REMOTE DESKTOP ACCESSING USING MIDLET

by

SATHYA.P

Register Number - 71206621047

of

KUMARAGURU COLLEGE OF TECHNOLOGY

COIMBATORE

A PROJECT REPORT

Submitted to the

FACULTY OF INFORMATION AND COMMUNICATION ENGINEERING

In partial fulfillment of the requirements

for the award of the degree

of

MASTER OF COMPUTER APPLICATIONS

ANNA UNIVERSITY

July, 2009

BONAFIDE CERTIFICATE

Certified that this project report titled **"Remote Desktop Access Using Midlets"** is the bonafide work of **"Ms.Sathya.P"** (Register Number: **71206621047**) who carried out the research under my supervision. Certified further, that to the best of my knowledge the work reported herein does not form part of any other project report or dissertation on the basis of which a degree or award was conferred on an earlier occasion on this or any other candidate.

N· Jayakanthan
Supervisor

Head of the Department

Submitted to Project and Viva Examination held on 06-07-2009

6/7/09
Internal Examiner

6/7/09
External Examiner

y to Success

COSMOSOFT
TECHNOLOGIES
LIMITED

141, Arcot Ro
Kodambakkam, Chennai - 600 0
Ph : +91-44-2372 3206/07/08 Fax : +91-44-2372 3
www.ctlindia.c
E-mail: info@ctlindia.c

TO WHOMSOEVER IT MAY CONCERN

29/05/09

This is to certify that the project entitled **"Remote Desktop accessing using Midlets"** is a bonafide work done in our organization by

Miss. Sathya P (RegNo:71206621047)

in partial fulfillment of the requirement for the award of **M.C.A. (Master of Computer Applications)** degree in **Kumaraguru College of Technology.** This project work is executed using **Java** technologies during the period of **December 22, 2008 to May 29, 2009**.

During the tenure of the project her **PERFORMANCE** was found to be **GOOD.**

For Cosmosoft Technologies Limited

COSMOSOFT TECHNOLOGIES LIMITED
CHENNAI-600 024

Authorised Signatory

ABSTRACT

The project **"Remote Desktop Access Using Midlets "** is developed for the one who are in need to get connected to the remote System. The main aim of the application is to remotely access and control different applications on a static IP PC by connecting to it over a GPRS link from a J2ME enabled cell phone.

Unhindered connectivity to a computer is the need of the hour. The ability to connect to a remote machine and control it will have infinite advantages. J2ME (Java 2 Micro Edition) is a platform created to enhance application development for mobile phones. Using this platform we have created an application which can control any desktop computer or a server by sending the required commands over an HTTP Connection.

ACKNOWLEDGEMENT

First and foremost I thank God for his good will and blessings showered on me throughout the project. The success of this project needs cooperation and encouragement from different quarters. Words are inadequate to express my profound and deep sense of gratitude to those who helped me in bringing out this project successfully.

I wish to express my deep unfathomable feeling of gratitude and indebtedness to **Prof. R. Annamalai,** Vice Principal – Kumaraguru College of Technology, Coimbatore for the successful completion of the project work.

I am very gladly taking this opportunity to express a special word of thanks to **Dr.M.Gururajan,** Head of the Department, Kumaraguru College of Technology, Coimbatore for encouraging me to do this work.

I am very much indebted to **Mrs.V.Geetha,** Assistant Professor, Kumaraguru College of Technology, Coimbatore for the complete assistance, guidance and support given to me throughout my project.

I would express heartfelt thanks to our internal guide **Mr.N.Jayakanthan,** Lecturer, Kumaraguru College of Technology as with out his best guidance it would not have been possible for me to successfully complete this project who also gave his innovative ideas at crucial times and tremendous encouragement.

It is my pleasure to express my profound gratitude to **Cosmosoft Technologies Limited**, Chennai for admitting into this project. I am thankful to **Mr. Veerajaya Babu** of Cosmosoft Technologies Limited, for his excellent guidance, timely suggestions and constant support in all my endeavors.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

LIST OF ABBREVIATIONS

Acronyms	**Full Form**
RDAUM	Remote Desktop Access Using Midlets
TCP	Transmission Control Protocol
UDP	User Datagram Protocol
IP	Internet Protocol
DFD	Data Flow Diagram
J2SE	Java 2 Standard Edition
J2ME	Java 2 Micro Edition
LCDUI	Limited Connected Device User Interface
MIDlet	Mobile Information Device Profile
VNC	Virtual Network Connectivity
MVC	Model View Controller
PDA	Personal Digital Assistant
JDBC	Java Database Connectivity

CHAPTER-1
INTRODUCTION

1.1 COMPANY PROFILE

Cosmosoft Technologies Limited

Cosmosoft provides software solutions for manufacturing, transportation and automobile industries. They are the solution provider for Delphi - TVS, Britannia, L&T, Mitsubishi, Visteon, Monsanto, Axles India, LG Care, Anand Group, Mahindra Holidays, Appollo Sindhoori, Srinivasa Exports International and many other such companies. Cosmosoft has software packages for inventory, finance, payroll, asset management and a complete ERP solution. Cosmosoft services include Customized solutions, Network Management Solutions, Web Solutions and Payroll - TDS outsourcing. They are program development partner for Tata Consultancy Services (TCS) and channel partner to SAP.

Projects Division (PPD) of Cosmosoft is the backbone of Cosmosoft with respect to technological feasibility analysis and development of prototype model(s) for various new advancements happening in today's technologies. This division strengthens Cosmosoft to be technically competent and enables us for future bids. Every financial year, this division also influences the organization to serve hundreds of project trainees, who will participate in various phases of development of these projects.

1.2 OUTLINE OF PROJECT

Many applications like web hosting services, network servers, automated systems need to be monitored continuously. And to monitor them 24/7 by being physically present at the location is not viable. Therefore we propose to control such applications remotely by J2ME enabled mobile devices.

The possible applications are:

i) Launching Winamp, WordPad, games or other similar applications.

ii) Shutting down and restarting or logging off from your machine.

iii) Formatting hard drives.

iv) Running Internet Explorer with the required URL.

v) Remote Database access including Query Processing and viewing the results on the cell phone.

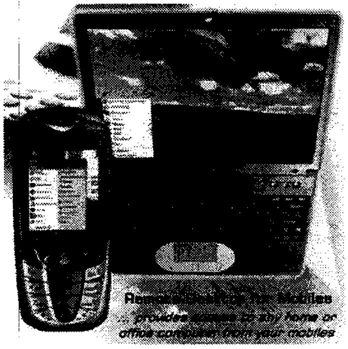
vi) Remote Desktop Connection from the Cell Phone.

2.2 DISADVANTAGES OF EXISTING SYSTEMS

- Always need a computer to access the remote desktop.
- Most of the existing systems are platform dependent.

2.3 PROPOSED SYSTEM

Java 2 ME has made it possible to write platform-independent applications for mobile devices and deploy them fairly easily. Now that we have a number of computing platforms strewn all over--office, home, andpocket--we need to think of interconnecting them. As far as desktop devices are concerned,wired internet has provided the necessary linkage for quite some time now. With the advent of GPRS, it has now become fairly easy to access data residing on desktops through mobile devices. Remote Desktop for Mobiles is a communication tool that gives you the unique ability to connect to your desktop computer through mobile devices and interact with it remotely. Remote Desktop Control displays the screen of another computer (via Internet or network) on your own screen. The program allows you to use your mouse and keyboard to control the other computer remotely. It means that you can work on a remote computer, as if you were sitting in front of it.


Remote Desktop for Mobiles
... provides access to any home or
office computer from your mobiles

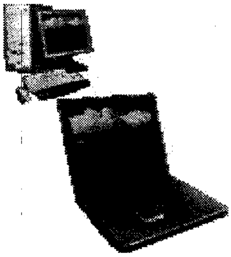
2.4 ADVANTAGES OF PROPOSED SYSTEM

- Access home computer from office and while traveling
- Provides home network and remote access technology
- Platform independent, works across heterogeneous networks
- Use Mobile phone keys to perform various operations and also to type text .

CHAPTER-2
SYSTEM ANALYSIS

2.1 EXISTING SYSTEM

Currently if any customer support is required, we have to go to the client spot and fix the bug. Or we can use Remote Desktop Connection which again needs a computer to view the remote computer's desktop.



The existing systems are:

Remote Desktop Connection:

Using this we can have access to a Windows session that is running on our computer when we are at another computer. This means, for example, that you can connect to your work computer from home and have access to all of your applications, files, and network resources as though you were in front of your computer at work.

Remote Desktop Web Connection:

This is a Web application that consists of an ActiveX control, sampleASP pages, and other files. When deployed on a Web server, Remote Desktop Web Connection allowsusers to create a connection with the remote desktop of another computer inside Internet Explorer, even if the Remote Desktop Connection program, formerly known as the Terminal Server client, is not installed on the user's computer.

CHAPTER-3
SYSTEM DESIGN

System design refers to the description of a new system based on the information that is collected during the analysis phase and the process by which it is developed. It is the creative process of inventing and developing new inputs, database procedures and outputs to meet the system objectives. System design builds on the information gathered during system analysis. It describes what output is to be produced, in what format, what input should be fed to obtain the required output, format of the input, etc. System design is a part of the system development process. It refers to the process of planning of an entirely new system or one to replace or enhance an existing one. System design encompasses the following steps

- Review of appropriate
- Determining the requirements for a new system
- Design the new system

INPUT DESIGN

The user interface design is very important for any application. The interface design describes how the software communicates within itself, to system that interpreted with it and with humans who use it. The input design is the process of converting the user-oriented inputs into the computer-based format.

Here the input is the usage of correct keys in the mobile phones. To have the knowledge of keys a help window is created.

OUTPUT DESIGN

A quality output is one, which meets the requirements of the end user and presents the information clearly. In any systems results of processing are communicated to the user and to other systems through outputs. In the output design it is determined how the information is to be displayed for immediate need. It is the most important and direct source information to the user. Efficient and intelligent output design improves the system's relationship with the user and helps in decision-making.

Output Design Concepts: Output is the most visible component of the information system.

- It is the basis of management's final assessment of the system;
- It is designed by rapidly constructing prototypes.
- During system design outputs are modeled as data flows.
- Outputs may introduce new aspects to the system.

DATABASE DESIGN:

The Database design directly affects the performance of the Software application. The database is normalized so as to avoid redundant data. Normalization reduces the wastage of valuable memory space. The databases for this project have been created with utmost care so as to follow the principles of normalization. The database system must give assurance for security of information; despite the system crashes due to attempts of unauthorized access. In my project I am using MySQL.

Attribute name	Data type	Size	Constraints	Description
USER_NAME	Varchar	50	Primary key, Not null	Name of the User
USER_PASS	Varchar	50	Not null	Password of the User
USER_RIGHTS	Integer	1	Not null	Rights given to the User

Table 3.1 User_Table

CODE DESIGN

The goal is to translate the design of the system produced during the design phase into code in a given programming language which can be executed by a computer and that performs the computation specified by the design. The coding phase affects both testing and maintenance profoundly. There are many different criteria for judging a program, including readability, size of the program, execution time and required memory. The main objective of the coding are minimize the effort required to complete the program, minimize the number of statements, minimize the memory maximize the clarity of program and output.

The Model – View – Controller architecture is a widely – used architectural approach for interactive applications. It divides functionality among objects involved in maintaining and presenting data to minimize the degree of coupling between the objects. The architecture maps traditional application tasks—input, processing, and output—to the graphical user interaction model.

The MVC architecture divides applications into three layers: model, view, and controller – and decouples their respective responsibilities. Each layer handles specific tasks and has specific responsibilities to the other areas.

The Codes uses has certain advantages they are

- **Multiple views using the same model**

  The separation of model and view allows multiple views to use the same enterprise model. Consequently, an enterprise application's model components are easier to implement, test, and maintain, since all access to the model goes through these components.

- **Clarity of design**

  By glancing at the model's public method list, it should be easy to understand how to control the model's behavior. When designing the application, this trait makes the entire program easier to implement and maintain.

- **Efficient modularity**

  Changes to one aspect of the program aren't coupled to other aspects, eliminating many nasty debugging situations. Also, development of the various components can progress in parallel, once the interface between the components is clearly defined.

- **Ease of growth**

  Controllers and views can grow as the model grows and older versions of the views and controllers can still be used as long as a common interface is maintained.

3.1 SYSTEM ARCHITECTURE

This project is a VNC-like application that allows the visualization and the control of a remote computer connected to the internet from a mobile device. It runs in any mobile device with Java Micro Edition technology enabled.

The javaserver is the component that converts the instructions from your mobile device into instructions in your desktop computer. Run the javaserver with the commandline argument like "-p 10099" (or any other port number). When we run the server for first time, need to answer some questions in order to provide basic authentication information (like e-mail and password) to the server. These information are saved in the database which is used for authentication.

When we start the iDesktop Client, a form asking to fill an e-mail address and a password will appear. The e-mail and password are the same as the one you've put when you setup the iDesktop Server. The e-mail and password are ways of making sure that only you will access your remote computer. When we hit the "Connect" command,the mobile device will try to connect to the javaserver perform the authentication and we should be able to control your desktop computer remotely.

The architecture can be broadly divide into 3 layers as :

1) The J2ME client.(MIDlet running on it)

2) The HTTP server (Java Servlet running on it)

3) The Static IP machine (on the Internet) with it's set of applications

Model-View-Controller Architecture

Model-View-Controller (MVC) is an architectural pattern used in software engineering. Successful use of the pattern isolates business logic from user interface considerations, resulting in an application where it is easier to modify either the visual appearance of the application or the underlying business rules without affecting the other.

In MVC, the **Model** represents the information (the data) of the application and the business rules used to manipulate the data, the **View** corresponds to elements of the user interface such as text, checkbox items, and so forth, and the **Controller** manages details involving the communication to the model of user actions such as keystrokes and mouse movements.

Model

The model represents enterprise data and the business rules that govern access to and updates of this data. Often the model serves as a software approximation to a real-world process, so simple real-world modeling techniques apply when defining the model.

View

The view renders the contents of a model. It accesses enterprise data through the model and specifies how that data should be presented. It is the view's responsibility to maintain consistency in its presentation when the model changes.

Controller

The controller translates interactions with the view into actions to be performed by the model. In a stand-alone GUI client, user interactions could be button clicks or menu selections, whereas in a Web application, they appear as GET and POST HTTP requests. The actions performed by the model include activating business processes or changing the state of the model. Based on the user interactions and the outcome of the model actions, the controller responds by selecting an appropriate view.

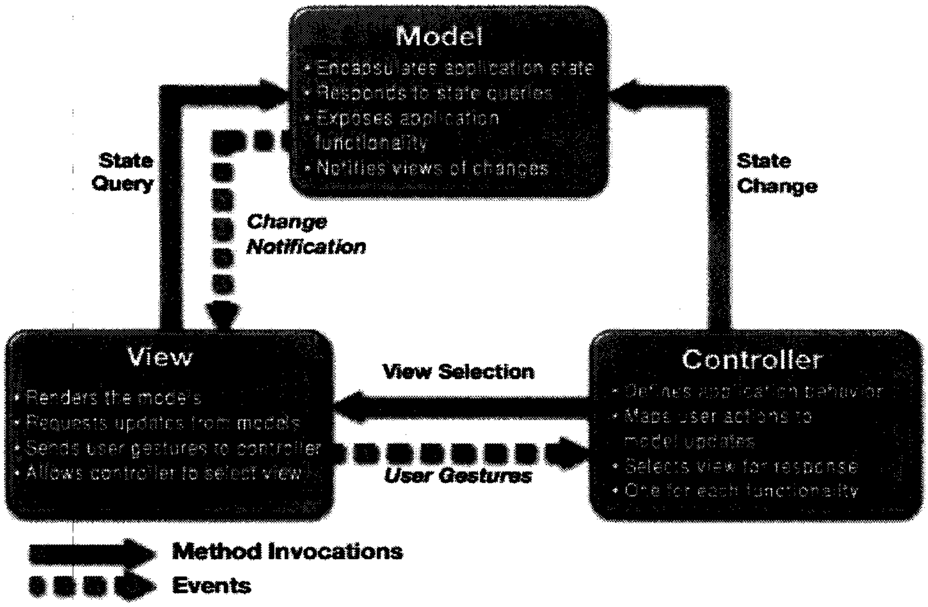

Model
• Encapsulates application state
• Responds to state queries
• Exposes application functionality
• Notifies views of changes
State Query
Change Notification
State Change
View
• Renders the models
• Requests updates from models
• Sends user gestures to controller
• Allows controller to select view
View Selection
User Gestures
Controller
• Defines application behavior
• Maps user actions to model updates
• Selects view for response
• One for each functionality
Method Invocations
Events

Figure 3.1 MVC Architecture

3.2 UML Diagrams:

The Unified Modeling Language is now the most widely used graphical representation scheme for modeling object-oriented systems. It has indeed unified the various popular notational schemes. Those who design systems use the language (in the form of diagrams) to model their systems.

An attractive feature of the UML is its flexibility. The UML is extensible (i.e., capable of being enhanced with new features) and is independent of any particular OOAD process. UML modelers are free to use various processes in designing systems, but all developers can now express their designs with one standard set of graphical notations.

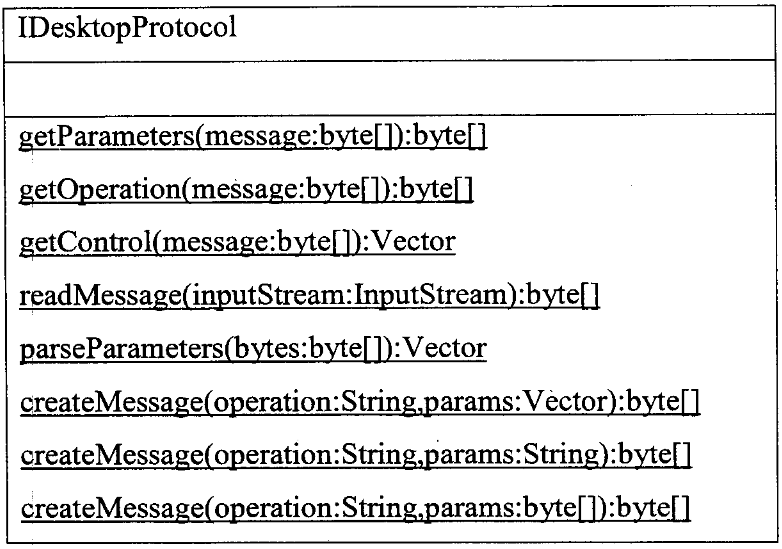
The UML is a complex, feature-rich graphical language. In our "Software Engineering Case Study" sections, we present a simple, concise subset of these features. We then use this subset to guide you through a first design experience with the UML intended for novice object-oriented programmers in a first- or second-semester programming course.

3.2.1. Class diagram

A UML class diagram, also referred to as object modeling, is the main static analysis diagram. These diagrams show the static structure of the model. A class diagram is a collection of static modeling elements such as classes and their relationships connected as graph to each other and to their contents.

DbSetup
Property:Properties Btn_default:JButton Close:JButton jLabel1:JLabel jLabel2:JLabel jLabel3:JLabel jLabel4:JLabel jPanelPortInfo:JPanel Save:JButton txtFielddriver:JTextField txtFieldpwd:JTextField txtFielduid:JTextField txtFieldurl:JTextField
Dbsetup() load():void initComponents():void saveActionPerformed(evt:ActionEvent):void closeActionPerformed(evt:ActionEvent):void btn_defaultActionPerformed(evt:ActionEvent):void formInternalFrameActivated(evt:InternalFrameEvent):void

Figure 3.2 Database Setup



IDesktopProtocol

getParameters(message:byte[]):byte[]

getOperation(message:byte[]):byte[]

getControl(message:byte[]):Vector

readMessage(inputStream:InputStream):byte[]

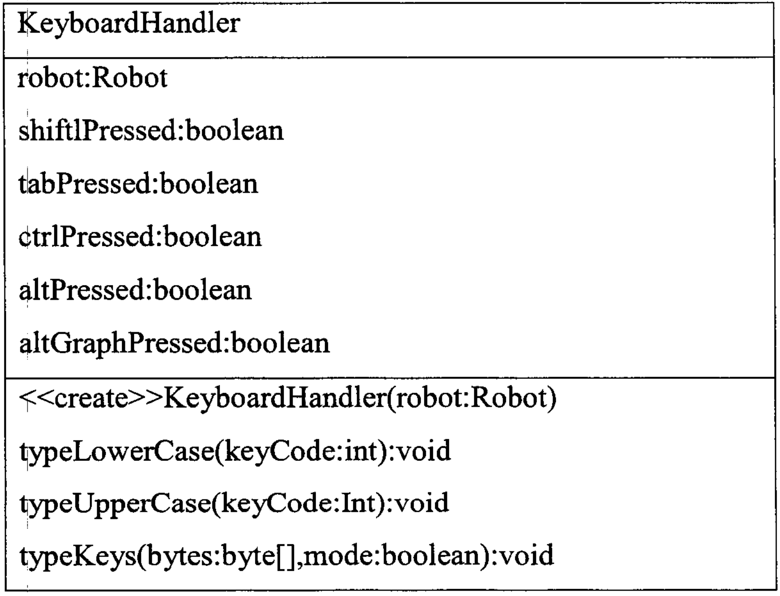
parseParameters(bytes:byte[]):Vector

createMessage(operation:String,params:Vector):byte[]

createMessage(operation:String,params:String):byte[]

createMessage(operation:String,params:byte[]):byte[]

Figure 3.3 Protocol



KeyboardHandler

robot:Robot

shiftlPressed:boolean

tabPressed:boolean

ctrlPressed:boolean

altPressed:boolean

altGraphPressed:boolean

<<create>>KeyboardHandler(robot:Robot)

typeLowerCase(keyCode:int):void

typeUpperCase(keyCode:Int):void

typeKeys(bytes:byte[],mode:boolean):void

Figure 3.4 Keyboard Handlers

ServerSideHandler
Socket:Socket inputStream:InputStream outputStream:OutputStream dboperation:JDBCConnectot
<<create>>ServerSideHandler(socket:Socket,dboperation:JDBCConnector) processMessage(operation:String,params:byte[],mode:boolean):void ViewMode():void sendMessage(message:byte[]):void getSocket():Socket getIutputStream():IutputStream getOutputStream():OutputStream performCloseoperation():boolean

Figure 3.5 Server Side Handlers

3.2.2 Data Flow Diagrams[DFD]:

"A network representation of a system. The system may be automated, manual, or mixed. The DFD portrays the system in terms of its component pieces, with all interfaces among the components indicated."

- Tom DeMarco

hence DFDs:

focus on the *movement* of data between external entities and processes, and between processes and data stores.

Level 0 :

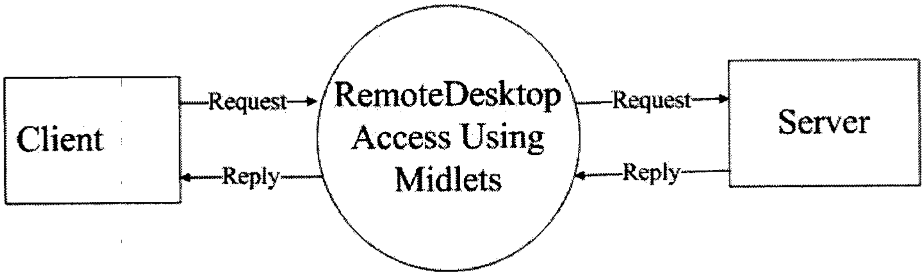


Figure 3.6 Level 0 Diagram

Level 1:

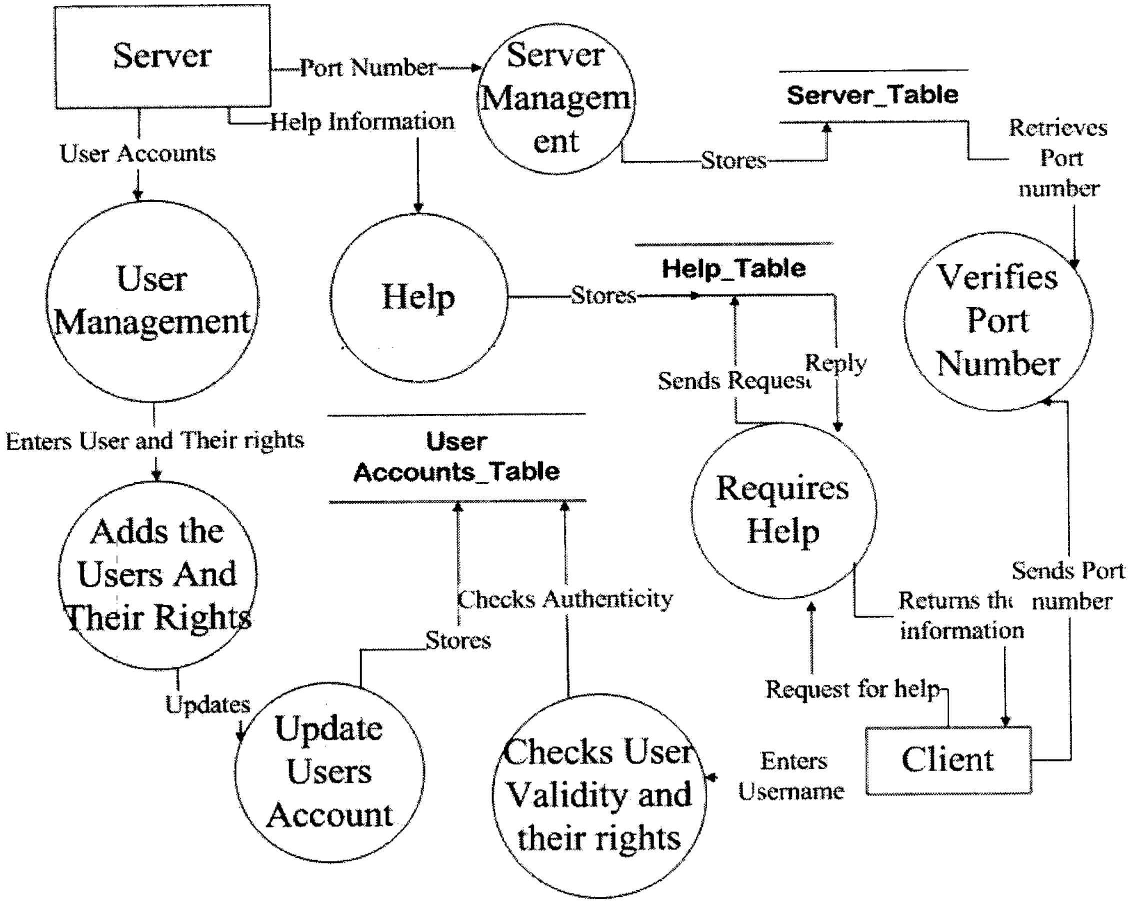


Figure 3.7 Level 1 Diagram

Level 2:

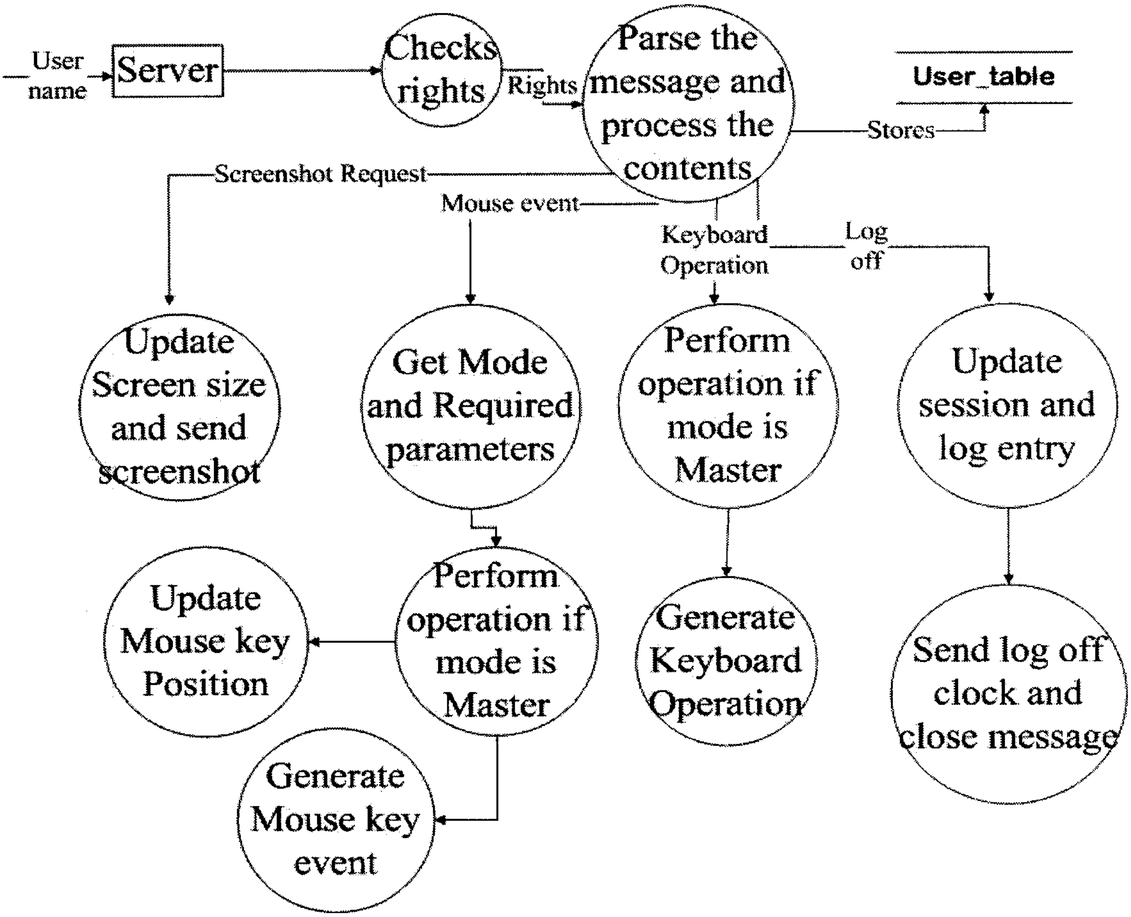

User name
Server
Checks rights
Rights
Parse the message and process the contents
User_table
Stores
Screenshot Request
Mouse event
Keyboard Operation
Log off
Update Screen size and send screenshot
Get Mode and Required parameters
Perform operation if mode is Master
Update session and log entry
Update Mouse key Position
Perform operation if mode is Master
Generate Keyboard Operation
Send log off clock and close message
Generate Mouse key event

Figure 3.8 Server Side Operation

Level 2:

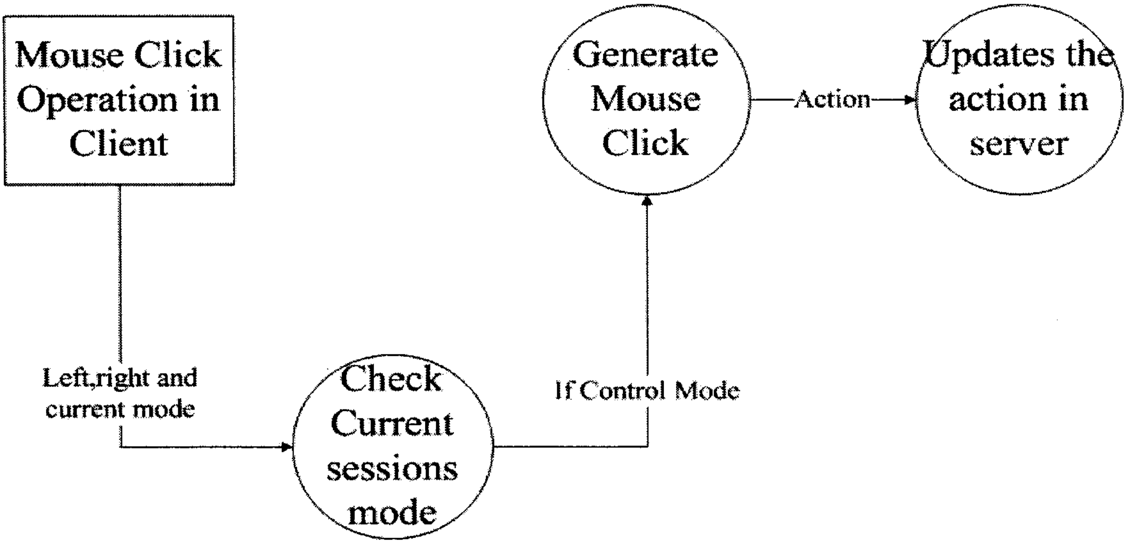

Mouse Click Operation in Client
Left,right and current mode
Check Current sessions mode
If Control Mode
Generate Mouse Click
Action
Updates the action in server

Figure 3.9 Mouse Click

Level 2:

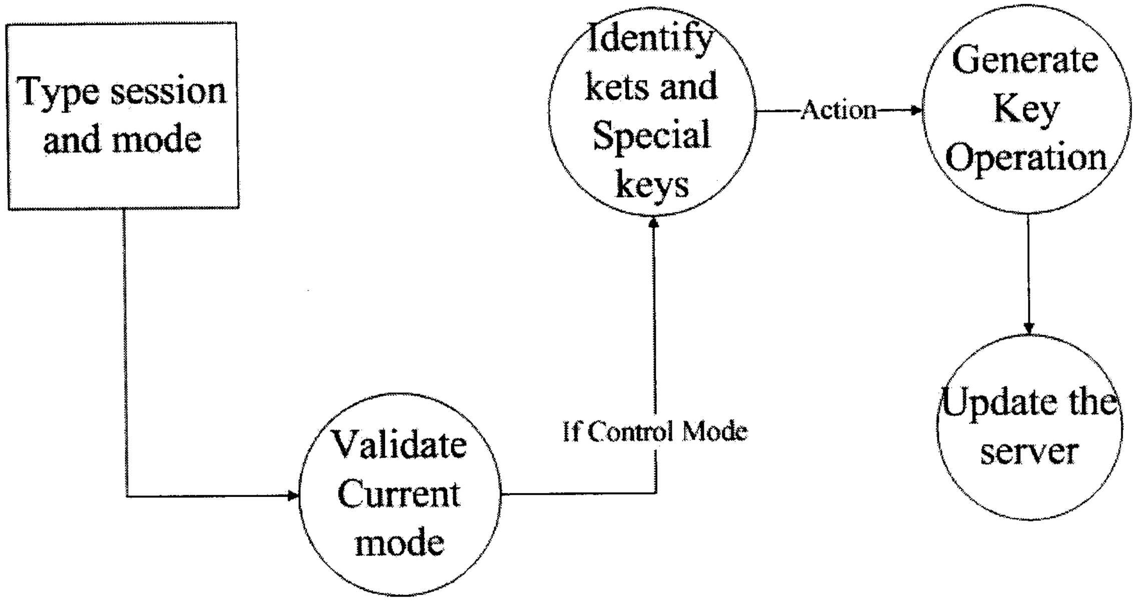


Figure 3.10 Keyboard Operation

CHAPTER – 4
DEVELOPMENT ENVIRONMENT

4.1 INTRODUCTION:

System development is a series of operations performed to manipulate data to produce output from computer system. This aim at translating the design of the system produced during the design phase into code in user programming language. A modular approach is used for the development of the software.

The development phase for the project was created from the specifications created during the design phase. A principal activity of the development phase is coding and testing the computer program that make up the computer program component of the overall system. Other important activities include implementation, planning, equipment acquisition and system testing. The development phase concludes with the report and review.

4.2 MODULE DESCRIPTION

- ✓ GUI Design using Java Swings
- ✓ J2ME Application Development using LCDUI Package
- ✓ Client - Side handler
- ✓ Mobile Device keyboard layout configuration
- ✓ Remote Desktop Canvassing
- ✓ User Authentication
- ✓ Handlers for Keyboard
- ✓ Mouse and Screen using Robot class
- ✓ JDBC Connectivity
- ✓ Auto Execution of Database Schema
- ✓ FAQ management using JDBC

4.2.1 GUI Design using Java Swings

This module involves in building the user interaction screen for the software. The GUI development is a common module for any software, because the presentation is very important for a project and also the GUI should be user friendly to the End-Users and the user should be guided throughout the process of testing .The screen has to be developed such that it should import the front and rear of java packages, so that the GUI should be rich and the same time the GUI should be flexible to the maximum extent. An application with a graphical user interface is written to respond to events posted into an event queue by the supporting window size. Swing components of java is used to build GUI layouts. The Java Swing API Provides a set of "lightweight" (all-Java language) components that, to the maximum degree possible, work the same on all Platforms Here the Swing package of java is used in the GUI development. The Swing classes have certain flexibilities through which the Swing scores better that AWT and Applets. The various components used are Frame, Panels, Buttons, Menu, InputDialogBox, File Choosers, Text Area, Layouts etc.,

4.2.2 J2ME Application Development using LCD UI Package

The LCDUI API provides a set of features for implementation of user interfaces for MIDP applications. In this module, the J2ME screen gui is designed using the lcdui package. The main criteria for the midlet GUI have been designed with mobile information devices in mind (i.e., mobile phones and pagers). These devices differ from desktop systems in many ways, especially how the user interacts with them.This module involves in development of various user interface screens in the mobile devices using lcdui package with following classes, listed in alphabetical order, from the javax.microedition.lcdui package: Alert, Command, DateField, Display, Displayable, Form, Gauge, List, TextBox, TextField, Ticker,CommandListener.Various screens developed are Authentication Screen, Media Capture Screen, Media Storage Screen,
RMS Modification Screen, Media Publishing Screen, Media Viewing Screen etc.,

4.2.3 Data Persistence using RMS

The Mobile Information Device Profile provides a mechanism for MIDlets to persistently store data and later retrieve it. This persistent storage mechanism is modeled after a simple record oriented database and is called the Record Management System(RMS).

4.2.4 Mouse and Screen using Robot class

This class is used to generate native system input events for the purposes of test automation, self-running demos, and other applications where control of the mouse and keyboard is needed. The primary purpose of Robot is to facilitate automated testing of Java platform implementations. Using the class to generate input events differs from posting events to the AWT event queue or AWT components in that the events are generated in the platform's native input queue. For example, Robot.mouseMove will actually move the mouse cursor instead of just generating mouse move events.Note that some platforms require special privileges or extensions to access low-level input control. If the current platform configuration does not allow input control, an AWTException will be thrown when trying to construct Robot objects. For example, X-Window systems will throw the exception if the XTEST 2.2 standard extension is not supported (or not enabled) by the X server.

4.2.5 Handlers for Keyboard

This method is used to convert the key codes that have more than one physical key associated with them.It will receive the key codes from remote mobile device and generate corresponding key.While generating the key code it will the remote user rights from database because our proposed will system support more than one remote access.

4.2.6 Server - Side Screen handler

This module is used to capture the desktop screen and scaled to the spicified size from the request.

Procedure to archive:

1.Create the image that hold the server desktop screen.

2.Create the Buffered Image as a template and it used to background image.

3.Buffered Image cropped into required size.

4.Draw server desktop image over the Buffered image.

5.Create image, convert into byte and send it.

4.2.7 Remote Desktop Canvasing

This module involves in building the image that recevied from byte image arrary.And creates the mouse cursor and draw on the canvas.

4.2.8 Auto Execution of Database Schema

This module aims in creating the database schema automatically, when the software loads for the first time.The database schemas table creation statements (SQL Scripts) will be read from the properties file and the SQL statements are executed on the connected database automatically. Also the initial values for some of the database tables are automatically inserted into the database.Java Beans are used to access the database. This makes this software as the Model View Controller architecture. Since the database access in done through Beans, code reusability is achieved which makes the project component based .

4.2.9 FAQ management using JDBC

In this module SMS Query(Expected SMS Content to be Received) and Solution(Matching Reply Messages to be sent) is retrieved from a swing GUI screen and stored in the database. Also data management functionality is provided .JDBC API is used for accessing and processing data stored in a data source (usually a relational database) using the Java programming language.

This module involves in designing the database model for a real time system and establishing the database connection between the database and the software using JDBC API. The JDBC API allows to invoke SQL commands from Java programming language methods. JDBC API in used in a java class which can be called from any other java class to access the database directly.

Here we are using MySQL as the RDBMS, to which the client should be connected before Creating tables and transferring the records in to the table.

The database is designed with Pointbase, MYSQL, Oracle 10g XE database. The connection parameters are specified in the *properties file* which has the following informations: Driver Name for connection, Driver URL, Username, Password, Database Name. JDBC connection is established by the following code :

```
Connection con = null;
Statement stmt = null;
Class.forName(driver);
Con = DriverManager.getConnection(jdbcURL, user, password);
con.setAutoCommit(false);
stmt = con.createStatement();
stmt.execute(query);
con.commit();
con.close();
```

The Java class is to written in a generic way so that it reads data from any database through a *properties file.*

With the help of this module this project can connect to any RDBMS without any code modifications with a few changes in the property file like the url, driver, username and password. Database connections are done in this module. This module is designed as Java class.

4.1.10 JDBC Connectivity

This module involves in designing the database model for a real time system and establishing the database connection between the database and the software using JDBC API. The JDBC API allows to invoke SQL commands from Java programing language methods. JDBC API in used in a bean which can be called from a servlet or JSP page to access the database directly.

Here we are using MYSQL as the RDBMS, to which the client should be connected before Creating tables and transferring the records in to the table.

The database is designed with Pointbase, MYSQL, Oracle 10g XE database. The connection parameters are specified in the **properties file** which has the following informations: Driver Name for connection, Driver URL, Username, Password, Database Name. JDBC connection is established by the following code :

```
Connection con = null;
Statement stmt = null;
Class.forName(driver);
Con = DriverManager.getConnection(jdbcURL, user, password);
con.setAutoCommit(false);
stmt = con.createStatement();
stmt.execute(query);
```

```
con.commit();
con.close();
```

The program reads data from any database through a properties file. With the help of this module this project can connect to any RDBMS without any code modifications with a few changes in the property file like the url, driver, username and password.

CHAPTER-5
TECHNICAL SPECIFICATIONS

5.1 ENVIRONMENT DETAILS:

The minimum requirements for the system to run effectively are:

5.1.1 Hardware Requirements

Monitors	800 x 600 mínimum resolution at 256 colours mínimum
Memory	Approximately 256 MB of on board memory.
I/O	two or three button mouse and standard 101-key keyboard.
Processor	At least 1.3 GHz processor
OS	Windows XP/ 2000
Java	Java SE 6.0 or above
Database	PointBase/ MySQL/ Oracle 10g XE/ SQL Server
Tools (IDE)	NetBeans 5.5.1 or above

5.1.2 Software Requirements

Operating system	Windows XP
Language	J2SE,J2ME
Database	MySQL
Tools(IDE)	NetBeans 6.0 or above
Web Server	Tomcat 6.0.4 or above

5.2 TECHNOLOGIES USED

5.2.1 Languages:

J2SE:

One of the most dynamic programming languages used by computer programmers today is Java. Especially in the case of servers and mainframes and big-picture systems, Java makes business processes happen. Technically and historically, Java is the creation of Sun Microsystems. This venerable computer company produces both hardware and software, and Java is one of its software offerings. Other companies now create Java applications, but Sun was there first.

One well-known type of Java application is the applet, a sort of fast-working subroutine that is largely platform-independent and can work within other frameworks. Applets are mini applications that perform a variety of functions, large and small, mundane and dynamic, within the framework of larger applications. Technically, an applet is like an application lite, as its name suggests. A software developer who is proficient at developing applets is in significant demand.

A prime example of an applet is a browser-based plugin, such as the one that allows users to see Flash movies or hear audio files by clicking on a Web page link. Such a click keeps the user in the browser environment, while simultaneously launching the application-within-an-application that is the applet, showing the requested video or playing the requested sound. Many online games are applet-based as well.

The Java 2 Platform has three basic Editions: Micro, Standard, and Enterprise. They have increasing amounts of functionality and flexibility, with the Micro Edition being used for small-range applications and the Enterprise Edition being used for large, server-based functions. In the middle is the Standard Edition, or J2SE. J2SE has applications up and down the requirements ladder, filling needs for both individual and complicated users.

One of the primary uses of J2SE is the development of Java applications for individual computers. Web-based activity sometimes lives and dies by the successful integration of applets into e-commerce and other Web-specific functions. J2SE applets and other applications make these functions run smoothly; without them, many transactions and other Internet interactions would not take place. In this way, J2SE is a tremendous enabler of Web activity.

Another important functionality made possible by J2SE is JavaBeans. These are reusable applications that can be developed and assembled easily in order to create more sophisticated applications. Basically, they are the building blocks of personalized J2SE applications. Java may be the base technology, but JavaBeans are what makes J2SE functionality and individuality really click.

J2ME:

Java Platform, Micro Edition (Java ME) provides a robust, flexible environment for applications running on mobile and other embedded devices—mobile phones, personal digital assistants (PDAs), TV set-top boxes, and printers. Java ME includes flexible user interfaces, robust security, built-in network protocols, and support for networked and offline applications that can be downloaded dynamically. Applications based on Java ME are portable across many devices, yet leverage each device's native capabilities.

5.2.2 Database-MySQL:

MySQL is the world's most popular open source database software, with over 100 million copies of its software downloaded or distributed throughout its history. With its superior speed, reliability, and ease of use, MySQL has become the preferred choice for Web, Web 2.0, SaaS, ISV, Telecom companies and forward-thinking corporate IT Managers because it eliminates the major problems associated with downtime, maintenance and administration for modern, online applications.

Many of the world's largest and fastest-growing organizations use MySQL to save time and money powering their high-volume Web sites, critical business systems, and packaged software — including industry leaders such as Yahoo!, Alcatel-Lucent, Google, Nokia, YouTube, Wikipedia, and Booking.com.

MySQL is a multithreaded, multi-user SQL database management system (DBMS) which has more than 11 million installations. The program runs as a server providing multi-user access to a number of databases. Libraries for accessing MySQL databases are available in all major programming languages with language-specific APIs.

The flagship MySQL offering is MySQL Enterprise, a comprehensive set of production-tested software, proactive monitoring tools, and premium support services available in an affordable annual subscription.

MySQL is a key part of LAMP (Linux, Apache, MySQL, PHP / Perl / Python), the fast-growing open source enterprise software stack. More and more companies are using LAMP as an alternative to expensive proprietary software stacks because of its lower cost and freedom from platform lock-in.

Advantages of MySQL:

- The best and the most-used database in the world for online applications
- Available and affordable for all
- Easy to use
- Continuously improved while remaining fast, secure and reliable
- Fun to use and improve
- Free from bugs
-

5.2.3 Tools-NetBeans:

A free, open-source Integrated Development Environment for software developers. You get all the tools you need to create professional desktop, enterprise, web, and mobile applications with the Java language, C/C++, and even dynamic languages such as PHP, JavaScript, Groovy, and Ruby. NetBeans IDE is easy to install and use straight out of the box and runs on many platforms including Windows, Linux, Mac OS X and Solaris.

The NetBeans IDE 6.5 provides several new features and enhancements, such as rich PHP, JavaScript and Ajax editing features, improved support for using the Hibernate web framework and the Java Persistence API, and tighter GlassFish v3 and MySQL integration.

Advantages:

- Java Desktop Applications
- PHP Development
- Java EE Web Applications
- C and C++ Development
- Visual MobileDevelopment

CHAPTER-6
TESTING

6.1 INTRODUCTION

Testing is very important in determining the reliability and efficiency of software, and hence it is very crucial stage in software development. Tests are conducted on the software to evaluate its performance at different levels.

Testing is a process of executing a program with the intent of finding whether the software achieves the desired result. Software testing is a critical element of software quality assurance and represents the ultimate review of specification, design and coding. If testing is conducted successfully, it will uncover errors in the software. As a secondary benefit, testing demonstrates that software functions appear to be working according to specification and that performance requirements appear to have been met. Different types of data are fed into the system and the end result is verified with the expected results. System testing is the stage of implementation, which is aimed at ensuring that the system works accurately and efficiently before live operation commences.

Testing is vital to the success of the system. Testing cannot show the absence of defects. It can only show the software errors present in the developing system. Testing is a set of activities that can be planned in advance and conducted systemtically. The users are trained to operate the developed system. Both hardware and software securities are made to run the developed system successfully in future.

6.1.1 Unit Testing

Unit testing deals with a unit as a whole. This would test the interaction of many functions but confine the test within one unit. The exact scope of a unit is left to interpretation. Here we test each module individually and integrate the overall system. Unit testing focuses verification efforts even in the smallest unit of software design in each module. This is also known as “Module Testing”. The modules of the system are tested separately. This testing is carried out in the programming style itself. In the testing each department is focused to work satisfactorily as regard to expected output. There are some validation checks for the fields.

Each module in RDAUM is checked.All the functions in the modules were checked individually and found to be function properly.

6.1.2 Integration Testing:

During Integration testing,we choose portions of the structure tree of the software to put together.Each sub tree should have some logical reason for being tested.It may be a difficult part of the code or it may be the essential to the function of the rest of the product.

Remote Desktop Access Using Midlets follows top-down integration testing.Modules were linked to the main menu in a sequence as required for the integration.This process is continued from the page level to the module level and then to the system level,In the final stage,the whole system is taken together and tested for integration.A change in one place should be reflected through out the system.

In RDAUM the change is made in the mobile screen and is found to be reflected in the remote desktop also.

6.1.3 System Testing:

This primary concern of system testing is the compatibility of individual modules,the integration of each module in the system.It tests to find discrepancies between the system and its original objective.

The whole system is tested using the java enabled mobile by installing the .jad file in it.The informations are sent through GPRS service from the GPRS enabled mobile.

6.1.4 Output Testing

After performing validation testing, the next step is output testing of the proposed system. Since the system cannot be useful if it does not produce the required output. Asking the user about the format in which the system is required tests the output displayed or generated by the system under consideration. Here the output format is considerable in two ways. One is on screen format and another one is printed format. The output format on the screen is found to be corrected as the format was designed in the system phase according to the user needs.

In RDAUM the output testing was done and it is found that the expected output was reached.

CHAPTER-7
CONCLUSION

In **“Remote Desktop Access Using Midlets”** the ability of the J2ME enabled devices to connect to FTP, HTTP or even the use of raw sockets has given it the power to break all computational barriers in spite of the resource limitations, making small handsets synonymous(in computational power) to any high performance server.

In the present application we have managed to control an application on a remote machine using the mobile phone. But the same can be extended to control any computer on the network.

Hence with this application in mind it is not difficult to imagine controlling huge computing clusters from your fingertips anywhere, anytime ......

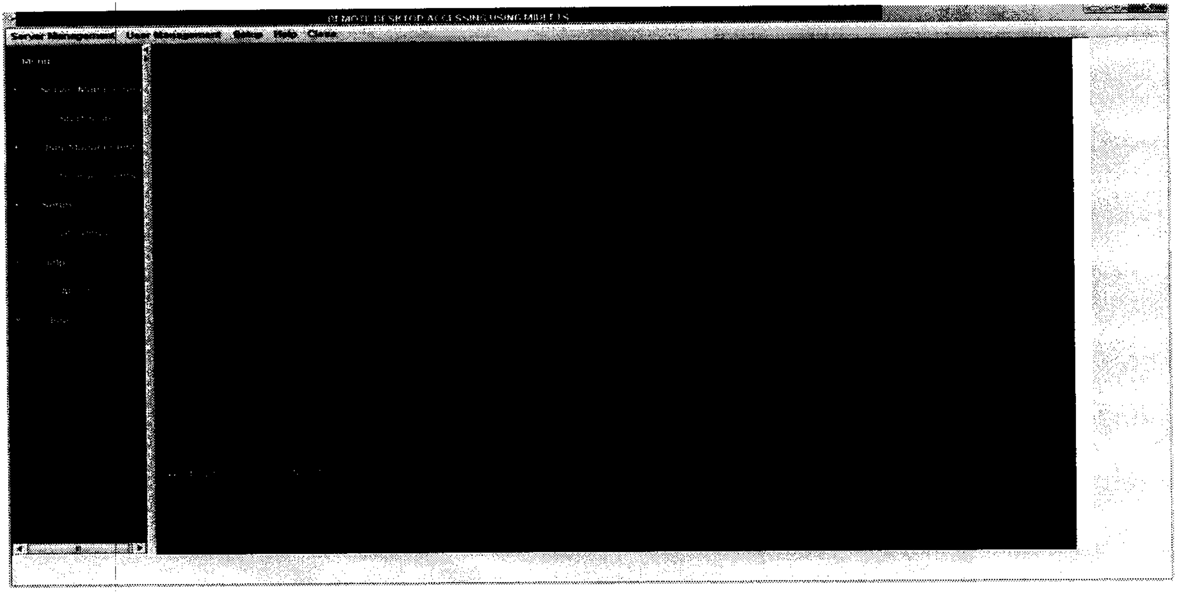
CHAPTER-8
FUTURE ENHANCEMENT

In the present application called **“Remote Desktop Access Using MIDlets”** we have managed to control an application on a remote machine using the mobile phone. But the same can be extended to control any computer on the network.This will enable us to access, edit and add the information on any computer on the network.Through this we can bring the world within our computers.

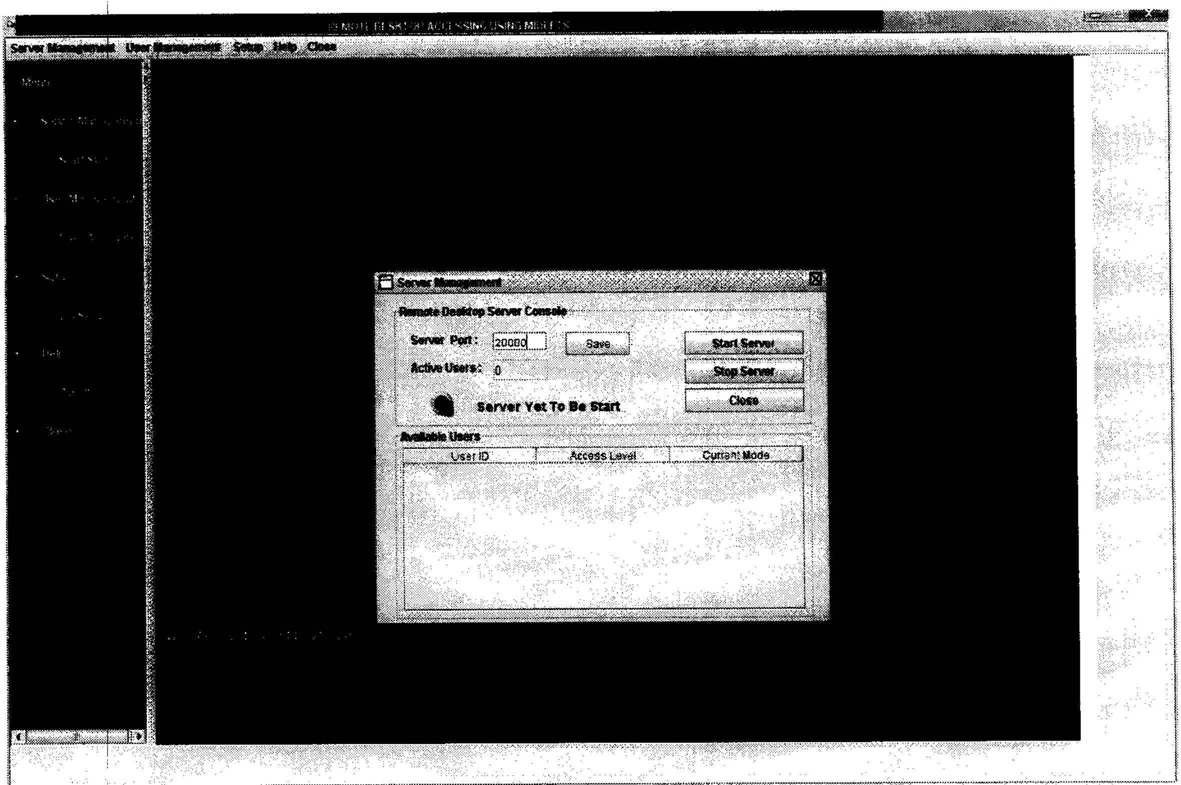
CHAPTER-9

APPENDICES

1. Server Screen:



2. Giving Port Number:



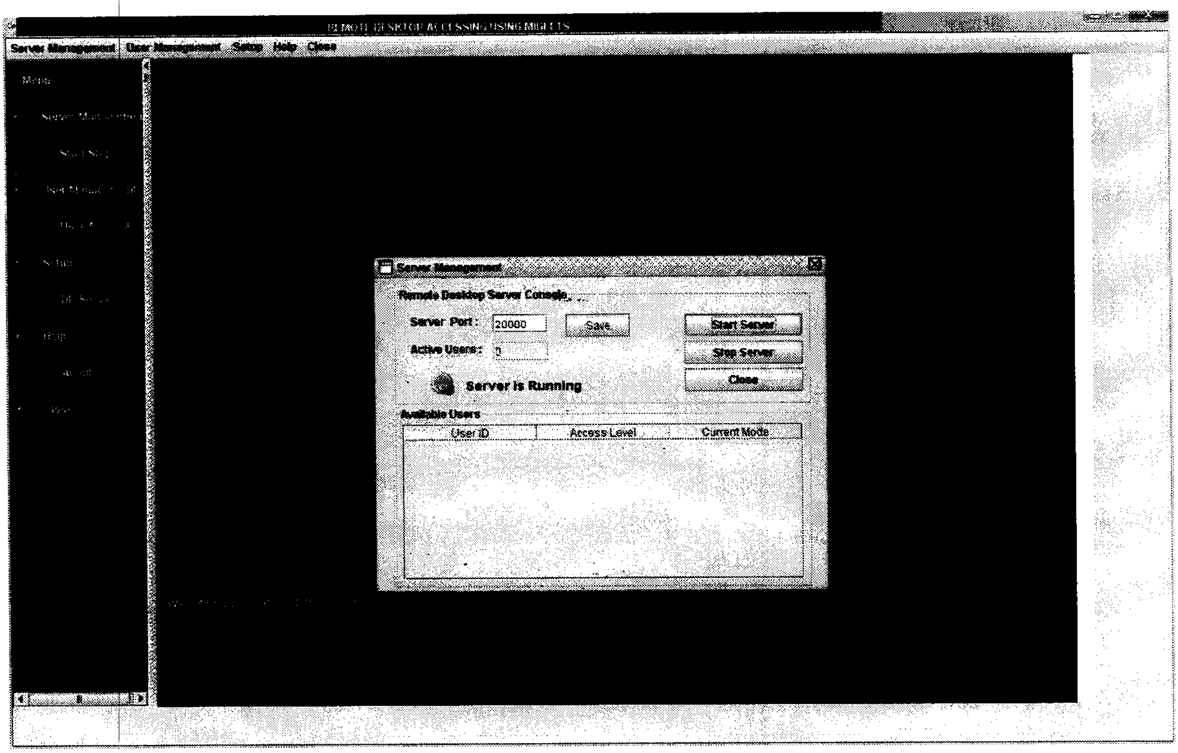
Server Port : 20080

Active Users: 0

Server Yet To Be Start

User ID	Access Level	Curtant Mode

3. Starting the Server:



Server Management

Remote Desktop Server Console

Server Port : 20000

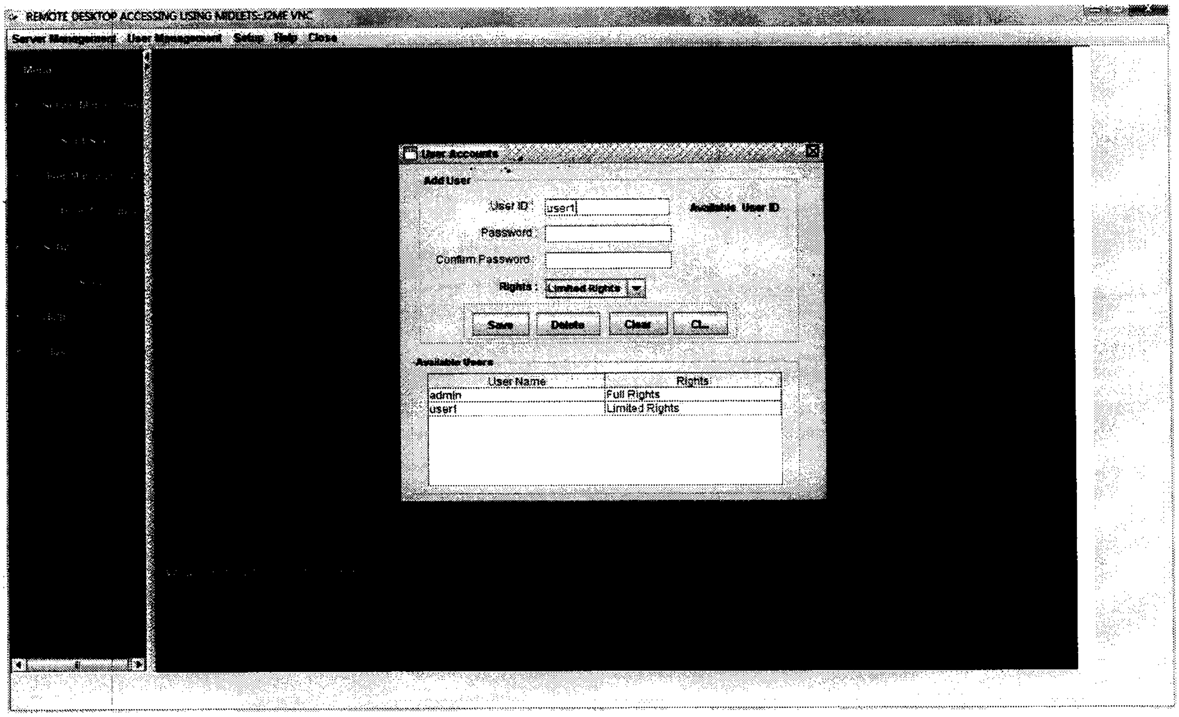
Active Users : 0

Server is Running

Available Users

User ID	Access Level	Current Mode

4. User Accounts Form:



User Accounts

Add User

User ID : user1

Available User ID

Password :

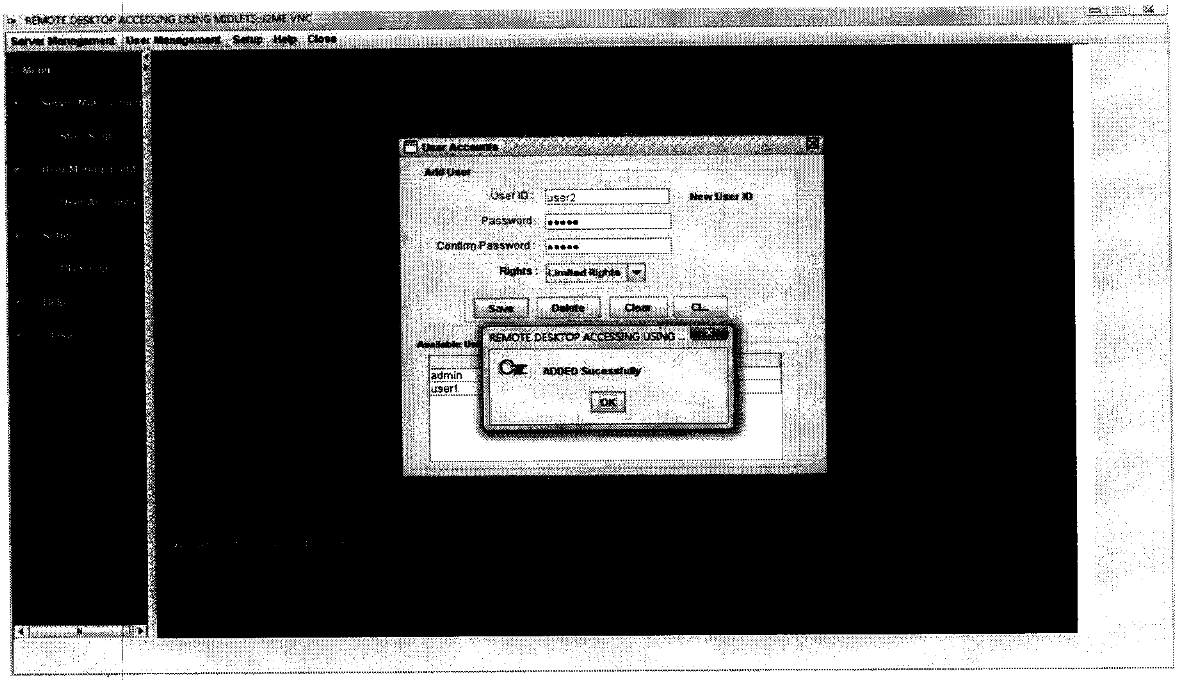
Confirm Password :

Rights : Limited Rights

Available Users

User Name	Rights
admin	Full Rights
user1	Limited Rights

5. Adding a user:



User ID : user2 New User ID

Password : *****

Confirm Password : *****

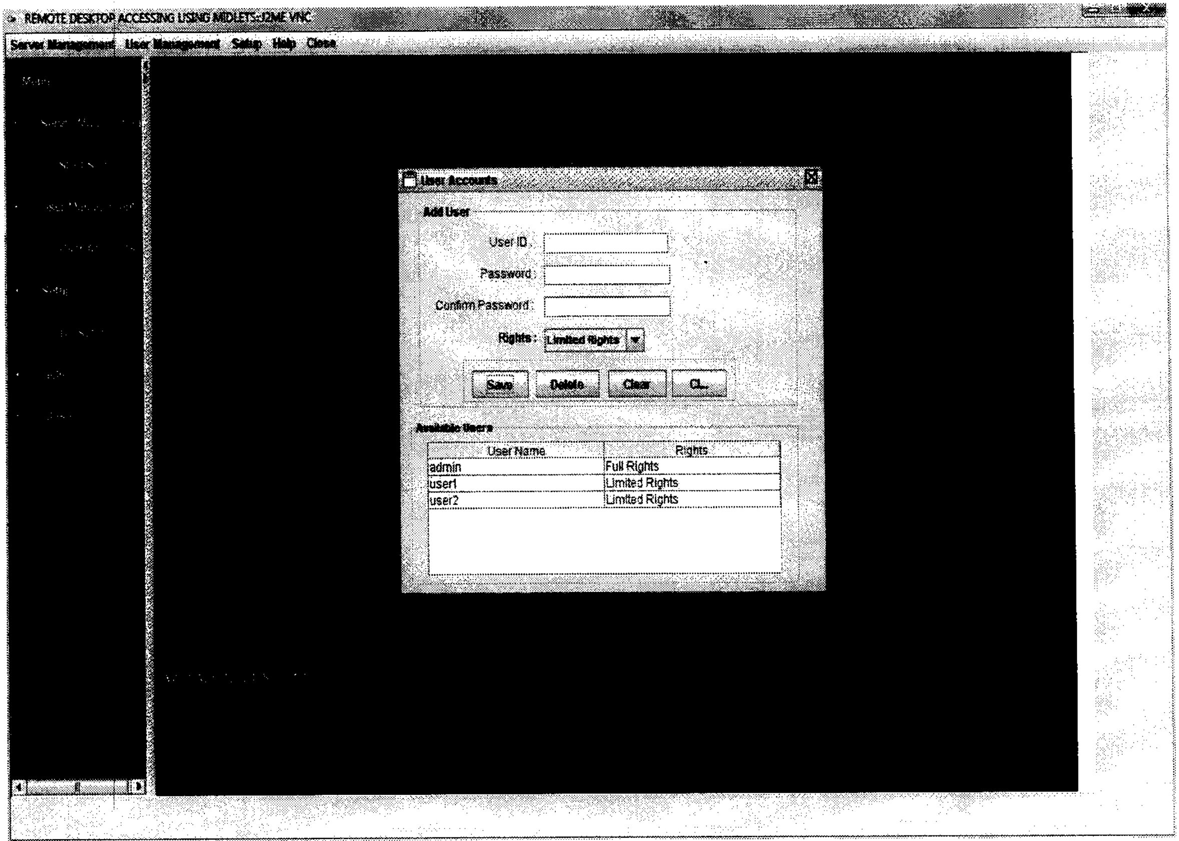
Rights : Limited Rights

Available Us

admin
user1

REMOTE DESKTOP ACCESSING USING ...

ADDED Sucessfully



User ID :

Password :

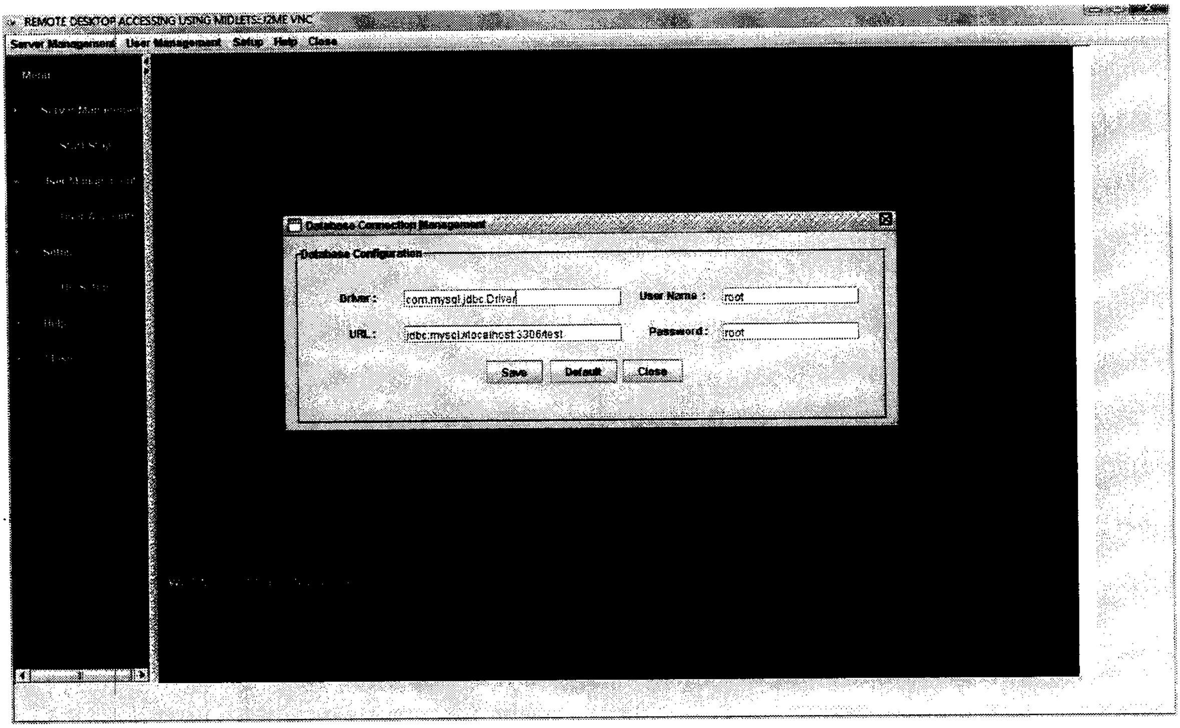
Confirm Password :

Rights : Limited Rights

Available Users

User Name	Rights
admin	Full Rights
user1	Limited Rights
user2	Limited Rights

6. Database Setup:



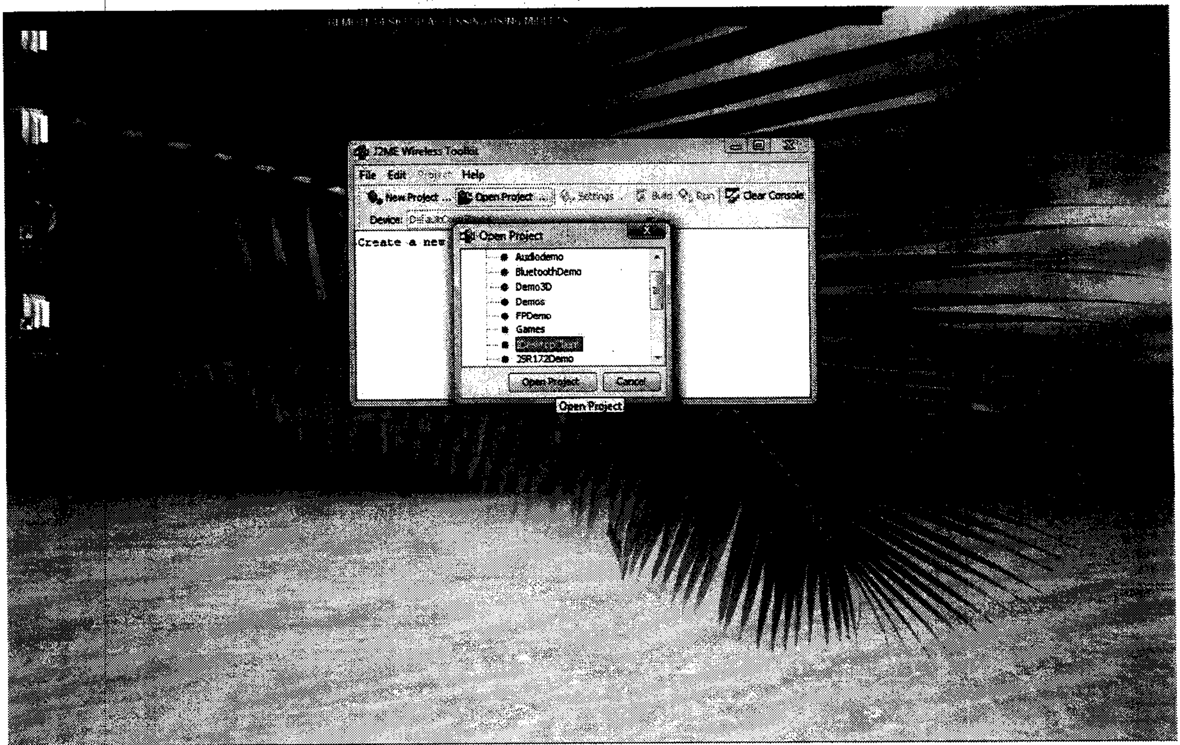
Driver : com.mysql.jdbc.Driver

User Name : root

URL : jdbc:mysql://localhost:3306/test

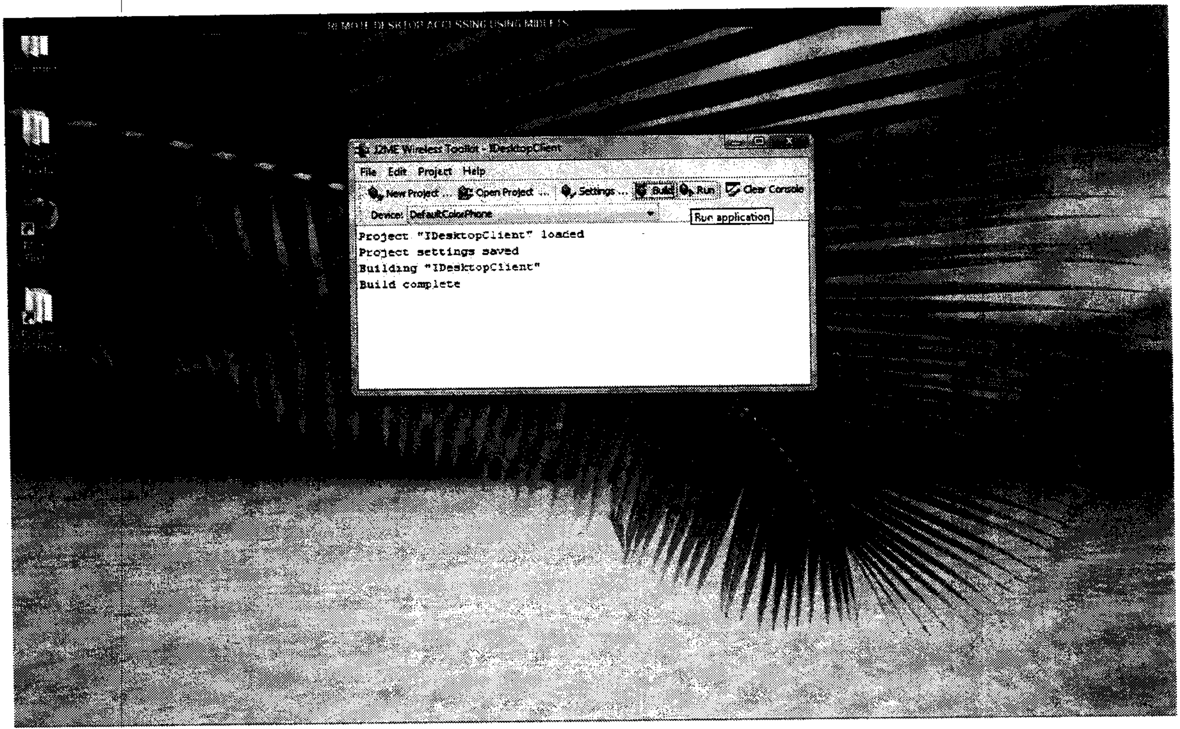
Password : root

7. Starting the client:



Create a new

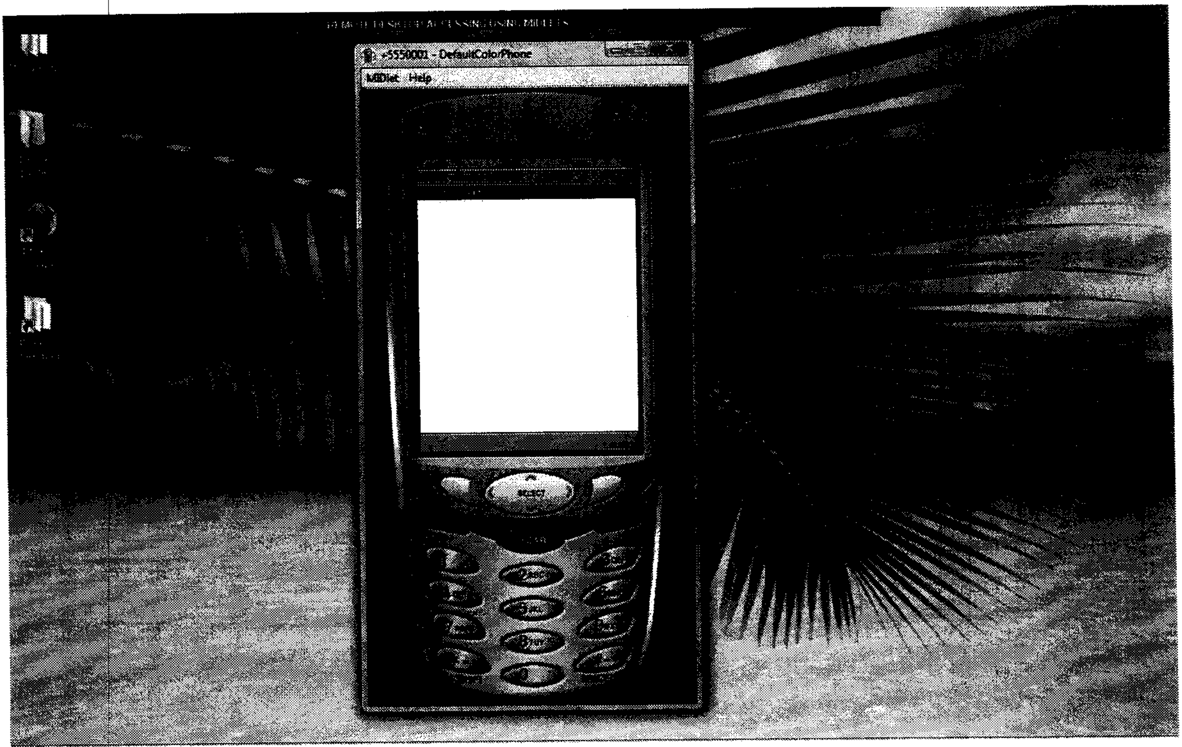
8. Loading the Client:



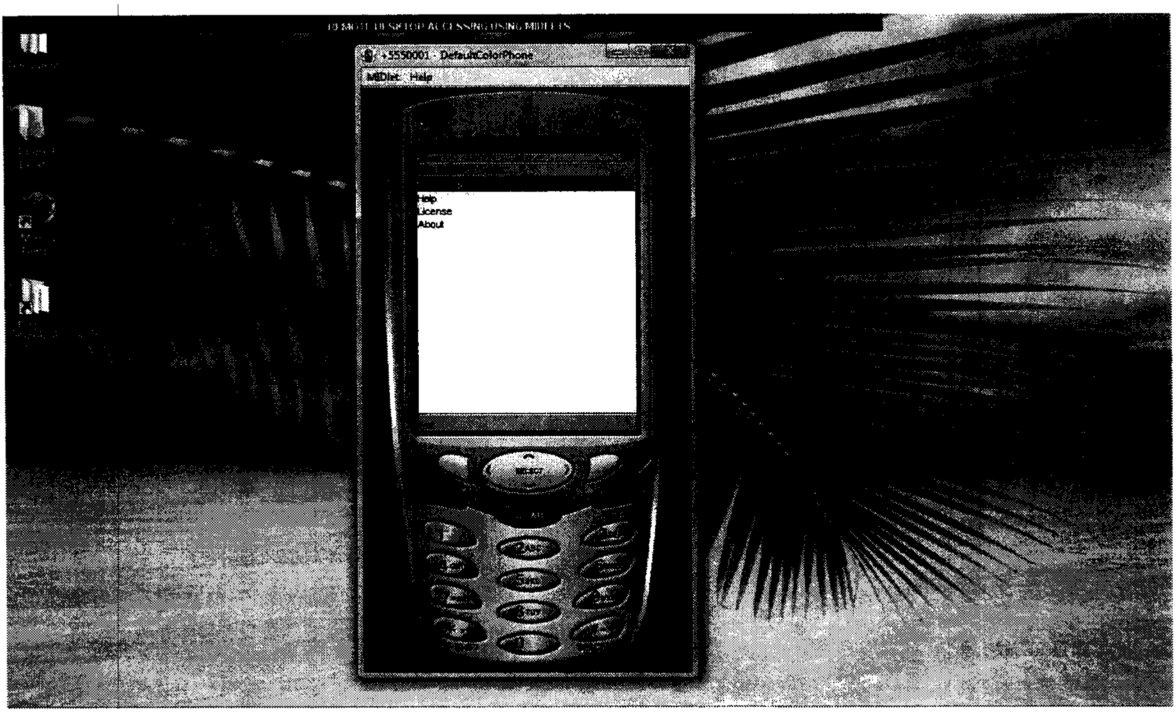
Device: DefaultColorPhone

```
Project "IDesktopClient" loaded
Project settings saved
Building "IDesktopClient"
Build complete
```

9. Selecting the application:

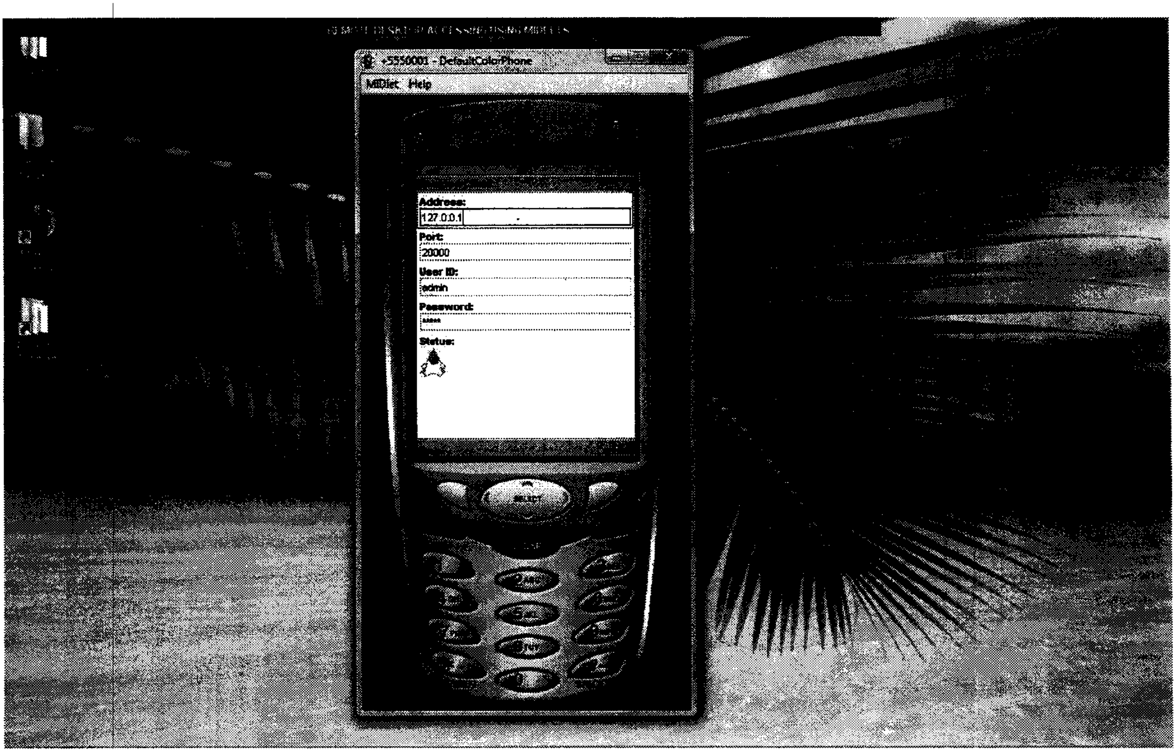

REMOTE DESKTOP ACCESSING USING MIDLETS
+5550001 - DefaultColorPhone
MIDlet Help
SELECT

10. Selecting the VNC system



Help
License
About

11. Setting the IP address and port number:



Address:
127.0.0.1

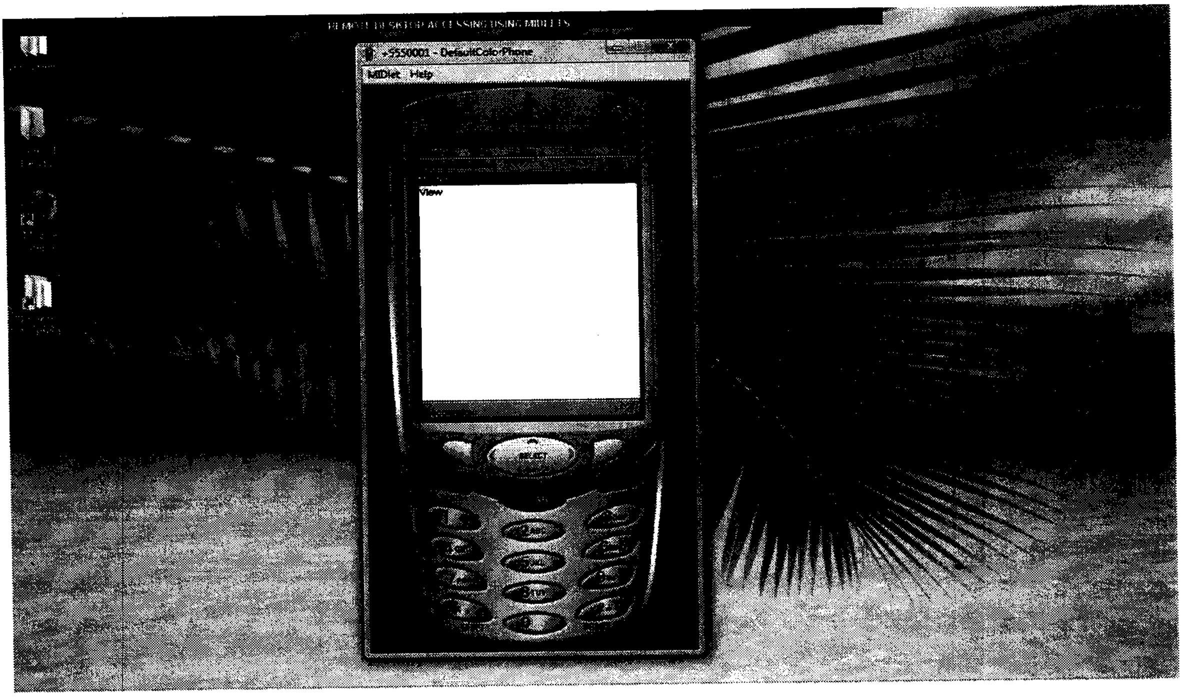
Port:
20000

User ID:
admin

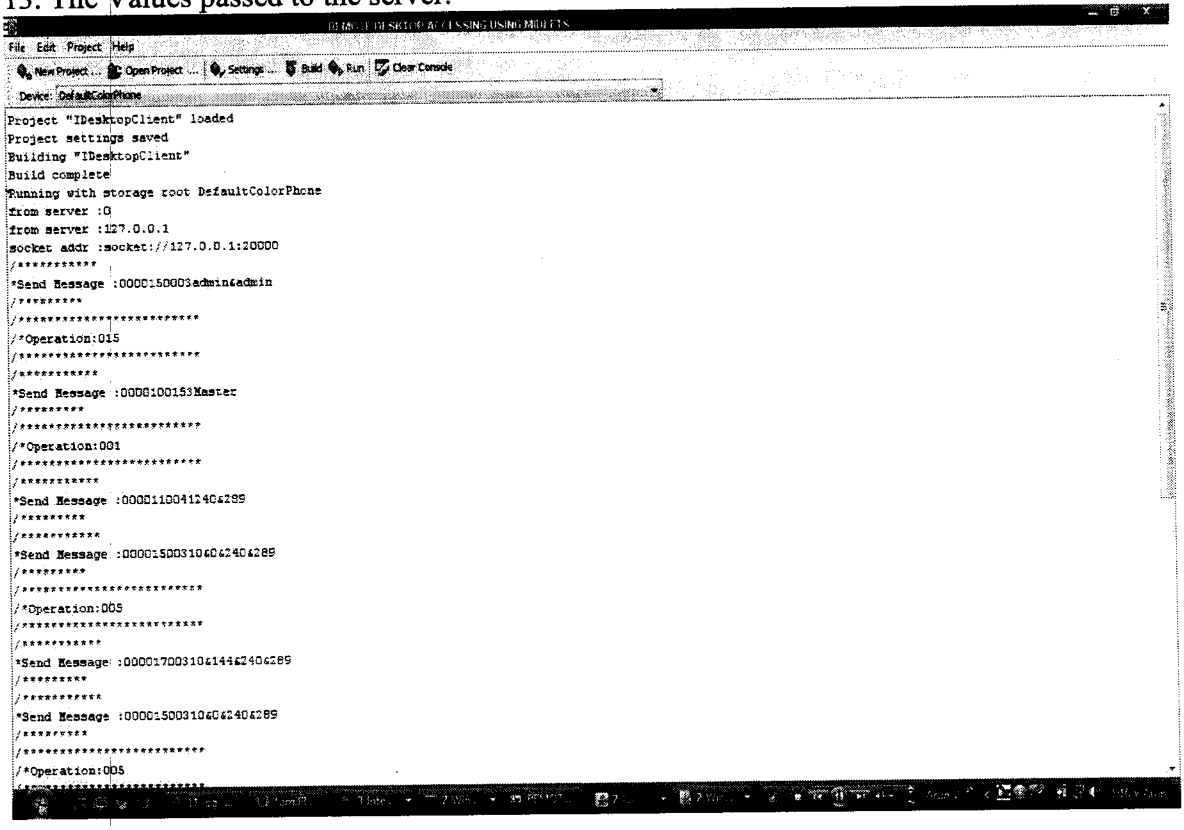
Password:
*****

Status:

12. Mode Selection:


REMOTE DESKTOP ACCESSING USING MIDLETS
+5550001 - DefaultColorPhone
MIDlet Help
View

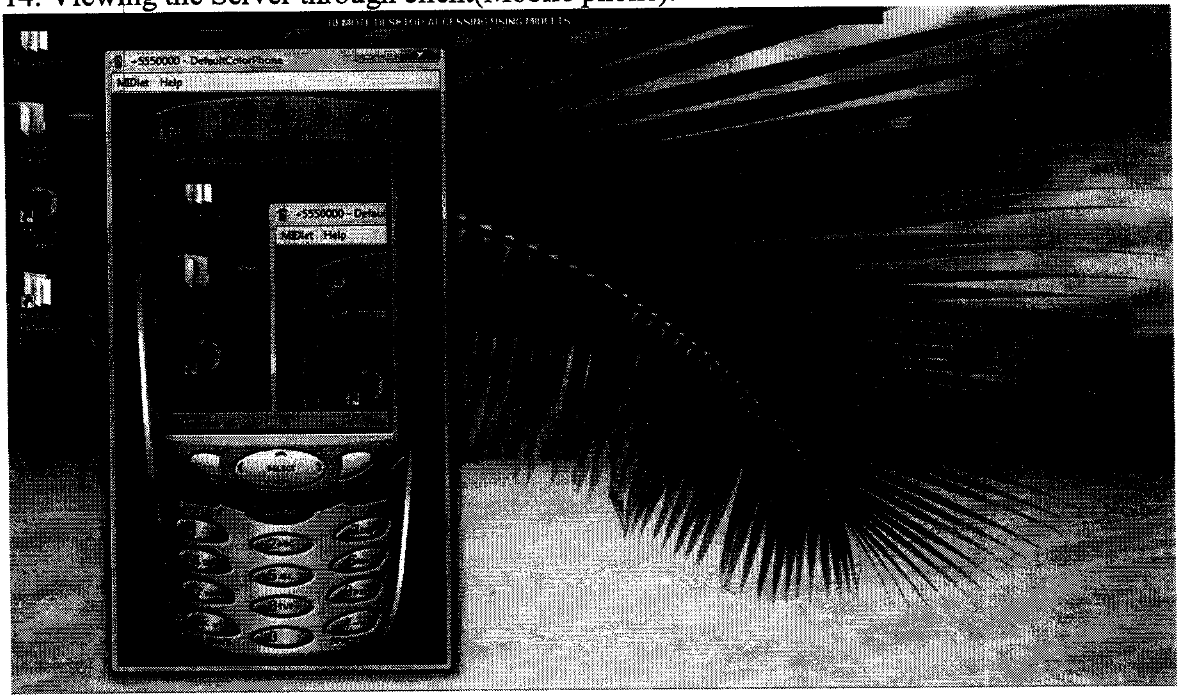
13. The Values passed to the server:


REMOTE DESKTOP ACCESSING USING MIDLETS
File Edit Project Help
New Project ... Open Project ... Settings ... Build Run Clear Console

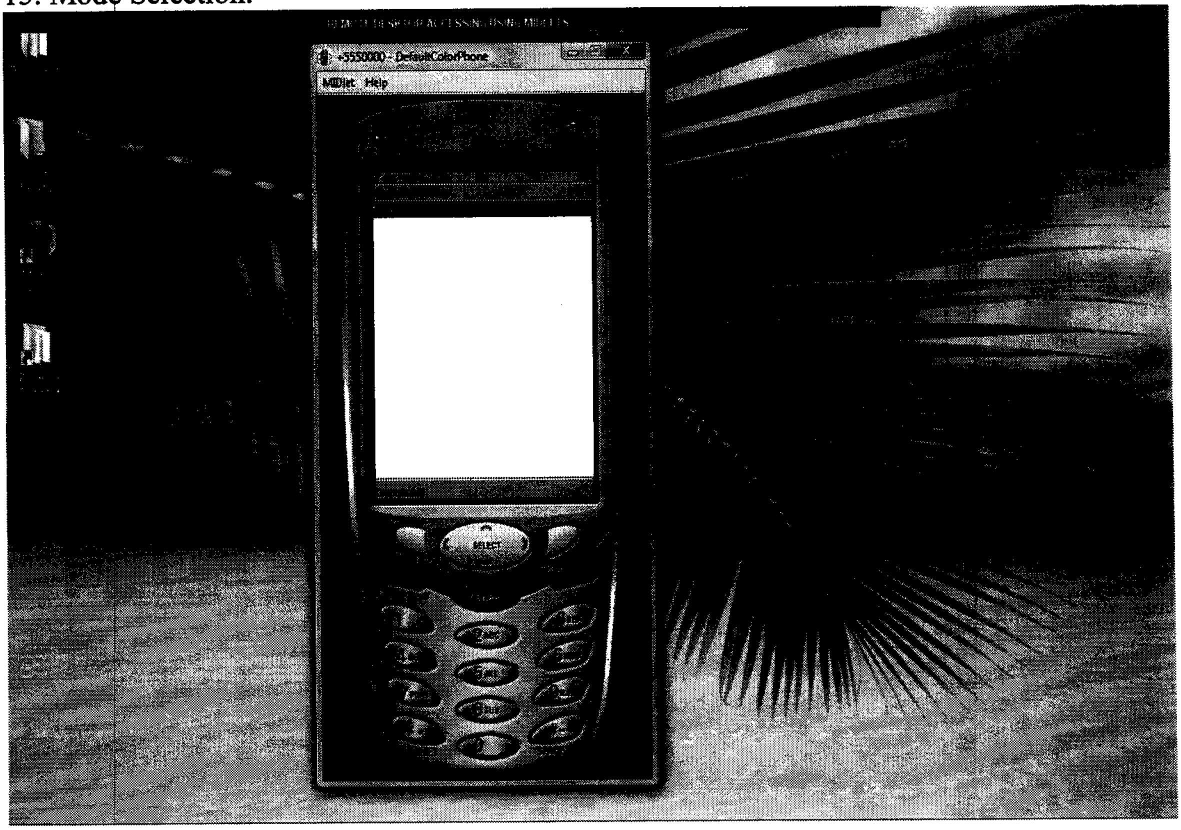
Device: DefaultColorPhone

```
Project "IDesktopClient" loaded
Project settings saved
Building "IDesktopClient"
Build complete
Running with storage root DefaultColorPhone
from server :0
from server :127.0.0.1
socket addr :socket://127.0.0.1:20000
/***********
*Send Message :0000150003admin&admin
/*********
/************************
/*Operation:015
/************************
/***********
*Send Message :0000100153Master
/*********
/************************
/*Operation:001
/************************
/***********
*Send Message :000011004124O&289
/*********
/***********
*Send Message :00001500310&0&240&289
/*********
/************************
/*Operation:005
/************************
/***********
*Send Message :00001700310&144&240&289
/*********
/***********
*Send Message :00001500310&0&240&289
/*********
/************************
/*Operation:005
```

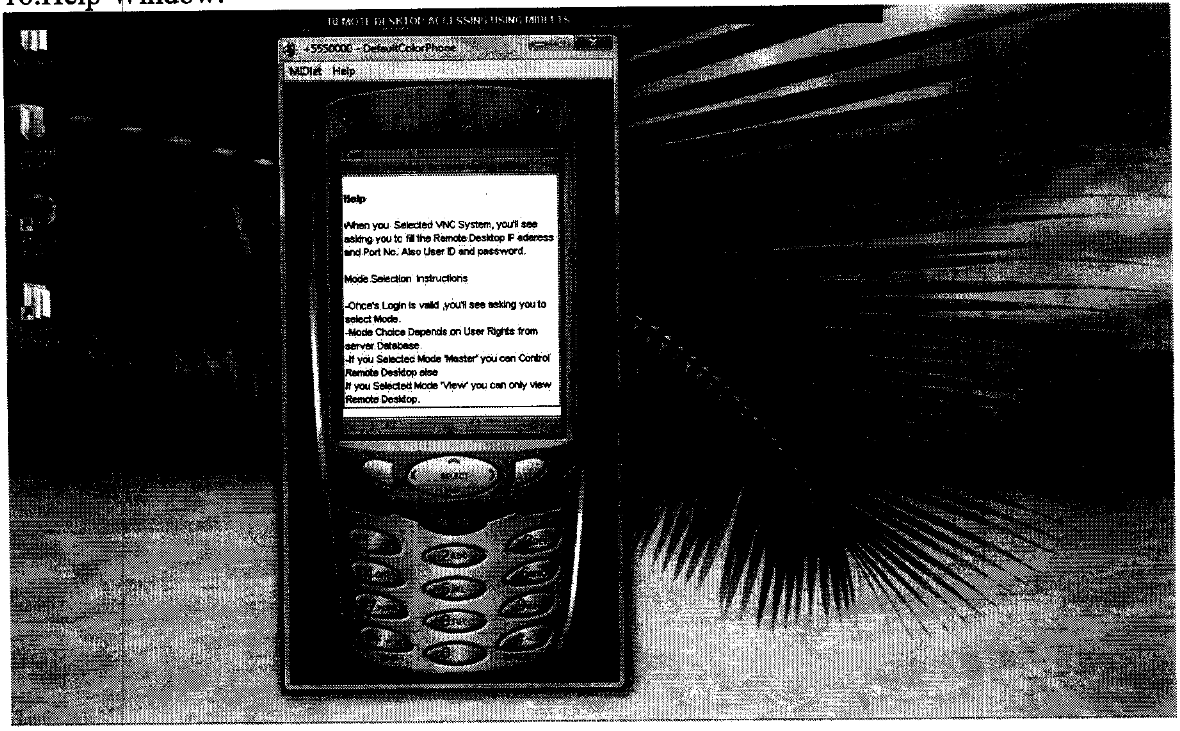
14. Viewing the Server through client(Mobile phone):



15. Mode Selection:



16.Help Window:



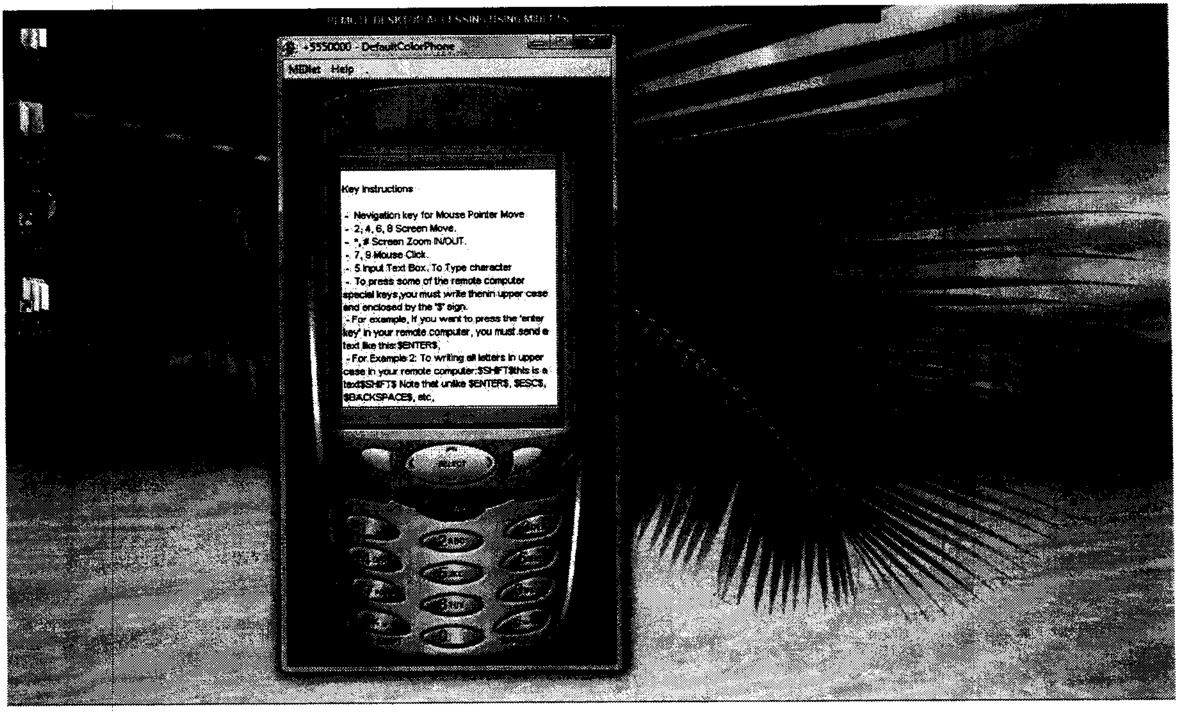
REMOTE DESKTOP ACCESSING USING MIDLETS
+5550000 - DefaultColorPhone
MIDlet Help
SELECT

Help

When you Selected VNC System, you'll see asking you to fill the Remote Desktop IP adaress and Port No. Also User ID and password.

Mode Selection Instructions

-Once's Login is valid ,you'll see asking you to select Mode.
-Mode Choice Depends on User Rights from server Database.
-If you Selected Mode 'Master' you can Control Remote Desktop else
If you Selected Mode "View" you can only view Remote Desktop.



REMOTE DESKTOP ACCESSING USING MIDLETS
+5550000 - DefaultColorPhone
MIDlet Help
SELECT

Key Instructions

- Nevigation key for Mouse Pointer Move
- 2, 4, 6, 8 Screen Move.
- *, # Screen Zoom IN/OUT.
- 7, 9 Mouse Click.
- 5 Input Text Box. To Type character
- To press some of the remote computer special keys,you must write thenin upper case and enclosed by the '$' sign.
- For example, If you want to press the 'enter key' in your remote computer, you must send a text like this:$ENTER$,
- For Example 2: To writing all letters in upper case in your remote computer:$SHIFT$this is a text$SHIFT$ Note that unlike $ENTER$, ESC, $BACKSPACE$, etc,

10. REFERENCES

BOOKS:

1. Elias M.Awad (2000) **"System Analysis and Design"**, Galgotia Publications.
2. Roger S.Pressman (2000) **"Software Engineering"**, McGraw-Hill Publishing Company Limited.
3. Herbert Schildt(2002) "**Java 2: The Complete Reference"**, Tata McGraw-Hill Edition.

WEB SITES:

1. www.w3schools.com
2. www.sun.com
3. www.netbeans.org
4. www.apache.tomcat.org
5. www.codeproject.com