

An Academic Kit for Integrating Mobile Devices into the CS Curriculum

Qusay H. Mahmoud, Thanh Ngo, Razieh Niazi, Pawel Popowicz, Robert Sydoryshyn,
Matthew Wilks

Department of CIS, University of Guelph, Guelph, ON, N1G 2W1, Canada

Dave Dietz, Waterloo, ON, N2L 5Z5, Canada

ABSTRACT

In this paper we present our freely available academic kit to help universities in integrating mobile devices into the Computer Science (CS) curriculum. The kit was designed and developed at the Centre for Mobile Education and Research at the University of Guelph, and includes instructors' resources for introducing and teaching mobile application development. The first release of the kit includes the teaching material for a full introductory course on mobile application development, and concrete teaching modules for integrating mobile devices into courses on software engineering, game design and development, web services, information security, and operating systems.

Categories and Subject Descriptors: K.3.2 [Computing Milieux]: Computer and Information Science Education – *computer science education, curriculum, human factors, literacy*.

General Terms: Algorithms, Design, Experimentation, Languages.

Keywords: Mobile application development, mobile devices, BlackBerry, programming for fun, teaching computer programming, teaching tools.

1. Introduction

With the widespread use of mobile devices such as smartphones and pocket personal computers, there is a great need for innovation in CS education to reflect today's reality. For example, students in introductory programming courses usually develop and test their programming assignments on a platform similar to the one on which they will be tested by the instructor or teaching assistant. However, this is not the case for mobile applications which are developed on one platform such as Microsoft Windows or Linux and deployed on a totally different platform such as a cellular phone. A great opportunity exists for introducing students to different programming models which will not only be very practical but can also inspire students to be excited about learning. Those students, equipped with the most current training in this field, may play a key role in driving innovations in the mobile space.

We have recently integrated mobile devices such as the BlackBerry wireless device into our lower and upper division programming courses [2], advanced courses, and senior capstone projects [6]. We have achieved excellent results in terms of student performance, excitement, and satisfaction. The feedback we have received from the students in the affected courses as well as from other faculty

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. To copy otherwise, or republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee.

ITiCSE '09, July 6–9, 2009, Paris, France.

Copyright 2009 ACM 978-1-60558-381-5/09/07...\$5.00.

members at other institutions is very encouraging and has inspired us to design and develop an academic kit that is generously funded in part by Research In Motion (RIM) to help universities in integrating mobile devices in the CS curriculum.

This paper presents our academic kit for integrating mobile devices in the CS curriculum. We provide details on the novel features of the kit and what it includes. We describe the course on mobile application development and what it covers, as well as the individual teaching modules. If you are looking for information and strategies for integrating mobile devices in the CS curriculum, please see *Integrating Mobile Devices into the CS Curriculum* [2].

The rest of this paper is organized as follows. Section 2 presents an overview of the kit. The details of the course on mobile application development are presented in Section 3. The teaching modules are presented in Section 4. Finally, conclusions and future work are presented in Section 5.

2. The Academic Kit

An academic kit has been created to facilitate the integration of mobile devices into CS and engineering curriculums for universities worldwide. The kit takes a mobile approach to teaching CS content with a focus on Java ME [3] and BlackBerry application development. The academic kit contains 20 weeks worth of teaching material including lesson slides, labs, tutorials, quizzes and assignments. The topics covered feature Java ME, BlackBerry, GUI, event handling, multimedia, security, and networking.

The content found in the academic kit will assist instructors and students in learning about mobile applications. We offer material for a full course on mobile application development, as well as teaching modules for software engineering, web services, game design and development, information security, and operating systems.

2.1 Availability

An early access release of the kit was made available to instructors teaching mobile application development courses in Fall 2008. The feedback we have received so far is very encouraging. Since late 2008, the kit has been available for free download at [5] and will be made available on CDs to ITiCSE 2009 attendees.

2.2 Software Tools

The software tools needed for using the kit, such as a development and emulation environment, can be downloaded free of charge. For Java ME [3], we recommend Sun's Java Wireless Toolkit, which provides an environment for getting started with Java ME development. Likewise, RIM's BlackBerry Java Development Environment provides a fully integrated development environment and simulation tool [4] for emulating the device's functionality. The simulators are useful for testing applications; however, we believe that letting students experiment with the physical devices (if your budget allows it) provides a totally different and superior experience.

3. Mobile Application Development Course

The ACM Computing Curricula [1] lists *wireless and mobile computing* as an elective course under the "Net-Centric Computing" body of knowledge, and as a possible consequence few CS departments offer a course on mobile computing at the undergraduate level and for those who do, the focus of such a course is on theory of wireless local area networks and protocols. We believe that a course on mobile application development should be part of today's CS curriculum.

The mobile application development course we have designed and developed is geared towards introducing and teaching CS students, who have had some programming experience with Java, to the special requirements for developing mobile applications. Mobile applications carry with them unique properties on a technical and business level. To be successful in this field, students must be knowledgeable of these considerations and challenges so that they can overcome them. The course can be offered in a second, third, or fourth year level.

3.1 Lecture Slides

The lecture slides cover a variety of topics ranging from the basics of Java ME and introduction to the BlackBerry to more advanced topics such as network connections and multimedia. These slides allow instructors to easily integrate a smartphone such as the BlackBerry into their courses while covering topics such as event handling and network connections by introducing these topics from a mobile device standpoint. Topics such as the event handling, creating threads, and establishing network connections, just to name a few, are all part of mobile application development. Rather than teaching students how to program these on the desktop environment, the slides demonstrate how to accomplish these tasks on the BlackBerry platform instead. Each set of slides focuses on a specific topic and is intended to cover a typical week of three hours worth of lectures.

The first set of slides is an introduction to Java ME, covering the basics including Java ME profiles, configurations, packages, MIDlets, as well as example code for starting your first Java ME application. The next set of slides covers best practices when developing for mobile devices, more specifically the BlackBerry. The differences between developing a desktop and mobile application are presented. Design challenges such as small screen size, processing power, unpredictable network connectivity, and limited battery life are presented along with the solutions to addressing these characteristics.

The next set of slides introduces the BlackBerry, differences between a Java ME application and a BlackBerry application, and how to convert an existing JAVA ME application into a BlackBerry application. Event handling within BlackBerry applications and the use of listeners are also discussed. GUI components within the BlackBerry API such as labels, buttons, dialogs, menus, the many different types of fields, etc. are covered in a separate set of slides. This set of slides also demonstrates how the `UIApplication` and `MainScreen` classes are used to create BlackBerry applications with a user interface. The slides also demonstrate the use of layout managers, custom fields, as well as the BlackBerry's unique way of displaying content on the screen. Data management and multimedia are covered in their own separate sets of slides.

Another set of slides covers multimedia topics. Finally, establishing network connections using HTTP, WiFi, Bluetooth, USB, and sockets are all covered in one set of slides.

All of the topics mentioned are accompanied by sample code along with a detailed explanation describing the code to further illustrate

the concepts and enhance the learning process. Students are able to start developing right away by using this code and making modifications as needed.

3.2 Tutorials

Tutorials were constructed to provide students with the self-inclined opportunity to practice some hands-on exercises dealing with mobile devices and specifically the BlackBerry wireless device.

Similar to hands-on labs, tutorials walk students through the process of creating and learning mobile applications. However, distinct from labs, tutorials provide more detailed information and explanation of the processes as well as solutions. Tutorials are meant to assist students in understanding and reinforcing the course material. There are nine tutorials which cover Java ME and the Wireless Toolkit, programming with the BlackBerry APIs, event handling, graphics and sound, and networking.

The tutorials are meant to introduce the student to generic programming with Java ME and ultimately lead the student to create complete Java ME/BlackBerry applications.

The tutorials are unique in the sense that they cover the necessary topics that would allow a student to not only create a basic BlackBerry application, but to create an application that is interactive, engaging, and useful. This is emphasized with such subjects as menus, events, graphics, sounds, and networking. These are the components that make practical applications more valuable, effective and user friendly.

As a precursor to learning how to program for the BlackBerry the student must first learn some introductory concepts such as the differences in developing mobile applications and understanding the Java ME architecture and features which the BlackBerry platform is built on. This information is covered in the first tutorial "Introduction to Java ME and the Java Wireless Toolkit". The Java Wireless Toolkit (WTK) is an easy-to-use tool for building Java ME applications that was chosen because it is freely available, simple, and flexible.

The tutorials are an additional resource that helps to alleviate the confusion and gaps in the learning material through hands-on exercises and step-by-step instructions. The idea is that the tutorials can create a more robust learning experience by providing a practical approach to lessons and a more in-depth experience than labs.

3.3 Labs

The focus in designing the labs has been on bridging the theory learned in the classroom with hands-on experiments enabling students to get started with developing real mobile applications. By completing the labs, students will be able to design and develop significant mobile applications.

The first two labs have been designed to get started with developing Java ME applications and deploying them on either Java-enabled mobile devices. The rest of the labs have been designed to work with BlackBerry APIs helping students to develop cool BlackBerry applications. All of the labs are in sync with the lecture slides and tutorials.

Each lab consists of a set of experiments followed by exercises. In the experiments, a practical scenario is presented and a complete application is developed based on that scenario. In this section, we also describe details of the code. In the exercise section, students are asked to add more features to the applications developed in the experiment section.

3.4 Assignments

The assignments are meant to enhance the student's skills with developing applications for mobile devices and will eventually result in a functioning game that they can play on a BlackBerry. As the course progresses the assignments will build up components to a Pacman-style game, and then the game will be assembled for a completed version.

In the first assignment, the students will build a mortgage calculator on a regular Java MIDP 2.0 environment, which familiarizes them with mobile application development as shown in Figure 1. Briefly, some limitations of Java ME can be seen (for example: No Math.pow). The second example shows how to use multiple forms for creating a quiz application as shown in Figure 1.

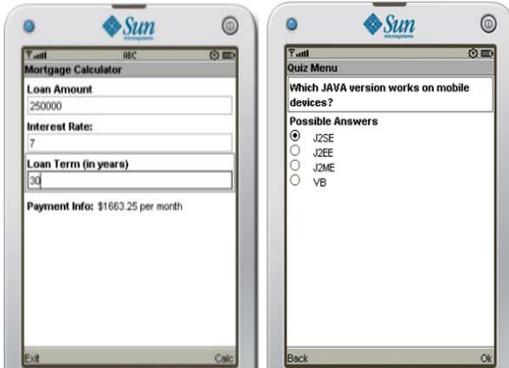


Figure 1: Sample solutions from Assignment 1

With the following assignment we move from Java ME MIDP to the BlackBerry APIs. The Mortgage Calculator assignment is done again, this time using the BlackBerry APIs. This allows for a student to see the difference between the two styles. This assignment demonstrates simple input and output using APIs developed by Research In Motion for their BlackBerry smartphone. Also, the second part to this assignment is building a Tic Tac Toe game as shown in Figure 2. Many important components are learned when constructing Tic Tac Toe such as the use of Managers like FlowFieldManager, and centering items on the screen.

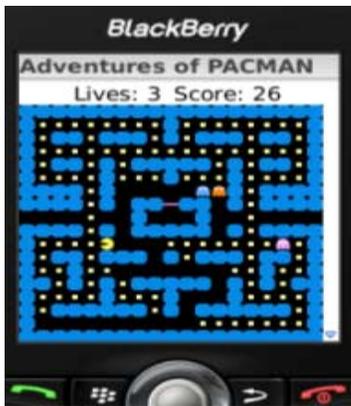


Figure 2: Sample solution from Assignments 4 and 5

The third assignment demonstrates how to create a game style user interface and move a character around within it. The first part involves building a maze with BlackBerry components and images, and then having a player navigate through it to the finish. The second part is creating the Pacman GUI. This shows the student how to import images, compose user interfaces with images, replace

images within a field manager, capture key presses as well as apply basic game logic.

In assignment 4, the student builds upon the previous assignment by creating all of the logic required, such as win scenarios, lose scenarios, and adding more users (AI). Assignment 5 involves creating the AI players and bringing the game together. This teaches threading and cross-thread operations (updating and modifying GUIs from threads other than the one it was created on) as well as seeing a fully completed project.

3.5 Quizzes

Each week of course material contains an accompanying quiz for the student to complete. The completion of the quizzes allows for instructors to monitor the progress that the student is making in the topic and course overall. The quizzes contain multiple choice and true/false questions relating to the topic covered in the lecture. The quiz questions are designed so that if the instructor would like to offer the quizzes on a mobile device such as a BlackBerry the questions are easily portable to the handheld through a quiz generator that we have built [7] that will be available as part of the kit. Each quiz contains approximately ten questions that range in difficulty in order to assess the skills of each student and identify areas for additional study.

4. Teaching Modules

The Computer Science curriculum currently covers many topics, the result of which is that there is little to no room to add additional courses. However, through the use of teaching modules this limitation can be overcome. Teaching modules provide the ability for instructors to integrate a mobile application development component into their existing course. Core CS courses such as software engineering or operating systems can have an integrated teaching module relating to mobile application development. Teaching modules are designed for two to three week sessions. A teaching module will provide the instructor with lecture slides, practice labs, tutorials, assignments, and quizzes. An instructor can pick and choose which of the features of the teaching module they wish to utilize in their course.

The process of integrating a teaching module into a core CS course accomplishes two goals: the key focus of the course does not change, as a teaching module related to software engineering with mobile applications still contains the key fundamentals within software engineering; the process also adds greater flexibility to the course in exposing students to a different hardware and development platform. This exposure and along with learning the challenges of mobile application development can give students a new perspective on other topics covered in the course.

4.1 Software Engineering

Software engineering is a core course in the CS curriculum. As software engineering covers the development, operation, and maintenance process of software applications this means that special considerations must be taken into account when dealing with software engineering and mobile applications. The software engineering teaching module developed for the mobile application development courses offers instruction in the challenges involved in designing applications for mobile devices, development processes and models, design patterns, testing, and deployment. Week one covers the definition of mobile applications, challenges involved in designing and implementing mobile application and solutions to these challenges. An introduction of a wireless software engineering model is made which identifies unique patterns in mobile application software engineering over the traditional software engineering. Week two provides architectural designs for mobile

applications, UI styles, navigation patterns, and information on application certification methods. The final week in the teaching module discusses the importance of testing in relation to mobile applications. Different forms of testing are provided as well as a mobile application focused testing checklist that will assist the developer. Test cases and examples are provided for the student.

The lecture slides provided with this teaching module come with accompanying quizzes. The questions in the quizzes directly relate to the topic covered during that particular week's lecture. The package also comes with practice labs and assignments. The first lab will allow the student to compare different integrated development environments such as the Sun Java Wireless Toolkit and the BlackBerry Integrated Development Environment. The second lab focuses on work with persistent storage on mobile devices and developing network connections. The final lab provides the student with more exposure to various UI components and generating mobile applications. The work completed in the labs will give students in the course the foundation needed to complete the assignment which involves the creation of a phonebook application on a mobile device.

Integrating mobile application development in the software engineering course gives students a greater sense of knowledge regarding the special circumstances, challenges, and considerations involved in mobile application development. The students will gain more experience and become at minimum, knowledgeable of these issues. As society becomes more mobile, students must gain this knowledge to stay competitive in computer science positions in the workforce.

4.2 Web Services

A Web service is a software system designed to support interoperable machine-to-machine interaction over a network. It is commonly a Web API that is available over the internet. Web services are interoperable because they are built upon XML-based technologies such as SOAP and WSDL. Web services will play an integral part in networking on the Web and because mobile devices rely so heavily on networking and connectivity, Web services are an important concept for students to understand. Furthermore, Web services are an important tool in helping alleviate some of the heterogeneity problems that are so copious in the mobile market.

We created a teaching module for Web services and mobile devices that can be integrated into a typical CS course. The Web service module includes three weeks worth of lessons and labs as well as an assignment and mini-project.

The first week is an introduction to Web services covering XML, architecture, styles, profiles, specifications, similar efforts, legacy systems, contract development, advantages and disadvantages. By the end of the first week, students should have a concrete understanding of Web services and be able to create and consume Web services in Java.

The second week of the module deals with Web services for mobile devices. While the concept of Web services does not change when dealing with mobile devices, the interaction between a mobile device and a Web service is different than say a laptop or PC. For this reason, week two addresses the unique characteristics of a mobile device that affects their communication with Web services. The topics include mobile limitations, communication types, device platforms, and design considerations. In the lab, students will gain practical experience creating a JAVA ME client application that invokes Web services to provide the functionality behind the application.

The third week of the module is a detailed overview of the BlackBerry architecture and how to develop Web service

applications for the BlackBerry smartphone. Furthermore, it discusses the BlackBerry Mobile Data System (MDS), BlackBerry security, the MDS platform, and the process and tools used to create BlackBerry Web service applications. In the lab, students will create BlackBerry Web service applications using dedicated BlackBerry development tools.

The assignment simulates a contract job where students are asked to work in pairs and solve a problem. The job requirements are to create two BlackBerry applications for a CRM software company to help their employees work out of the office more effectively. The assignment allows the students to apply what they have learned by the end of week three of the Web service module.

The mini-project puts students into small groups and leaves the purpose of the application up to the students, provided that it relates to mobile Web services. The mini-project allows the students to be creative while also following the criteria for which the project will be assessed. Students will work together to experience a more realistic design process and present their final work.

This Web service module is meant to give students a wide range of experience working with Web services and mobile devices. Students will enjoy learning about Web services by taking a mobile approach that is more interactive and engaging than traditional models.

4.3 Game Design and Development

This teaching module will give the student a greater understanding about how to develop games for mobile devices while taking into account various hardware restrictions. Integrating this into a course would prove beneficial as more mobile devices are being manufactured and newer models become more feature-rich and more powerful. The future will allow for powerful forms of mobile entertainment such as games and multimedia functionality. This game design teaching module includes lecture slides, labs, and assignments.

The lectures start off covering an introduction to the topic, differences in phones, why there are so many different phone models, tips on developing a successful game, various approaches, challenges and limitations, performance, and mobile platforms. The second week goes into more detail about available platforms, tools and SDKs. The third lecture covers various APIs. Some challenges in current mobile devices are that when developing, there are many different screen resolutions and processor speeds which can greatly impact the games' performance and consumer appeal. Also, Java and Research In Motion provide many graphical APIs to be utilized with multimedia content such as games. APIs include: Java2D, Java3D, SVG, and various RIM APIs.



Figure 3: Sample solution to the assignment in this module

The labs demonstrate how to develop games first on a Java enabled mobile device with MIDP 2.0. Then the next lab shows how to

develop a similar game using only BlackBerry APIs. The labs consist of two games that involve obstacle avoidance. Essentially, a car is traveling along a road and must avoid other cars in its path, otherwise you crash and lose. The labs reinforce many of skills in using components and graphical libraries.

The assignment portions are used to reinforce some of the skills taught throughout the module. Creating a role playing game is the assignment. The game resembles a Pokémon-style scenario where a player walks around the world and monsters attack them, which takes them to a combat screen. Students learn how to interact with the interfaces, use multiple forms, graphics, and various components. A sample solution of this assignment is shown in Figure 3.

4.4 Information and Network Security

The Security teaching modules consists of three weeks worth of slides for lectures, three labs and one assignment. The focus of these slides will be on mobile devices and more specifically BlackBerry smartphones..

Slides within the first week provide an introduction to security and mobile devices discussing general information with regards to security and mobile devices. They cover security in general and how it applies to mobile devices, and various threats that can occur to mobile devices. In the second week, the slides focus on developing secure applications and the purpose of doing this. It looks at security APIs for mobile devices and the different standards that are available for the development of mobile device applications.. The third week covers BlackBerry APIs that are available only for BlackBerry devices, including the process required for code signing in order to access and use some of the more security sensitive functions on BlackBerry devices.

The labs will allow students to familiarize themselves with security APIs and BlackBerry APIs since that is not normally covered in a programming course. The assignment will allow students to get even more familiar with security and BlackBerry APIs and give them an opportunity to apply the knowledge learned during the course into an application. The purpose of covering mobile devices and application security in this course is to allow students to gain an understanding of the development considerations that need to be made for mobile devices and to gain experience developing applications and integrating those concepts.

4.5 Operating Systems

Design and capabilities of a Mobile Operating System is very different than a general purpose OS running on desktop machines. This is because: (1) mobile devices have constraints on their physical characteristics such as screen size, memory, and processing power; (2) scarce availability of battery power; and (3) limited amount of computing and communication capabilities. In addition, given the wide variety of mobile devices with different capabilities and constraints available, there are different types of mobile operating systems such as Symbian, BlackBerry, Windows Mobile, iPhone, and Mobile Linux. Thus, it is essential for students to become aware of the OS landscape in the mobile space.

This teaching module consists of two parts. In the first part, a general overview on mobile operating systems is provided. Students learn about their structures and some common mobile OS features. The OSs we cover include: Palm, Symbian, Linux, Windows Mobile, BlackBerry, iPhone, and Google Android.

In the second part, students learn about the BlackBerry smart phone OS, its architecture, and features. We go through the various services available on BlackBerry OS and students learn about their features, events and APIs in more detail. The BlackBerry OS is event-driven which enables heavy use of input devices like the trackball, keypad and scroll wheel. It has a multi-tasking environment and supports multi-threading applications. In fact, threads are an important part in developing BlackBerry applications. Thus, it is essential for students to have a solid knowledge about multi-threading and multi-tasking features in the BlackBerry OS.

5. Conclusions and Future Work

In this paper we have presented the design and development of an academic kit to help universities worldwide in integrating mobile devices into the CS curriculum. The kit features a full course on mobile application development, as well as teaching modules for courses on software engineering, Web services, game design and development, information security, and operating systems. We provide instructor's resources that include lecture slides, tutorials, labs, quizzes, and assignments. Solutions to the assignments are also available to instructors. The kit will be freely available (on a CD) to all ITiCSE 2009 attendees, as well as on the Web at [5].

Our future work includes adding more materials for advanced topics as well as teaching modules for other courses such as databases. In addition, we are working on material to help faculty in assessing the value of the kit. . Finally, we are working on applications and services to facilitate mobile learning that will be freely available soon.

6. Acknowledgments

We are grateful to Research In Motion (RIM) for their generous support of this work and the Centre for Mobile Education Research.

7. References

- [1] ACM Computing Curricula (ACM-CC 2001): http://www.acm.org/education/education/education/curric_vols/cc2001.pdf. Accessed: August 19, 2008
- [2] Mahmoud, Q.H., and Dyer, A., "Integrating BlackBerry Wireless Devices into Computing Programming and Literacy Courses", *Proceedings of the 45th Southeast Conference*, Winston-Salem, NC, USA, March 2007, pp. 495-500.
- [3] Java Platform, Micro Edition (Java ME): <http://java.sun.com/javame>. Accessed: August 19, 2008.
- [4] RIM's BlackBerry Simulator: <http://na.blackberry.com/eng/developers/downloads/simulators.jsp>. Accessed: August 19, 2008.
- [5] Centre for Mobile Education and Research: <http://cmer.cis.uoguelph.ca>. Accessed: January 19, 2009.
- [6] Mahmoud, Q.H., "Integrating Mobile Devices into the Computer Science Curriculum". The 38th Annual Frontiers in Education Conference (FIE 2008), Saratoga Springs, NY, USA, October 22-25, 2008., pp. S3E-17-S3E-22.
- [7] Niazi, R., and Mahmoud, Q.H.: *A Web-based Tool for Generating Quizzes for Mobile Devices*. A poster at the 39th ACM Technical Symposium on Computer Science Education (SIGCSE), Portland, OR, USA, March 12-15, 2008.