Undergraduate Research Opportunity Programme (UROP)

Project: E290 - MOBILE COMMERCE APPLICATION DEVELOPMENT

Supervisor
Professor Kong Pe Hin Hinny
Nanyang Technological University
Email: ephkong@ntu.edu.sg

Student
Bui Hung Dao
Nanyang Technological University
Email: daobh@pmail.ntu.edu.sg

Abstract

Mobile commerce (m-commerce) makes use of wireless transactions to provide personalized and location-based services to users of high-speed mobile devices and mobile networks around the world. Some technologies available for m-commerce application development such as J2ME, BREW and .NET Mobile are discussed and compared in the report and J2ME proves itself a powerful wireless programming tool. The project was to program and develop a Mobile Commerce Application in J2ME. The so-called “Movie Ticket Booking” application we created in this project allows users to browse and book the movie ticket online through their Personal Digital Assistant (PDAs) or mobile phones. The advanced features of application include checking the validity of the credit card number and also storing the transaction details in the record that users can view later through their handheld devices at anytime. Further improvements can be made for the application such as allowing users view the remaining seats of each time slots before booking and making a automatic alert before movie time as well as deleting the records automatically after movie time. Since the knowledge of J2ME is very important for future mobile commerce application, further study is still needed communications. You can also conduct e-commerce by making online purchases and bids.

In addition, more and more companies are developing wireless applications. The most powerful technology used to develop such applications is Java 2 Micro Edition (J2ME). According to recent investigation, it is estimated that in Japan alone over 20 million J2ME enabled mobile phones were manufactured in 2001 [1]. The opportunity for those interested in writing for the J2ME platform speaks for itself.

The objective of this project is to program and develop the Mobile Commerce Application using J2ME.

2 Related Work

Java 2 Platform, Micro Edition (J2ME) which Sun Microsystems defines as a “highly optimized Java run-time environment” targets a wide range of consumer products, including pagers, cellular phones, screen-phones, digital set-top boxes and car navigation systems.”

Recently, a committee called the J2ME Executive Committee was founded. This includes 16 outstanding delegates from BEA, IBM, Insignia, Matsushita, Motorola, Nokia, Palm, Philips, RIM, Siemens, Sony, Sun, Symbian, Sony Ericsson, Texas Instruments, and Zucotto Wireless. This is the group of Members guiding the evolution of Java technology in the Java Community Process (JCP). [8]

In addition, Siemens has announced that its new Siemens Mobility Toolkit (SMTK) is now available at
no cost to third party developers focused on creating applications based on J2ME. The SMTK is compliant with MIDP, as well as CLDC, to provide a complete J2ME application runtime environment targeted at mobile information devices, such as mobile phones. The SMTK includes a phone emulator, API (Application Program Interface) class libraries and documentation, a programmer's reference manual, source code examples, and IDE plug-ins. [9]

The above facts show the related works of leading mobile companies in supporting J2ME. J2ME is still in its infancy and will have good future.

3 Technology Overview

A. Technologies Available

Having done a study about technologies available for wireless programming, I found the following major technologies: Java 2 Platform, Micro Edition (J2ME) of Sun Microsystems; Binary Runtime Environment for Wireless (BREW) of Qualcomm; and .NET Mobile of Microsoft. They all are advanced tools and platforms to development wireless application with different programming languages to use such as Java, C/C++ and ASP.NET.

On the other hand, the traditional WAP (Wireless Application Protocol) is also a technology that already supports many wireless devices. It enables mobile users easily access and interact with information and services. However, WAP is only a protocol for browsing the Web on mobile devices and hence is not our scope.

3.1 J2ME

J2ME is a technology that allows programmers to use the Java programming language and related tools to develop programs for mobile wireless information devices such as cellular phones and personal digital assistants (PDAs). J2ME consists of programming specifications and a special virtual machine, the K Virtual Machine (KVM), that allows a J2ME-encoded program to run in the mobile device.

J2ME uses configurations and profiles to customize the Java Runtime Environment (JRE). As a complete JRE, J2ME is comprised of a configuration, which determines the JVM used, and a profile, which defines the application by adding domain-specific classes.

3.2 BREW

 Qualcomm's BREW (Binary Runtime Environment for Wireless) can be viewed as:

- A set of APIs that enable developers to create software applications for wireless devices (wireless phones for now), and
- A means of selling and delivering applications to end-users.

However, currently BREW is only used with wireless devices equipped for code division multiple access (CDMA) technology. With BREW, developers can create portable applications that will work on handsets equipped with CDMA chipsets.

3.3 NET Mobile

.NET Mobile is an extension to Microsoft ASP.NET and the Microsoft's .NET Framework. It is a set of server-side Web Forms Controls to build applications for wireless mobile devices, like web phones and PDAs.

These controls produce the different output for different devices by generating WML 1.1, HTML 3.2, or compact HTML. The following Diagram shows how the .NET Mobile works

Mobile applications with .NET Mobile are developed using Microsoft Mobil Internet Toolkit (MMIT)

3.4 Comparisons:

J2ME advantages over other technologies include:

- Superior user interfaces with graphics
- The ability to function off-line out of wireless coverage
- Peer-to-peer networking
- Improved security and consistency of applications across platforms and devices.
- Have low cost of entry
- Have strong market penetration (and growing)

Some experts in Java reveal that J2ME can work in more environments than any other single application platform at a high level. If you want to talk to the microcode, you want assembler, C or C++, but for building useful applications that ride on a variety of operating systems, it’s hard to beat J2ME. How about BREW? It’s fine, but please takes note that it’s only been around since January 2001. And, to date, only about 20 simple games and database programs have been written in it.
B. Technologies and Softwares Employed

Following is the list of softwares and tools together with their functions employed in the project:

- Client: J2SE Platform, 1.4.1; J2ME Wireless Toolkit 2.0 Beta 1; Palm OS Emulator; TextPad 4.5.0.
- Server: Tomcat 4.1 server;
- Database Server: MySQL Database; MySQL Front v2.5.

4 Application Development

In this section, I will talk about the mobile commerce application called Movie Ticket Booking that we created.

Movie Ticket Booking Application which can run on PDAs or mobile phones allows users to browse and order movie tickets through their handheld devices. Its client is a MIDlet running on wireless devices. The MIDlet sends requests to the server via an http connection over a wireless network. The server connects to the database server, retrieves the required data and then sends back the requests’ results to the MIDlet.

First, let’s talk about the MIDlet’s client. Movie Ticket Booking contains quite a number of screens. To begin with, users will browse to a list of movies currently being on the cinema of the week. They can choose to view each movie description including content, director, stars, poster, rating, etc or to browse which cinema that they want to watch that movie through the MIDlet’s menu. In the latter, a list of cinemas will be available for users to choose, follow by the days and time slots of the movie. Choosing all the above details will let users to the transaction form where they fill all their particulars and order details such as name, credit card number, number of tickets, and so on. The MIDlet will check whether users fill in all the required fields and if not, it will give a warning and the transaction, of course, is not processed. One of the powerful features of Movie Ticket Booking is that a warning will be thrown if users’ credit card number is invalid. However for demonstration purpose, the actual transaction is not processed. That means the application will not deduct any amount of money from your account. The only thing it does is check whether the credit card number is valid. For the real transaction to process, it is needed to contract with a third party like Verisign.com or Worldpay.com to perform the transaction.

Come back to the MIDlet, after filling in all the required and valid data, the order will be processed. The MIDlet connects to the database server to store the transaction, update the remaining seats available for the time slot of the particular movie that users just book. One more advanced feature is that users can view the details of their booking including movie, time, date, cinema, etc. by choosing record in the main menu. This act like a reminder for users to remember what they have booked. The record can be deleted at any time they want by simply pressing delete button.

Diagram 1 shows the program flow of the Movie Ticket Booking application. The arrows indicate the action when users press a button.

Diagram 1: Program Flow of the Application

During the development of the application, object-oriented programming approach is chosen. By this way, we design each problem by a separate class. And all these classes are connected together by a main MIDlet named “MovieTicketBooking.java”. Object-oriented approach helps us to easily debug the application. Also it makes the later improving and updating process much more convenient. Several methods are also used in the MIDlet such as

- showTitle(): connects to the database and gets the movies available
• **readTitle()**: displays the movies’ title on the MIDlet

• **getImage()**: gets the movies’ posters from the server in byte form and display on the MIDlet.

Totally, the Movie Ticket Booking contains 8 .java files (1 MIDlet and 7 class files) as the client and 12 .java servlet files as the server. Following are some important class files of the application and their function:

• **MovieDesc.java**: sends the request to the servlet and gets back movies’ descriptions such as content, director, stars, poster, etc.

• **CreditCardChecking.java**: checks whether the credit card number filled in by users is valid

• **bookingDB.java**: creates the RecordStore to store the transaction details for users’ later view.

Second, let’s move on the server-side of the Application. Once the client has packaged together the request method (get, post or header) and content of request and sent it over the network, it is now up to the server to interpret the request and generate a response.

The server-side contains all the servlets which executes the entire request sending from MIDlet. An http connection was established between the client and the server when a request is sent. The servlet then makes a connection to the database, retrieves the data and sends back to the client. The doGet method of the servlet is executed every time it receives a get request from the client. The servlet can print the results as html or plain text format. The client then gets these outputs from the servlet and displays when required on the MIDlet.

Figure 1 and 2 shows the application’s main screen and thank screen, respectively.

![Figure 1: The main screen](image)

![Figure 2: The thank screen after transaction](image)

**5 Discussion**

The Movie Ticket Booking application was finished coding and debugging on time and work successfully without any error. Since the time of finish, we have done some more improvements of features for the application including:

• Managing to display image on the MIDlet which turns to be a complicated task for wireless devices. The image available for PDAs must be in PNG (Portable Network Graphics) format and is read in byte-form from the server host and displayed on the MIDlet.

• Checking whether the credit card number is valid as mentioned earlier. Although it is just an initial step in credit card verifying process, it plays a crucial part in e-commerce. That’s why we added this feature to make the application more marketable.

• Adding the record screen which allows users to view what they have ordered. This record also acts like a reminder for user to go to cinema enjoying their movies.

• Adding the ticker scrolling across every screen to indicate what users have selected so far. This allows users to keep track what they have chosen so that they can make necessary corrections before performing the transaction.

Currently, we are working on the web programming for the “Administration Pages” for the application. These pages allow the application’s administrators to view, add and modify the database. This is extremely necessary because the database (i.e. current movies showing of the week as well as the day and time slots of showing) is needed to modify periodically. And of course, we do not want them to access the database server directly to do that for security reason. This work is also one of the efforts to make our product more marketable.

Although Movie Ticket Booking is quite a powerful application, further improvements still can be made:

• We can allow users to view the available seats of each time slot before booking. In this way, users can know which time slots they can book and which cannot. So we do not have to check whether number of seats available is less than number of
tickets that users book and hence do not have to give a warning screen if that happens.

- Also we can further improve the record by making an automatic alert (auto switching on, ringing, etc.) before the showing time of the movie that users have booked. And this record will be deleted automatically from the devices after the showing time of the movie.

The last but not least, to make the application a real marketable product, we can contract with 3rd-party e-payment companies such as Verisign.com or Worldpay.com for the process of verifying credit card and payment. And after that, our product definitely is ready for use in the market.

6 Conclusion

Nowadays, the most current trend is to use PDAs and mobile phones to browse the internet through web pages that are specifically formatted for wireless devices. Users can also conduct e-commerce on wireless devices by making online purchases and bids. J2ME is a handy yet powerful that we need to create wireless applications.

This project aimed to develop a mobile commerce application using J2ME. The objective of the project is fully satisfied. We have created a useful application that let users do mobile commerce through their wireless devices in general or book their movies’ tickets in particular. J2ME have proved that it is a powerful technology day by day. Sun and its happy partners are estimating that 1 billion J2ME phones will be in use in less than five years [4]. J2ME will become the development standard for handsets. J2me is new and still in its infancy yet powerful enough.

Mobile commerce is creating a whole new meaning for mobile phones and services. Key drivers for mobile commerce service adoption will be ease of use and security. Fully understanding J2ME is very important as it plays a key role in wireless technology development. Hence for future application of mobile commerce, further study in this problem is strongly recommended.

Reference


