers, such as symbols, are added to maps, they are added based on longitude and latitude.</p> <p>Add a map</p> <p>Display a map, using the Google Maps Android API.</p> <p>Add a <code>&lt;fragment&gt;</code> element to your activity's layout file, <code>activity_maps.xml</code>. This element defines a <code>SupportMapFragment</code> to act as a container for the map and to provide access to the <code>GoogleMap</code> object. The tutorial uses the Android support library version of the map fragment, to ensure backward compatibility with earlier versions of the Android framework.</p> <pre data-bbox="533 1252 1913 1408">&lt;fragment xmlns:android="http://schemas.android.com/apk/res/android"     xmlns:tools="http://schemas.android.com/tools"     android:id="@+id/map"     android:name="com.google.android.gms.maps.SupportMapFragment"</pre>

## Exhibit D - Claim Chart for U.S. Patent No. 9,467,838 Against HTC

US9467838	Exemplary Supporting Evidence Regarding Accused Products
	<pre data-bbox="535 235 1911 389"> android:layout_width="match_parent" android:layout_height="match_parent" tools:context="com.example.mapwithmarker.MapsMarkerActivity" /&gt; </pre> <p data-bbox="535 422 1911 495">In your activity's onCreate() method, set the layout file as the content view. Get a handle to the map fragment by calling <code>FragmentManager.findFragmentById()</code>. Then use <code>getMapAsync()</code> to register for the map callback:</p> <pre data-bbox="535 527 1911 966"> @Override protected void onCreate(Bundle savedInstanceState) {     super.onCreate(savedInstanceState);     // Retrieve the content view that renders the map.     setContentView(R.layout.activity_maps);     // Get the SupportMapFragment and request notification     // when the map is ready to be used.     SupportMapFragment mapFragment = (SupportMapFragment) getSupportFragmentManager()         .findFragmentById(R.id.map);     mapFragment.getMapAsync(this); } </pre> <p data-bbox="535 998 1911 1071">Implement the <code>OnMapReadyCallback</code> interface and override the <code>onMapReady()</code> method, to set up the map when the <code>GoogleMap</code> object is available:</p> <pre data-bbox="535 1104 1911 1412"> public class MapsMarkerActivity extends AppCompatActivity     implements OnMapReadyCallback {     // Include the onCreate() method here too, as described above.     @Override     public void onMapReady(GoogleMap googleMap) {         // Add a marker in Sydney, Australia,         // and move the map's camera to the same location.         LatLng sydney = new LatLng(-33.852, 151.211); </pre>

**Exhibit D - Claim Chart for U.S. Patent No. 9,467,838 Against HTC**

US9467838	Exemplary Supporting Evidence Regarding Accused Products
	<pre> googleMap.addMarker(new MarkerOptions().position(sydney)     .title("Marker in Sydney")); googleMap.moveCamera(CameraUpdateFactory.newLatLng(sydney)); } </pre> <p>See, also, <a href="https://developers.google.com/maps/documentation/android-api/marker">https://developers.google.com/maps/documentation/android-api/marker</a></p>
<p>[22C] and identifying the particular user-selectable symbol based, at least in part, on the spatial coordinates represented by the selected position.</p>	<p>HTC infringes directly and/or indirectly by performing, inducing others to perform, and/or contributing to the performance of identifying the particular user-selectable symbol based, at least in part, on the spatial coordinates represented by the selected position. See, e.g., 1P, 1A, 1G and 1H. Upon information and belief, the first device identifies a symbol based on the determined coordinates.</p> <p>Each marker corresponds to a spatial coordinate (i.e. longitude/latitude) as well as an x/y coordinate on the map display.</p> <p>See, e.g.,</p> <p><a href="https://developers.google.com/maps/documentation/android-api/map-with-marker">https://developers.google.com/maps/documentation/android-api/map-with-marker</a>  <a href="https://developers.google.com/maps/documentation/android-api/marker">https://developers.google.com/maps/documentation/android-api/marker</a></p>
<p>23[A]. The method of claim 22, wherein identifying the particular user-selectable symbol based, at least in part, on the spatial coordinates represented by the selected position comprises: searching a database of entities for an entity located nearest to the spatial coordinates represented by the</p>	<p>HTC infringes directly and/or indirectly by performing, inducing others to perform, and/or contributing to the performance of the method of claim 22, wherein identifying the particular user-selectable symbol based, at least in part, on the spatial coordinates represented by the selected position comprises: searching a database of entities for an entity located nearest to the spatial coordinates represented by the selected position, wherein the entities represented by data in the database include the one or more second devices included in the group, wherein the database data include locations of the respective entities, and wherein the database is searchable by location. See, e.g., 1G, 1H, and 22. Upon information and belief, the Accused Products include a database located at the first device or send a search request to a server for performing the searching methods of this limitation.</p> <p>See, e.g.,</p>

**Exhibit D - Claim Chart for U.S. Patent No. 9,467,838 Against HTC**

US9467838	Exemplary Supporting Evidence Regarding Accused Products
<p>selected position, wherein the entities represented by data in the database include the one or more second devices included in the group, wherein the database data include locations of the respective entities, and wherein the database is searchable by location;</p>	<p>Selection with Markers:  <a href="https://developers.google.com/maps/documentation/android-api/marker">https://developers.google.com/maps/documentation/android-api/marker</a></p> <p>Queries with GeoTagging database:  <a href="https://developers.google.com/android/reference/com/google/android/gms/location/places/GeoDataClient">https://developers.google.com/android/reference/com/google/android/gms/location/places/GeoDataClient</a></p>
<p>[23B] and based on a result of searching the database, identifying the particular second device as the entity located nearest to the spatial coordinates represented by the selected position, wherein the particular user-selectable symbol corresponds to the particular second device.</p>	<p>HTC infringes directly and/or indirectly by performing, inducing others to perform, and/or contributing to the performance of, based on a result of searching the database, identifying the particular second device as the entity located nearest to the spatial coordinates represented by the selected position, wherein the particular user-selectable symbol corresponds to the particular second device. See, e.g. 1G and 1H. Upon information and belief, the Accused Devices include a database located either at the first device or send a search request to a server for performing the searching methods of this limitation.</p> <p>See, e.g, Android Maps and Geotagging APIs:</p> <p>Instantiate the Places API clients</p> <p>The following interfaces provide the primary entry points to the Google Places API for Android:</p> <ul style="list-style-type: none"> <li>• The <a href="#">GeoDataClient</a> provides access to Google's database of local place and business information.</li> <li>• The <a href="#">PlaceDetectionClient</a> provides quick access to the device's current place, and offers the opportunity to report the location of the device at a particular place.</li> </ul> <p>The <a href="#">LocationServices</a> interface is the main entry point for Android location services.</p> <p>To use the APIs, instantiate GeoDataClient, PlaceDetectionClient, and FusedLocationProviderClient in your fragment's or activity's <a href="#">onCreate()</a> method, as shown in the following code sample:</p>

## Exhibit D - Claim Chart for U.S. Patent No. 9,467,838 Against HTC

US9467838	Exemplary Supporting Evidence Regarding Accused Products
	<pre data-bbox="535 337 1915 852"> protected void onCreate(Bundle savedInstanceState) {     super.onCreate(savedInstanceState);     setContentView(R.layout.activity_main);      // Construct a GeoDataClient.     mGeoDataClient = Places.getGeoDataClient(this, null);      // Construct a PlaceDetectionClient.     mPlaceDetectionClient = Places.getPlaceDetectionClient(this, null);      // Construct a FusedLocationProviderClient.     mFusedLocationProviderClient =     LocationServices.getFusedLocationProviderClient(this); </pre> <p data-bbox="535 927 1915 992">See, e.g., <a href="https://developers.google.com/maps/documentation/android-api/current-place-tutorial">https://developers.google.com/maps/documentation/android-api/current-place-tutorial</a>  <a href="https://developers.google.com/android/reference/com/google/android/gms/location/places/GeoDataClient">https://developers.google.com/android/reference/com/google/android/gms/location/places/GeoDataClient</a></p>
24[A]. The method of claim 23, wherein the entity is a first entity, and wherein the method further comprises performing by the first device: receiving user input via user interaction with the interactive display of the first device, the user input	HTC infringes directly and/or indirectly by performing, inducing others to perform, and/or contributing to the performance of the method of claim 23, wherein the entity is a first entity, and wherein the method further comprises performing by the first device: receiving user input via user interaction with the interactive display of the first device, the user input specifying a location and a symbol corresponding to a second entity other than the first device and the one or more second devices included in the group. See, e.g., 1G, 1H, 22, and 23. For example, the user of the first device selects a symbol on the map by touching the display. The selection occurs on a touch display of the first device which includes logic for detecting the touch position and corresponding the touch position to a position on the georeferenced map. See, e.g., 1P an 1A. For example, the user of the first device can select other symbols on the touch display.

**Exhibit D - Claim Chart for U.S. Patent No. 9,467,838 Against HTC**

US9467838	Exemplary Supporting Evidence Regarding Accused Products
specifying a location and a symbol corresponding to a second entity other than the first device and the one or more second devices included in the group;	
[24B] and based on the user input, adding the user-specified symbol to the interactive display at a position on the second georeferenced map corresponding to the user-specified location of the second entity.	HTC infringes directly and/or indirectly by performing, inducing others to perform, and/or contributing to the performance of, based on the user input, adding the user-specified symbol to the interactive display at a position on the second georeferenced map corresponding to the user-specified location of the second entity. See, e.g., 1G and 1H. For example, the user of the first device selects a symbol on the map by touching the display. The selection occurs on a touch display of the first device which includes logic for detecting the touch position and corresponding the touch position to a position on the georeferenced map. See, e.g., 1P and 1A. For example, the user of the first device can select other symbols on the touch display and a user can drop a marker on a map and/or share a location by specifying another user/device, specifying a method for sharing, and specifying additional data such as text to be sent with the location/marker.
25. The method of claim 24, further comprising performing by the first device: transmitting the user-specified symbol and location of the second entity to the one or more second devices included in the group for addition of the user-specified symbol to respective interactive displays of the one or more second devices at respective positions on respective georeferenced maps corresponding to the user-specified location of the second entity.	<p>HTC infringes directly and/or indirectly by performing, inducing others to perform, and/or contributing to the performance of the method of claim 24, further comprising performing by the first device: transmitting the user-specified symbol and location of the second entity to the one or more second devices included in the group for addition of the user-specified symbol to respective interactive displays of the one or more second devices at respective positions on respective georeferenced maps corresponding to the user-specified location of the second entity. See, e.g., 1P, 1A, 22, 23, and 24. For example, the user of the first device can select other symbols on the touch display and a user can drop a marker on a map and/or share a location by specifying another user/device, specifying a method for sharing, and specifying additional data such as text to be sent with the location/marker.</p> <p>See, e.g., Placing a Marker:  <a href="https://developers.google.com/maps/documentation/android-api/marker">https://developers.google.com/maps/documentation/android-api/marker</a></p> <p>based on queries with GeoTagging database:  <a href="https://developers.google.com/android/reference/com/google/android/gms/location/places/GeoDataClient">https://developers.google.com/android/reference/com/google/android/gms/location/places/GeoDataClient</a></p>

**Exhibit D - Claim Chart for U.S. Patent No. 9,467,838 Against HTC**

<b>US9467838</b>	<b>Exemplary Supporting Evidence Regarding Accused Products</b>
maps corresponding to the user-specified location of the second entity.	Sharing a link: <a href="https://support.google.com/maps/answer/144361?co=GENIE.Platform%3DDesktop&amp;hl=en">https://support.google.com/maps/answer/144361?co=GENIE.Platform%3DDesktop&amp;hl=en</a>
26. The method of claim 25, wherein the user input further specifies information associated with the second entity, and wherein the method further comprises performing by the first device: transmitting the user-specified information associated with the second entity to the one or more second devices included in the group.	HTC infringes directly and/or indirectly by performing, inducing others to perform, and/or contributing to the performance of the method of claim 25, wherein the user input further specifies information associated with the second entity, and wherein the method further comprises performing by the first device: transmitting the user-specified information associated with the second entity to the one or more second devices included in the group. See, e.g., 1P, 1A, 1B, 22, 23, 24, and 25. For example, the user of the first device can select other symbols on the touch display and a user can drop a marker on a map and/or share a location by specifying another user/device, specifying a method for sharing, and specifying additional data such as text to be sent with the location/marker.
27. The method of claim 26, wherein the information comprises a category of the second entity.	HTC infringes directly and/or indirectly by performing, inducing others to perform, and/or contributing to the performance of the method of claim 26, wherein the information comprises a category of the second entity. See, e.g., 1P, 1A, 22, 23, 24, 25, and 26. For example, the user of the first device can select other symbols on the touch display and a user can drop a marker on a map and/or share a location by specifying another user/device, specifying a method for sharing, and specifying additional data such as text to be sent with the location/marker.
28. The method of claim 27, wherein the category comprises a vehicle, a person, an event, a site, a building, or a facility.	HTC infringes directly and/or indirectly by performing, inducing others to perform, and/or contributing to the performance of the method of claim 27, wherein the category comprises a vehicle, a person, an event, a site, a building, or a facility. See, e.g., 1P, 1A, 22, 23, 24, 25, and 26. For example, the user of the first device can select other symbols on the touch display and a user can drop a marker on a map and/or share a location by specifying another user/device, specifying a method for sharing, and specifying additional data such as text to be sent with the location/marker. The location can specify a person, an event, or some other location, building, or facility.



**Exhibit D - Claim Chart for U.S. Patent No. 9,467,838 Against HTC**

<b>US9467838</b>	<b>Exemplary Supporting Evidence Regarding Accused Products</b>
29. The method of claim 26, wherein the information comprises an image.	HTC infringes directly and/or indirectly by performing, inducing others to perform, and/or contributing to the performance of the method of claim 26, wherein the information comprises an image. See, e.g., 1P, 1A, 22, 23, 24, 25, and 26. For example, the user of the first device can select other symbols on the touch display and a user can drop a marker on a map and/or share a location by specifying another user/device, specifying a method for sharing, and specifying additional data such as text to be sent with the location/marker. When information is shared by message, the user may also transmit an image.
30. The method of claim 26, wherein the information comprises at least one type of information selected from the group consisting of text and video.	HTC infringes directly and/or indirectly by performing, inducing others to perform, and/or contributing to the performance of the method of claim 26, wherein the information comprises at least one type of information selected from the group consisting of text and video. See, e.g., 1P, 1A, 22, 23, 24, 25, and 26. For example, the user of the first device can select other symbols on the touch display and a user can drop a marker on a map and/or share a location by specifying another user/device, specifying a method for sharing, and specifying additional data such as text to be sent with the location/marker. When information is shared by message, the user may also transmit text and/or video.
31. The method of claim 26, further comprising performing by the first device: identifying user interaction with the interactive display selecting the symbol corresponding to the second entity, and based thereon, displaying the information associated with the second entity.	HTC infringes directly and/or indirectly by performing, inducing others to perform, and/or contributing to the performance of the method of claim 26, further comprising performing by the first device: identifying user interaction with the interactive display selecting the symbol corresponding to the second entity, and based thereon, displaying the information associated with the second entity. See, e.g., 1P, 1A, 22, 23, 24, 25, and 26. For example, the user of the first device can select other symbols on the touch display and a user can drop a marker on a map and/or share a location by specifying another user/device, specifying a method for sharing, and specifying additional data such as text to be sent with the location/marker.
32. The method of claim 31, wherein the first device uses an Internet Protocol to transmit the user-specified symbol, location, and	HTC infringes directly and/or indirectly by performing, inducing others to perform, and/or contributing to the performance of the method of claim 31, wherein the first device uses an Internet Protocol to transmit the user-specified symbol, location, and information associated with the second entity. See, e.g., 1P, 1A, 22, 23, 24, 25, 26, and 31. For example, the user of the first device can select other symbols on the touch display and a user can drop a marker on a map and/or share a location by specifying another user/device, specifying a method for sharing, and specifying additional data such as text to be sent with the

**Exhibit D - Claim Chart for U.S. Patent No. 9,467,838 Against HTC**

<b>US9467838</b>	<b>Exemplary Supporting Evidence Regarding Accused Products</b>
information associated with the second entity.	location/marker. See, e.g., 1P and 1A above regarding the hardware and software for transmitting data over the Internet. See, e.g., 1C and 1D regarding the employment of location services and location sharing to obtain and send locations over the Internet. For example, both Find My Friends and Find My Family use location services and location sharing to obtain locations and send those locations to corresponding servers over the Internet for transmission to recipient devices/users. For example, sending messages on the Accused Devices using Android Device Manager, Find My Phone, Find My Device, Google Latitude, Google Plus, Google Hangouts, Google Maps, Google Assistant, Google Search, Google Messages, Android Messenger, Google Allo, Google Duo, Gmail, Google Chrome, and/or associated services includes sending communication over IP.
33. The method of claim 26, further comprising performing by the first device: adding data representing the spatial coordinates of the location of the second entity and data representing the information associated with the second entity to the database.	HTC infringes directly and/or indirectly by performing, inducing others to perform, and/or contributing to the performance of the method of claim 26, further comprising performing by the first device: adding data representing the spatial coordinates of the location of the second entity and data representing the information associated with the second entity to the database. See, e.g., 1P, 1A, 22, 23, 24, 25, and 26. For example, the user of the first device can select other symbols on the touch display and a user can drop a marker on a map and/or share a location by specifying another user/device, specifying a method for sharing, and specifying additional data such as text to be sent with the location/marker.
34[A]. The method of claim 24, wherein the portion of the interactive display is a first portion, wherein the position of the symbol corresponding to the particular second device is a first position, and wherein receiving the user input specifying the	HTC infringes directly and/or indirectly by performing, inducing others to perform, and/or contributing to the performance of the method of claim 24, wherein the portion of the interactive display is a first portion, wherein the position of the symbol corresponding to the particular second device is a first position, and wherein receiving the user input specifying the location of the second entity comprises: detecting user selection of a second portion of the interactive display corresponding to a second position on the second georeferenced map. See, e.g., 1G and 1H. For example, the user of the first device selects a symbol on the map by touching the display. The selection occurs on a touch display of the first device which includes logic for detecting the touch position and corresponding the touch position to a position on the georeferenced map. See, e.g., 1P and 1A.

**Exhibit D - Claim Chart for U.S. Patent No. 9,467,838 Against HTC**

<b>US9467838</b>	<b>Exemplary Supporting Evidence Regarding Accused Products</b>
location of the second entity comprises: detecting user selection of a second portion of the interactive display corresponding to a second position on the second georeferenced map;	
[34B] and based at least in part on coordinates of the second position on the second georeferenced map and on the second georeferenced map data relating positions on the second georeferenced map to spatial coordinates, determining spatial coordinates of a location represented by the second position on the second georeferenced map, wherein the location represented by the second position is the location of the second entity.	HTC infringes directly and/or indirectly by performing, inducing others to perform, and/or contributing to the performance of, based at least in part on coordinates of the second position on the second georeferenced map and on the second georeferenced map data relating positions on the second georeferenced map to spatial coordinates, determining spatial coordinates of a location represented by the second position on the second georeferenced map, wherein the location represented by the second position is the location of the second entity. See, e.g., 1P, 1A, 1G and 1H. Upon information and belief, the first device determines spatial coordinates representing selected positions on the map.
35. The method of claim 23, wherein the database	HTC infringes directly and/or indirectly by performing, inducing others to perform, and/or contributing to the performance of the method of claim 23, wherein the database is stored on the first device. See, e.g., 1P,

**Exhibit D - Claim Chart for U.S. Patent No. 9,467,838 Against HTC**

<b>US9467838</b>	<b>Exemplary Supporting Evidence Regarding Accused Products</b>
is stored on the first device.	1A, 1G and 1H. Upon information and belief, the first device keeps one or more databases for the purposes of Android Device Manager, Find My Phone, Find My Device, Google Latitude, Google Plus, Google Hangouts, Google Maps, Google Assistant, Google Search, Google Messages, Android Messenger, Google Allo, Google Duo, GMail, and Google Chrome.
36. The method of claim 23, wherein the database is stored on the first server.	HTC infringes directly and/or indirectly by performing, inducing others to perform, and/or contributing to the performance of the method of claim 23, wherein the database is stored on the first server. See, e.g., 1P, 1A, 1G and 1H. Upon information and belief, the Accused Products communicate with servers keeping databases for the same purposes discussed in claims 1 and 23, e.g., Android Device Manager, Find My Phone, Find My Device, Google Latitude, Google Plus, Google Hangouts, Google Maps, Google Assistant, Google Search, Google Messages, Android Messenger, Google Allo, Google Duo, GMail, and Google Chrome. See, e.g., 1E and 1F.
37[A]. The method of claim 1, further comprising performing by the first device: receiving user-specified information transmitted by a particular second device, the user-specified information including a user-specified location and a user-specified symbol corresponding to an entity other than the first device and the one or more second devices included in the group;	HTC infringes directly and/or indirectly by performing, inducing others to perform, and/or contributing to the performance of the method of claim 1, further comprising performing by the first device: receiving user-specified information transmitted by a particular second device, the user-specified information including a user-specified location and a user-specified symbol corresponding to an entity other than the first device and the one or more second devices included in the group. See, e.g., 1P and 1A above regarding the hardware and software for transmitting data over the Internet. See, e.g., 1C and 1D regarding the use of location services and location sharing to obtain and send locations over the Internet. For example, Android Device Manager, Find My Phone, Find My Device, Google Latitude, Google Plus, Google Hangouts, Google Maps, Google Assistant, Google Search, Google Messages, Android Messenger, Google Allo, Google Duo, GMail, and Google Chrome use location services and location sharing to obtain locations and send/receive those locations to corresponding servers over the Internet for transmission to recipient devices/users. For example, receiving messages on the Accused Devices using Android Device Manager, Find My Phone, Find My Device, Google Latitude, Google Plus, Google Hangouts, Google Maps, Google Assistant, Google Search, Google Messages, Android Messenger, Google Allo, Google Duo, GMail, and Google Chrome and associated servers for the same includes sending communication over IP.
[37B] and adding the user-specified symbol to the interactive display at a position on the second	HTC infringes directly and/or indirectly by performing, inducing others to perform, and/or contributing to the performance of adding the user-specified symbol to the interactive display at a position on the second georeferenced map corresponding to the user-specified location. See, e.g., 1P and 1A above regarding the hardware and software for transmitting data over the Internet. See, e.g., 1C and 1D regarding the use of

**Exhibit D - Claim Chart for U.S. Patent No. 9,467,838 Against HTC**

<b>US9467838</b>	<b>Exemplary Supporting Evidence Regarding Accused Products</b>
georeferenced map corresponding to the user-specified location.	location services and location sharing to obtain and send locations over the Internet. For example, Android Device Manager, Find My Phone, Find My Device, Google Latitude, Google Plus, Google Hangouts, Google Maps, Google Assistant, Google Search, Google Messages, Android Messenger, Google Allo, Google Duo, Gmail, and Google Chrome use location services and location sharing to obtain locations and send/receive those locations to corresponding servers over the Internet for transmission to recipient devices/users. For example, receiving messages on the Accused Devices using Android Device Manager, Find My Phone, Find My Device, Google Latitude, Google Plus, Google Hangouts, Google Maps, Google Assistant, Google Search, Google Messages, Android Messenger, Google Allo, Google Duo, Gmail, and Google Chrome as associated services includes sending communication over IP. For example, a first device receives shared locations and markers from second devices and displays them on a map on the first device.
38. The method of claim 37, further comprising performing by the first device: identifying user interaction with the interactive display selecting the user-specified symbol corresponding to the entity, and based thereon, displaying information associated with the entity, wherein the user-specified information further includes the information associated with the entity.	HTC infringes directly and/or indirectly by performing, inducing others to perform, and/or contributing to the performance of the method of claim 37, further comprising performing by the first device: identifying user interaction with the interactive display selecting the user-specified symbol corresponding to the entity, and based thereon, displaying information associated with the entity, wherein the user-specified information further includes the information associated with the entity. See, e.g., 1P and 1A above regarding the hardware and software for transmitting data over the Internet. See, e.g., 1C and 1D regarding the use of location services and location sharing to obtain and send locations over the Internet. For example, Android Device Manager, Find My Phone, Find My Device, Google Latitude, Google Plus, Google Hangouts, Google Maps, Google Assistant, Google Search, Google Messages, Android Messenger, Google Allo, Google Duo, Gmail, and Google Chrome use location services and location sharing to obtain locations and send/receive those locations to corresponding servers over the Internet for transmission to recipient devices/users. For example, receiving messages on the Accused Devices using Android Device Manager, Find My Phone, Find My Device, Google Latitude, Google Plus, Google Hangouts, Google Maps, Google Assistant, Google Search, Google Messages, Android Messenger, Google Allo, Google Duo, Gmail, and Google Chrome and associated services includes sending communication over IP. For example, a first device receives shared locations and markers from second devices and displays them on a map on the first device. A user of the first device may then interact with the received shared location and/or marker and the map data and specify additional information.
39. The method of claim 1, wherein the message including the identifier	HTC infringes directly and/or indirectly by performing, inducing others to perform, and/or contributing to the performance of the method of claim 1, wherein the message including the identifier corresponding to the group is a first message, and wherein the method further comprises performing by the first device:

**Exhibit D - Claim Chart for U.S. Patent No. 9,467,838 Against HTC**

<b>US9467838</b>	<b>Exemplary Supporting Evidence Regarding Accused Products</b>
<p>corresponding to the group is a first message, and wherein the method further comprises performing by the first device: sending, to a particular second device via the first server, a second message related to remotely controlling the particular second device to perform an action, wherein the particular second device is configured to perform the action based on receiving the second message.</p>	<p>sending, to a particular second device via the first server, a second message related to remotely controlling the particular second device to perform an action, wherein the particular second device is configured to perform the action based on receiving the second message. See, e.g., 1H. The user of the first device sends an action by message, via corresponding servers, to remotely control the second device upon receiving a message from the server corresponding to the action by message.</p>
<p>40. The method of claim 39, wherein the second message indicates the action to be performed, and wherein the action is selected from the group consisting of playing audio, initiating a phone call, vibrating, converting text to speech, changing sound intensity, and displaying information.</p>	<p>HTC infringes directly and/or indirectly by performing, inducing others to perform, and/or contributing to the performance of the method of claim 39, wherein the second message indicates the action to be performed, and wherein the action is selected from the group consisting of playing audio, initiating a phone call, vibrating, converting text to speech, changing sound intensity, and displaying information. See, e.g., 1H. For example, a user may play a sound, initiate a call, and display information. Upon information and belief, the second device vibrates, converts text to speech, and changes sound intensity in accordance with the claim.</p>
<p>41. The method of claim</p>	<p>HTC infringes directly and/or indirectly by performing, inducing others to perform, and/or contributing to</p>

**Exhibit D - Claim Chart for U.S. Patent No. 9,467,838 Against HTC**

<b>US9467838</b>	<b>Exemplary Supporting Evidence Regarding Accused Products</b>
40, wherein playing audio comprises playing an audio message announcing an emergency.	the performance of the method of claim 40, wherein playing audio comprises playing an audio message announcing an emergency. See, e.g., 1H and 40. For example, a user may play a sound, initiate a call, and display information to announce an emergency, e.g. lost device.
42. The method of claim 1, wherein the message including the identifier corresponding to the group is a first message, and wherein the method further comprises performing by the first device: receiving a second message sent by a particular second device, wherein the second message indicates an action to be performed by the first device; and performing the indicated action.	HTC infringes directly and/or indirectly by performing, inducing others to perform, and/or contributing to the performance of the method of claim 1, wherein the message including the identifier corresponding to the group is a first message, and wherein the method further comprises performing by the first device: receiving a second message sent by a particular second device, wherein the second message indicates an action to be performed by the first device; and performing the indicated action. See, e.g., 1H. The first device, because it is running Android Device Manager, Find My Phone, Find My Device, Google Latitude, Google Plus, Google Hangouts, Google Maps, Google Assistant, Google Search, Google Messages, Android Messenger, Google Allo, Google Duo, Gmail, and Google Chrome, receives and performs actions identified in 1H.
43. The method of claim 42, wherein the indicated action is selected from the group consisting of playing audio, initiating a phone call, vibrating, converting text to speech, changing sound intensity, and displaying	HTC infringes directly and/or indirectly by performing, inducing others to perform, and/or contributing to the performance of the method of claim 42, wherein the indicated action is selected from the group consisting of playing audio, initiating a phone call, vibrating, converting text to speech, changing sound intensity, and displaying information. See, e.g., 1H. For example, a user may play a sound, initiate a call, and display information. Upon information and belief, the second device vibrates, converts text to speech, and changes sound intensity in accordance with the claim.

**Exhibit D - Claim Chart for U.S. Patent No. 9,467,838 Against HTC**

<b>US9467838</b>	<b>Exemplary Supporting Evidence Regarding Accused Products</b>
information.	
44[A]. The method of claim 1, further comprising performing by the first device: presenting another symbol on the second georeferenced map corresponding to a fixed location and associated with a telephone number;	HTC infringes directly and/or indirectly by performing, inducing others to perform, and/or contributing to the performance of the method of claim 1, further comprising performing by the first device: presenting another symbol on the second georeferenced map corresponding to a fixed location and associated with a telephone number. See, e.g., 1G and 1H. For example, the first device's map can display icons representing fixed entities which correspond to a location and a telephone number.
[44B] and receiving user selection of the other symbol and, based thereon, initiating a telephone call to the telephone number associated with the symbol.	HTC infringes directly and/or indirectly by performing, inducing others to perform, and/or contributing to the performance of receiving user selection of the other symbol and, based thereon, initiating a telephone call to the telephone number associated with the symbol. See, e.g., 1G and 1H. For example, the user of the first device can select the icon of a fixed entity to retrieve a telephone number and call the telephone number associated with the fixed entity. See, e.g., 1C and 1D for additional map and location data.
45. The method of claim 1, further comprising performing, by the first device: presenting a symbol corresponding to a facility, wherein the facility is selected from the group consisting of a hospital, a police station, and a fire station, and wherein the symbol corresponding to the	HTC infringes directly and/or indirectly by performing, inducing others to perform, and/or contributing to the performance of the method of claim 1, further comprising performing, by the first device: presenting a symbol corresponding to a facility, wherein the facility is selected from the group consisting of a hospital, a police station, and a fire station, and wherein the symbol corresponding to the facility is positioned on the second georeferenced map at a position corresponding to a location of the facility. See, e.g. 1H and 44A-B. See, e.g., 1C and 1D for additional map and location data.



**Exhibit D - Claim Chart for U.S. Patent No. 9,467,838 Against HTC**

<b>US9467838</b>	<b>Exemplary Supporting Evidence Regarding Accused Products</b>
facility is positioned on the second georeferenced map at a position corresponding to a location of the facility.	
46. The method of claim 45, further comprising performing, by the first device: identifying user interaction with the interactive display selecting the symbol corresponding to the facility, and based thereon, displaying information associated with the facility.	HTC infringes directly and/or indirectly by performing, inducing others to perform, and/or contributing to the performance of the method of claim 45, further comprising performing, by the first device: identifying user interaction with the interactive display selecting the symbol corresponding to the facility, and based thereon, displaying information associated with the facility. See, e.g., 1G and 1H. For example, the user of the first device can select the icon of a fixed entity to retrieve additional information about the entity, including a telephone number. See, e.g., 1C and 1D for additional map and location data.
47. The method of claim 46, wherein the information associated with the facility comprises a uniform resource locator (URL) of a web site associated with the facility.	HTC infringes directly and/or indirectly by performing, inducing others to perform, and/or contributing to the performance of the method of claim 46, wherein the information associated with the facility comprises a uniform resource locator (URL) of a web site associated with the facility. See, e.g., 1G and 1H. For example, the user of the first device can select the icon of a fixed entity to retrieve additional information about the entity, including a website's address. See, e.g., 1C and 1D for additional map and location data.
48. The method of claim 45, further comprising performing, by the first device: identifying user interaction with the interactive display	HTC infringes directly and/or indirectly by performing, inducing others to perform, and/or contributing to the performance of the method of claim 45, further comprising performing, by the first device: identifying user interaction with the interactive display selecting the symbol corresponding to the facility and user interaction with the display specifying an action, and based thereon, loading a web page associated with the facility. See, e.g., 1G and 1H. For example, the user of the first device can select the icon of a fixed entity to retrieve additional information about the entity, including a website's address for selection. See, e.g., 1C

**Exhibit D - Claim Chart for U.S. Patent No. 9,467,838 Against HTC**

<b>US9467838</b>	<b>Exemplary Supporting Evidence Regarding Accused Products</b>
selecting the symbol corresponding to the facility and user interaction with the display specifying an action, and based thereon, loading a web page associated with the facility.	and 1D for additional map and location data.
49[A]. The method of claim 1, further comprising performing by the first device: identifying user interaction with the interactive display selecting a subset of the user-selectable symbols corresponding to a subset of the one or more second devices included in the group;	HTC infringes directly and/or indirectly by performing, inducing others to perform, and/or contributing to the performance of the method of claim 1, further comprising performing by the first device: identifying user interaction with the interactive display selecting a subset of the user-selectable symbols corresponding to a subset of the one or more second devices included in the group. See, e.g., 1B. For example, Google Contacts may be grouped by lists. Groups made by linking or associating within Find My Device and Google Maps, taken separately, constitute subsets or subgroups of a group of contacts within the Accused Device. Within the Accused Device, if multiple accounts have been setup for Find My Device, each Account constitutes a subset of second devices. Moreover, groups and subsets may be identified in each of Android Device Manager, Find My Phone, Find My Device, Google Latitude, Google Plus, Google Hangouts, Google Maps, Google Assistant, Google Search, Google Messages, Android Messenger, Google Allo, Google Duo, GMail, and Google Chrome.
[49B] and identifying user interaction with the interactive display specifying an action and, based thereon, assigning the subset of second devices to a sub-net.	HTC infringes directly and/or indirectly by performing, inducing others to perform, and/or contributing to the performance of the identifying user interaction with the interactive display specifying an action and, based thereon, assigning the subset of second devices to a sub-net. See, e.g., [49A] above.
50[A]. The method of claim 49, further comprising performing	HTC infringes directly and/or indirectly by performing, inducing others to perform, and/or contributing to the performance of the method of claim 49, further comprising performing by the first device: identifying user interaction with the interactive display selecting the sub-net and user interaction with the display

**Exhibit D - Claim Chart for U.S. Patent No. 9,467,838 Against HTC**

<b>US9467838</b>	<b>Exemplary Supporting Evidence Regarding Accused Products</b>
by the first device: identifying user interaction with the interactive display selecting the sub-net and user interaction with the display specifying an action;	specifying an action. See, e.g., 1C and 49A above. For example, a user of a first device can initiate a call with a member or members of the grouped identified.
[50B] and based thereon, sending fourth data to the subset of second devices via the first server or initiating a phone conference with the subset of second devices.	HTC infringes directly and/or indirectly by performing, inducing others to perform, and/or contributing to the performance of, based thereon, sending fourth data to the subset of second devices via the first server or initiating a phone conference with the subset of second devices. See, e.g., 50A above. For example, communication to the recipient devices includes sending data to a server. See, e.g., 1A, 1C, 1D, and 1H above.
51. The method of claim 1, wherein the first server is the second server.	HTC infringes directly and/or indirectly by performing, inducing others to perform, and/or contributing to the performance of the method of claim 1, wherein the first server is the second server. See, e.g., 1A, 1C, 1D, and 1H above, including a server corresponding to Android Device Manager, Find My Phone, Find My Device, Google Latitude, Google Plus, Google Hangouts, Google Maps, Google Assistant, Google Search, Google Messages, Android Messenger, Google Allo, Google Duo, Gmail, and Google Chrome and associated services..
52. The method of claim 1, wherein the first set of second devices and the second set of second devices are identical.	HTC infringes directly and/or indirectly by performing, inducing others to perform, and/or contributing to the performance of the method of claim 1, wherein the first set of second devices and the second set of second devices are identical. See, e.g., 1A, 1F, and 1G, where the first and second sets of second devices can be the same.
53. The method of claim 1, wherein the message further includes an identifier corresponding to the first device.	HTC infringes directly and/or indirectly by performing, inducing others to perform, and/or contributing to the performance of the method of claim 1, wherein the message further includes an identifier corresponding to the first device. See, e.g., 1A and 1B, where the message includes the identifier. Upon information, additional metadata associated with the Accused Device is included within the message.

**Exhibit D - Claim Chart for U.S. Patent No. 9,467,838 Against HTC**

<b>US9467838</b>	<b>Exemplary Supporting Evidence Regarding Accused Products</b>
54[P]. A system comprising:	HTC directly and/or indirectly infringes by providing a system [comprising].  See 1P above.
[54A] a first device programmed to perform operations comprising:	HTC directly and/or indirectly infringes by a first device programmed to perform operations [of claim 54].  See 1A above.
[54B] joining a communication network corresponding to a group, wherein joining the communication network comprises transmitting a message including an identifier corresponding to the group;	HTC directly and/or indirectly infringes by joining a communication network corresponding to a group, wherein joining the communication network comprises transmitting a message including an identifier corresponding to the group.  See 1B above.
[54C] participating in the group, wherein participating in the group includes sending first location information to a first server and receiving second location information from the first server, the first location information comprising a location of the first device, the second location information comprising	HTC directly and/or indirectly infringes by: participating in the group, wherein participating in the group includes sending first location information to a first server and receiving second location information from the first server, the first location information comprising a location of the first device, the second location information comprising one or more locations of one or more respective second devices included in the group.  See 1C above.

**Exhibit D - Claim Chart for U.S. Patent No. 9,467,838 Against HTC**

<b>US9467838</b>	<b>Exemplary Supporting Evidence Regarding Accused Products</b>
one or more locations of one or more respective second devices included in the group;	
[54D] presenting, via an interactive display of the first device, a first interactive, georeferenced map and a first set of one or more user-selectable symbols corresponding to a first set of one or more of the second devices, wherein the first set of symbols are positioned on the first georeferenced map at respective positions corresponding to the locations of the first set of second devices, and wherein first georeferenced map data relate positions on the first georeferenced map to spatial coordinates;	<p>HTC infringes directly and/or indirectly by presenting, via an interactive display of the first device, a first interactive, georeferenced map and a first set of one or more user-selectable symbols corresponding to a first set of one or more of the second devices, wherein the first set of symbols are positioned on the first georeferenced map at respective positions corresponding to the locations of the first set of second devices, and wherein first georeferenced map data relate positions on the first georeferenced map to spatial coordinates.</p> <p>See 1D above.</p>
[54E] sending, to a second server, a request for second georeferenced map data different from the first georeferenced map data;	<p>HTC infringes directly and/or indirectly by: sending, to a second server, a request for second georeferenced map data different from the first georeferenced map data.</p> <p>See 1E above.</p>

**Exhibit D - Claim Chart for U.S. Patent No. 9,467,838 Against HTC**

<b>US9467838</b>	<b>Exemplary Supporting Evidence Regarding Accused Products</b>
[54F] receiving, from the second server, the second georeferenced map data;	<p>HTC infringes directly and/or indirectly by receiving, from the second server, the second georeferenced map data.</p> <p>See 1F above.</p>
[54G] presenting, via the interactive display of the first device, a second georeferenced map and a second set of one or more user-selectable symbols corresponding to a second set of one or more of the second devices, wherein the second set of symbols are positioned on the second georeferenced map at respective positions corresponding to the locations of the second set of second devices, and wherein the second georeferenced map data relate positions on the second georeferenced map to spatial coordinates;	<p>HTC infringes directly and/or indirectly by presenting, via the interactive display of the first device, a second georeferenced map and a second set of one or more user-selectable symbols corresponding to a second set of one or more of the second devices, wherein the second set of symbols are positioned on the second georeferenced map at respective positions corresponding to the locations of the second set of second devices, and wherein the second georeferenced map data relate positions on the second georeferenced map to spatial coordinates.</p> <p>See 1G above.</p>
[54H] and identifying user interaction with the interactive display selecting one or more of the second set of user-	<p>HTC infringes directly and/or indirectly by identifying user interaction with the interactive display selecting one or more of the second set of user-selectable symbols corresponding to one or more of the second devices and positioned on the second georeferenced map and user interaction with the display specifying an action and, based thereon, sending third data to the selected one or more second devices via the first server.</p>

**Exhibit D - Claim Chart for U.S. Patent No. 9,467,838 Against HTC**

<b>US9467838</b>	<b>Exemplary Supporting Evidence Regarding Accused Products</b>
selectable symbols corresponding to one or more of the second devices and positioned on the second georeferenced map and user interaction with the display specifying an action and, based thereon, sending third data to the selected one or more second devices via the first server.	See 1H above.