

Personal Digital Assistant Based Ubiquitous Programming

Nidhi Manjula, Uttam Kumar Kesri, Deepika Sharma

Abstract— The entitled paper “PDA Based Ubiquitous Programming” aims is to develop a software for mobile which is use to execute all kind of programming using a mobile phone with the support of server at backend. Now a days to execute any program or code, we need a well equipped laboratory, which is platform dependent, but to enhance uses of mobile phone and to make us able to do programming platform independently we need to develop such software. Because Today mostly people uses his cell phones/PDA or any handheld devices only for call, messages, listening songs or pictures but this software makes a person able to do programming in any language(Java ,C and C++ etc) using his mobile phone. And at the run time he can get any kind of errors if that exist in any piece of code or can receive exact output of that program.

Keywords — PDA, J2ME, IDE, MIDlet, Servlet, Microsoft Visual Studio 2005 Cross Tools.

I. INTRODUCTION

The term PDA stands for “personal digital assistants”. Mobile application development is used to develop application software which is used for small low-power handheld devices such as personal digital assistants (PDA), enterprise digital assistants or mobile phones. These applications software’s are installed on phones during manufacturing or downloaded by customers from various mobile software distribution platforms, or web applications are delivered over HTTP which use server-side or client-side processing (e.g. JavaScript) to provide an “application-like” experience within a Web browser.

Mobility projects are based on the concept of a single source root that results in a single distribution JAR file. It is possible to simulate multiple source roots by setting up one project for each source root and then using project dependencies. It may be helpful to think of the source root containing the MIDlet as the application project, while all other source roots should be considered library projects. This software can work only on availability of GPRS services. It cannot run or compile code without using this. It is dependent on this. The most critical factor affecting the quality of the software is design of a system and it has a major impact on the later phases. It

provides the design document as an output of this phase. This design document works as a plans for the solution and is used on later phases.

II. TECHNOLOGY USED

A. Java 2, Micro Edition (J2ME):-

This is the application environment used for wireless devices. There are two configuration used for J2ME, one is Connected Limited Device Configuration (CLDC) and another one is the Connected Device Configuration (CDC).

J2ME Connected, Limited Device Configuration (CLDC) specifies the Java environment for mobile phone, pager and wireless devices. It is designed for 16 bit or 32 bit small computing devices. 160 – 512 KB of memory available for CLDC devices. Typically has limited power or battery operated. network connectivity, often wireless, intermittent, low-bandwidth (9600bps or less)

J2ME Connected Device Configuration (CDC)

- Describes the Java environment for digital television set-top boxes, high end wireless devices and automotive telemetric systems.
- device is powered by a 32-bit processor
- 2MB or more of total memory available for Java
- network connectivity, often wireless, intermittent, low-bandwidth (9600bps or less)

These two configurations differ only by their memory and display capabilities. J2ME profile has implementation features for a series of computing devices. These are the Foundation Profile, Game Profile, Mobile Information Device Profile, PDA Profile, Personal Profile, Personal Basis Profile, and RMI Profile.

J2ME Architecture:-

Most of small computing devices are based on the modular design of the J2ME architecture. It has several layers located

Manuscript received April, 2014.

Nidhi Manjula, Faculty Of Engineering, Mody University Of Science & Technology, Sikar India, 8239184912

Uttam Kumar Kesri, Faculty Of Engineering, College Of Engineering, Anna University, Chennai, India, 9801048999.

Deepika Sharma, Faculty Of Engineering, Somany Institute of Technology and Management, Rewari, India , 9799458674

above the operating system, which is referred as the Connected Limited Device Configuration (CLDC). J2ME architecture doesn't replace the operating system of a small computing device. Because it is installed on top of the operating system which forms the run-time environment for devices. It consist of three layers-

First layer that is called configuration layer includes Java Virtual Machine (JVM), which directly interacts with the native operating system .it also handles interactions between the profile and the JVM. The second layer is the profile layer. It consists of set of application programming interfaces (APIs) which are used for the small computing device. The third layer is the Mobile Information Device Profile (MIDP). It contains Java APIs for user network connections, persistence storage, and the user interface. It also has access to CLDC libraries and MIDP libraries.

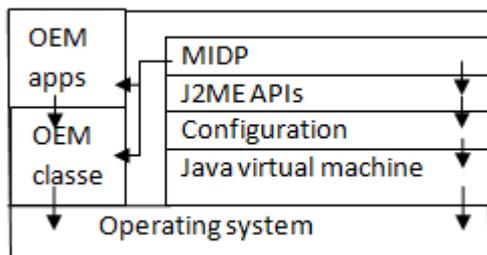


Figure1. Layers of J2ME Architecture

B. MIDlet:-

A MIDlet is an application program that uses the Mobile Information Device Profile (MIDP) for the Java ME environment. Mostly applications include games running on mobile devices and phones which have small graphical displays, numeric keypad and limited network access over HTTP. A MIDlet is designed to operate on MIDP small computing device. A MIDlet is defined with at least a single class that is derived from the javax.microedition.midlet. MIDlets have a "compile once, run anywhere" capability.the MIDlet has become the most ubiquitous application program for mobile.

Java's main challenge was to provide the use of various applications on devices with constrained resources.but Cell phones has small displays, slow central processing units (CPU), small memory, ordinary keypads and minimal connectivity features. So MIDlets have capability to facilitate all the applications to mobile phones.

A MIDlet is normally deployed as a suite which is composed of a Java Archive (.jar) file and a Java Application Descriptor (.jad) file. There are various MIDlet installation methods, as follows:

- **Direct Method:** A direct connection is established between the development computer and devices. so the most commonly used connection medium is data cable and beside this use of wireless connections, like Bluetooth and infrared (IR), is also possible.
- **Over the air (OTA) provisioning:** This method is ideal for large scale deployments because anyone can access the MIDlet at any time.

C.Servlets

Java Servlets are objects that process request and construct responses. It allows a software developer to generate dynamic content to a Web server using the Java platform.

Servlets are the Java equivalent to non-java dynamic web content technologies like PHP, CGI and ASP.NET. Servlets can maintain many server transactions, HTTP cookies, session variable or URL rewriting.

The Servlet Application Program Interface is contained in java package hierarchy javax.servlet, that defines the interaction of a Web container and a servlet. A Web container is the component of a Web server that interacts with the Servlets. It is responsible for managing the lifecycle of Servlets.

The basic servlet package defines Java objects to represent servlet requests and responses, as well as objects to reflect the Servlets configuration parameters and execution environment. The package javax.servlet. http defines HTTP-specific subclasses of the generic servlet elements, including session management objects that track multiple requests and responses between the Web server and a client. Servlets may be packaged in a WAR file as a Web Application.

Lifecycle of a Servlets:- The Servlet lifecycle consists of the following steps:

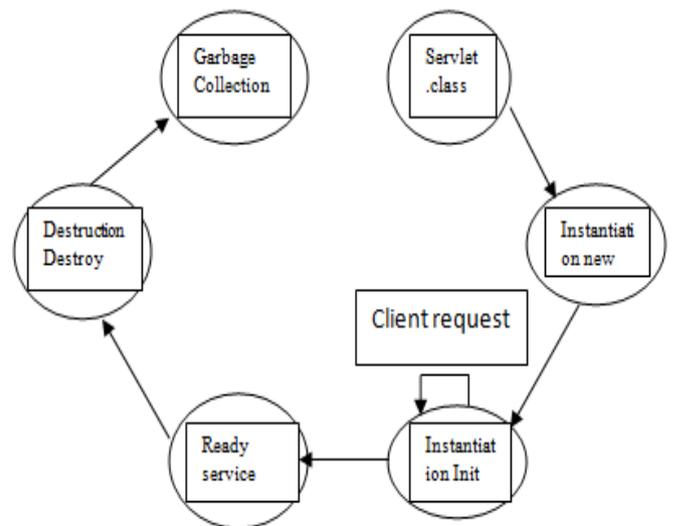


Figure2. Life cycle of servlet

D. Microsoft Visual Studio 2005 cross tools

These are the set of tools, which has been used to compile the c & c++ codes at runtime.

III. DESIGN

The purpose of the design phase is to plan a solution of the problem specified by the requirements document. This phase is the first step in moving from the problem domain to the solution domain. In other words, starting with what is needed; design takes us toward how to satisfy the needs.

The design of a system is perhaps the most critical factor affecting the quality of the software; it has a major impact on the later phases particularly testing and maintenance. The output of this phase is the design document. This document is similar to a blueprint or plans for the solution and is used later during implementation, testing and maintenance.

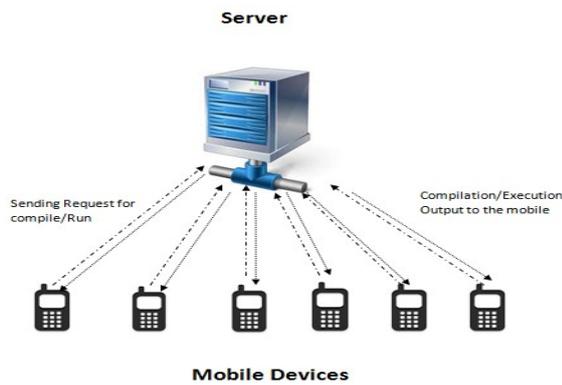


Figure3. Upper Layer Architecture

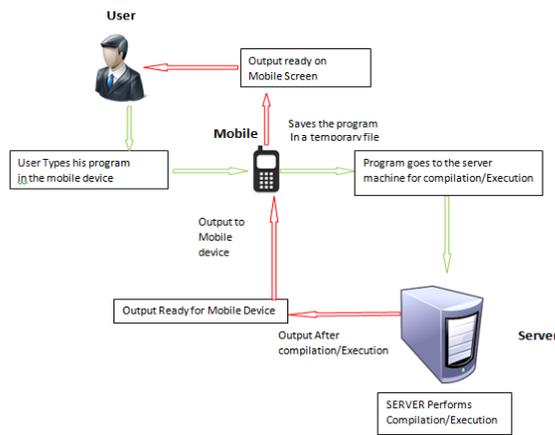


Figure4. Detail Architecture

IV. TESTING

During the development of this project all the design and source code were tested by running and compiling the Java programs and after that only they are given the final touch. Testing is the major control measures used during software development. Its basic function is to detect errors in the software. During requirements analysis and design, the output is a document that is usually textual and non-executable. After the coding phase, computer programs are available that can be executed for testing purposes. This implies that testing not only has to uncover errors introduced of testing is to uncover requirement, design and coding errors in the programs consequently; different levels of testing are used.

Testing performed:

Test case 1:-

In connectivity testing a testing is performed to test the timing and connectivity between Servlet and MIDlet.

Test case 2:-

To making successfully connection between Servlet and midlet, connectivity time testing is performed. For the connection timing taken is 7 second.

Test case 3:-

Next testing which is performed that is for compilation of java program by using a Web Application program. Time taken to execute this testing is 5 second.

Test case 4:-

Next testing which is performed that is for testing that is it possible to compile a java program using a MIDlet. And on performing this type of testing positive result is found. Total time for execution of a program is 8 second.

Test case 5:-

In this case next testing which is performed that is for Running of java program by using a Web Application. And total time taken is 5 second.

Test case 6:-

Next testing which is performed that is for checking that is Running of java program by using a MIDlet is executing or not, and a positive result is achieved. Total execution time taken is 8 second.

Test case 7:-

After designing a complete GUI (Graphical User Interface) for the mobile phone, now next testing comes that if a java program is compiling using the mobile phone or not. And a positive result is gained for the GUI compilation testing. Total time taken is 14 second.

Test case 8:-

Now next testing which is performed that is for using the GUI running of java program. And it is also successfully done. Total time taken is 15 second.

Serial number (Test Case)	Test cases	Status
1.	Connectivity	OK
2.	Connectivity time	OK (7 Second)
3.	Compilation of java program by Web Application	OK (5 Second)
4.	Compilation of java program by MIDlet	OK (8 Second)
5.	Running of java program by Web Application	OK (5 Second)
6.	Running of java program by MIDlet	OK (8 Second)
7.	Using GUI compilation testing	OK (14 Second)
8.	Using GUI Run testing	OK (15 Second)

V. RESULTS

1. Login Screen

It is the main screen of this application software. This is designed to execute all kind of programming using a mobile phone. Now user will select menu option to go on programming option screen include the labels "(a)" and "(b)" as part of the artwork. Please verify that the figures and tables



2. Menu Screen

It is the menu screen of this application software. When it is installed in a mobile then with several options a separate icon is generated which is used to do programming in mobile phone.



3. Programming Button

In this option is available after the installation of software. For login in main software click on this option to view login screen.



4. Programming Option Screen

After choosing the programming option, this screen appears. Now user can select any language among Java, C, C++, in which user want to execute the code.



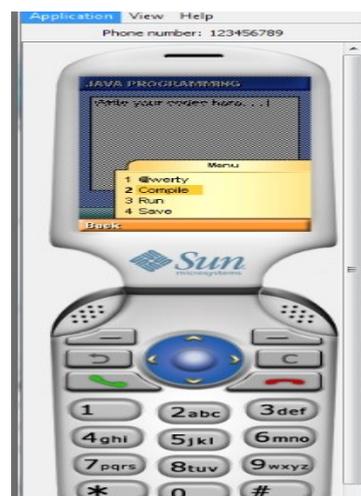
5. Java Coding Screen

Suppose user wish to do programming in java ,so java option is selected. after selecting this option a text editor opens as shown in above snapshot.



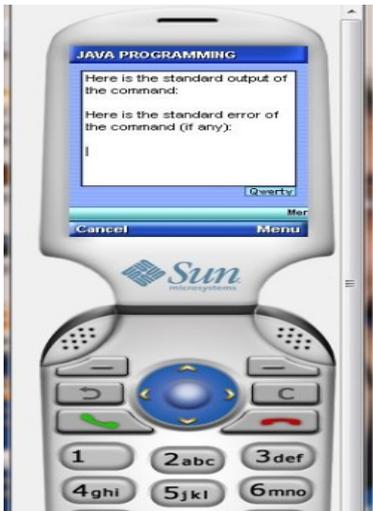
6. Java Program Compile Screen

After writing the code in java programming language, first of all user will compile the code by selecting the compile option. it is showing the java program compile screen.



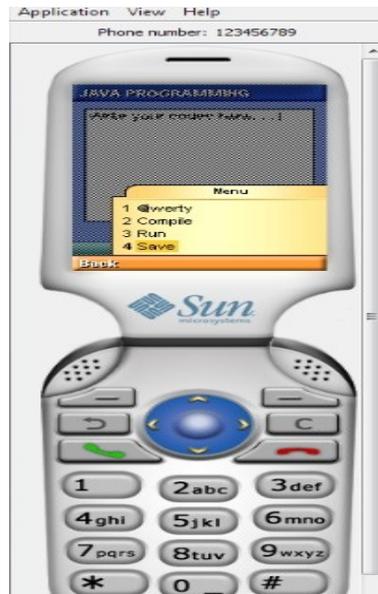
7. Java Program Compile output

After compilation of code, if any syntax or programming error exists then on this screen error is shown to user.



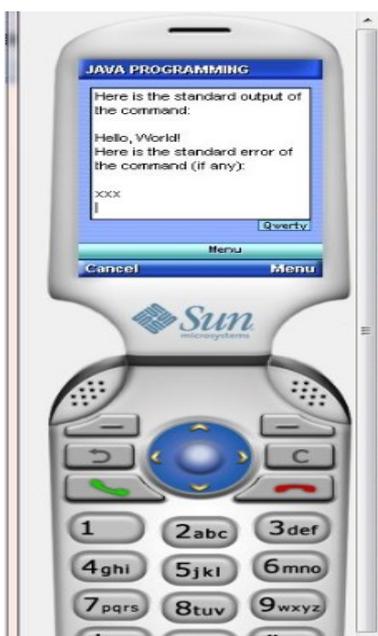
8. Java Program Run Screen

Now for running the code user will select Run option



9. Java Program Run Output

If any kind of run time error is generated then it is shown on this screen otherwise correct output display at here.



10. Java Program Save Screen

V. CONCLUSION & FUTURE SCOPE

In the beginning it was the hypothesis to develop such software which can execute Mobile Programming coding platform independently. This project is a mobile application. It includes several efficient modules in solving the purpose to execute all kind of programming using a mobile phone. After going through steps of developing this project it is concluded that main objective is achieved which was to enhance uses of mobile phone and to make us able to do programming platform independently. After implementation of this software "Mobile Programming" there is no need to use a desktop or laptop for coding purpose, a person just need to install this software in to his mobile phone and he can access all the functionality of this one (only on availability of GPRS services).

REFERENCES

- [1] Griswold, William G., et al. "Activecampus: Experiments in community-oriented ubiquitous computing." *Computer* 37.10 (2004): 73-81.
- [2] Carroll, Aaron E., Sunil Saluja, and Peter Tarczy-Hornoch. "Development of a Personal Digital Assistant (PDA) based client/server NICU patient data and charting system." *Proceedings of the AMIA Symposium*. American Medical Informatics Association, 2001.
- [3] Isakow, Anna, and Hao Shi. "Review of J2ME and J2MEbased Mobile Applications." *International Journal of Communication and Network Security* 8.2 (2008): 189-198.
- [4] Fraternali, Piero. "Tools and approaches for developing data-intensive Web applications: a survey." *ACM Computing Surveys (CSUR)* 31.3 (1999): 227-263.
- [5] Adnan, Syed Farid Syed, Habibah Hashim, and Idnin Pasya. "Implementation of MIDlet application on probability of Alzheimer's disease via mobile phone." In *Computer Applications and Industrial Electronics (ICCAIE), 2011 IEEE International Conference on*, pp. 113-116. IEEE, 2011.
- [6] Blom, Sören, Matthias Book, Volker Gruhn, Ruslan Hrushchak, and A. Kohler. "Write once, run anywhere a survey of mobile runtime environments." In *Grid and Pervasive Computing Workshops, 2008. GPC Workshops' 08. The 3rd International Conference on*, pp. 132-137. IEEE, 2008.
- [7] Thorn, Tommy. "Programming languages for mobile code." *ACM Computing Surveys (CSUR)* 29, no. 3 (1997): 213-239.
- [8] Lange, Danny B., and Oshima Mitsuru. *Programming and Deploying Java Mobile Agents Aglets*. Addison-Wesley Longman Publishing Co., Inc., 1998.
- [9] Mueller, John Paul. *Mastering Web Development with Microsoft Visual Studio 2005*. John Wiley & Sons, 2006.