Windows Media Player Remote Control

(Wi.M.P.R.C)

By

Khairul Anwar Bin Ali (2462)

Final Draft submitted in partial fulfillment

of the requirements for the

Bachelor of Technology (Hons)

(Information System)

JULY 2005

Universiti Teknologi PETRONAS

Bandar Seri Iskandar

31750 Tronoh

Perak Darul Ridzuan

CERTIFICATION OF APPROVAL

A Windows Media Player Remote Control

by

Khairul Anwar Bin Ali

Final Draft submitted to the
Information Technology Programme
Universiti Teknologi PETRONAS
In partial fulfillment of the requirement for the
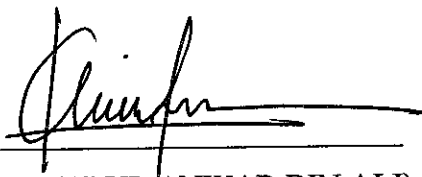BACHELOR OF TECHNOLOGY (Hons)
(INFORMATION SYSTEM)

Approved by,

(Ms Nazleeni Samiha Haron)

UNIVERSITI TEKNOLOGI PETRONAS
TRONOH, PERAK
June 2005

CERTIFICATION OF ORIGINALITY

This is to certify that I am responsible for the work submitted in this project, that the original work is my own except as specified in the references and acknowledgements, and that the original work contained herein have not been undertaken or done by unspecified sources and persons.

(KHAIRUL ANWAR BIN ALI)

ABSTRACT

Mobile remote control is an application embedded in mobile phone with Symbian Software Ltd Java or all Java-based enable. They can control various types of devices and software via Bluetooth or Infra Red connection. Most of the mobile phone application uses the Java platform known as Java 2 Mobile Edition as their programming language. In this paper, the author presents an approach to develop a Windows Media Player Remote Control (Wi.M.P.R.C) which can control basic function of the application such as play or pause. The main purpose of this project is to develop an application to control Windows Media Player via Bluetooth. This application will reduce the usage amount of keyboard and mouse to control the Windows Media Player and also to fully utilize the function of mobile phone for not just to talk.

ACKNOWLEDGEMENT

Bismillah ar-Rahmani Ar-Raheem

In the Name of Allah, The Most Compassionate, the Most Merciful

First and foremost I would like to recite my greatest gratitude to the Most Merciful Allah for giving me the opportunity in completing this manuscript on time and without much hassle or problem. Without His observance in giving me the chance in finishing the report, there might be major problem which can resulted in delay of turning in the report in the time constrain.

One of the greatest pleasures of writing a report is acknowledging the efforts of many people whose names may not appear on the cover, but without that hardworking cooperation, friendship and understanding, producing the report could have been impossible.

I would like to dedicate my special gratitude to my Final Year Project supervisor; Ms Nazleeni Samiha Haron. His insights, advices and encouragement have been invaluable throughout this project.

Also, a lot of appreciation goes to each UTP IT/IS Lecturers and FYP Committee that has guide and give advices to me regarding this FYP including all other party who involved directly or indirectly, for their assistance and precious time spent over me.

Not to be left behind, my special thanks also goes to my beloved family (Encik Hj Ali Abdul Rahman, Puan Hjh. Atma Ishak, Norasmalia, Azlan, Julia, Hafizul and Najihah) of course; all of my wonderful housemates who collectively create work culture that supports and nurtures my continued learning process and personal development.

Your guidance and assistance is highly appreciated. Thank You.

TABLE OF CONTENTS

LIST OF FIGURES:

LIST OF TABLES

ABBREVIATION AND NOMENCLATURES

FYP – Final Year Project

HCI – Human Computer Interaction

J2ME – Java 2 Mobile Edition

OS – Operating System

RAD – Rapid Application Development

UTP – Universiti Teknologi Petronas

BT - Bluetooth

WMP – Windows Media Player

WiMPRC – Windows Media Player Remote Control

WiMPRCsvr – Windows Media Player Remote Control Server

WiMPRCcli – Windows Media Player Remote Control Client

WWW – World Wide Web

CHAPTER 1

INTRODUCTION

1.1 Background Of Study

A mobile phone application can be defined as the wireless program that embedded into mobile peripheral such as mobile phone, palm handled and pocket pc. Mobile application nowadays grows rapidly as the mobile phone manufacturer continuously release more advance technology to the mobile phone. Application such as java games, multimedia player, office tools are the most demanding in the market today's. Mobile application only needs small size of memory and will suit into mobile devices.

Remote control application is the most popular application in mobile phones nowadays. Either it can controls television or computer. In this study, the author will develop a remote control application which can controls the Windows Media Player which is a standard media player build in with the Windows operating system. Windows Media Player Remote Control (WiMPRC) consists of two applications, which allow your mobile device to operate as a generic remote control which the desktop running on the user computer or server. Potentially, any kind of desktop application can be controlled – however in the current moment, it just focuses on Windows Media Player. The two applications mention are Windows Media Player Remote Control Server (WiMPRCsvr) which installed in the user pc to operate as a server and Windows Media Player Remote Control Client (WiMPRCcli) is a MIDlet which is installed in mobile phone to behave as a remote control. A MIDlet is a class that extends the class javax.microedition.MIDlet. In other words, it is the superclass of all MIDP applications.

The WiMPRCcli MIDlet application operates on any MIDP 1.0 compatible device that has at least color support and screen size greater than 90 x 90. Almost all new Java enable mobile phones today fulfill these requirements. The MIDlet communicates with the WiMPRCsvr using Bluetooth Communication Port. The following functionality can be used to remotely control the WMP.

1

WiMPRCsvr is a desktop-application deployable on Windows NT or higher platform. Since it was developed using Java technology, it is possible to run under other operating system such as Linux or Solaris. It provides a link between the server's WMP with the mobile phone embedded with the remote control.

1.2   Problems Statement

Nowadays, MP3 and MPEG file is very common to all computer users, whereby they can listen to the music and also watch the video clips and movies. Most of them are using Windows Media Player as the default media player because the application comes with the Windows operating system. There are several problems has occur in controlling the WMP. The lists are as below.

1.2.1   Problem Identification

i. Users have to be in front of the PC to control the Windows Media Player.

   a. Users have to be in front of the computer because they are attached to keyboard and mouse in controlling Windows Media Player.

ii. Users have to spend a lot of time using keyboard and mouse.

   a. Users spending much time using keyboard and mouse to control the Windows Media Player functions.

iii. New user need more time in navigating and using the new application.

   a. Novice users will have to spend time to adapt and get use to the new application.

iv. New user need more time to adjust or to understand the Windows Media Player skins.

   a. User will easily get confuse by the skins when navigating using mouse, due to the response time to find the play and next button.

1.2.2 Significant of the Project

The application is build to give the user more control of the Windows Media Player and to reduce the time spending in front of PC with the keyboards and mouse in controlling it. It can also increase the usability of the current application from the mobile phone. This project also acts as a framework for others J2ME developers in developing another wireless remote control. This application is better than others (Infra Red and Radio Frequency) because it uses the Bluetooth technology, which allows us to control up to 100 meters radius with a 1Mbps transfer rate. This application is much better than the wireless keyboard and mouse in term of size, coverage and functionality. The size is much smaller than keyboard and mouse, it can cover up to 10 times from the wireless keyboard and mouse and it just one press away to execute command.

3

1.3 Objectives

i. To create a remote control application to be implemented in any mobile phone with Java and Bluetooth features.

ii. To control basic function of Windows Media Player via Bluetooth.

iii. To develop an application that allows users to remotely control Windows Media Player.

iv. To minimize time to control Windows Media Player.

v. To use the Bluetooth technology as a base for connection between the mobile phone and PC.

vi. To enhance the functionality of mobile phone.

1.4 Scope Of Study

This project focuses on developing a remote control using Java 2 Micro Edition. The scopes of study which the author has to cover are Java 2 Micro Edition, and Bluetooth Technology.

The author is responsible to have a truly understanding of J2ME programming to develop the interface and functionality within the mobile application. The author uses J2ME rather than C++ is because J2ME can be implemented with any kind of mobile phone with Java enable. Meanwhile, the C++ only works with smart mobile phones which have Symbian Operating System.

The author also has to understand the Bluetooth technology and the architecture. The Bluetooth will act as the connection tools to connect the client (WiMPRCcli) with the server. The study will also include the implementation of Bluetooth in J2ME.

The author also include the BEMUSED application as a reference to develop the system flow and interaface. The BEMUSED application include the server and client which is developed using C# programming.

4

1.4.1 Relevancy of the project

This project is very useful for the users who want more to do with their mobile phone as they can perform real-time remote control the Windows Media Player or any other application in their computer without having them to be in front of the monitor. Hence, this project can become an additional tool for user who always on the run and do not have time to be in front of their monitor.

1.4.2 Feasibility of the Project within the Scope and Time Frame

By using RAD as a methodology, the stages from analysis, design construction and implementation can be compressed together due to the short time frame which time to develop the product less than 4 months. All the research and development is done during that time period. The author also has limited understanding of the programming language (J2ME) and Bluetooth technology and need more time to fully understand it.

CHAPTER 2

LITERATURE REVIEW

At this early stage, the project will basically look into the research of the utilization of J2ME system simulation. A thorough literature review would be done through reference books, internet and journals to further understand in order to determine the development procedure and to learn the basic process that is used to develop the project.

2.1   Windows Media Player

Windows Media Player is the most popular application among Windows® users because it is included with the operating system. This software not only can play video, but also music and photo. These additional features attract many Windows® user to use this application compared to others.

"By giving you easy access to all of your multimedia files (music, photos and video) and helping you accomplish the tasks you're likely what to do with them, WMP 10 at least equals and in our estimation surpasses the current leader in the field." – [1]

"PC Magazine has given the player the coveted PC Magazine Editor's Choice Award and a rating of 4.5 out of 5.0." – [2]

2.2   Remote Control

Remote control can decrease time in controlling the application and also comfort which allow you to control certain application and hardware anywhere within the radius. By using the Bluetooth technology, user can control the application or hardware in a wide range of area. Although the Bluetooth technology is a rarely new, but it grows rapidly. Bluetooth is similar to Radio Frequency (RF), which allow you to use it without line-of-sight issues, anywhere in a room.

"A personal remote gives a presenter several advantages over a standard projector remote. First and foremost, it provides more options. With few exceptions, the projector's remote is designed to access the projector's menu system rather than handle PowerPoint or other slideshow programs. Presentation features are always a secondary concern for these remotes. Second, a personal remote becomes the remote you use for all of your presentations, which allows you to get comfortable with it. Third, if you always carry it with you, you'll never have to rely on a remote being available on location." – [3]

2.3   Bluetooth technology

Bluetooth provides the best wireless connection compared to InfraRed technology. Bluetooth can provide up to 100m range of connection. Meanwhile the InfraRed can only provide 1m parallel connection.

"The effective distance range for Bluetooth to be in optimal performance is about 10m and can span up to 100m at higher power levels. Bluetooth has an advantage over IR because the furthest distance IR can reach is only 1m." – [4]

"The complementary of Bluetooth is the Infrared device of the Infrared Data Association (IrDA). During the last few months, it has been argued that Bluetooth has been created as a substitute for Infrared. Though, this is not quite true: Bluetooth has been invented as an enhancement of Infrared, especially in terms of low cost, small volume, low power, and the accomplishment of Infrared's limitations, such as the its unidirectional connections, the connectivity range of only meter or its limitation to point-to-point connections. Although Bluetooth has actually coped with these constraints, e.g. by omni directional connections, the connectivity range of up to 10 meters, or its capability for point-to-multipoint connections, the two technologies are quite complementary. While Bluetooth is very applicative for networking, Infrared is more appropriate for the direct connections, e.g. for exchanging business cards." – [5]

2.4   Java 2 Micro Edition

In my findings, the java technology is the best and suitable programming language to use in developing the mobile application. The programming language is called Java 2 Micro Edition which is one of the Java platforms for consumer and embedded devices such as mobile phone, pager and personal organizer.

"A Java API enables application to run on different types of hardware, operating systems, and classes of device. The Java language provides several other benefits such as rapid development of application because of the better abstraction and high-level programming constructs provided by an object-oriented programming language." – [6]

"The JavaTM 2 Platform, Micro Edition (J2METM) is the Java platform for consumer and embedded devices such as mobile phones, PDAs, TV set-top boxes, in-vehicle telematics systems, and a broad range of embedded devices." – [7]

Simplicity™ Version 3.0 for Palm OS® makes it easy to build great applications using Java technology that will run on palmOne™ devices, Treo™ smartphones, and many other devices that use the Palm OS® Platform. – **Data Representation Inc.**

Imagine an enterprise solution that allows Java developers to have true WYSIWYG integration of front and back-end applications. Dream no more! With Simplicity for Mobile Servers, now you can develop, view, and test Servlets and MIDlets simultaneously. – **Data Representation Inc.**

CHAPTER 3

METHODOLOGY/PROJECT WORK

3.1 Procedure Identification

In developing the system, I will use the five (5) phases of RAD (Rapid Application Development) with evolutionary prototyping that would be the most suitable development methodology for the project. The phases involved in this RAD are shown in Figure 3.1.

```
┌─────────────────────┐
│      Analysis       │
└─────────────────────┘
          │
          ▼
┌─────────────────────┐
│   Project Design    │
└─────────────────────┘
          │
          ▼
┌─────────────────────┐
│ Project Development │
└─────────────────────┘
          │        ▲
          ▼        │
┌─────────────────────┐
│   Project Testing   │
└─────────────────────┘
          │
          ▼
┌─────────────────────┐
│  Final Deliverable  │
└─────────────────────┘
```
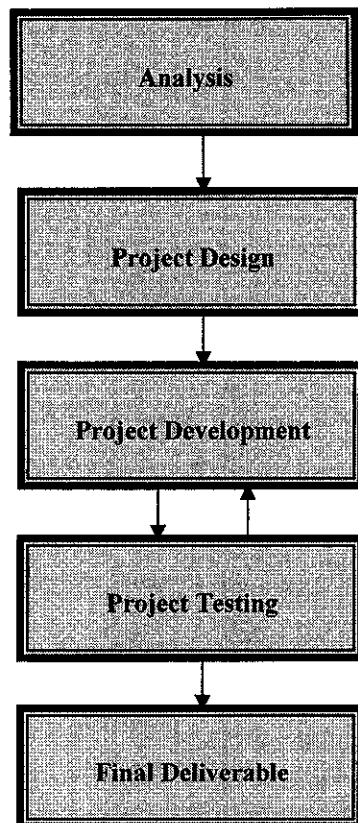
Figure 3.1 Rapid Application Development Phases

Evolutionary prototyping means that the any complete application of this will be based on the prototype itself. In other word, the prototype can be enhanced to be the final

9

finished product. If the prototype can be enhanced to be the final product, the time, cost and energy can be saved. These are some advantages of evolutionary prototyping. This technique is chosen because of the time constraint and limited knowledge of J2ME which the author encounter in developing the system. The five phases in RAD are further explained below.

3.2 Method

3.2.1 Analysis / Research

Analysis stage involves the activity of gathering information. The information gathered includes the problems that may arise, all the possible techniques, tools that are needed and the project milestone to develop the remote control. The purpose of doing the analysis or information gathering is to obtain as much information as possible to be able to understand better about the existing mobile application before the author can develop a new one. Among methods used to gather information are by conducting research on the sources available on the Internet websites, references, white papers and journals.

3.2.2 Project Design

During this stage, the project plan will be developed and it will be divided into several task - (refer to **Appendix 1** for project timeline) and each of the task will be given the estimate period to finish it as a milestone. These are to ensure that the project will be delivered on time. Besides that, the design phase is concerned with how the product will function according to the requirements. In this stage, the author have come out the design for the system work flow, the interface for server and client.

3.2.3 Project Development

The project development phase starts with constructing the interface for the server and then for the client. This is the entire backbone framework for the WMP Remote Control. The construction will be based on the techniques and milestone that are identified during the previous phase (project design phase). The tools required to develop this application is the J2SDK and J2ME Wireless Toolkit. For the graphic, Adobe Photoshop 8.0 and Microsoft Paint are used.

3.2.4 Project Testing

The remote control will not need to be fully developed to be tested but the project testing can be carried out in the prototype phase even before remote control's framework is completed. Project development stage and project testing will run concurrently after the development of the system is started. This is to ensure that each functionality meets the expectation.

3.2.5 Final Deliverable

For this phase the final report will be prepared and the product will be presented to the examiners. Lastly the final dissertation will be submitted to the supervisor. This phase is the last phase in the methodology for this project.

3.3 Tools

3.3.1 Developer's Specification

3.3.1.1 Software

The table below shows the lists of software are used for the mobile remote control development and their usage:

Table 3.1: Software for the Development and Their Usage

Software	Usage
Java 2 Software Development Kit 1.4	Java Compiler
J2ME Wireless Toolkit 2.2	Application testing platform
Microsoft Word	Preparation of documents
Microsoft Project	For schedule planning
Microsoft Power Point	For presentation/learning slide
Internet Explorer	For browsing the Internet in analysis phase
Adobe Photoshop 8.0	For image editing
MS Paint	For image editing

3.3.1.2 Development and Construction Hardware

Table 3.2 shows the hardware requirement specification of the computer for the development of the Windows Media Player Remote Control which exclude the monitor, keyboard and mouse:

Table 3.2: Minimum Hardware Requirement

Device	Requirement
Operating System	Microsoft Windows NT or higher
Processor	Intel® Pentium® III CPU 1 GHz or higher
Memory	128 MB of memory
Disk Space	10GB of free space
Bluetooth Dongle	Any standard

12

3.3.2 User's Specification

3.3.2.1 Software

Windows Media Player Remote Control Server needs to be installed in order to communicate with the Windows Media Player Remote Control Client.

3.3.2.2 Hardtware

The table below shows the minimum hardware requirement that the user must have in order to use or communicate with the remote control:

*N*ote: The list reflects the minimum requirement for personal computer

Table 3.3: User's Hardware Specification

Hardware	Specification
Operating system	Windows NT or higher
Processor	Pentium III 1.0 GHz or higher
RAM	128 MB or higher
Disk Space	At least 40 Mb free space
Monitor	15 inch (800 x 600 resolution)
Graphic card	At least High Color (16 bit)
Bluetooth Dongle	Any standard
Mouse and Keyboard	Any standard

CHAPTER 4

RESULTS AND DISCUSSION

4.1 Introduction

This section will discuss the findings on the J2ME and Bluetooth technology that has been identified and learned by the author in constructing WiMPRC, the remote control for Windows Media Player.

4.2 Findings

4.2.1 Analysis Phase

The author had first identified the problem regarding the development of the mobile application. Based on the previous problem statement, users still using the traditional way to control WMP which limited the flexibility in controlling it. They do not know what can be done to overcome it.

The author suggests a solution to cater this problem, which is, developing a remote control which integrated with mobile phone to control the WMP. The Bluetooth technology will be the medium to connect the computer with the mobile phone. The overall project progress and milestone is provided in the Appendix.

Within the Analysis phase, involved a lots of research before proceed to the next phase. The research includes Java 2 Mobile Edition, Bluetooth Technology and Human Computer Interaction as the main scope in developing this Final Year Project.

The auhor choose J2ME compared to others programming language because of several reasons. From the research and analysis that has been donem there are

several advantages of J2ME. For example, the advantages of J2ME between .Net framework are:

a) J2ME is much cheaper than the .Net framework. J2ME can be downloaded at www.java.sun.com for free.

b) J2ME can support all mobile platfrom which different from .Net framework which only support Pocket PC and Windows CE.

c) J2ME has a large market focus because it ranged from consumer to enterprise. Meanwhile, the .Net framework only marketed to enterprise.

Java 2 Micro Edition (J2ME) Platform is by far the most advanced and successful mobile application platform available today. However, with mobile commerce growing into a multibillion-dollar industry, serious competition is on the horizon from Microsoft. Microsoft's latest mobile commerce offering is the .Net Compact Framework (.Net CF). What exactly is .Net CF? How does it measure up to J2ME?

In Juntao Yuan article (2002), he has compared and elaborated the J2ME (J2ME/CDC and J2ME/CLDC) with .Net CF platforms. To summarize the comparison, please refer to Table 2.2.

Table 4.1: Comparison Between Three Mobile Application Platforms

	.Net Compact Framework	J2ME Connected Device Configuration	J2ME Connected Limited Device Configuration
Device requirement	Powerful, expensive	Powerful, expensive	Cheap, pervasive
Cost	High	High	Medium

15

Market focus	Enterprise	Enterprise	Consumer and enterprise
Language support	C#, VB.Net	Java	Java
Platforms	Pocket PC, Windows CE	Major mobile platforms except Palm OS	All mobile platforms
Byte code compatibility	Standard .Net CLR	Standard Java 2	Not compatible with J2SE or CDC
API compatibility	Subset of .Net	Subset of J2SE plus standard optional packages	Partial compatibility with CDC with additional standard optional packages
Native APIs	P/Invoke; consistent across supported devices	JNI; device- and OS-specific	N/A
Development tools	VS.Net 2003	Command line, vendor SDKs, CodeWarrior, and WebSphere	Command line, vendor SDKs, all major Java IDEs
Specification process	Single company	Community	Community
Service gateway	N/A	Run gateways as OSGi servlets; run gateway clients via vendor-specific SDKs	Run gateway clients via vendor-specific SDKs
Security model	Simplified .Net model	Full Java security manager	Limited Java 2 model supplemented by OTA specification
Client installation	ActiveSync, Internet Explorer download	Sync, download	Formal OTA specification
Life cycle management	N/A	OSGi for gateway apps, J2EE Client Provisioning Specification for generic clients	Included in OTA spec, works with J2EE Client Provisioning Specification

In Bluetooth technology, the author has to study and understand a several topics before he can proceed to the next phase. The topics include the Bluetooth protocol and profiles, Bluetooth Architecture, Bluetooth Object Exchange Protocol, Device Discovery and Service Discovery.

Human Computer and Interaction is important too because the author wants the application to be user friendly. All the design is made by the author himself by referring others mobile application and old projects.

4.2.2 Project Design Phase

Within the design phase, the system flow chart and user interfaces are listed as below:

4.2.2.1 The WiMPRCcli System Flow
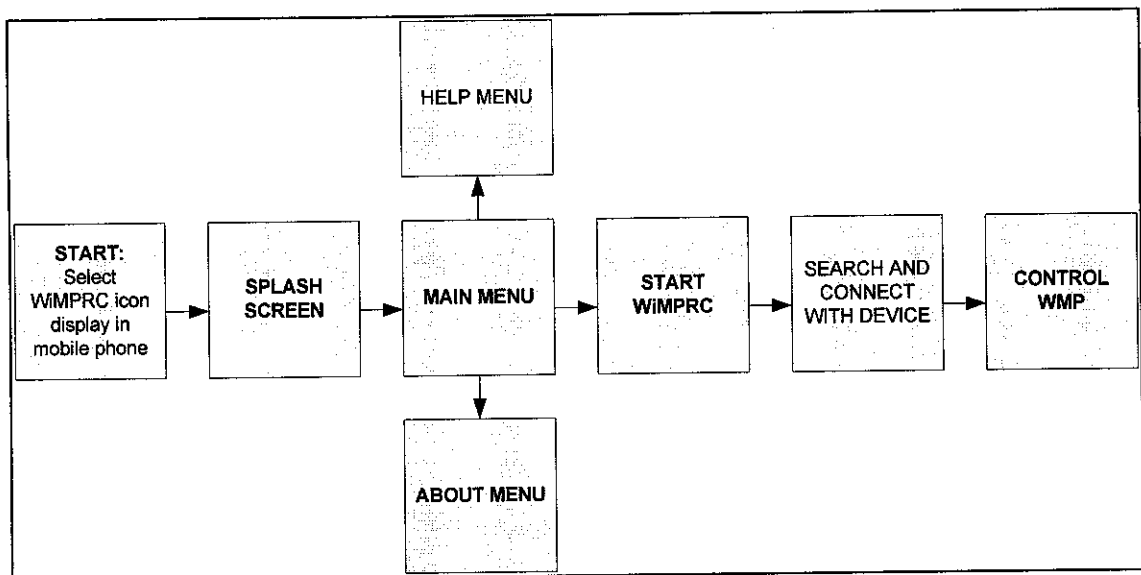
Figure 4.1 – WIMPRCcli System Flow Diagram

Firstly, user has to select the WiMPRC icon in their mobile phone. Then the user will be directed to the splash menu, which display the WiMPRC logo. The after four second, the menu will change to main menu which gives three options to user, Start WiMPRC, Help and About. In Help Menu, there is a brief desciption about the application and in About Menu exist the brief description about the author. When user select the Start WiMPRC option, user can search devices and

select the device he/she want to connect with. After the connection has been established, user can control the WMP as he/she desire.

4.2.2.2 The BEMUSED Server System Flow
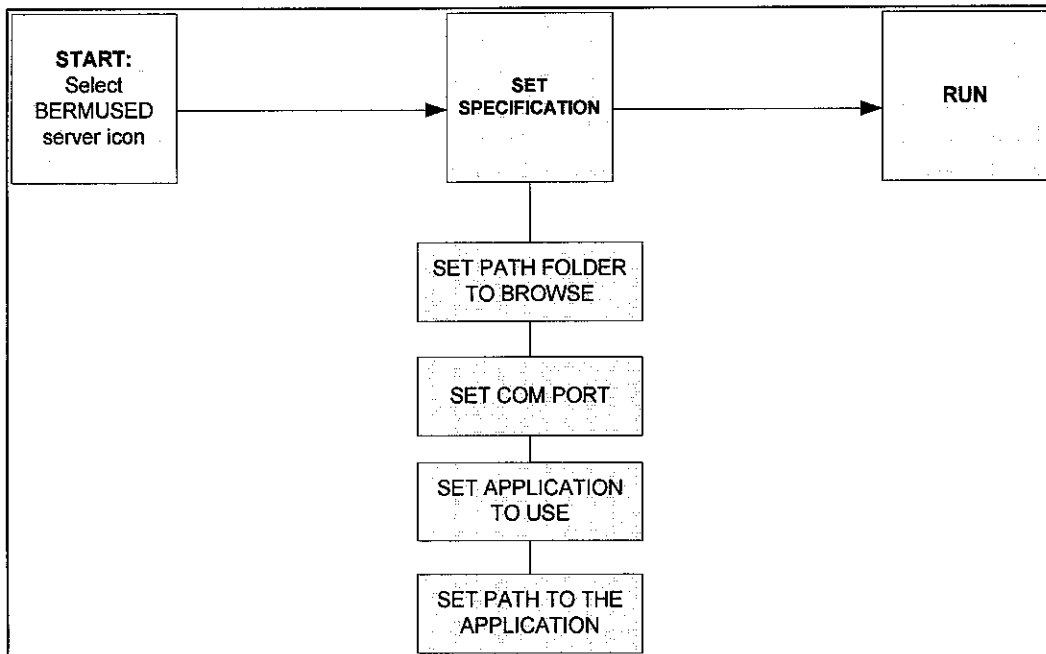
<u>**Figure 4.2 – BEMUSED Server System Flow Diagram**</u>

Firstly, user has to run the BEMUSED server and then he/she has to do some configuration. The configurations are set the path folder which user want to browse, set the COMMUNICATION PORT (COM PORT), set which application they want to use, currently just WMP and set the application path. User can determine the COM port from the Bluetooth properties. Make sure the Bluetooth is switch on. When user enters a command from WiMPRCcli, the BEMUSED server will receive and try to translate the command. If it recognize the command, it will execute it, and if otherwise, it will not execute.

4.2.2.3 The WiMPRC Use Case Diagram
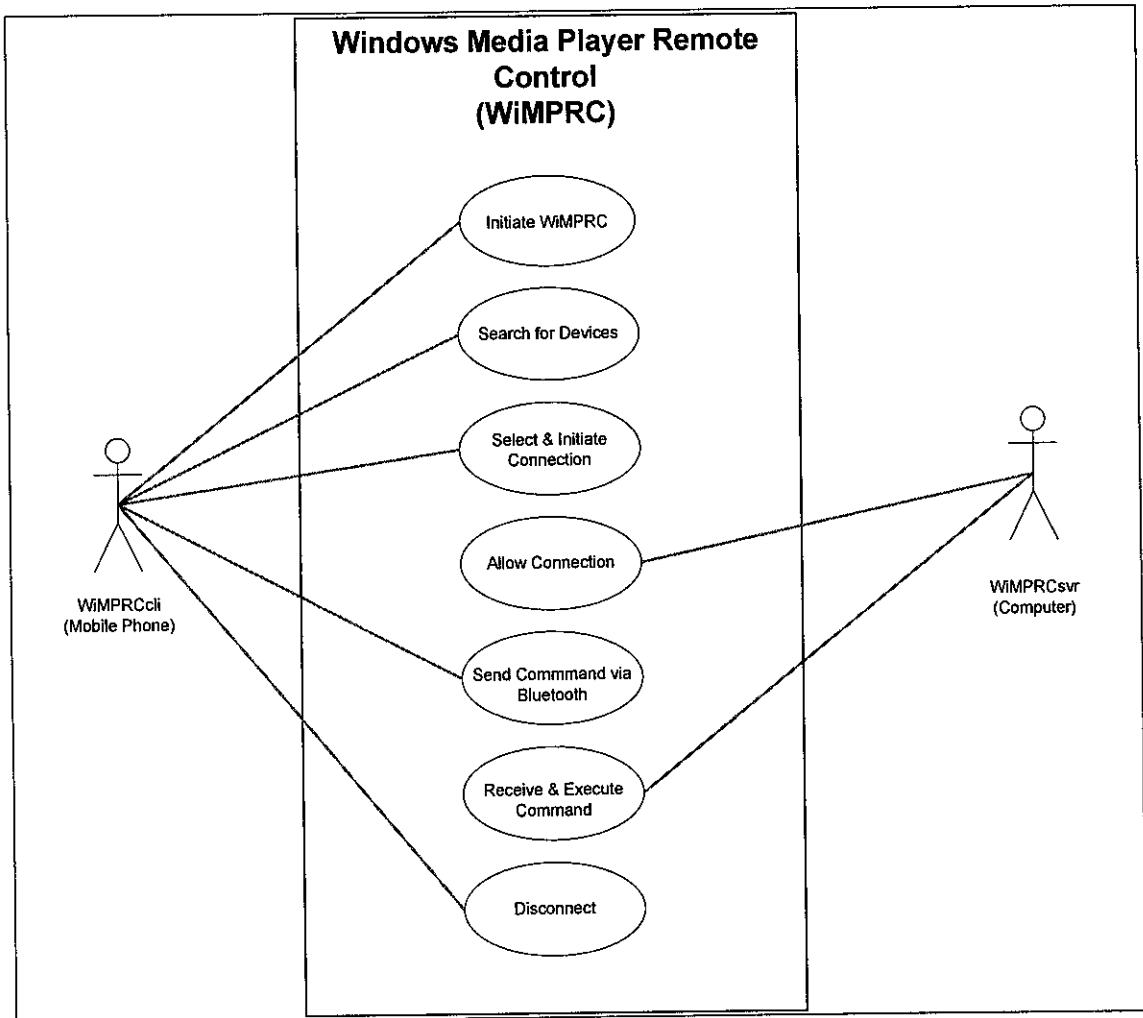
Figure 4.3 – WIMPRC Use Case Diagram

i. UC#1 – Run WiMPRC

User has to run the WiMPRC before he/she can use the application. WiMPRC consists of WiMPRCcli (in mobile phone) and WiMPRCsvr (in computer), which user has to run both of them to create the connection and run the application. In the serve menu, user has to configure the Serial Com Port, the music folder and the application they want to use (currently just Windows Media Player). Then, user has to enable the Bluetooth from mobile phone and computer.

ii. UC#2 – Search For Devices

When the user initiate the WiMPRCcli, its will automatically search for available device within the Bluetooth range. Then, user has to select which device they want to connect with.

iii. UC#3 – Select and Initiate Connection

When user has selected the device he/she want to connect with, the WiMPRCcli will send request to the device (computer) to connect with. Then the device will prompt the PIM Synchronization (which act as a PIN number to allow the connection) and user has to type in any number before send it to the WiMPRCcli. When WiMPRCcli receive the PIM Synchronization, he/she just has to key in the number before. If the number entered is correct, the connection has established.

iv. UC#4 – Send Command

Once the connection has established, user can send command to the server, which he/she can select it from the menu. The command include play, stop, pause, next track, previous track, volume up, volume down, minimize and full screen.

v. UC#5 – Receive and Execute Command

As soon as the server receives the command, it will interpret it and execute the command to the WMP. WMP will act as the user want simultaneously.

vi. UC#6 – Disconnect

User can disconnect the connection by selecting the disconnect function in the option menu or he/she just can simply switch off the Bluetooth either in mobile phone or computer. After the disconnection, user will not be able to control the WMP.

4.2.3 Project Development Phase

3.3.2.1 Screen Shot (WiMPRC)
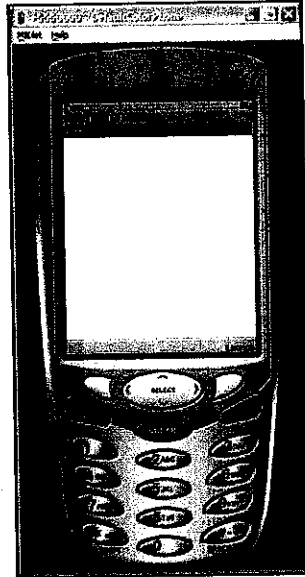
Figure 4.4 – Start Screen

User has to select the WiMPRC and select launch.
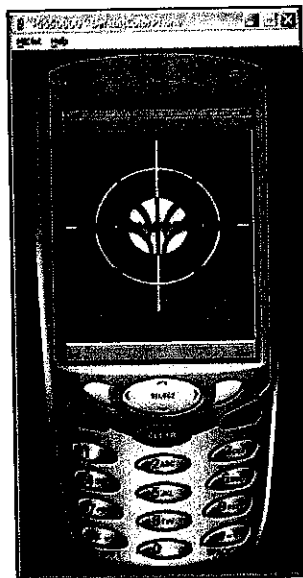
Figure 4.5 – Splash Screen

Then the splash screen will be displayed. The screen will last about four second before it directed to the Main Menu.

Figure 4.6 – Main Menu

In main menu, user will be provided with three options, Start WiMPRC, Help and About. User can select any option they desire. To start the application, user must select the Start WiMPRC option.
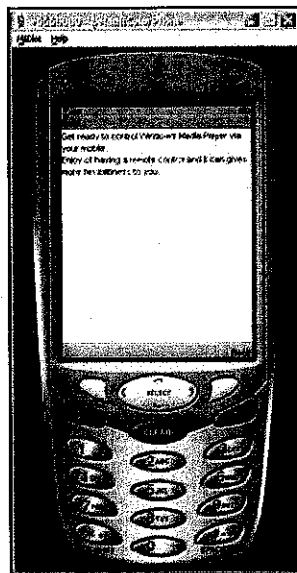
Figure 4.7 – Help Menu

In Help Menu briefly describe the application usage and the vision of the author.
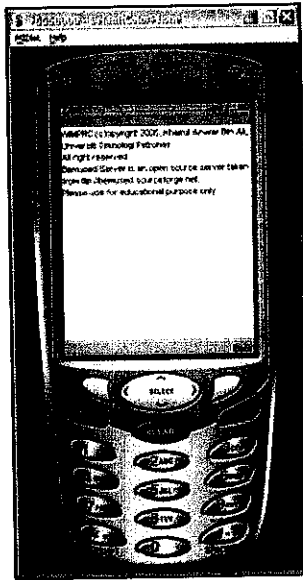
Figure 4.8 – About Menu

Meanwhile, in About Menu it provides user with the description about the author and the BEMUSED server.
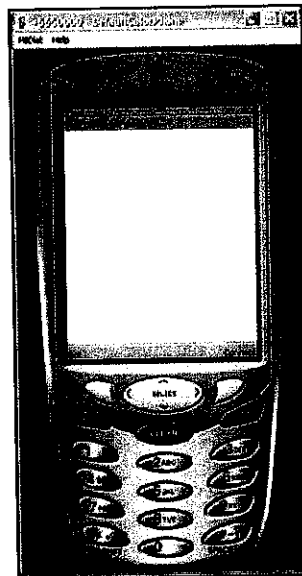
Figure 4.9 – Search Menu

When user select the Start WiMPRC option, the application will start searching for the nearby Bluetooth devices.
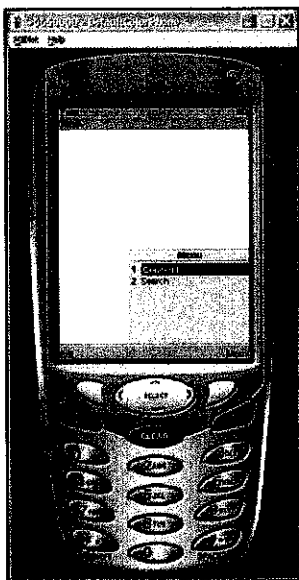
Figure 4.1.1 – Connect Menu

When the WiMPRC found nearby devices, user can select only one device he/she want to connect with. When the connection has established, user can now control the WMP.
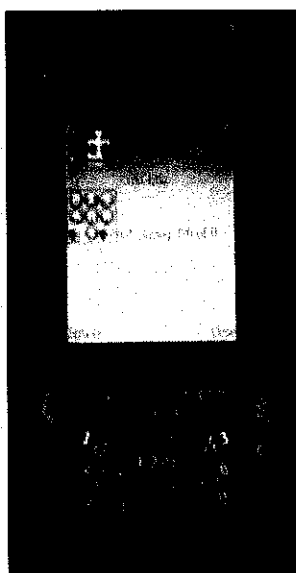
Figure 4.1.2 – Controller Menu

When the WiMPRC has established the connection, the controller menu will be appeared. In the menu, it has *next, play, back, fast forward, fast*

backward, pause, stop, volume up and volume down. User can control it by using keypad. For example keypad number 2 for *play*.
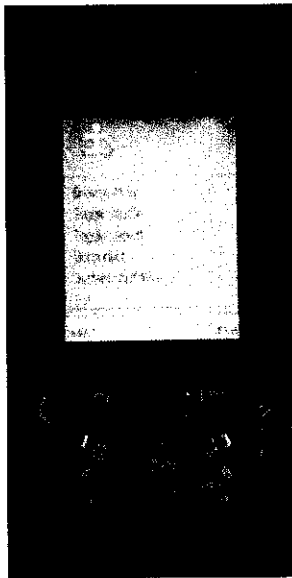
Figure 4.1.3 – Controller Option Menu

When the user select the option in Controller Menu, it will prompt out the Controller Option Menu. In this menu, user can select several functions such as *browse music, toggle shuffle, toggle repeat, disconnect and shutdown.* Toggle shuffle will function as to on the shuffle or to off it. Same goes to the toggle repeat.

4.2.4　Project Testing Phase

When all the design and development has completed, the author proceeed to the project testing phase. Because of the time constrained, the author only managed to complete the application in client side, which call as WiMPRCcli. The application is installed in the mobile phone.

As an alternative to test the application, the author has decided to use the BEMUSED server as the server for WiMPRC. The BEMUSED server is an open source server which can be downloaded from www.bemused.sourceforge.net for educational purpose only.

During the testing phase, the application does not work as it suppose to do. This is because, the server could not quite compatible with the WiMPRC. The remote could only open the WMP when the play button is selected. The author has try in others PC, and the result is still the same. If the PC is running on below 256 Mb, it will restart itself after a few minutes when the connection has been established.

The testing phase involve in three section, first the menu flow and link. Then the connection with the BEMUSED server. The last one is the controlling the WMP with the application. All the testing is done by the author using Nokia 6630 and Sony Ericson T630 as the final testing.

4.2.5　Final Deliverable Phase

For this phase the final report will be prepared and the product will be presented to the examiners. Lastly the final dissertation will be submitted to the supervisor. This phase is the last phase in the methodology for this project.

26

4.3 Results

Table 4.2: Test Schedule

TEST TOPIC	TEST CONDUCTED	RESULT
Installation	Transfer and install the WiMPRCcli in mobile phone	Pass
Search Devices	WiMPRC start to search nearby Bluetooth devices	Pass
Connecting with the device	WiMPRC request to connect with the server	Pass
Controlling the WMP	WiMPRC send control command to the server (e.g: Play)	Fail: the server do not response as required. Only open up the WMP.
Disconnect	WiMRC disconnect the current connection with the server	Pass

The idea of having an remote control for application in computer is very positive. As for this project, it can controls Windows Media Player. The author only manage to complete the client part which is installed in the mobile phone. Becauce of the time constraint, the server could not be develop. The overall completion of this project is about 70%.

27

CHAPTER 5

CONCLUSION AND RECOMMENDATION

5.1 Conclusion

Windows Media Player Remote Control gives many benefits to the user because most of the Windows operating system's users use their Windows Media Player as the main application to run movies and songs. It gives user more flexibility in controlling the Windows Media Player. Normally users have to sit in front of the computer to navigate the Windows Media Player using keyboard and mouse. But with the Windows Media Player Remote Control, users can navigate Windows Media Player up to 100m radius from the computer at any convenient. Users can also minimize time spend to navigate and control the Windows Media Player. For example, usually user has to click on the play icon to start playing the songs or movies, which take times. But it is different when using Windows Media Player Remote Control, user just have to press one button to execute command in Windows Media Player.

5.2   Recommendation

There are a couple of recommendations which can be implemented to enhance the WiMPRC. The recommendation is divided into two categories which is the client (WiMPRCcli) and the server (WiMPRCsvr).

In the client side, the author recommends a few functionalities can be implemented to enhance the WiMPRC. In interface design, the author recommends that the file browser menu, remote menu, and playlist menu will be in separate tabwindows. This will gives user more understanding and increase response time to control the remote control. For the remote menu, the author recommends the future developers to create more skins to this menu which user can easily select and change depends on their interest. In the browser tabwindows it will displays the list of audio and video file in the selected folder, which users can easily browse and play any file they want. The folder can be assign in the WiMPRCsvr menu. The author also recommends that this remote control application will have the functionality to control any application users want to such as Microsoft Office Power Point, WinAmp, Power Dvd and many more. The future developer also can add the voice command function to replace the key-pressing function because nowadays, many mobile phones come with voice command function. When this function is available, user has more flexibility in controlling applications.

Next recommendation goes to the WiMPRCsvr side. The author recommends the server can detect the Bluetooth Serial Port automatically when the server started. Next one is, the event viewer dialog window. This event viewer will display all the commands that have been query from the client and the commands that have been executed from the server itself. In the interface module, future developer should build an application browser for user to choose the application they wish to control in an drop-down menu. This will decrease time in browsing through the hard drive to select the application to use. But, the browsing button is still remaining, for any custom application to select.

REFERENCES

[1] MusicMatch Jukebox 9,

Http://www.pcmag.com/article2/0,1895,1641331,00.asp)

[2] PC Magazine Author,

Http://www.pcmag.com/article2/0,1759,1641825,00.asp)

[3] Michael Goldstein,
Http://www.presentations.com/presentations/technology/article_display.jsp?v
nu_content_id=1000614671

[4] Http://www.freewebs.com/grpe/infraredvsbluetooth.htm

[5] Nnamdi Charles Akalugwu, Justas Bugnevicius, Christine Evers, Farhad

Pourtaran - International University Bremen,

Http://www.faculty.iu-bremen.de/birk/lectures/PC101-

2003/17bluetooth/bluetooth/intro.html

[6] C Bala Kumar – Book, Bluetooth Application Programming with the Java

APIs.

[7] http://developers.sun.com/

[8] Http://www.codeproject.com/netcf/pocketremote.asp

[9] Http://elektron.its.tudelft.nl/~jkohne76/

[10]   Mohd Sazizam Bin Mohd Saad (Dec 2004) - Mobile Remote Control

(MORECO)

[11]   Mohd Khairul Zarir Bin Ahmed Lokman (July 2004) – Mobile Game:

Digital Companion as Student Organizer

[12]   Final Year Project Guideline – Universiti Teknologi Petronas

[13]   Http://fred.hexbox.de/bemused/

[14]   Http://www.benhui.net/modules.php?name=Bluetooth

[15]   Http://elektron.its.tudelft.nl/~jkohne76/faq.php

[16]   Http://bemused.sourceforge.net/related

[17]   Http://www.microjava.com/articles/links

[18]   Http://java.sun.com/products/sjwtoolkit/download-2_2.html

[19]   Http://www.forum.nokia.com/tools

[20]   Http://www.benhui.net/modules.php?name=Bluetooth&page=bluechat

[21]   Http://www.forum.nokia.com/main/1,6566,034-301,00.html

[22]   Http://www.forum.nokia.com/main/0,,034-4,00.html

[23]   Michael Juntao – Book, Enterprise J2ME: Developing Java Mobile
        Applications

APPENDICES

APPENDICES 1:

Gannt Chart

APPENDIX 1: GANTT CHART

ACTIVITIES	WEEKS													
	1	2	3	4	5	6	7	8	9	10	11	12	13	14
ANALYSIS PHASE	█			█										
RESEARCH ON J2ME	▒	▒												
RESEARCH ON BLUETOOTH			▒	▒										
PROJECT DESIGN PHASE				█			█							
CLIENT FLOW SYSTEM				▒	▒	▒								
SERVER FLOW SYSTEM				▒	▒	▒	▒							
PROJECT DEVELOPMENT PHASE							█					█		
WIMPRC DESIGN INTERFACE							▒	▒	▒	▒	▒	▒		
WIMPRCcli FUNCTIONALITY									▒	▒	▒	▒	▒	
PROJECT TESTING PHASE													█	
TEST APPLICATION AND MODIFY ERRORS													▒	
FINAL DELIVERABLE														█
FINAL PRESENTATION PREPARATION														▒
FINAL REPORT SUBMISSION														▒

APPENDICES 2:

Sample Survey Forms

Survey Form For Windows Media Player

1. Have you ever use Windows Media Player?
   a) Yes   b) No

2. How frequent do you use Windows Media Player?
   a) Always   b) Sometimes   c) Never

3. Have you try Windows Media Player in Skin Mode?
   a) Yes   b) No

4. How do you rate your first time navigating Windows Media Player in skin mode?
   a) Hard   b) Normal   c) Easy

5. Do you think you will need more time to understand and get use with the skins?
   a) Yes   b) No

6. Any suggestion/ recommend on how to increase the efficiency in using Windows Media Player?
   - Build or develop easy to understand skin
   - Develop a remote control like t.v. remote

APPENDICES 3:

Sample Source Code

```java
import java.io.IOException;
import javax.microedition.io.StreamConnection;
import javax.microedition.lcdui.Display;
import protocol.MusicPlayer;
import protocol.ProtocolStatus;
import protocol.StatusUpdateListener;
import WiMPRC.WiMPRCProtocol;
public class Controller implements StatusUpdateListener {
        Main owner;
        SearchForm searchForm;
        BrowseForm browseForm;
        PlaylistForm playlistForm;
        ControlForm controlForm;
        JumpForm jumpForm;
        Settings settings;
        StreamConnection currentConnection;
        WiMPRCProtocol protocol;
        public Controller(Main owner) {
                this.owner = owner;
                protocol = new WiMPRCProtocol(Display.getDisplay(owner));
                protocol.registerStatusUpdateListener(this);
                settings = new Settings();
                searchForm = new SearchForm(this);
                browseForm = new BrowseForm(this);
                playlistForm = new PlaylistForm(this);
                controlForm = new ControlForm(this);
                jumpForm = new JumpForm(this);
                }
        Settings getSettings() {
                return settings;
        }
        public void start() {
                Display.getDisplay(owner).setCurrent(searchForm);
        }
        public void showPlaylist() {
                protocol.refreshPlaylist();
                Display.getDisplay(owner).setCurrent(playlistForm);
        }
        public void showBrowser() {
                showBrowser(true);
        }
        public void showBrowser(boolean refresh) {
                if (refresh) protocol.fileBrowser.readDir();
                Display.getDisplay(owner).setCurrent(browseForm);
        }
        public void showController() {
                Display.getDisplay(owner).setCurrent(controlForm);
        }
        public void exit() {
                owner.destroyApp(false);
                owner.notifyDestroyed();
        }
        public void showJumper() {
                Display.getDisplay(owner).setCurrent(jumpForm);
        }
        void destroy() {
                disconnect();
                settings.save();
                if (protocol != null) {
                        protocol = null;
                }
        }
        void connectTo(String url, String name) {
                settings.setLastUsedDevice(url, name);
                settings.save();
                protocol.setConnectionURL(url);
        }
        void disconnect() {
                protocol.closeConnection();
        }
```

```java
public void WiMPRCStatusChanged(ProtocolStatus status) {
    Display display = Display.getDisplay(owner);
    if (status.connected && (display.getCurrent() == searchForm)) {
            display.setCurrent(controlForm);
    }
    else if (status.connected == false && (display.getCurrent() != searchForm)) {
            display.setCurrent(searchForm);
    }

    WiMPRCProtocol.PlaybackStatus s = protocol.status();
    controlForm.setStatus(s);

    playlistForm.updatePlaylist(protocol.currentTrackNum(), protocol.playlist());
    browseForm.updateDirList(protocol.fileBrowser.dirs(), protocol.fileBrowser.files(),
protocol.fileBrowser.dirChanged());
    }

    MusicPlayer getPlayer() {
            return protocol;
    }

}
```

MAIN MENU.JAVA

```java
import javax.microedition.lcdui.*;
import java.util.Vector;
public class MainMenu extends List implements CommandListener
{   private Main main;
    private Command Select;
    private Command Exit;
    protected MainMenu(Main main)
    {
      super("Main Menu", IMPLICIT);
            goMainMenu();
      this.main = main;

            Select = new Command ("Select", Command.OK, 1);
            Exit = new Command ("Exit", Command.EXIT, 2);
            addCommand(Select);
            addCommand(Exit);
            setCommandListener(this);
    }
    public void goMainMenu()
    {
            append("Start WiMPRC", null);
            append("Help", null);
            append("About", null);
    }
    public void commandAction(Command command, Displayable displayable)
    {
            if (command == Select)
            {
                    if (getString(getSelectedIndex()) == "Start WiMPRC")
                    {
                            main.activateController();
                    }
                    if (getString(getSelectedIndex()) == "Help")
                    {
                            main.activateHelp();
                    }
                    if (getString(getSelectedIndex()) == "About")
                    {
                            main.activateAbout();
                    }
            }
      if (command == Exit)
        main.destroyApp(true);
    }
}
```

```java
import javax.microedition.lcdui.*;
import javax.microedition.midlet.*;
public class Main
extends MIDlet {
        private StringItem stateItem;
        private Command exitCommand;
        private Command connectCommand;
        private Command disconnectCommand;
        private Splash splash;
        private Controller controller;
        private MainMenu mainmenu;
        private Help help;
        private About about;
        public Main() {
                System.out.println("Main.Main()");
                splash = new Splash(this);
        }
        public void startApp() {
                System.out.println("Main.startApp()");
                Display.getDisplay(this).setCurrent(splash);
        }
        public void pauseApp() {
                System.out.println("Main.pauseApp()");
        }
        public void destroyApp(boolean unconditional) {
                controller.destroy();
        }
        public void activateController() {
                controller = new Controller(this);
                controller.start();
        }
        public void activateMainMenu() {
                mainmenu = new MainMenu(this);
                Display.getDisplay(this).setCurrent(mainmenu);
        }
        public void activateHelp() {
                help = new Help(this);
                Display.getDisplay(this).setCurrent(help);
        }
        public void activateAbout() {
                about = new About(this);
                Display.getDisplay(this).setCurrent(about);
        }
        private void destroy() {
        }
        private void updateStatus() {
        }
}
```

```java
import javax.microedition.lcdui.*;
import java.io.IOException;
public class Splash extends Form implements Runnable, CommandListener
{           private Main main;
            private boolean running = true;
            private ImageItem SplashItem;
            private Command Continue;
            private long cycleStartTime;
            protected Splash(Main main)
            {
                        super ("");
                        this.main = main;
                        initResources();
                        Thread t = new Thread(this);
                        t.start();
                        append(SplashItem);
                        Continue = new Command ("Start", Command.OK, 1);
                        addCommand(Continue);
                        setCommandListener(this);
            }
            public void initResources()
            {
                        Image Splash = null;
                        try
                        {
                                    Splash = Image.createImage("/splash.png");
                        }
                        catch(IOException ioe)
                        {
                                    append("unable to load image");
                        }
                        SplashItem = new ImageItem(null, Splash, ImageItem.LAYOUT_CENTER, null);
                        cycleStartTime = System.currentTimeMillis();
            }
            private static final int TIME_LOAD = 4000;
            public void run()
            {
                        try
                        {
                                    while(running)
                                    {
                                                long timeSinceStart = (System.currentTimeMillis() - cycleStartTime);
                                                if (timeSinceStart > TIME_LOAD)
                                                {
                                                            running = false;
                                                            main.activateMainMenu();

                                                }
                                    }
                        }
                        catch(Exception e)
                        {
                                    System.out.println("App exception: " + e);
                                    e.printStackTrace();
                        }
            }
            public void commandAction(Command command, Displayable displayable)
            {
                        if (command == Continue)
                        {
                                    running = false;
                                    main.activateMainMenu();

                        }
            }

}
```