Designing Applications for Mobile Phones: The Mobile Subject Assistant

Sam Jebeile
University of Wollongong

School of Information Technology and Computer Science
University of Wollongong
New South Wales, Australia
Email: sj74@uow.edu.au

Abstract

The rapid adoption rate of mobile phones coupled with advancements in supporting technologies suggests mobile phones may now be a viable tool for classroom use. In this paper we introduce the Mobile Subject Assistant, a proposed mobile phone based course tool. The tool is intended to assist in the delivery of university level subjects with an emphasis on student participation, collaboration and group or project work. We discuss a number of issues related to the design of interfaces for mobile phone applications and briefly describe the design of the interface to the Mobile Subject Assistant.

Keywords

Mobile technologies, mobile phones, collaborative learning, course tool.

INTRODUCTION

Mobile phone technology has developed rapidly over the past decade and new age mobile phones now support much of the software that was once restricted to Personal Digital Assistants (PDAs) and Personal Computers (PCs). With facilities such as instant text and multimedia messaging, e-mail, sound and video recording, streaming video playback and personal organisation features, not only are mobile phones becoming faster and more efficient, their size and aesthetics are appealing to a new generation of users. Consequently, mobile phones are no longer limited to users in the corporate sector.

The Australian Mobile Telecommunications Association reported that at least 65% of Australians were using mobile phones in 2003 (AMTA 2003). In addition it has been estimated that 85% of College students in the United States will have mobile phones in 2005 (Wuthrich, Kalbfleisch et al. 2003). It is therefore reasonable to assume that a large majority of students in any one course will soon have a mobile phone.

New age mobile phones provide some of the functionality of a PC with the added benefits of being familiar to the user, being fashionable, supporting the "anywhere, anytime" concept and in some cases being considerably smaller that their PDA counterparts. PDAs have been sometimes touted as a possible mobile classroom solution (Roschelle and Pea 2002; Corlett, Sharples et al 2004; Cortez, Nussbaum et al 2004), but have failed to enter mainstream use. One shortcoming is that, although PDAs support many of the facilities of mobile phones and more, they have been scarcely adopted by the general population.

Due to the nature of mobile phones, developing software for use on them has many limitations. Issues such as limited screen size, text input, speed and storage are important factors that must be considered. This paper firstly identifies some of the user interface (UI) issues associated with developing a mobile phone application. Secondly the paper describes the Mobile Subject Assistant, a proposed course tool being developed specifically for use on mobile phones. Finally the UI issues faced in the design of the application are discussed.

DESIGNING FOR MOBILE PHONES

When developing applications specifically for mobile phones, developers must consider two distinct aspects that affect the user interface: the physical operation of the device and usability issues associated with the application itself. Factors that will physically affect the way in which the application operates, include, for example, the user interface and modes of text input. In addition mobile phone usage trends cannot be ignored. If any of these aspects is ignored, it may result in an application that is easy to use on a mobile phone, but users have no reason to use it, or an application that users would find useful, but is difficult to use on a mobile phone.

Kaikkonen and Roto (2003) list the following general limitations of mobile phones compared to other systems, such as PCs:

- Display size is small
- There is a lot of variation in display dimensions

- The number of color displays has just started to grow
- Text input is slower than with a full PC keyboard
- Usually there is no mouse for activating an object. This limits the possible user interface components and slows down object activation
- Some devices support only vertical scrolling
- Softkeys are commonly used for activating commands, but the number and purpose of softkeys varies between devices from different manufacturers
- Accesskey functionality allows users to activate commands also with the phone's number keys
- Data transfer between the terminal and the server is slow
- The amount of cookie data that can be stored in a mobile device is very limited
- Context of use is harder to predict than with an office PC application
- The user may have to pay for each piece of data.

The remainder of this section discusses the significance of some of these limitations with respect to the design of the visual display, input mechanisms, navigation and useability of a mobile phone application.

Visual display

The screen size of a mobile phone is one of its most limiting factors. Only so much information can be displayed on the screen, with some newer mobile phones supporting resolutions of up to 176 x 208 pixels whilst older or smaller phones support lower resolutions. Not only is screen size a factor, but colour also plays a major role. Most current application development targets users with a colour screen, with the level of colour depth varying. Thus monochrome phones are not considered in this paper.

Display Size

Applications should be developed supporting as low a resolution as possible. The Siemens SL55 is an example small-screen phone that supports a resolution of 101 x 80 pixels (Siemans, 2004). It's lower resolution means both that less information can be displayed and the information that is displayed could be displayed much smaller than the designer intended. It is also important to note that the phone operating system may take up space, as seen in Figure 1. The Operating System (OS) menus take up approximately 30% of the screen. If an application is being developed with the operating system screens visible, a significant amount of space is lost. Most custom applications, however, will use the entire screen, replacing any OS menus with application based menus.

Figure 1: Siemens SL55 Screen

Colour

Colour phones currently support two main colour depths on their primary display screen: 12 bit colour (4096 colours) and 16 bit colour (65536 colours). As mobile phones are still being produced using a 12 bit colour depth, it is important for applications to support such colour. Applications should be tested on phones (or emulators) with both colour depths to ensure that the desired effect is still being conveyed in either circumstance.

Displaying Information

Masoodian and Lane (2003) performed a study on displaying travel information in both text and graphical formats. Their findings showed that, overall, users' familiarity with textual displays meant that they preferred a textual display over a graphical display. The results of this study indicate that, although the possibility of displaying graphics is available, it may be necessary to allow for textual navigation in order to make the user more comfortable.

Input

Considering the large variety of phones available today, it is no surprise that there are a number of input varieties available. These include visual input (images and videos), keypad or navigational input and, of course, oral input. Each input method has a number of implications for application design.

The main method of input on a mobile phone is in the form of text via its keypad. The most common method of text input is the MultiTap method, whereby a user presses a button on the keypad multiple times to display a single letter, with each button on the keypad supporting multiple letters. For example, the '2' button on a standard keypad would map to the letters 'a', 'b' and 'c'. If a user desired to press 'a', they would press the '2' button once; 'b' would involve pressing the '2' button twice, and so on. As the average number of keypresses to input 7 words using the common MultiTap method is 70 (Wigdor and Balakrishnan 2003), the amount of text input should be limited to that which is absolutely necessary. This is less critical if the application is to support an alternate (and faster) means of inputting text, such as the TiltText method developed by Wigdor and Balakrishnan (2003).

Navigation

The main form of navigation on a mobile phone is also via the keypad and/or additional navigational tools such as a joystick or multi-directional key. Generally, phones support a basic 12-button keypad, with other optional navigational buttons available depending on the phone model. Thus, all applications must support navigation using at least the 12-button keypad.

Usability

Usability is concerned with a broad range of issues and includes not only how mobile phones are used, but also the demographics of who uses them, when they are used and for what purpose. Thornton and Houser (2004) demonstrated the trends in usability differ for different user groups, concluding that users tend to be more comfortable when using their phone in a familiar manner. As such, it is important that an interface be developed with the user in mind. Research into how a target user group uses their mobile phone will help determine both how to display information and also how to effectively develop application features so that users will be familiar with their manner of operation and hence comfortable in using them.

THE MOBILE SUBJECT ASSISTANT

The Mobile Subject Assistant (MSA) is proposed a mobile phone based course tool designed to assist a course instructor the delivery of university level subjects. The MSA is strongly focussed on supporting collaborative learning, emphasising student participation and group or project work The MSA provides an instructor with a means of communicating with their students via their mobile phone. It provides an instructor with a means of assessing class performance using a mobile quiz tool and also provides a shared calendar to post important reminders. Finally, the MSA allows an instructor to support group projects following the "anywhere, anytime" concept and provides each project group with their own calendar and communication system.

The most interesting aspect of the MSA, from a HCI perspective, is the quiz tool. The quiz tool is designed to be used at the end of lectures or tutorials to test student understanding of a particular topic. This would effectively provide an instructor with immediate feedback on how well students understood the content of the particular session. However, as discussed below, the quiz questions must be limited to selectable-choice-answer type questions (multiple choice, true/false etc), thus the information on how well students absorbed the topic is also limited.

The MSA itself is designed as two main applications: the MSACoordinator and the MSAClient. The MSACoordinator is a tool that allows an instructor to coordinate their subject with regards to the MSA system. This tool uses a web based interface that allows an instructor to create quizzes, add calendar entries, send messages etc. It also allows an instructor to set up student projects and groups, providing students with a tool to support collaborative projects. The MSAClient is an application running on a mobile phone that allows a student to access the resources of their MSA enabled subjects from anywhere at anytime. This means that a student has access to calendar entries, project facilities and subject messages whenever they log onto the system from their mobile phone. The student will also have access to quizzes when the instructor opens such quizzes using the MSACoordinator.

HCI Issues

While designing the interface for the MSA, a number of questions arose:

- What is the maximum number of navigation levels can be included without detrimentally affecting useability?

- Should text based or graphical navigation be used?
- How should, for example, quiz questions be displayed?
- How should quiz answers be input?

The solution to each of these questions is designed to take into consideration previously discussed limitations to mobile phone applications.

Navigation

Masoodian and Lane (2003) point out that many standard mobile phone features are not generally used as they are buried in a large, hierarchical navigation system. Thus the MSA is designed to support a maximum of four levels of navigation after the user logs in, chooses their subject and/or project and has been presented with a subject/project menu. However, most options function on two or three levels. For example, when performing a quiz, a user would traverse two levels down, however when using the calendar they could possibly traverse four levels down:

Quizzes	→	Quiz 1	→	Question 1		
Messages						
Calendar →	Today's Entries	→	Change Date	→	View Entry	
Projects						

The design of the MSA aims to ensure that a user understands the options available, and that no options are hidden within a large hierarchical menu. Navigation will be text based i.e. users' select text options to take them into menus and sub menus etc. As discussed earlier, this is a more familiar environment for the majority of phone users. Most application output, such as quiz questions, will also be textual.

Display

The MSA is designed to be primarily text based. As Masoodian and Lane (2003) identified, users are more familiar with text based navigation and hence are more comfortable in a text based environment. Due to the nature of mobile phones and screen sizes, a text based environment is ideal for the MSA. The MSA is also designed to use a simple display. In order to enable a user to easily understand where they are and the task at hand, each quiz question along with its possible answers, for example, will be displayed on a separate screen. Each screen is designed to support one item, whether it is a quiz or a calendar entry. This helps a user identify where they are and what they are doing.

User Input

We stated above that it is desirable to limit textual input. Answers to quiz questions, for example, must be selectable only and involve no text input. This also allows quizzes to be completed faster and with less chance for error by the students. However textual input can not be avoided altogether. The project based messaging system designed to assist students collaborate within project groups requires text based input by the user.

CONCLUSION

This paper has described a number of important issues that must be considered when developing applications for mobile phones. It has also describes the Mobile Subject Assistant project, its purpose and a number of design issues that were faced regarding the interface, navigation and user input.

REFERENCES

AMTA (2003). *AMTA Annual Report.* Griffith, ACT, Australian Mobile Telecommunications Association Limited:

Corlett, D., M. Sharples, et al. (2004). A Mobile Learning Organiser for University Students. *Proceedings of The 2nd IEEE International Workshop on Wireless and Mobile Technologies in Education*, Jhongli, Taiwan, IEEE Computer Society.

Cortez, C., M. Nussbaum, et al. (2004). Teaching Science with Mobile Computer Supported Collaborative Learning (MCSCL). *Proceedings of The 2nd IEEE International Workshop on Wireless and Mobile Technologies in Education*, Jhongli, Taiwan, IEEE Computer Society.

Kaikkonen, A. and V. Roto (2003). Designing Applications for Handheld Devices: Navigating in a Mobile XHTML Application. *Human factors in computing systems*, Ft. Lauderdale, Florida, USA, ACM Press.

Masoodian, M. and N. Lane (2003). An Empirical Study of Textual and Graphical Travel ItineraryVisualization using Mobile Phones. *Fourth Australasian User Interface Conference*, Adelaide, Australia, Australian Computer Society, Inc.

Roschelle, J. and R. Pea (2002). A walk on the WILD side: How wireless handhelds may change CSCL. *Computer Support for Collaborative Learning* 2002, University of Colorado, Boulder, CO, USA, International Society of the Learning Sciences.

Siemens http://www.my-siemens.com.au/, 10/09/2004.

Thornton, P. and C. Houser (2004). Using Mobile Phones in Education. *Proceedings of The 2nd IEEE International Workshop on Wireless and Mobile Technologies in Education*, Jhongli, Taiwan, IEEE Computer Society.

Wigdor, D. and R. Balakrishnan (2003). TiltText: Using Tilt for Text Input to Mobile Phones. *16th Annual ACM Symposium on User Interface Software and Technology*, Vancouver, Canada.

Wuthrich, C., G. Kalbfleisch, et al. (2003). "On-line instructional testing in a mobile environment." *The Journal of Computing in Small Colleges* 18(4): 23-29.

COPYRIGHT