Client Mobile Software Design Principles for Mobile Learning Systems

doi:10.3991/ijim.v3i1.753

Qing Tan and Kinshuk
Athabasca University, Athabasca, Canada

Abstract—**In a client-server mobile learning system, client mobile software must run on the mobile phone to acquire, package, and send student's interaction data via the mobile communications network to the connected mobile application server. The server will receive and process the client data in order to offer appropriate content and learning activities. To develop the mobile learning systems there are a number of very important issues that must be addressed. Mobile phones have scarce computing resources. They consist of heterogeneous devices and use various mobile operating systems, they have limitations with their user/device interaction capabilities, high data communications cost, and must provide for device mobility and portability. In this paper we propose five principles for designing Client mobile learning software. A location-based adaptive mobile learning system is presented as a proof of concept to demonstrate the applicability of these design principles.**

Index Terms—**Multiplatform Adaptation, Client Mobile Software, Design Principles, Location Based Adaptive Mobile Learning**

I. INTRODUCTION

People use their mobile phones not only as phone sets but also as powerful data communication devices. The number of mobile devices accessing the Internet has been increasing rapidly over recent years. Studies have shown that in the United States the numbers have grown significantly, demonstrating a quarterly growth increase of 154% in Q4 of 2007 versus the same period in 2006 [1]. Similarly, another study of mobile Internet usage identified that in Japan there were over 53 million users that accessed the Internet on their mobile devices in June of 2007 [2].

Utilizing widely used mobile phones for learning has created a new dimension of learning modalities, namely Mobile Learning or mLearning. mLearning is a natural extension of eLearning, which enables learning anytime and anywhere. In comparison to eLearning, mobile learning has its own unique and outstanding features, particularly location awareness. A location-based service (LBS) is the service that adds value to target location provided by LoCation Service (LCS) [3]. LBS is accessible with mobile devices via the mobile network and utilizes the ability to make use of the geographical position of the mobile device [4]. Employing the location awareness of a mobile phone we have proposed a location based optimal grouping algorithm for an adaptive mobile learning system. The algorithm creates optimal groups by identifying the individual mobile learners' geographic locations and other learning factors, such as their learning profiles and learning behaviors. The adaptive mobile learning system automatically provides grouping suggestions to the online individual mobile learners. Thus, for example, the individual mobile learners within a designated geographic radius could be grouped together to take advantage of collaborative learning. Further, such a grouping can enhance the students' learning in the pervasive learning environment [5]. Another useful and interesting application we have developed provides for the creation of authentic examples for mLearning itself. The examples are attached to their location information, which indicates where the examples were experienced or happened. Other mobile learners can learn from the examples and use the attached location information to guide them to go the locations and to experience the examples. This use of location can greatly help learners to understand the learning concept [6].

However, there are challenges in using the proposed and developed location-based mobile learning applications. In particular, the learning applications require client mobile software to run on mobile phones. Among the natural constraints of a mobile phone, such as scarce resource limitations, heterogeneity and use of multiplatform mobile operating systems, user/device interaction limitations, high cost of data communications, and device mobility and portability, the primary challenge is to ensure the client mobile software can work across platforms [7][8][9]. In this paper, we will discuss client mobile software issues, propose five design principles, and present a practical demonstration of the design principles.

In the following section, we explain relevant client mobile software issues in detail. In the third section, we will propose the five design principles. In the fourth section, we will describe a client mobile software application for a location based mobile learning system as a practical demonstration of the design principles. Finally, in the last section, we will provide some conclusions.

II. CLIENT MOBILE SOFTWARE ISSUES

There are many third-party mobile software applications that have been developed for personal and industrial use, which include mobile learning systems. However, most of this software can only be used with a particular mobile operating system supported by specific mobile phone brands and many times with only particular models of the brands. There are a number of proprietary mobile operating systems such as Symbian OS, RIM BlackBerry, and Windows Mobile, iPhone OS, Palm OS, Android, and so on, that are popular in the market. Although most smart-phones are Mobile Information Device Profile (MIDP) 2.0 ready - to support cross platform Java 2 Micro Edition (J2ME) programs - there are still differences within their Java Virtual Machines (JVM). The result is that there is no guarantee that a J2ME application developed on one platform would be compatible with other platforms.

A location based mobile learning application system requires the learner's mobile phone to run its client mobile software in order to acquire, package, and send its device information and location data to the server and to offer location based services to the mobile learner. The impact of heterogeneous mobile phones and multiplatform mobile operating systems has greatly limited the development and use of mobile learner applications. Various versions of the client mobile software must be developed for compatibility with the wide range of mobile phones available in the market. This can be a costly venture for mobile learning content providers. Alternatively, if developers of client mobile software only target specific phones, they face great difficulty in marketing their solutions. A recent example of this conundrum relates to the development of a transport tracking and logging mobile application solution by a local Edmonton, Alberta-based firm. The application requires that the transport drivers enter logging data specifically through a Treo 650 mobile phone using the Palm OS. The challenge in terms of marketing this application is the cost of the Treo 650 phone since the client mobile software is only supported on this particular phone.

Most mobile phones available in the market have very limited resources: slow CPU speed, small amounts of RAM, and small display footprint. The limited resources imply that client mobile software is restrained in terms of computation speed and memory usage. Usually client mobile software is developed using a compact version SDK, such as the J2ME development environment. This development environment has less software resources and "lite" versions of the Java APIs. As a result, in order to achieve high response performance, mobile software demands the use of better design and application architectures and best practices program coding. There is little ability to cope with so-called spaghetti code. Further, human interactions via a mobile phone are usually inconvenient for the user due to the limited size of the keypad and /or touch screen. This can have a large impact on the design of a mobile software interface and the functions of mobile applications.

The Client software for mobile applications usually relies on client side data to drive functionality offered by the server side. The limited nature of the mobile computing resource implies that as much computation occur on the server side as is possible. The communications between the client side and the server side applications largely depends on the cellular network. The data plans currently available from most mobile system suppliers allow for limited data transmission bandwidth usually at significant cost. The limited bandwidth available with mobile data communications via cellular network has become an overriding factor in mobile application design.

People enjoy the mobility and portability that is allowed by use of a mobile phone. The devices are now carried everywhere and used anywhere and anytime. A mobile application must use the built-in features of the phone; enhance the phone's capability but not reduce the portability. A mobile application that needs third party's hardware will greatly decrease mobility and portability.

III. CLIENT MOBILE SOFTWARE DESIGN PRINCIPLES

To overcome the issues discussed in the last section, we propose the following five design principles for client mobile software and mobile learning system development:

1. Multiplatform adaptation
2. Little resource usage
3. Little human/device interaction
4. Small data communication bandwidth use
5. No additional hardware

A. *Multiplatform Adaptation*

Client mobile software must work in multiplatform environments. For example, with the adaptive mobile learning system discussed below, the proposed location-based optimal grouping algorithm is implemented on the application server side. This algorithm is designed to offer grouping suggestions to the mobile learners. The algorithm requires client-side software to run on the learners' mobile phones. The client-side application supplies the learners' geographic location data. In practice, it is unlikely that the institution implementing the adaptive mobile learning system would provide mobile phones to their students. As a result, the mobile learners will download and run the client mobile software on their own MIDP 2.0 ready phones. The use of a variety of phones and the multiplatform nature of the operating systems that will be used are, therefore, inevitable.

B. *Little Resource Usage*

The resource constraints of mobile phones mean that any application should employ the built-in functions and features of the phone. The application must be designed to use the computation capabilities of the phone as little as possible. Instead, computation must use the server side as much as possible. The phone is thus adopting the role of a mobile terminal but not a mobile workstation. Our adaptive mobile learning system, for example, uses client software, running in background mode to acquire, package, and send the phone's location data to the application server. The bulk of the computation takes place on the server side.

C. *Little Human/Device Interaction*

A mobile user needs to interact with his/her mobile phone to perform necessary actions. Because of the small screen or/and keypad, human/device interaction is inconvenient and even difficult in some cases. A design principle is, therefore, the less human interaction is required the better the mobile application is.

D. *Small Data Communication Bandwith Use*

In order to dedicate most of the computational load to the server side more communication between the client and the server may be required. Mobile data communication, via the cellular network, is very expense and consumes phone's scarce resources, such as battery power. A design principle for a client-server mobile application system is that there should be as little communication between the client phone and the server computer as possible. Unlike with conventional LAN-

WAN based computer applications, the cost of communications paramount.

E. *No Additional Hardware*

The mobility and portability of mobile phones is of great utility to users. Currently available phones have been integrated with much useful hardware and software to enhance the mobile phone's features. The mobile phone is not only a communications device but it also offers a powerful multi-functional tool for people's daily use. Any additional hardware that is required by a mobile application will reduce the compactness and portability of phone. It will also create extra cost for the user and give rise to the lack of compatibility issue. Consequently, the principle of no additional hardware is very important in the context of mobile applications.

IV. CLIENT MOBILE SOFTWARE DESIGN PRINCIPLES IN PRACTICE

In this section, we will describe a prototype client mobile software application that demonstrates the use of the five design principles in practice. An adaptive mobile learning system has been developed with a number of features and functions that are supported by the location based services available with most mobile devices. A location-based optimal grouping algorithm utilizes the phone's location awareness to group together online mobile learners. The client software executes the following tasks to support the algorithm's functions.

- Acquire mobile phone's device information, such as International Mobile Equipment Identity (IMEI), International Mobile Subscriber Identity (IMSI), device type and type of mobile operating system.
- Acquire mobile phone's cellular network field data, such as CellID, Location Area Code (LAC), Mobile Network Code (MNC), and mobile phone number.
- Acquire Global Positioning System (GPS) coordinates, such as latitude, longitude, and altitude.
- Package the acquired client side data into a required format.
- Send the client side data to the application server via the communications link, be it General Packet Radio Service (GPRS) cellular network, Wi-Fi, Bluetooth or via Universal Serial Bus (USB) connection.

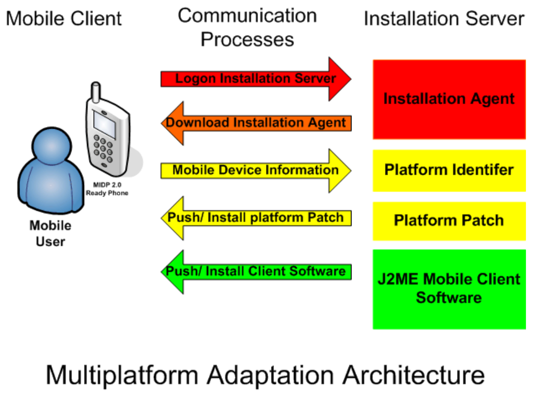

Mobile Client
Communication Processes
Installation Server
Logon Installation Server
Download Installation Agent
Mobile Device Information
Push/ Install platform Patch
Push/ Install Client Software
Installation Agent
Platform Identifer
Platform Patch
J2ME Mobile Client Software
MIDP 2.0 Ready Phone
Mobile User
Multiplatform Adaptation Architecture

Figure 1. The Multiplatform Adaptation Architecture

A. *Multiplatform Adaptation*

The adaptive mobile learning system application requires the following very basic requirements from a mobile phone:

- MIDP 2.0 with Connected Limited Device Configuration (CLDC) 1.0 ready mobile phone

This basic requirement covers more than 90% of the smart phones currently available. The multiplatform adaptation architecture is shown in Figure 1.

A pre-authorized mobile user logs on to the installation server using the mobile phone's built-in browser over the Internet. An installation agent on the server is downloaded to the phone. The installation agent is a J2ME application that runs on any J2ME 2.0 ready mobile phone without optional Java Specification Requests (JSR) support. After the installation agent runs, it collects and sends the mobile phone's device information, such as IMEI, mobile phone number, mobile brand and model, and mobile operating system to the installation server. Using the device information, a platform identifier agent identifies the mobile device and the mobile's environment. The platform identifier agent analyzes the mobile's environment and can push and install missing environmental components such as a required mobile JVM for J2ME platform patch. For example, required JSRs can be identified as missing and pushed and installed on the mobile device automatically. As a result the mobile phone's environment is automatically upgraded to support the client mobile software. Lastly, the client mobile software itself is pushed and installed on the phone. The entire installation process takes place automatically with minimal involvement of the user. This multiplatform adaptation architecture can not only be used for J2ME client mobile software applications but can also be applied to applications developed using another programming language, such as C or C++. The installation agent, however, must be a J2ME application because J2ME is the only cross platform programming language which is supported by most of the mobile phone manufactures.

An associated research project will investigate all MIDP 2.0 ready mobile phones, to inventory the available mobile operating systems and their mobile JVMs. The research will explore options in the development of patches for the various platforms. The aim is to provide multiplatform support for over 90% of the mobile phones currently in the market.

B. *Little Resource Usage*

The location-based optimal grouping algorithm only requires the client mobile software to perform limited functions – primarily to acquire, package, and send device and location data to the application server. The remaining computational tasks take place on the server computer. The optimal grouping suggestions are sent to the mobile learners through the mobile learning web page. The web browser is automatically launched and loaded with the mobile learning web page as depicted in Figures 2 and 3.

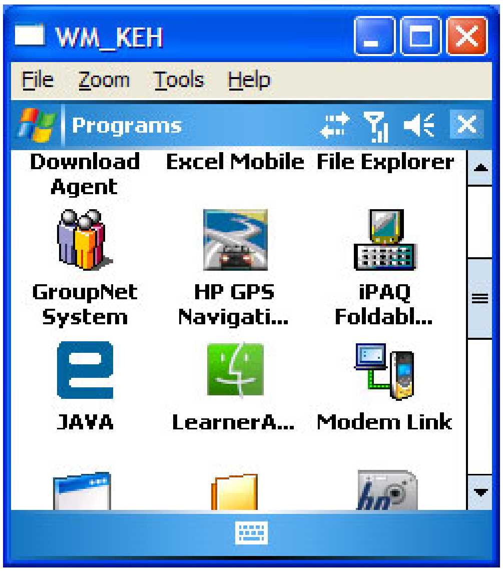
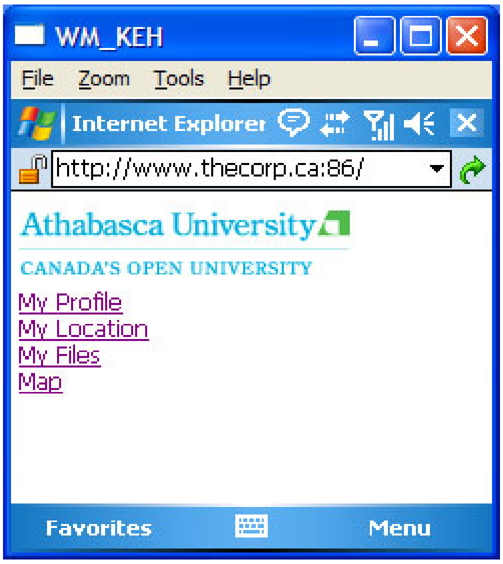


Figure 2. Launch Client Mobile Software



http://www.thecorp.ca:86/

Athabasca University

CANADA'S OPEN UNIVERSITY

My Profile
My Location
My Files
Map

Figure 3. Mobile Learning Web Page

C. Little Human/Device Interaction

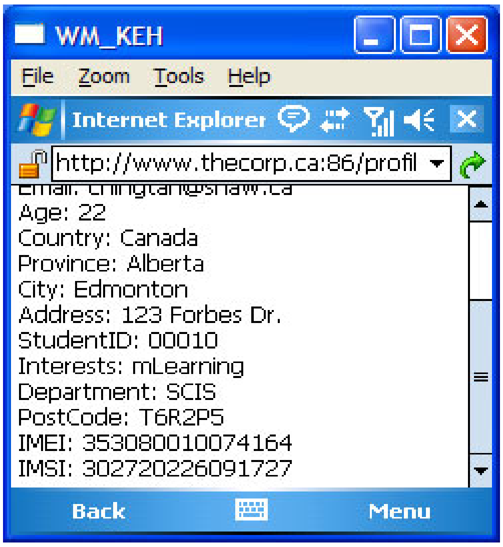
The adaptive mobile learning system uses the mobile device information, i.e. IMEI and/or the mobile phone number for user authentication. The authentication process is automatically invoked without the mobile learner's involvement. The IMEI and mobile phone number are the unique identification for a mobile phone. The unique IDs of the mobile phone are registered and stored in the mobile learner's profile table of the database on the application server during the installation process. When the client mobile software runs, the unique IDs are included in each data package that is sent to the server. The server uses the unique IDs, as shown in Figures 4 and 5, to automatically authenticate the user without involving the mobile learner. This greatly reduces the human/device interaction.

D. Small Data Communication Bandwidth Use

The adaptive mobile learning system application applies the location information of a mobile learner for grouping. Knowing that the mobile learner is online and keeping track of the changes in the learners' locations provides enough information for the grouping algorithm. Thus, while the mobile learner is online, the client mobile software only needs to communicate with the server in cases where:

- the cellular network base station Cell ID changes;
- the changes in GPS coordinates are out of the predefined geographic group distance range (in urban areas 5 Kms or in rural areas 10 Kms); or
- there have been no changes for some time and the client syncs with the server to keep a user's current status active (every 15 minutes).

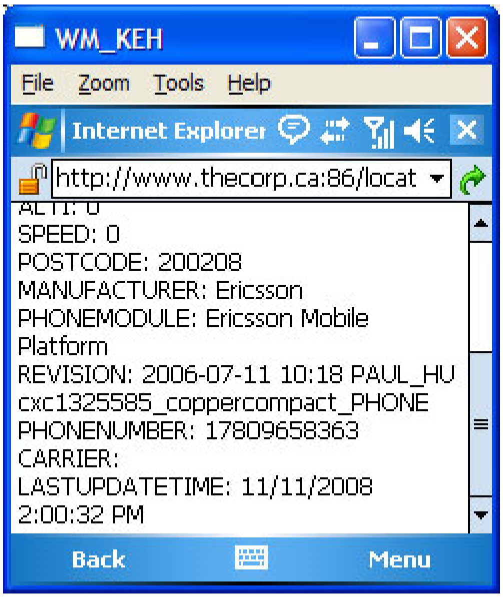
This design approach reduces the data communication from the mobile phone to the application server. A depiction of the mobile learners' locations as shown on a map and associated profile links are illustrated in Figures 6 and 7.



http://www.thecorp.ca:86/profil

Email: chingtan@shaw.ca
Age: 22
Country: Canada
Province: Alberta
City: Edmonton
Address: 123 Forbes Dr.
StudentID: 00010
Interests: mLearning
Department: SCIS
PostCode: T6R2P5
IMEI: 353080010074164
IMSI: 302720226091727

Figure 4. Registered IMEI



http://www.thecorp.ca:86/locat

ALTI: 0
SPEED: 0
POSTCODE: 200208
MANUFACTURER: Ericsson
PHONEMODULE: Ericsson Mobile Platform
REVISION: 2006-07-11 10:18 PAUL_HU cxc1325585_coppercompact_PHONE
PHONENUMBER: 17809658363
CARRIER:
LASTUPDATETIME: 11/11/2008 2:00:32 PM

Figure 5. Registered Phone Number

E. No Additional Hardware

Use of existing features of the mobile phone, such as the built-in digital camera and Bluetooth support to obtain the location information, reduces the cost and enhances the usability of the system. While many research projects and development activities have used Radio-frequency identification (RFID) technology to provide location information, a RFID reader becomes a requirement. Because there is this requirement for additional hardware RFID based mobile applications are employed only in special circumstances. Instead of an RFID approach, one can use Quick Response (QR) Code to mark spots of interest. A mobile user then uses the existing phone's digital camera to scan and decode the information on the QR Code [10]. A mobile digital camera is built in to almost all the smart phones available today. It is anticipated that QR Code based applications will find wide use in future because there is no additional hardware required. A proof of concept example showing the use of QR Code is depicted in Figure 8.

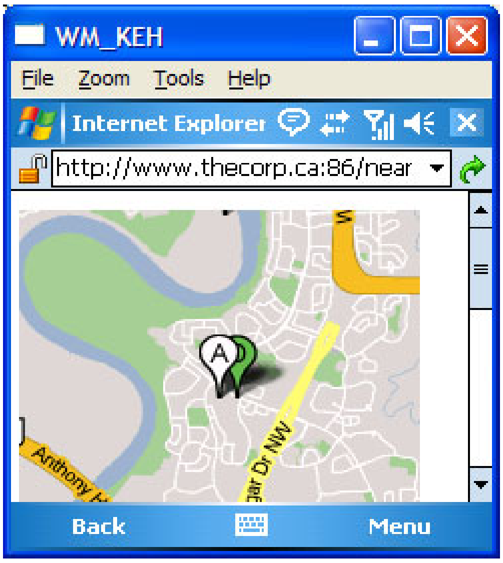
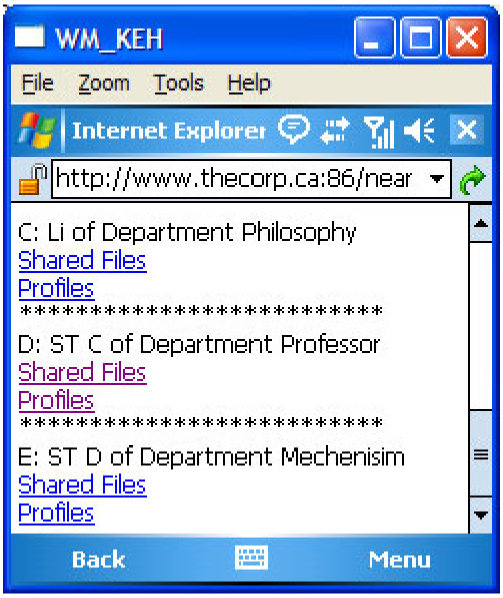


Figure 6. Mobile learners' locations



C: Li of Department Philosophy
Shared Files
Profiles
****
D: ST C of Department Professor
Shared Files
Profiles
****
E: ST D of Department Mechenisim
Shared Files
Profiles

Figure 7. Grouped mobile learner's profiles

V. CONCLUSIONS

This paper has proposed five design principles for developing client mobile software in mobile learning systems. A demonstration proof of concept client software application which implements an adaptive mobile learning system algorithm has been developed using the five design principles. The design principles are thus used as a guide to the design of real client mobile software applications and can be used to evaluate existing applications. This should contribute to the design of better mobile software with the goal of producing more practical and usable mobile learning systems. This research has wider implications, since majority of mobile applications share their characteristics with mobile learning systems, and therefore these principles should be equally applicable to majority of mobile applications. Future research will focus on further real world mobile software examples as production-ready demonstrators of the five design principles in client-server mobile application development.

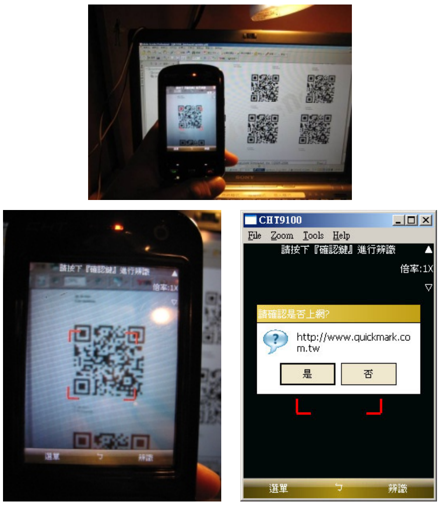


Figure 8. A proof of concept demonstration of using QR Code with a mobile phone camera

REFERENCES

[1] A. Lipsman. "Number of U.S. Computers Accessing the Internet via Mobile Broadband Soars 154 Percent in 2007," *comScore*. From http://www.comscore.com/press/release.asp?press=20 99, March, 2008

[2] J. Gavin, "Mobile Phone Web Users Nearly Equal PC Based Internet Users in Japan." *comScore*. From http://www.comscore.com/press/release.asp?press=1742, September, 2007

[3] S. Wang, J.W. Min and B. K. Yi, "Location Based Services for Mobiles: Technologies and Standards," *IEEE International Conference on Communication (ICC)*, Beijing, China, 2008

[4] K. Virrantaus, et al. "Developing GIS-Supported Location-Based Services," *Second International Conference on Web Information Systems Engineering (Wise'01)* Volume 2, December 3 -6, Kyoto, Japan, 2001

[5] M. Chang, Q. Tan, T.C. Liu, and O. Lin, "Multi-Agent Architecture-based Location-aware Service Project for Ubiquitous Learning." *In Proceeding of the ICCE Conference 2008*, Taipei, R.O.C, 2008

[6] Y.H. Kuo, Q. Tan, Kinshuk, Y.M. Huang, T.C. Liu, and M. Change. "Collaborative Creation of Authentic Examples with Location for u-Learning," *In Proceedings of the IADIS e-Learning 2008 International Conference*, Amsterdam, Netherlands, 2008

[7] C. Mascolo, L. Capra, and W. Emmerich, "Mobile computing middleware," *In Advanced lectures on networking* pp. 20-58

Springer-Verlag New York, Inc., 2002 (doi:10.1007/3-540-36162-6_2)

[8] D. Grigoras. “Challenges to the Design of Mobile Middleware Systems,” *In Parallel Computing in Electrical Engineering, International Symposium* on pp. 14-19. doi: 10.1109/ PARELEC.2006.23, 2006

[9] S. Vaughan-Nichols, “Wireless middleware: glue for the mobile infrastructure. Computer” 37(5), 18-20. doi: 10.1109/ MC.2004.1297229, 2004

[10] H. Susono, T. Shimomura, “Using Mobile Phones and QR Codes for Formative Class Assessment,” *In Current Developments in Technology-Assisted Education* (2006) p.1006-1010, 2006

AUTHORS

Dr. Qing Tan is with the School of Computer and Information Systems, Athabasca University, Canada (e-mail: qingt@athabascau.ca).

Dr. Kinshuk is with the School of Computer and Information Systems, Athabasca University, Canada (e-mail: kinshuk@athabascau.ca).

Manuscript received 3 December 2008. Published as submitted by the authors.