Wireless Devices for Mobile Commerce: User Interface Design and Usability

Peter Tarasewich, College of Computer and Information Science, Northeastern University

Appears in Mobile Commerce: Technology, Theory, and Applications, 2002,
Brian E. Mennecke and Troy J. Strader, (Eds.) Hershey, PA: Idea Group Publishing, pp. 26-50.

INTRODUCTION

An increasing number of technologies and applications have begun to focus on mobile computing and the wireless Web. *Mobile commerce* (m-commerce) encompasses all activities related to a (potential) commercial transaction conducted through communications networks that interface with wireless (or mobile) devices (Tarasewich, Nickerson, and Warkentin, 2001). Ultimately, researchers and developers must determine what tasks users really want to perform anytime from anywhere and decide how to ensure that information and functionality to support those tasks are readily available and easily accessible.

A well-designed and usable interface to any application is critical. For example, properly designed websites help ensure that users can find information that they are looking for, perform transactions, spend time at the site, and return again. Given the uniqueness of the wireless environment, usability becomes even harder to ensure for m-commerce applications. The purpose of this chapter is to provide the reader with an overview of current wireless device interface technologies. It will provide guidance on designing usable m-commerce applications that take advantage of the benefits and respect the limitations of these devices. This chapter will also explore the interface design and usability challenges that the m-commerce environment still presents for users, researchers, and developers.

This chapter is organized as follows. The first section describes the benefits and limitations of various wireless device interfaces. The next section looks at how the usability of wireless devices affects the feasibility and success of m-commerce applications. The third section discusses some of the additional challenges that developers face when designing applications for wireless devices. The final section reiterates the need for good wireless application design, and describes some of the safety and security issues related to wireless device interface design.

WIRELESS DEVICES AND THEIR INTERFACES

The devices currently most important to m-commerce can be classified according to the categories listed in Table 1. There is some feeling that devices will become completely generic, and take the place of items like televisions, pagers, radios, and telephones (Dertouzos, 1999), but the question remains as to what form the devices will ultimately take. This important issue will be investigated further in the section on mobile system developer issues later in the chapter. But first we look at the current interfaces of these devices, their strengths, and their limitations. The discussion is separated into input and output interactions. Research that has been performed with various types of interface devices will be discussed in the next section on usability.

Table 1 – Wireless Device Categories
Laptop Computer
Handheld (e.g., Palm, Pocket PC, Blackberry)
Telephone
Hybrid (e.g. “smartphone” PDA/telephone combination)
Wearable (e.g., jewelry, watches, clothing)
Vehicle Mounted (in automobiles, boats, and airplanes)
Specialty (e.g., the now defunct Modo)

Input Interaction with Wireless Devices

Input interaction concerns the ways in which users enter data or commands. Common technologies used for input interaction with wireless devices include keyboards, keypads, styluses, buttons, cameras, microphones, and scanners. Each of these will be discussed in turn, emphasizing the benefits and limitations of each in the mobile environment.

The keyboard still remains popular as a form of input for many types of computing devices. The QWERTY configuration of keys (named for the sequence of keys at the upper left of the keyboard), while not the most efficient layout possible, remains a standard because of its wide user acceptance. Laptop computers have carried the concept of QWERTY keyboards forward, although keys are usually made smaller to conserve room. Devices such as phones and handhelds, however, have generally foregone the integration of a full keyboard because of the desire to create a device that is as small and light as possible. The exception to this is the Blackberry device, which includes a miniature keyboard. The problem with this keyboard is that a user must adjust to smaller keys, oftentimes learning to type messages with both thumbs. Data entry and error rates can suffer with smaller keys as well.

Smaller mobile devices usually rely on a more limited keypad for input. Most mobile phones use a standard 12-button numeric keypad, sometimes augmented by several special purpose keys (such as “clear” and “ok”). Each of the keys 2 through 9 also corresponds to a set of three or four letters. There are several approaches to entering text using a keypad. In the first, known as the multi-press input method, the user must hit a numeric key that also corresponds to the desired letter. For example, the letter “s” would require that the “7” key (labeled with “pqrs”) be depressed four times. A capital “S” would then require eight or more keystrokes. A user must also pause or press an additional key to move onto the next letter. A different method that uses two-key input requires selecting a letter’s group with the first key press and the location of the desired key with the second. For example, the letter ‘E’ (the second character on the “3” key which is labeled “def”) requires the key press sequence 3-2. Another approach uses dictionaries of words and linguistic models to “guess” the word intended by a series of keystrokes. For example, the sequence 8-4-3 (corresponding to “tuv”-“ghi”-“def”) might produce the word “the” out of all possible letter combinations.

One way to eliminate the use of a keypad for text entry is to attach a temporary keyboard to the device being used. Several vendors have developed miniature and/or full-size folding keyboards for this purpose. A more radically designed alternative is the Matias Half Keyboard (Figure 1), which contains only those keys from the left-hand side of a traditional keyboard. When the space bar is pressed, the same keys function as the right-hand side. Another alternative is a fabric keyboard, being developed by ElectroTextiles, that can be rolled up for storage (Figure 2). Researchers are also developing “non-keyboards” in the form of gloves (Goldstein et al., 1999) or “finger rings” (Fukumoto and Tonomura, 1997) that sense finger movements of users typing on a virtual keyboard and use software to interpret the movements. Essential Reality (www.essentialreality.com) is producing a glove called P5 that can be programmed to respond to users’ hand gestures with combinations of keystrokes and mouse clicks. A potential problem with these types of devices is the additional training time that might be needed to use the device effectively.

Another way to eliminate the use of a keypad (and keyboard as well) is to use a stylus to write input directly on the screen of the device, a process known as gesture recognition. With this method, the device must recognize each character or symbol that is written, which can take a good deal of processing time and oftentimes suffers from inaccuracy. Palm has developed a proprietary system for character recognition (called Graffiti) that seems more accurate than other recognition systems, but forces the user to conform to a writing style for letters that is somewhat different than normal. Another gesture recognition technique is Jot (often used with Pocket PC devices). In both cases, the user must learn which pen strokes represent a particular character to the device, rather than the device interpreting the handwriting of the user. As an alternative to keypads, Smart

Design (www.smartdesign.com) is developing a system called Thumbscript that replaces a keypad on phones with a nine-point grid. Users tap a keystroke sequence on the grid for each character (Roman letter or Asian character) that they wish to input.

As an alternative to gesture recognition, keyboards (or other key configurations) can be created virtually on a screen, with each key being "pushed" by touching it with a stylus. These so-called "soft-keyboards" are sometimes implemented in sections (e.g., the alphabetic characters separated from numbers and other characters) to save screen space and create larger keys. Styluses can also be used to activate icons, menu choices, or hyperlinks displayed on a screen. Virtual keyboards currently suffer from a lack of tactile feedback often found on keyboards and some keypads, although feedback can be provided through sounds generated as keys are "pressed."

Mobile device input can also be achieved through "mouse buttons," thumbwheels, and other special-purpose buttons. The user interface of the telematics system OnStar consists of just three buttons, labeled "call", "help", and "off". Mobile phones often have dedicated buttons with labels such as "call", "ok", and "clear" in addition to a numeric keypad. Mouse buttons are toggle switches that allow one-dimensional cursor movement. An alternative to a mouse button is the "navi-roller," which allows scrolling by rolling and selection by clicking. Small joysticks, which allow two-dimensional cursor movement, are sometimes found integrated into the keyboards of laptop computers, and more recently on mobile phones. Handheld devices usually have a mouse button and a few other special-purpose buttons, but no keyboard or keypad. CyMouse by Maui Innovative Peripherals (maui-innovative.com) is an eight-ounce headset that acts as a wireless mouse. A version called Miracle Mouse is aimed at providing more control options to people with physical disabilities. The now defunct Modo device (Figure 3), which featured one-handed operation, had a thumbwheel to move between selections and to scroll text up and down. Pressing the wheel activated the current selection. Some other handheld devices also feature a similar built-in thumbwheel. However, the location of the thumbwheel limits which hand can hold the device for one-handed operation.

Using human speech as input to mobile devices is also becoming increasingly practical as voice recognition technology continues to improve. Whether or not voice interfaces will ultimately succeed as a primary form of input depends on how well certain limitations of the technology can be overcome. These limitations include the need to train devices to recognize a user's voice, the relative slowness of voice versus other input means, and the difficulty in using visual information (e.g., graphics) with voice input. Benefits of voice input include the ability of users to interact with the device in their natural language. Voice input allows those users who cannot type or use a stylus to interact with a device. It may also be a viable interface alternative for devices too small for buttons or for those without a screen. However, voice input suffers from possible privacy and social issues. For example, users may feel uncomfortable speaking input aloud instead of typing or writing it, and certain places (e.g., libraries) might restrict the use of voice input to maintain a quiet environment. One option that allows a voice interface with mobile devices, but does not require direct Internet access from the mobile device, is Voice Extensible Markup Language (VXML). This standard allows consistent access to Web applications from both the wired and wireless environments.

With the shrinking size of camera lenses and the increasing sophistication of digital photography, video is becoming more common as a form of input with mobile devices. Some laptops, phones, and handheld devices have built-in or attachable cameras. DoCoMo has been developing specialized mobile Internet appliances, some of which are cameras that can take pictures, adorn them with overlays, and send them to users with similar devices or i-mode phones. Video might also be used as input through the recognition of hand gestures or facial expressions.

Similarly, scanners may also become part of the wireless environment. They can be used for reading text, bar codes, or other symbols. Wireless devices that scan UPC symbols as input could be part of in-store mobile commerce applications used for comparison-shopping or for purchasing merchandise without the need of a cash register and sales attendant.

Finally, input can come from technologies that sense location, or from those that can receive information from their environment based on their location. The Global Positioning System (GPS), a set of satellites owned and operated by the U.S. Department of Defense, allows any device equipped with a GPS receiver to determine its geographic location within about 10 meters. All mobile phones sold in the U.S. will be required to have the ability to determine their location. Bluetooth technology, which allows short-range communications, will allow mobile devices to receive information automatically when they are in close proximity of another Bluetooth-equipped device. As we will discuss later, location is a key factor in designing useable mobile applications. However, privacy issues dealing with the use of location data must also be addressed.

Output Interaction with Wireless Devices

Output interaction concerns the ways in which users receive data, prompts, or the results of a command. Common technologies used for output interaction with wireless devices include video screens and speakers. Both of these will be discussed in turn, emphasizing the benefits and limitations of each in the mobile environment.

The liquid crystal display (LCD) screen is the primary technology used to produce output in the form of images and text on current wireless devices. Screen size varies greatly from one type of device to another. Most mobile phones have small (1" to 2" square) screens that can display 4 to 8 lines of 10 to 20 alphanumeric characters each. Handheld devices have relatively larger screens (about 3" by 4") that are more suitable for graphics as well as text, but are still limited by low screen resolutions (usually 240 by 320 pixels). Most phones and handhelds have monochrome screens, although more are being sold with color screens, which can increase device usability. Laptops have fairly large color screens (up to 15" diagonal) with resolutions that compare favorably to desktop monitors. Vehicle-mounted devices have screens ranging from smaller than the size found on phones to the size found on small laptops, depending on the intended purpose of the device (e.g., displaying song titles versus a map of a city).

The current limitations of screens on wireless devices are their size, resolution, and color capabilities, all of which are usually less than those found on desktop computers. These limitations make it difficult to display large amounts of text and graphic-based output (e.g., maps, charts, or Web pages). There are also tradeoffs in improving the screen characteristics of mobile devices. Increasing screen size will increase the size and weight of a device. Color screens with high resolutions use more power than their monochrome counterparts, resulting in increased battery weight and/or less time before the battery needs to be recharged (although research into better batteries continues).

There are, however, some recent technological developments that may address some of the disadvantages of current wireless device screens. Flexible screens are on the horizon, which may eventually allow screens that can be rolled or folded up. E Ink (www.eink.com) and Gyricon Media (www.gyriconmedia.com) are developing displays with electronic ink technology (e-paper), first in black and white, but possibly in color in the future. The screens hold an image until voltage is applied to produce a new image, using less overall power than LCD screens.

Monocular units or goggles can be used with magnifying glasses to enlarge small displays (less than an inch diagonal) so that they look like an 800 x 600 resolution monitor. Goggle-type products include InViso's eShade (www.inviso.com/products), Sony's Glasstron (www.ita.sel.sony.com/products/av/glasstron), and Olympus' Eye-Trek (www.eye-trek-olympus.com, see Figure 4). Microvision (www.mvis.com) is developing a device that projects an image, pixel by pixel, directly onto the viewer's retina. Heads-up displays, which have seen limited use in automobiles in the past, might also be used for vehicle-mounted devices. These types of devices allow viewing of color images with similar sizes and resolutions as those found on desktop computers. Potential concerns with these technologies include interference with users' other visual inputs, and the social acceptance of wearing and using such technologies.

Sound is the other primary form of output from a wireless device. Forms of this output range from words to music to various beeps, buzzes, and other noises. These can be created through speakers or through headphones. Newer laptops usually have a set of speakers built in for stereo sound production. Most smaller mobile devices have a single speaker at best. Stereo speakers can be used to generate sounds coming from a particular direction, which as we shall see later can be used to enhance usability. This same effect can be achieved through headphones, but at the cost of possible interference with a user's other audio input (i.e., sounds from the environment).

Sound output may be a viable interface alternative for devices without a screen, although there may be difficulties in presenting certain visual information (e.g., graphics). Voice output is also generally produced and comprehended slower than visual output. On the positive side, sound allows those users who cannot see a screen to receive output. Ultimately, it may be that multi-modal browsing, where voice and visual output are combined, may be best suited for wireless devices (Nah and Davis, 2001).

WIRELESS DEVICE USABILITY

This section looks at the usability of wireless devices and how usability affects the feasibility and success of m-commerce applications. Some of the recent research on interface design and usability for mobile and wireless devices will be discussed, along with usability issues present with wireless devices. The section will also consider whether or not current HCI standards can be applied to wireless devices, and what further research issues regarding the usability of wireless devices need to be addressed.

Usability can be defined as the quality of a system with respect to ease of learning, ease of use, and user satisfaction (Rosson and Carroll, 2002). It also deals with the potential of a system to accomplish the goals of the user. *Usability testing* asks users to perform certain tasks with a device and application while recording measures such as task time, error rate, and the user's perception of the experience. Methods for evaluating usability include empirical testing, heuristic evaluations, cognitive walkthroughs, and analytic methods such as GOMS (goals, operators, methods, and selection rules).

Many of these same usability methods can be applied successfully in order to test the usability of a particular application on a device, or compare usability across different devices or configurations. Chan and Fang (2001) reported on research in progress that is conducting a heuristic evaluation and cognitive walkthrough of fifteen m-commerce sites across three different device platforms (Palm, Pocket PC, and WAP phone). Their preliminary results indicate that many Web sites are trying to duplicate their wired Web architecture and design for the wireless Web, resulting in poor navigation and information overload.

Likewise, many of the current principles of interface design can be transferred to newer devices, although soundly applying these principles may be more difficult due to the unique nature of mobile systems and devices. Fundamental rules such as consistency, shortcuts for advanced users, the use of feedback, error prevention, easy reversal of actions, and minimization of short-term memory requirements (Shneiderman, 1998) will undoubtedly transfer to mobile applications. However, as shown in the previous section, the devices that the user might interact with are quite different than the desktop computers used in much of the interface design research to date. While further study is needed, it is likely that much of the specific research on effective screen design and information output cannot be generalized to mobile devices.

Furthermore, *context* will factor heavily into the use of mobile applications and devices, which is something that was not as much (if any) of a concern with stationary desktop applications. Mobile tasks and technology use are significantly different than their stationary counterparts. People can now literally be anywhere at anytime and use a mobile application, which was not true with the traditional (wired) Web since a physical connection was needed to the Internet. Location will need to be factored into the usability of an application and a device, as will the dynamic nature of the environment within which it is used. Conceivably, a mother could be walking down a street in

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.

WHAT WILL YOU BUILD? | sales@docketalarm.com | 1-866-77-FASTCASE

fastcase®
Smarter legal research.