UNITED STATES PATENT AND TRADEMARK OFFICE

Applicant(s): Rempell et al.

Application No.: TBD

Filed: HEREWITH

Title: SYSTEMS AND METHODS FOR PROGRAMMING MOBILE DEVICES

Group Art Unit: TBD

Examiner: TBD

Docket No.: XPR.002US0C8

PRELIMINARY AMENDMENT AS PART OF THE ORIGINAL DISCLOSURE UNDER 37 CFR §1.115(a)(1),

Commissioner for Patents
P.O. Box 1450
Alexandria, VA 22313-1450

Dear Commissioner:

The following amendments are presented with the filing of the present application, where:

Amendments to the Claims are reflected in the **Listing of Claims** and begin on the following page of this paper; and

Remarks/Arguments begin on the page following the **Amendments to the Claims**.

Amendments to the Claims

This listing of claims will replace all prior versions, and listings, of the claims in the application:

Listing of Claims:

Please amend this application as follows:

Claim 1 (Currently Amended): A system for generating code to provide content on a display of a device, said system comprising:

~~a database of web services obtainable over a network;~~

computer memory storing:

a) symbolic names required for evoking one or more web components each related to a set of inputs and outputs of a web service obtainable over a network, where the symbolic names are character strings that do not contain either a persistent address or pointer to an output value accessible to the web service, where each symbolic name has an associated data format class type corresponding to a subclass of User Interface (UI) objects that support the data format type of the symbolic name, and where each symbolic name has a preferred UI object, and

b) an address of the web service;

an authoring tool configured to:

define ~~an~~a UI object for presentation on the display,

~~select a~~ where said defined UI object corresponds to a web component ~~of a web service~~ included in said ~~database,~~

~~associate said object with said~~ computer memory selected ~~component,~~ from a group consisting of an input of the web service and an output of the web service, where each defined UI object is either:

~~produce device-specific code that, when executed on the device,~~ ~~provides said~~ 1) selected by a user of the authoring tool; or

2) automatically selected by the system as the preferred UI object corresponding to the symbolic name of the web component selected by the user of the authoring tool,

access said computer memory to select the symbolic name corresponding to the web component ~~on~~of the defined UI object,

associate the selected symbolic name with the defined UI object, where the selected symbolic name is only available to UI objects that support the defined data format associated with that symbolic name,

store information representative of said defined UI object and related settings in a database;

retrieve said information representative of said one or more said UI object settings stored in said database; and

build an application consisting of one or more web page views from at least a portion of said database utilizing at least one player, where said player utilizes information stored in said database to generate for the display of at least a portion of said one or more web pages,

wherein when the application and player are provided to the device and executed on the device, and

when the user of the device provides one or more input values associated with an input symbolic name to an input of the defined UI object, the device provides the user provided one or more input values and corresponding input symbolic name to the web service, the web service utilizes the input symbolic name and the user provided one or more input values for generating one or more output values having an associated output symbolic name,

and the player receives the output symbolic name and corresponding one or more output values and provides instructions for the display of the device to present an output value in the defined UI object.

Claim 2 (Currently Amended): The system of claim 1, where said ~~database~~system stores information in a registry, and wherein the registry includes definitions of input and~~/or~~ output related to said web service.

Claim 3 (Currently Amended): The system of claim 1, where said web component is a text chat, a video chat, an image, a slideshow, a video, or an RSS feed.

Claim 4 (Currently Amended): The system of claim 1, where said UI object is an input field for a chat.

Claim 5 (Currently Amended): The system of claim 1, where said UI object is an input field for a web service.

Claim 6 (Currently Amended): The system of claim 1, where said UI object is an input field usable to obtain said web component, where said input field includes a text field, a scrolling text box, a check box, a drop down-menu, a list menu, or a submit button.

Claim 7 (Currently Amended): The system of claim 1, where said web component is an output of a web service, is the text provided by one or more simultaneous chat sessions, is the video of a video chat session, is a video, an image, a slideshow, an RSS display, or an advertisement.

Claim 8 (Original): The system of claim 1, where said authoring tool is further configured to:

define a phone field or list; and

generate code that, when executed on the device, allows a user to supply a phone number to said phone field or list.

Claim 9 (Original): The system of claim 1, where said authoring tool is further configured to:

define a SMS field or list; and

generate code that, when executed on the device, allows a user to supply an SMS address to said SMS field or list.

Claim 10 (Currently Amended): The system of claim 1,

where said code includes ~~two~~three or more codes, where one of said ~~two~~three or more codes is device specific, and where ~~one~~two of said ~~two~~three or more codes ~~in~~is device independent.

Claim 11 (Original): The system of claim 1, where said code is provided over said network.

Claim 12 (Currently Amended): A method ~~for~~of displaying content on a display of a device ~~utilizing a database of~~ having a player and non-volatile computer memory storing symbolic names required for evoking one or more web ~~services~~components each related to a set of inputs and outputs of a web service obtainable over a network, where the symbolic names are character strings that do not contain either a persistent address or pointer to an output value accessible to the web service, where each symbolic name has an associated data format class type corresponding to a subclass of User Interface (UI) objects that support the data format type of the symbolic name, and where each symbolic name has a preferred UI object, and an address of the web service, said method comprising:

defining ~~an~~a UI object for presentation on the display~~;~~, where said UI object corresponds to a web component included in the computer memory, where said web

component is selected from a group consisting of an input of a web service and an output of the web service, where each defined UI object is either: 1) selected by a user of the authoring tool; or 2) automatically selected by the system as the preferred UI object corresponding to a symbolic name of the web component selected by the user of the authoring tool;

selecting ~~a~~ the symbolic name corresponding to the web component of ~~a web service included in said database~~ the defined UI object;

associating ~~said object~~ the selected symbolic name ~~with said selected component; and~~ with the defined UI object, where the selected symbolic name is only available to UI objects that support the defined data format associated with that symbolic name;

~~producing device-dependent code that, when executed on the device, provides said selected component on the display of the device.~~

storing information representative of said defined UI object and related settings in a database;

retrieving said information representative of said one or more said UI object settings stored in said database; and

building an application consisting of one or more web page views from at least a portion of said database utilizing the player, where said player utilizes information stored in said database to generate for the display of at least a portion of said one or more web pages,

wherein, when the application and player are provided to the device and executed on the device, and when the user of the device provides one or more input values associated with an input symbolic name to an input of the defined UI object, 1) the device provides the user provided one or more input values and corresponding input symbolic name to the web service, 2) the web service utilizes the input symbolic name and the user provided one or more input values for generating one or more output values having an associated output symbolic name, and 3) the player receives the output symbolic name and corresponding one or more output values and provides instructions for the display of the device to present an output value in the defined UI object.

DOCKET ALARM

Explore Litigation Insights

Docket Alarm provides insights to develop a more informed litigation strategy and the peace of mind of knowing you're on top of things.

Real-Time Litigation Alerts

Keep your litigation team up-to-date with **real-time alerts** and advanced team management tools built for the enterprise, all while greatly reducing PACER spend.

Our comprehensive service means we can handle Federal, State, and Administrative courts across the country.

Advanced Docket Research

With over 230 million records, Docket Alarm's cloud-native docket research platform finds what other services can't. Coverage includes Federal, State, plus PTAB, TTAB, ITC and NLRB decisions, all in one place.

Identify arguments that have been successful in the past with full text, pinpoint searching. Link to case law cited within any court document via Fastcase.

Analytics At Your Fingertips

Learn what happened the last time a particular judge, opposing counsel or company faced cases similar to yours.

Advanced out-of-the-box PTAB and TTAB analytics are always at your fingertips.

API

Docket Alarm offers a powerful API (application programming interface) to developers that want to integrate case filings into their apps.

LAW FIRMS
Build custom dashboards for your attorneys and clients with live data direct from the court.

Automate many repetitive legal tasks like conflict checks, document management, and marketing.

FINANCIAL INSTITUTIONS
Litigation and bankruptcy checks for companies and debtors.

E-DISCOVERY AND LEGAL VENDORS
Sync your system to PACER to automate legal marketing.