World Academy of Science, Engineering and Technology
International Journal of Information and Communication Engineering
Vol:6, No:12, 2012

Utilizing Adaptive Software to Enhance Information Management

J. Soini, P. Sillberg, and J. Raitaniemi

Open Science Index, Information and Communication Engineering Vol:6, No:12, 2012 publications.waset.org/13456/pdf

***Abstract*—**The task of strategic information technology management is to focus on adapting technology to ensure competitiveness. A key factor for success in this sector is awareness and readiness to deploy new technologies and exploit the services they offer. Recently, the need for more flexible and dynamic user interfaces (UIs) has been recognized, especially in mobile applications. An ongoing research project (MOP), initiated by TUT in Finland, is looking at how mobile device UIs can be adapted for different needs and contexts. It focuses on examining the possibilities to develop adapter software for solving the challenges related to the UI and its flexibility in mobile devices. This approach has great potential for enhancing information transfer in mobile devices, and consequently for improving information management. The technology presented here could be one of the key emerging technologies in the information technology sector in relation to mobile devices and telecommunications.

***Keywords*—**Emerging technologies, Flexible user interfaces, Information management, Information technology, Mobile technology.

I. Introduction

INFORMATION sharing and transfer are necessary elements of knowledge management in organizations. Accessibility and access to the company's back-end system information, corporate data and process infrastructure, anytime and anywhere, is essential. There is a growing need for more efficient checking, updating, and viewing of important company information in real time [1],[2],[3]. Contemporary telecommunications and mobile technology offer many different opportunities for meeting this critical business need. Using intelligent mobile devices and the applications made for them, users are able to collaborate and share information and capture knowledge when and where they need it. However, expectations and requirements for information are changing, and there are recognized needs for more flexibility, dynamic and personally tailored features in relation to the usability of mobile devices [4],[5]. On the other hand, the increasing fragmentation related to mobile devices and also the types of information to be transmitted, place major challenges on the implementation of device independent dynamicity. Moreover, there are some technical restrictions and limitations related to mobile services, which add new challenges to the widespread adoption and diffusion of mobile applications [6] as well as data transformation [7].

As a result of this situation, the MOP research project (MOP - adaptive software services) was initiated in early 2011 by the Tampere University of Technology (TUT) [8] in Finland. The research deals with exploiting the opportunities created by current mobile technology to meet the more extensive and increasing needs of users. The main aim in the ongoing project is to study technologies that can enable context-based adaptive user interfaces (UI) for mobile devices, and the project is examining the distribution options for an interface that can be adapted according to the situation. In the research, the primary focus is on technologies that enable run-time implemented user interfaces that adapt easily and quickly. This kind of feature offers new opportunities for exploiting the usage of mobile devices in a more versatile way that at present. The research will also try to help solve the problems and challenges faced by software developers when trying to create software for multiple platforms and devices that may not be natively compatible with each other, while still providing similar user experience on all target devices.

The purpose of this research is to indicate new opportunities for exploiting mobile technology in corporate information management. It focuses on increasing the potential of using the existing resources of the company – mobile devices in this case – as well as improving the availability, updating and transferability of real-time information, one of the most important factors from the aspect of information quality in an organization's operations. The proposed approach shows an innovative way to apply existing mobile technologies for enhancing fluent and suitable communication and information sharing in an organization. Mobile applications and services already released for this flexible and dynamic UI environment are not very widely available. However, in the research field, there have been several other previous approaches to studying UIs dynamically, such as Jou [9], who presents a way to make the web browsing experience on mobile devices better by transcoding the web tables into more suitable forms. Jin et al. [10] present a UI framework for the dynamic updating of a web page during UI interaction on local networks and over the Internet. Moreover, Ye and Herbert [11] have presented generic software architecture for an adaptive user interface development system, which is based on a standard abstract UI description language (XUL), which is shared among different platforms. Rosenthal [12] proposed a special case of adapter

J. O. Soini is with the Software Engineering Department, Tampere University of Technology, Pori unit, 28100 Pori, Finland (phone: +358-40-8262890; fax: +358-2-627 2727; e-mail: jari.o.soini@tut.fi).

P. K. Sillberg is with the Software Engineering Department, Tampere University of Technology, Pori unit, 28100 Pori, Finland (phone: +358-40-8262745; fax: +358-2-627 2727; e-mail: pekka.sillberg@tut.fi).

J. Raitaniemi is Director (Mobile Solutions) of Acando Oy, Valtakatu 6, 28100 Pori, Finland (phone: +358-44-3434100; fax: +358-424-723220; e-mail: janne.raitaniemi@acando.com).

Open Science Index, Information and Communication Engineering Vol:6, No:12, 2012 publications.waset.org/13456/pdf

software in what is called the WebKit hybrid (see also [13]). In addition, there are some more technology-oriented studies on this topic, which have given examples of using XML-based UI language in simple applications [14],[15],[16],[17].

The structure of this paper is as follows: the next section gives a brief introduction to the impact of mobile technology and also the opportunities it provides as a part of information management. Section II also explains the relevance of the research topics, i.e., dynamic user interfaces and also evaluates potential ways of exploiting it. Section III describes the studied and applied mobile technologies and also, with the help of examples, presents the preliminary results of the study. Section IV describes the approach of using adapter software for retrieving UI embedded with content and even for retrieving software updates. Section V includes a discussion and evaluation of the study results and also some proposals for future research on this topic. Finally, Section VI summarizes the paper.

II. Opportunities and Challenges of Mobile Technology in Information Management

In general, the technology management function in an organization is meant for understanding the value of a certain technology for the organization. In relation to this, it can be stated that information technology (ICT) can have a remarkable strategic impact on organizations [18]. ICT offers huge opportunities to organizations, and if exploited effectively, it can create competitive advantages and also economic value for organizations [19],[20]. Today an inseparable part – and one of the most rapid growing industries - of information technology is mobile technology. It is obvious that this sector has a huge potential to succeed in emerging technology markets in the near future. Organizations are realizing more and more clearly the huge potential that mobile information technology can offer [21]. There is a lot of evidence for positive effects such as acceleration of data availability, real-time data, quality of information, and decision making [3],[22]. There are also many indicators of the strategic implications of mobile technology, like increasing internal communication and knowledge sharing as well as flexibility in coordination [5],[23],[24]. The capability of utilizing the latest mobile technology and also awareness of new ways of applying, adapting and utilizing it, is just one of the most important and significant parts of a company's strategic technology management.

In modern business, one of the key issues is ubiquitous access to the organization's back-end system information, company data, and process infrastructure. Enabling access to corporate information anywhere and anytime is essential. Mobile devices are also becoming more suitable for mobile data use and the amount of more sophisticated mobile applications is quickly increasing [25]. We can state that in principle advanced mobile technology facilitates the real-time collecting and sharing of data, and its storage directly in the company's information systems, so that the updated data become available to everyone. However, in practice, there are several issues that restrict the fluent and seamless functionality in the kind of situation described above. Fragmentation could be an apposite term to describe the current situation in the mobile sector. There are many different brands, run-time platforms, networks and many other factors all contributing to the fact that there is a large variety of mobile devices on the market and, as a consequence, a number of problems with device-specific software [7]. It could be argued that it is almost impossible to write a single version of a mobile application that could run on every mobile device available. In addition, challenges related to data presentation (type and content of data may vary: text, image, sound, video, content streaming, presence, etc.) and device-specific features (screen resolution/display size, keyboard, mouse, touch screen, motion control, performance differences, cross-platform incompatibility, etc.) limit the full invocation of mobile technology in information management. As can be seen, fragmentation is wide-ranging and diverse in the mobile sector, and there are many technology-related issues, including the maturity of mobile information technology.

This situation described above creates an interesting research frame for exploration of this theme. Next, Section III presents one approach (dynamic user interfaces) for solving issues related to the fragmentation problem in mobile technology.

III. Dynamic User Interfaces

In the telecommunications sector and especially in mobile technology the main focus is increasingly moving away from improving the performance of data transfer and devices, toward applications, usability, and service configurability. In addition to workers needing ubiquitous access to critical information of the organization and knowledge exchange [2], there is a growing desire for more agile, adaptive, real-time use and features in mobile devices. One potential solution is the *dynamic user interface* of an individualized application, which also allows a company-specific customized data screen if necessary. The role of the *dynamic user interface* is significant in a business environment where real-time and up-to-date data - and also real-time reporting - is essential [1],[3].

The ongoing project entitled *Adaptive Software Services* (Mukautuvat Ohjelmisto Palvelut - MOP in Finnish) was initiated at the beginning of 2011 by the Tampere University of Technology (TUT) [8] with a consortium consisting of three Finnish ICT companies and the City of Pori. The two-year project is funded mainly by the Finnish Funding Agency for Technology and Innovation (Tekes) [26] and also partly by the participating organizations. The general goal of the project is to research software and information technologies and also deploy new techniques that can be useful in solving the problems and challenges related to the fragmentation issues in mobile technology described above. In general, the aim is to study how user interfaces can be adapted according to different needs and contexts. The project focuses on technologies that can enable context-based adaptive user

Open Science Index, Information and Communication Engineering Vol:6, No:12, 2012 publications.waset.org/13456/pdf

interfaces and also the distribution options for an interface that can be adapted according to the situation. More specifically, the research concentrates on the possibility of using *adapter software* and a centralized management system to update UIs dynamically. The main idea is that the adapter can update its UIs and extend functionalities by retrieving newer specification files from a system managing the UIs and their distribution. The research target is cross-platform solutions that provide software components and composed services that can be adapted in run-time for the preferred context or situation. The ultimate goal of the framework studied is to allow users to create applications, distribute applications to desired clients, use applications on any device with compatible adapter software, and to gather data. This study, and its results, supports the research related to enabler technologies and their implementation in the mobile information technology context.

IV. Flexibly Managed User Interfaces for Mobile Applications

This section describes the overall system architecture of an information system for creating and distributing data gathering applications. An example of the application creation process is also given. Fig. 1 shows the main components of the system, which are the UI Application Service and Adapter Software. The *UI Application Service* is used by the *Developer* or *Power User*, who creates and maintains the applications available for the users. The *Adapter Software* uses the Representational state transfer (REST) Application Programming Interfaces (API) of the *UI Application Service* to retrieve, update, and submit data. Finally, the *User* uses the system with the *Adapter Software*, which is installed into his/her device.

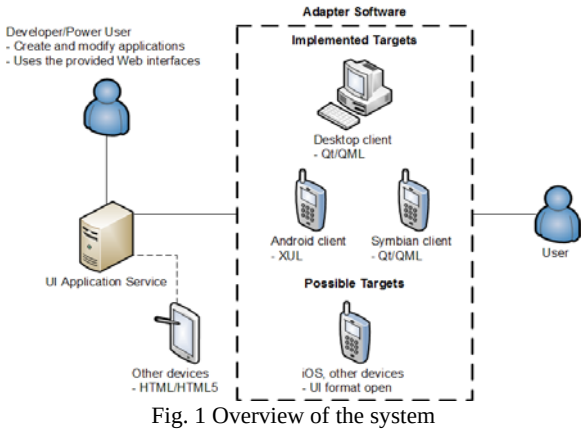


Fig. 1 Overview of the system

Currently there are three different platforms supported (*Implemented Targets* in Fig. 1): Android, Qt Symbian, and Qt Desktop. The Android client's UI format is Extensible User Interface Language (XUL) while both of the Qt versions use Qt Meta Language (QML) [27],[28],[29]. The *UI Application Service* is able to generate the correct UI files for each different type of *Adapter Software*. Also, each different *Adapter Software* is able to render the UI file according to the UI conventions of the platform/device. This allows the content of the applications distributed through the system to be similar on all devices but the look-and-feel and usage follows the UI concepts of the target device. It is also possible to support other devices, such as iOS, by implementing *Adapter Software* for the device. There have also been discussions about whether there should be a Hypertext Markup Language (HTML) [30] or HTML5 [31] compatible application interface. This would allow the usage of the system on an even greater range of devices.

The main goal of the information system is to provide an easy and rapid way to create and distribute query applications and data collection forms to the users. The typical workflow of using the system is started by the *Developer*, who creates an application. The new application requires a name and type. The name of the application is visible to the *User* when he or she browses through the application list in the *Adapter Software*. The application type determines how the application is rendered in the adapter software. Current variants are "Menu" and "Navigable." In menu applications the forms can be selected individually and in navigable applications the forms are displayed in a successive manner – like in wizard dialogs. The developer may also set the rights to access the application. The access restriction can be made by user group or simply on a user basis.

The next step is to create the forms of the application. An example of the form creation process is shown in Fig. 2. On the left side of the figure there is the *Developer view* or IDE view. The right side of the figure shows how the form would be rendered on the adapter software running on Android. The available components are shown on the left side of Fig. 2 and the preview of the current form on the right. There are now ten different components available but it is possible to include components such as Image or File for extending the usability of the system. The *Properties* section shows the editable fields of the selected component. The form has the name *XULTest* and it has three input fields named *Label1*, *Label2*, and *Label3*. The first two components are Textboxes and the last one is a Combobox. The components may also have a placeholder text set in the *Value* field.

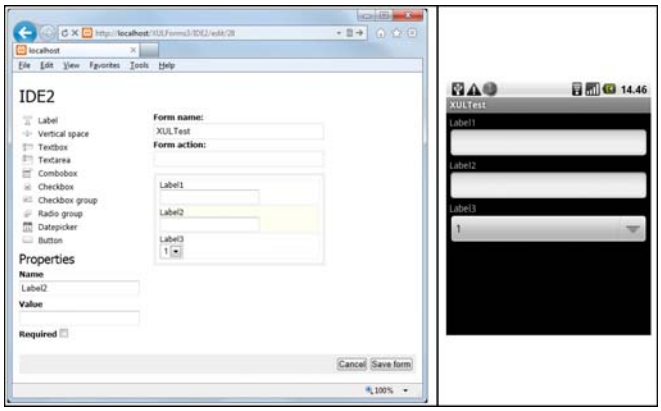


Fig. 2 Developer view IDE (left) and user interface on device (right)

Open Science Index, Information and Communication Engineering Vol:6, No:12, 2012 publications.waset.org/13456/pdf

The process continues by saving the form when it is finished and then creating as many new forms as are needed. When all the forms have been created, the users may start using the new application once their adapter software has retrieved the updated application definitions. The *UI Application Service* will also automatically create the database entries for data gathering and reporting purposes. The developer does not need to worry about how the data will be stored, as the UI Application Service will perform the task. The forms can also be edited if new components are required or corrections are needed to the old ones. The changes are propagated to the users once their adapter software updates itself.

The following Figs. 3, 4, and 5 illustrate an example of the creation of an application, what the application looks like on an Android device and what kind of data have been gathered. Fig. 3 has similar components as seen in Fig. 2 but this time the form contains slightly different input fields. The left side of the figure is the IDE view and the right side is the corresponding view on the device. This application example consists of a total of three forms.

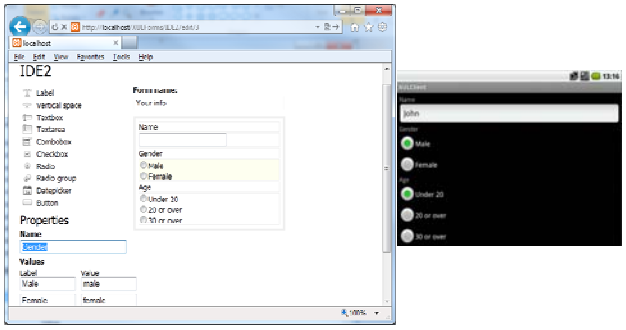


Fig. 3 Example of a query application (first form)

Fig. 4 (below) shows the other forms in the application. The corresponding IDE view is not shown for these forms but it follows the scheme seen in the earlier figures. It should be noted that the IDE or the adapter software does not do automatic paging but supports scrolling. The form seen in the IDE appears similarly in the device's adapter software, so it is up to the developer of the application to decide how to split the content. An example of this can be seen on the left side of Fig. 4, where the items seem to come to an abrupt stop, but in reality, the user can simply scroll down to see the rest of the form's content. The user may also navigate between the forms by pressing the *Back* and *Next* buttons. The *Next* button is replaced by the *Submit* button in the final form. After pressing *Submit*, the values given by the user are sent to the server.

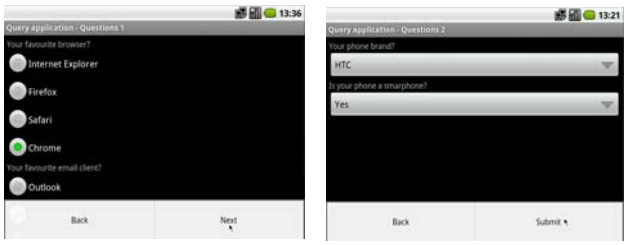


Fig. 4 Example of a query application (forms two and three)

All submissions of the application are listed in the IDE view with timestamp and user information. The full data of a particular submission can be seen by selecting it from the list. This is shown in Fig. 5 (below). The *Field* (name) column contains all the input fields which were created in the application's IDE view. The *Value* column contains the corresponding data value given by the user.

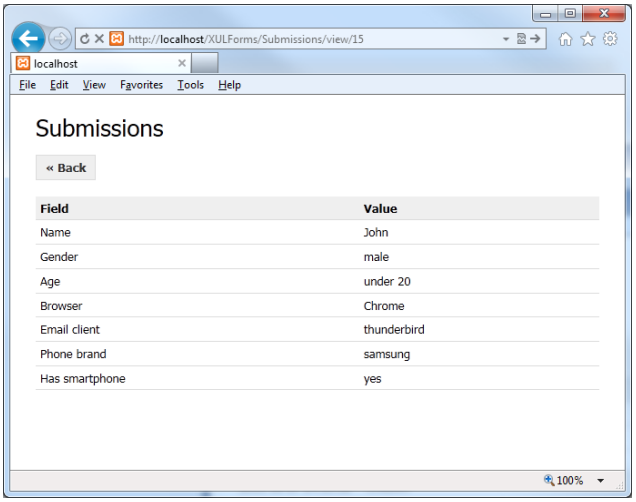


Fig. 5 Example of submission data gathered from the user

Figs. 6 and 7 (below) show two screenshots of the adapter software running on Qt Simulator. The figures show what the adapter software would look like for the user - in this case test-user@example.org.



Fig. 6 Qt/QML-based adapter software running on Qt Simulator. Listing of the user's available applications

In this example the developer has created two applications: *Travel Report* and *Browser Gallup*. The Travel Report application is used as an example. Both of these applications are listed in Fig. 6. The *Travel Report* application is currently selected, and the adapter software shows detailed information about the application (in this case, the last modification date is shown). By selecting the application again, the adapter software will load the forms of the application into a new page as shown in Fig. 7.

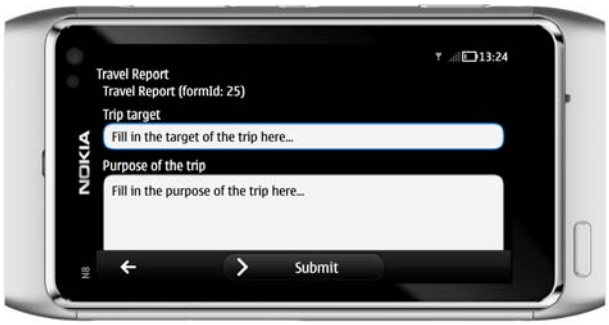


Fig. 7 Qt/QML-based adapter software running on Qt Simulator. Form view of the Application

Fig. 7 (above) shows the form of the *Travel Report* application. In this case, there is only one form but any number of forms may be created for the application. If there is more than one form the currently shown *Submit* button would be a *Next* button instead. Once the last form is reached, the button would be *Submit* again. When *Submit* is pressed, the adapter software gathers all the data from the input fields and sends the generated XML to the *UI Application Service*'s REST API.

V. Discussions

The adapter software approach seems to be a viable option when there is no need for more advanced components than those shown in the Fig. 2 IDE section. For example, when used for *query applications* (i.e. applications that are used to gather data from human users), the system has advantages as it is able to handle the application's creation, distribution, data collection, and data reporting. One other clear benefit of the system presented here is that this adapter type of model gives functionality to mobile network "offset" situations and guarantees that data are not lost from the mobile device. When the network connection is missing, then the Adapter application collects data to local storage for later usage. Of course, a web-based application might gather data on the server, but they have one problem in the network lost or "offline" situations. Old web-based applications do not handle "offline" situations at all and the data go missing. The new HTML5 standard gives some help in this, but when will it be supported by the browsers and how well? There is one further drawback in HTML5 local storage support, which is that the default amount of data is limited to 5 megabytes per application. Sometimes a mobile application database might be bigger, for example, when gathering some pictures.

Additionally, mobile sensors, for example, GPS, accelerometer, magnetometer, etc. could be easier to adapt to the application. Basically from the application developer or business point of view, we simply want to obtain e.g. location information from the user regarding where the data entry was made. In this model you simply put it in the text field of the GPS sensor included in your application and the adapter software handles the rest. For example, in a web application a special JavaScript API has to be used to put the location information in the text field. HTML5 and JavaScript have to be handled, but the web-based application does not guarantee the location information accuracy and the developer must have the knowledge to develop the web application. The adapter type of implementation, as here, does not require knowledge of HTML5 and JavaScript. One of the main concerns regarding the feasibility of the system is that the limited amount of available components restricts the developer from creating complex applications. However, there is a pilot program underway where these kinds of aspects will be tested and more detailed information on the feasibility of the system will come up during this pilot.

Therefore, the presented model helps shorten development cycles and organizations may develop a simple mobile application for targeted users very quickly. The approach studied and presented here can offer one solution for the increasing demands related to the agility, adaptability, and also individualistic use of mobile devices for information transfer. A short search has revealed that in the mobile application area there is a lack of *Adapter solutions* corresponding to those described here. However, one can assume that the technologies and solution presented here are indicative of potential emerging technologies in the mobile device industry in the imminent future.

VI. Summary

The paper deals with the strategic importance of mobile technology from the information management point of view. The paper discussed the opportunities and challenges that cutting-edge mobile technology can offer the information management field. The basis of the research was to try to find solutions for the challenges caused by the fragmentation typical of the mobile sector. In particular, the possibilities were studied of creating a framework for developing a flexible, device-independent and real-time updated user interface for mobile devices. The approach was based on the impression that flexible manageable user interfaces are becoming an important trend in mobile and ubiquity applications. The goal of the framework studied and presented here was to allow users to create applications, distribute applications for desired clients, use applications on any device with compatible adapter software, and to gather data. These kinds of information technologies and their features offer new opportunities for exploiting the usage of mobile devices in a more versatile way that at present. The proposed approach provides an innovative way to apply existing mobile technologies to enhance fluent and suitable communication and information sharing in organizations. The technology presented here could be one of the key emerging technologies, through which the use of mobile devices could be expanded in the information management sector in the future.

References

[1] K. Burden, "Business benefits of industry-specific mobile applications," Global Headquarters: 5 Speen Street Framingham (2005), MA 01701 USA P.508.872.8200 F.508.935.4015, Retrieved 7/9/2011 World Wide Web, http://www.idc.com

Open Science Index, Information and Communication Engineering Vol:6, No:12, 2012 publications.waset.org/13456/pdf

[2] G. B. Davis, "Anytime/anyplace computing and the future of the knowledge work," *Communications of the ACM*, vol. 45, no. 12, pp. 67-73, 2002.
[3] K. MacDonald, and J. Metzger, "Achieving tangible IT benefits in small physician practices," *California healthcare Foundation*, ihealth Reports, pp.1-36, Sep. 2002.
[4] A. Göker, and H. Myrhaug, "Evaluation of a mobile information system in context," *Information Processing & Management*, vol. 44, no. 1, pp. 39-65, 2008.
[5] S. L. Järvenpää, K. R. Lang, Y. Takeda, and V. K. Tuunainen, "Mobile commerce at crossroads," *Communications of the ACM*, vol. 46, no. 12, pp. 41-44, 2003.
[6] K. Siau, E. Lim, and Z. Shen, "Mobile commerce: Promises, challenges and research agenda," *Journal of Database Management*, vol. 12, no. 3, pp. 4-13, 2001.
[7] J. Van Gurp, A. Karhinen, and J. Bosch, "Mobile Service Oriented Architectures (MOSOA)," *Lecture Notes in Computer Science*, vol. 4025, pp. 1-15, 2006.
[8] Tampere University of Technology (TUT), Retrieved 15/3/12 World Wide Web, http://www.tut.fi/public
[9] C. Jou, "Automatic web table transcoding for mobile devices based on table classification," in *Proc. of IADIS Multi Conference on Computer Science and Information Systems*, pp. 143-150, Rome, July 2011.
[10] H. Jin, J. Y. Cho, and J. W. Park, "Dynamic user interface update using web based framework," in *Proc. of International Conference on Consumer Electronics (ICCE)*, pp. 10-14, Las Vegas, Jan. 2007.
[11] J. Ye, and J. Herbert, "Framework for Use Interface Adaption," *Lecture Notes in Computer Science*, vol. 3196, pp. 167-174, 2004.
[12] N. Rosenthal, "Hybrid Qt/WebKit applications: Pushing the Limits of Web Development [Video]," *Presented in MeeGo Conference 2010*, Dublin, Nov. 2010. Retrieved 14/12/2011 World Wide Web, http://conference2010.meego.com/session/hybrid-qtwebkit-applications-pushing-limits-web-development
[13] The WebKit Open Source Project (2011), Retrieved 14/12/2011 World Wide Web, http://webkit.org
[14] J. Bishop, and N. Horspool, "Cross-Platform development: Software that lasts," *IEEE Computer*, vol. 39, no. 10, pp. 26-35, 2006.
[15] N. Mitrovic, and E. Mena, "Adaptive user interface for mobile devices," Interactive Systems: Design, Specification, and Verification," *Lecture Notes in Computer Science*, vol. 2545, pp. 29-43, 2002.
[16] N. Mitrovic, J. A. Rovo, and E. Mena, "ADUS: Indirect generation of user interfaces on wireless devices," in *Proc. of 15th International Workshop on Database and Expert Systems Applications (DEXA)*, pp. 662- 666, Zaragoza, Aug.-Sep. 2004.
[17] V. T. Vasconcelos, S. J. Gay, A. Ravara, N. Gesbert, and A. Z. Caldeira, "Dynamic interfaces," *Presented in FOOL conference 2009*, Savannah, Georgia, Jan. 2009.
[18] D. Buhalis, "E-airlines: Strategic and tactical use of ICTs in the airline industry," *Information and Management*, vol. 41, no. 7, pp. 805-825, 2004.
[19] G. H. Griffiths, and P. N. Finlay, "IS-enabled sustainable competitive advantage in financial services, retailing and manufacturing," *The Journal of Strategic Information Systems*, vol. 13, no. 1, pp. 29-59, 2004.
[20] M. N. Melville, K. Kraemer, and V. Gurbaxani, "Review: Information technology and organizational performance: an integrative model of it business value," *Journal of MIS Quarterly*, vol. 28, no. 2, pp. 283-322, 2004.
[21] S. J. Barnes, "Enterprise mobility: concept and examples," *International Journal of Mobile Communications*, vol. 1, no. 4, pp. 341-359, 2003.
[22] A. Krause, D. Hartl, F. Theis, M. Stangl, K. Gerauer, and A. Mehlhorn, "Mobile decision support for transplantation patient data," *International Journal of Medical Informatics*, vol. 73, issue 5, pp. 461-464, 2004.
[23] CIO insight (2003), "When will mobility add strategic value?," Retrieved 14/12/2011 World Wide Web, http://www.cioinsight.com/article2/
[24] H. Sheng, F. Fui-Hoon Nah, and K. Siau, "Strategic implications of mobile technology: A case study using Value-Focused Thinking," *The Journal of Strategic Information Systems*, vol. 14, no. 3, pp. 269-290, 2005.
[25] L. Cosgrove, "Applied wireless: Making wireless work in the business," *CIO Focus*, pp. 1-39, 2005.
[26] Finnish Funding Agency for Technology and Innovation (Tekes), Retrieved 15/3/2012 World Wide Web, http://www.tekes.fi/eng
[27] Mozilla Developer Network (2012), XML User Interface Language, Retrieved 10/2/2012 World Wide Web, https://developer.mozilla.org/En/XUL
[28] Nokia Corporation (2012a), Qt Meta Language - Introduction to the QML Language, Retrieved 10/2/2012 World Wide Web, http://developer.qt.nokia.com/doc/qt-4.8/qdeclarativeintroduction.html
[29] Nokia Corporation (2012b) Qt - Cross-platform application and UI framework, Retrieved 10/2/2012 World Wide Web, http://qt.nokia.com
[30] World Wide Web Consortium (W3C), HTML 4.01 Specification, W3C Recommendation, Dec 24, 1999, Retrieved 10/2/2012 World Wide Web, http://www.w3.org/TR/html401/
[31] World Wide Web Consortium (W3C), 2011, HTML5 - A vocabulary and associated APIs for HTML and XHTML, W3C Working Draft, May 25, Retrieved 10/2/2012 World Wide Web, http://www.w3.org/TR/html5/

Open Science Index, Information and Communication Engineering Vol:6, No:12, 2012 publications.waset.org/13456/pdf