A
Project Report
On

Remote Computing using Java (RCJ)

S.S. BE-IV, 2012-13

Prepared At
Department of Computer Science,
Faculty of Tech & Engg, Maharaja Sayajirao University of Baroda.

Prepared By
Shlok Desai (exam no – 446)
Pranav Masariya (exam no - 461)
Viplav Kadam (exam no – 457)
Ankur Sorathia (exam no – 472)

THE MAHARAJA SAYAJIRAO UNIVERSITY OF BARODA
सत्यं शिवं सुन्दरम्

Faculty of Technology and Engineering,
Maharaja Sayajirao University of Baroda.

Abstract

Remote Computing using Java (RCJ) is a client/server software package allowing remote network access to graphical desktop. This software enables you to get a view of the remote machine desktop and thus control it with your local mouse and keyboard. It can be used to perform remote system control and administration tasks in Unix, Windows and other assorted network environments. It also provides features like file transfer, text-chatting with proper authentication using SSL .

This software requires a TCP/IP connection between the server and the viewer, which works on LANs. Each computer has a unique IP address and may also have a name in the DNS. User will need to know the IP address or name of the server when a viewer wants to connect to it.

The server is designed to make the client as simple as possible, so it is usually up to the server to perform any necessary translations. For example, the server must provide pixel data in the format the client wants. Each desktop is like a virtual X display, with a root window on which several X applications can be displayed. Servers mirror the real display to a remote client, which means that the server is not 'multi-user'. It does, however, provide the primary user of a PC with remote access to their desktop. The server processing includes retrieving the pixel information and sending it.

The input side is based on a standard workstation model of a keyboard, and multi-button pointing device. Input events are sent to the server by the client whenever the user presses a key or whenever the mouse is moved. It also requests for all the possible specific parameters that the server can handle, for instance the colour mode, pointer events and so on.

Table of Content

1.0 Introduction

1.1 Project Details

Remote Computing using Java is a Desktop Application which allows the user to access computer located remotely. It also provides features like text-chatting, file-transfer with proper authentication and security. It uses TCP-IP connection via LAN.

1.2 Purpose

Remote Computing using Java provides businesses the ability to login and access computers remotely. Utilizing software enables personnel to transfer files quickly and easily, and communicate by instant message or text chat inter-communication from any PC. As with many IT departments having the ability to remotely manage computers, expanding the breath of service level and reach that can be provided by a centralized support groups. e.g Client complains network operator about some configuration issues. Network operator using this software will solve the issue instantly from a remote distance and utilizes other modules like chatting and file-transfer for better communication. This saves time, transportation cost and customer is also satisfied with the instant service.

1.3 Project Scope

This project works for windows operating system, Linux operating system and all other operating system for which Java Virtual Machine is developed.

1.4 Objective

Using this application user can do:

- Reducing the need to travel to provide computer support.
- Provides quick response to remote issues.
- Centralizing and streamlines computer support.
- Can help in reducing overall costs.

User can’t do:

- User can’t use this software without LAN connection.
- Remotely Alt+Ctrl+Delete is not working.

1.5 Technology and Literature

JAVA: Application uses Java Language.

Java can create all kind of application that you could create with any other programming language. Java technology is high-level programming and a platform independent language. Java is designed to work in distributed environment on the internet. It works on the concept of object-oriented programming model. Java is a well-known technology which allows for software designed and written only once for a “virtual machine” to run on different computers supports various operating systems.

NETBEANS: Application uses NETBEANS as frontend.

Netbeans is easily available on Internet. Anyone who knows Internet can easily download the Netbeans. It is available in different versions. And also the video tutorials are available on internet that can guide you to install it on your computer.

Local Area Network:

A Local Area Network (LAN) is a network that connects computers and devices in a limited geographical area such as home, school, computer laboratory, office building or closely positioned group of buildings. Each computer or device on the network is a node. Current wired LANs are most likely to be based on Ethernet technology. The defining characteristics of LANs, in contrast to WANs (Wide Area Networks), include their higher data transfer rates, smaller geographic range, and no need for leased telecommunication lines.

Ethernet Physical Layer:

The Ethernet physical layer is the physical layer component of the Ethernet family of computer network standards. The Ethernet physical layer evolved over a considerable time span and encompasses quite a few physical media interfaces and several magnitudes of speed. The speed ranges from 1 Mbit/s to 100 Gbit/s while the physical medium can range from bulky coaxial cable to twisted pair to optical fiber. In general, network protocol stack software will work similarly on all physical layers.

Threads:

A thread is a program's path of execution. Most programs written today run as a single thread, causing problems when multiple events or actions need to occur at the same time. For example, a program is not capable of drawing pictures while reading keystrokes. The program must give its full attention to the keyboard input lacking the ability to handle more than one event at a time. The ideal solution to this problem is the seamless execution of two or more sections of a program at the same time which is allowed by threads. Multithreaded applications deliver their potent power by running many threads concurrently within a single program.

RMI:

Remote Method Invocation(RMI) provides Java programs abstraction in communication. In RMI we do not have to create sockets and care about closing the sockets. All things are done by RMI. We just have to get remote object from the RMI server.

Bits Per Pixel:

The number of bits of information stored per pixel of an image or displayed by a graphics adapter. The more bits there are, the more colours can be represented, but the more memory is required to store or display the image.

2.0 Project Planning

We have used traditional approach for the Project Development, because it is structured planning & it goes through step by step.

- Project initiation stage.
- Project planning and design stage.
- Project execution and construction stage.
- Project monitoring and controlling systems.
- Project completion.

3.0 System Requirement Study

3.1 Study of Current System

Currently available systems are:

- Windows Remote Desktop: Remote desktop uses Remote Desktop Protocol control remote computer, but it is for windows only.
- Team Viewer: This software is used widely across the globe. It is free for home users and non-free for commercial use. For commercial use licence has to be bought which is very costly.

3.2 User Characteristics

Two users at a time can use this application.

3.3 Hardware and Software Requirements

Minimum configuration for our system is as below:

Hardware Requirements: Computer

Software Requirements: 1) O.S. – Windows 2000/XP/Windows 7/Windows 8
2) Netbeans

3.4 Assumptions

Here we have assumed that the user has an internet connection and the jar file of the software should be properly installed in the computer.

4.0 System Analysis

4.1 Software Requirement Specification

Requirement Specification sets out the system services and constraints in detail which are abstract in requirement definition. It should state what the system should do, not how it should be implemented. It can be described as functional and non-functional requirements.

Functional requirements of proposed system are as below:-

R.1: Manage own account
R.1.1: Generate connection Id and password

Description: Whenever client starts the software on his machine, he should specify his own username and desired password respectively. User has to provide password every time so no need to remember it.
Input : Username and password
Output : Account created.

R.2: Controlling Remote computer
R2.1: Connecting remote computer

Description: After performing authentication by entering Username, password and IP-address of the remote pc, client is able to see the remote pc screen on his own computer constantly.
Input: Username, password and IP address of the remote pc to be connected.
Output: Screen of the remote computer is shown in a window.

R.2.2: Controlling with mouse

Description: Whenever client presses mouse keys on the window of remote computers screen this same event should be performed on the remote computer simultaneously and the resulted screen should be displayed.
Input: Clicking, pressing mouse buttons and dragging mouse wheel.
Output: The mouse events are performed on the remote computer.

R.2.3:Controlling with keyboard

Description: Whenever client presses keyboard keys during focussing of software’s window, the respected key event should be performed on the remote pc as if keys were pressed on the remote computer.
Input: Any keyboard keys.
Output: Respected key event should be performed on the remotely connected computer.

R.3: Managing resolution
Description: Client can decide to keep the resolution of window in which he wants to see the remote computer screen.
Input: Selection of needed resolution.
Output: The required resolution remote screen is visible.

R.4: Managing display of the screen

Description: Client is given a option to choose black & white, full colour mode, 256 colour or 15 bit colour mode display.
Input: Select desired colour option.
Output: The desired colour mode is visible.

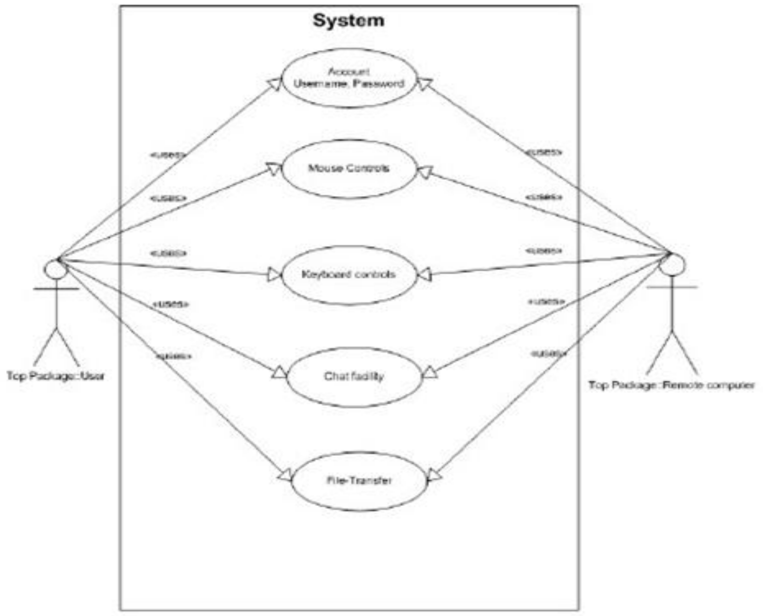
R.5: Chat facility

Description: Client should be able to chat with the remote computer.
Input: Open chat window.
Output: Client can chat with the remote computer.

R.6: File Transfer

Description: In this client gives the path of the file which he want to send to the remote computer and that file should be transferred.
Input: File which client wants to transfer.
Output: File is transferred to remote computer.

4.2 Use-Case Diagram



Use-Case Diagram

4.3 Features of the System

- User Friendly.
- Easily Maintained.
- Faster with remote reboot options.

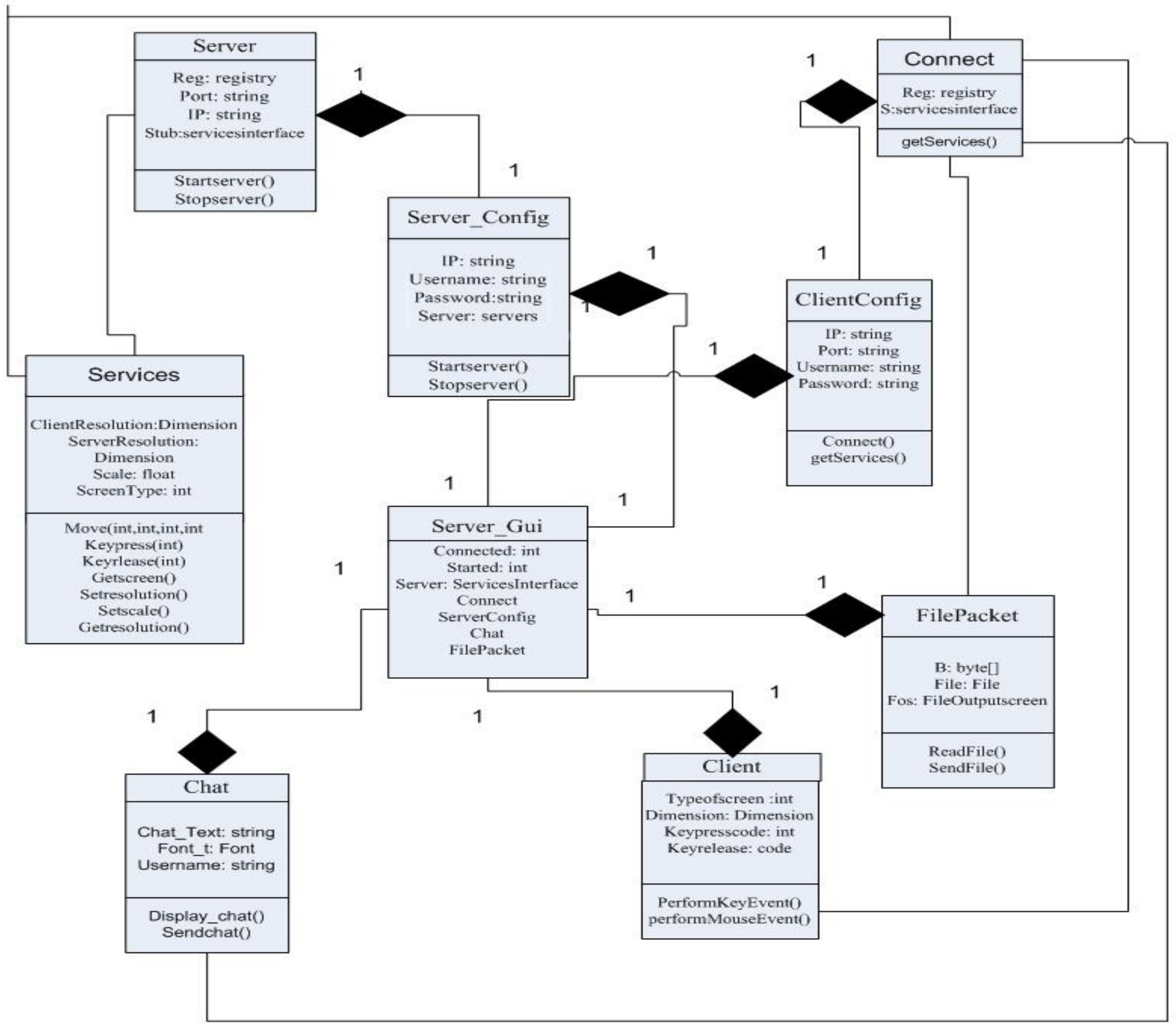
5.0 System Design

This presents the logical data model that defines entities required for this application.

Overview

The proposed system is developed using Java, it is platform independent. Thus it will run under any operating system like Windows or Linux. The main part of the application is a GUI that is used to browse other machines in the network. The GUI should be developed using Java Swing. The system should be modularized into a client part and a server part so that each part could be run separately. The server part should acts as the file server for the local client as well as the remote server. The client part should provide a GUI for easy user interaction. All request from the client are passed to the local server only. Requests to the remote server are forwarded by the local sever.

CLASS DIAGRAM



6.0 Implementation

6.1 GUI Coding

The GUI is built using JAVA Swing. Swing has many GUI components like jframe, jpanel, jbutton, jtextarea, jtextfield, jbutton, jcheckbox, jinternelframe, jcombobox, and many others. Swing provides different pluggable look and feel like nimbus look and feel, metal look and feel, GTk look and feel and the platform's look and feel. Our application uses nimbus look and feel for the GUI.

```
try
{
        for (LookAndFeelInfo info : UIManager.getInstalledLookAndFeels())
        {
                if ("Nimbus".equals(info.getName()))
                {
                        UIManager.setLookAndFeel(info.getClassName());
                        break;
                }
        }
}

Catch(Exception e)
{
}
```

The application uses both the socket programming and RMI for transfer of data between two computers.
The RMI (Remote Method Invocation) is that you don't need to create socket and other lower level things all you need to care about sending a parameter to the function on the Remote RMI server and receive a return value to your computer.

The application can connect to at most one remote server at time. The user can either remote control and chat or Transfer file and chat. IDE used for the coding is NetBeans. It is very user friendly ide supports intellisense debugging refactoring the code, templates and many more.

- First part Server status is a Text Area where the information regarding the server is displayed like IP address, port number.
- Second part consists of the buttons to start or stop server, connect or disconnect to remote server and to start remote controlling.
- The third part of the GUI is a text area which displays information regarding the connected remote server like IP address, port. File transfers output is displayed on this area.
- The fourth part consists of two buttons select file and send file. If you are connected to any server than you can send file from your computer to another computer.
- Server configuration window can be opened by clicking on the start server and will prompt you for the IP address of your computer and port on which you want to start

the server.

- Client configuration window will open when you click on the connect server button. Then it will prompt for the IP address and port number at which the remote server is running.
- Remote control window has one button that is start button. By clicking on the start button the GUI will prompt you for the resolution and type of the screen in which you want see your remote computer. After giving the configuration it will start transferring screen from remote computer to your computer.
- Chat window can be opened by selecting chat option from the Extras menu from main GUI. For chat you need server running at both the side and connected to each other. The chat area will automatically scroll down when new text is added to it.

6.2 Program-Modules Specifications

Below are the classes which make the GUI part of the application:

- Server_gui(main gui)
- Server_configuration(gui for server ip address and port)
- Client_configuration(to connect to remote pc this config must be set)
- Screen_config(resolution and type of screen)
- Chat
- Client (For remote conrolling)

Below are the classes that are for server:

- Services interface (declares service provided by the server to user)
- Services (implementation of the services listed in above interface)
- Server (for starting and stopping server)
- Connect (to connect or disconnect the remote computer)

The remote server sends the captured screen to the user who is controlling. The user can use his mouse or keyboard to control the remote pc.

When using mouse to control the mouse co-ordinates are transferred to the remote server and type event is passed to the server like dragging event, mouse move event, click event, press event or double click event.

When using keyboard to control remote pc it will pass the keycode to the server and the type of keyevent like keypress or keyrelease.

7.0 Test Cases

Server Side:

- Starting server on already occupied ports or incorrect ports like -8000 will not create server and give proper error message.
- If server is already started and trying to start it again will give appropriate message.
- Stopping server without starting server will give appropriate message.

Client Side:

- Connecting to a computer where server is not running will give message to start the server at other side.
- Connecting to a computer which is not part of the local area network will prompt you to give proper IP address.
- Connecting with proper IP address but incorrect port number will also give error message.
- If server is running at other side as well as IP address and port number are correct will also give error message if username or password are not correct.

Remote Controlling:

- If remote controlling window is running and if the server is disconnected or stopped will close the client window and will let you try again.
- Application will not let you transfer file to remote server If remote controlling window is opened and running.
- Application will let you chat with other side as well as control remote pc at one time.

File Transfer:

- To transfer file server should be started and connected otherwise will not send the file.
- Application will not let you open remote controlling window if file transfer is being done. But chat window can be opened at any time.
- File with size <2 gb can be transferred. Will take time on the basis of speed of the network.
- Folders are not transmitted.

Chat:

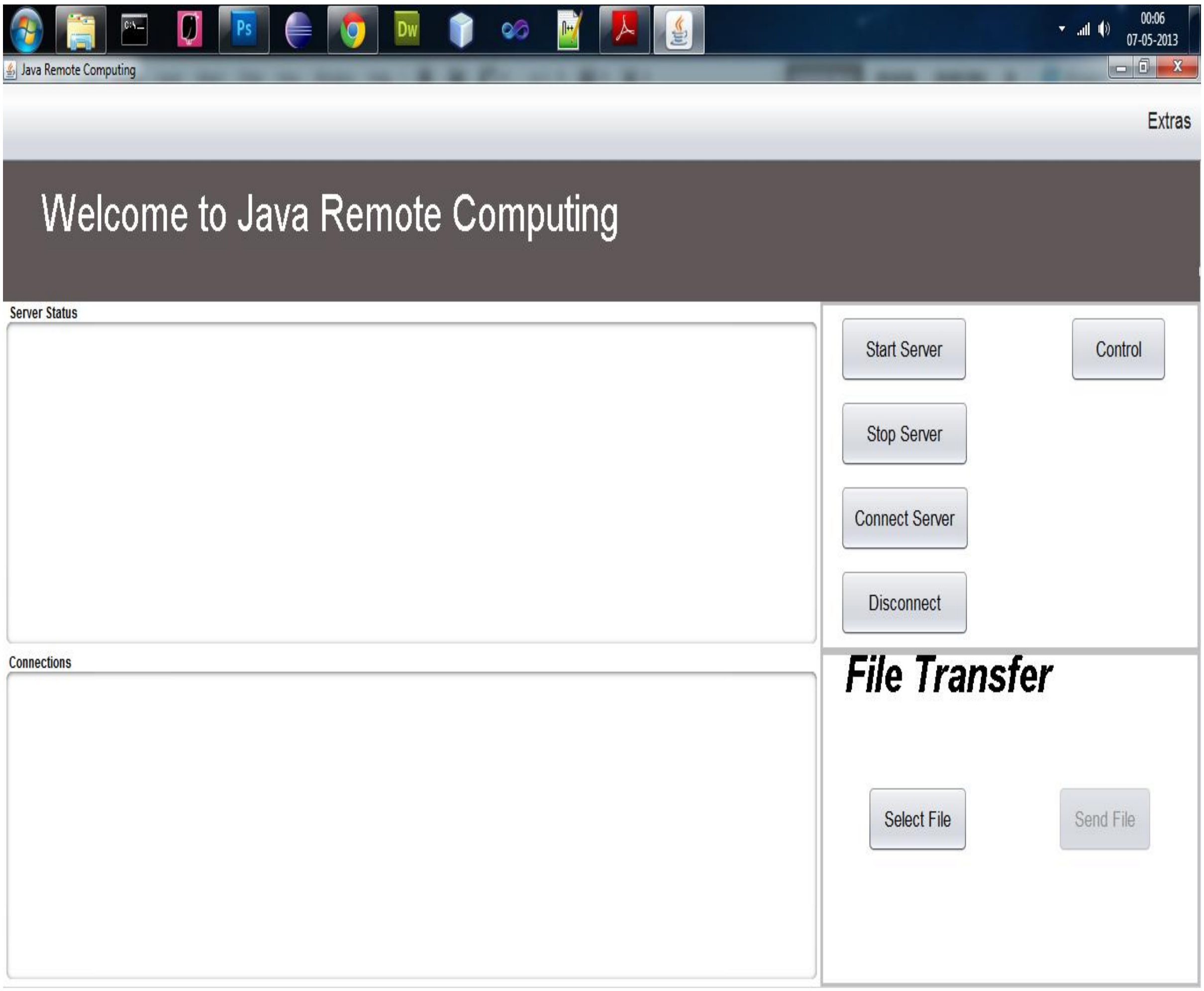
- If both side server are not created and connected to each other chat window will exit and give you proper message.
- If server is stopped or disconnected chat window will prompt you to start the Server or connect the server.

8.0 User Manual

(User must have JRE 6.0 or later version in his/her computer.)

8.1 Initial Instructions

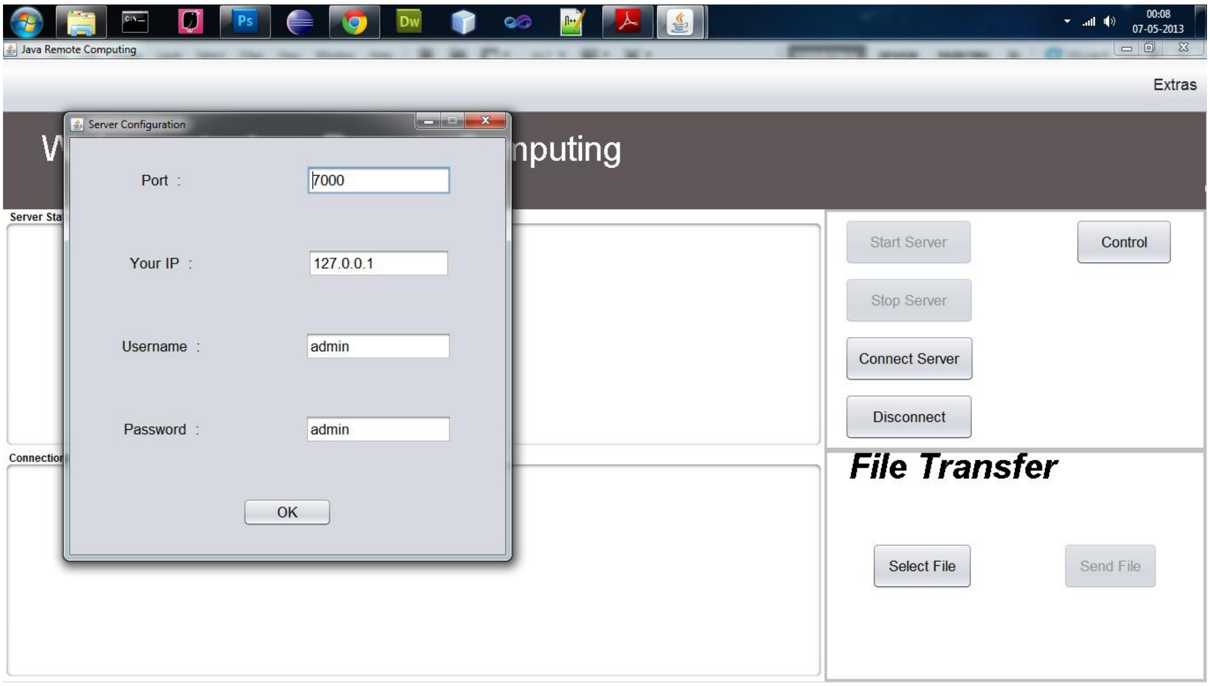
- Java Remote Computing(JRA) works only in a Local Area Network. It will not work on the internet.(Future version will !)
- To use JRA you must have JRA.jar file at both the end computer in LAN and your pc's firewall should allow this application. Otherwise it will create some errors.(It may not display screen)
- To start the application just double click on the .jar file or goto cmd and change the directory to the directory where your jar file is and execute "java -jar JRA.jar" command.
- Now wait until your GUI starts properly.



Initial GUI

8.2 Starting and Connecting Server

Once the GUI has been started first you have to start the server by clicking on the 'start server' button. It will prompt for the IP port number, user id and password. IP is just for user's conveniency .It will be displayed at the server status area so user doesn't have to every time check IP from network connections.


Server Configuration

Port : 7000

Your IP : 127.0.0.1

Username : admin

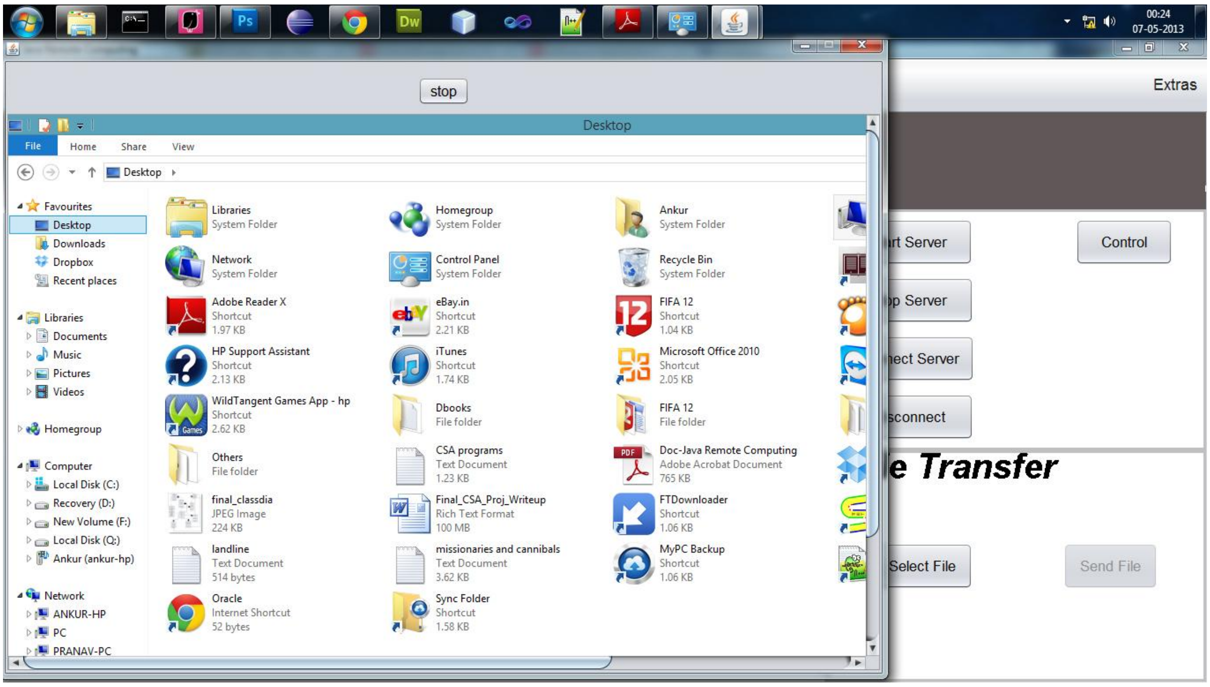
Password : admin

Starting the Server

- ID and Password.
- Once server has been started you can control it from another pc by connecting it.
- To control the remoter pc you must first start the server at the remote pc.
- Now click on the 'connect server' button, it will prompt you for IP, port ,user id and password which you just gave on the remote pc.
- If connection successfully message comes then you can use 'remote control' button to start remote controlling or you can send files from file transfer panel.

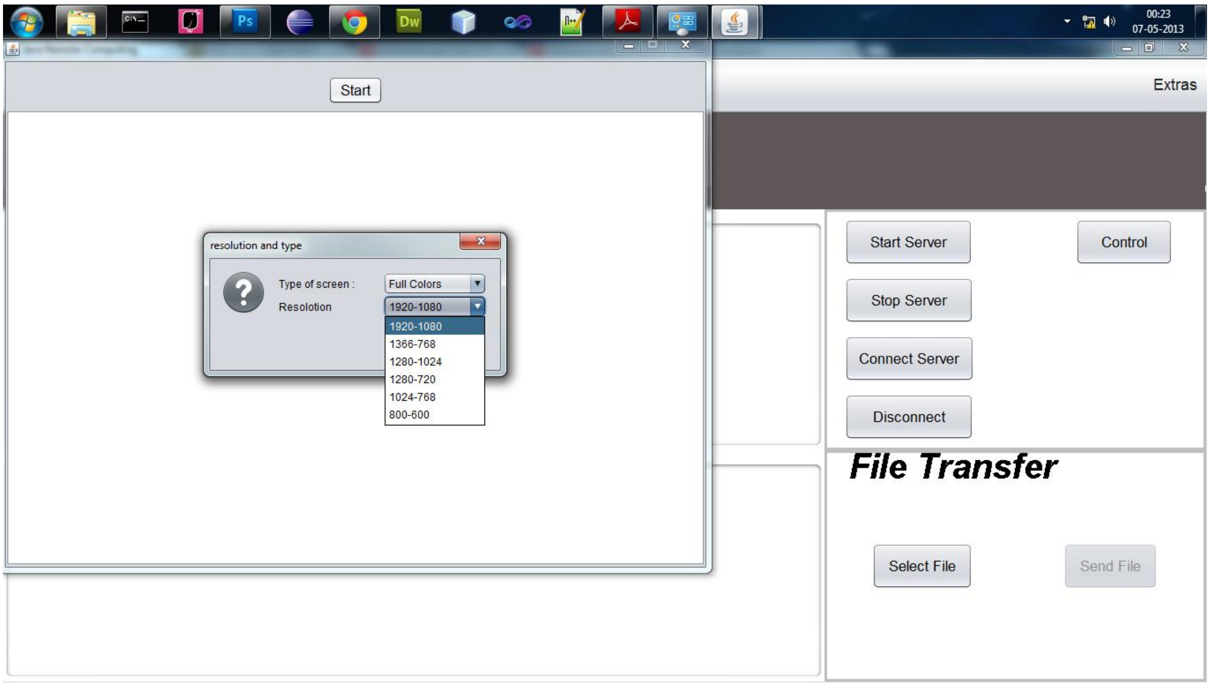
8.3 Remote Control

- ➢ Here u can access mouse as well as keyboard events remotely.



Remote Control

- ➢ You have been given options to change resolution and colour as per requirement.



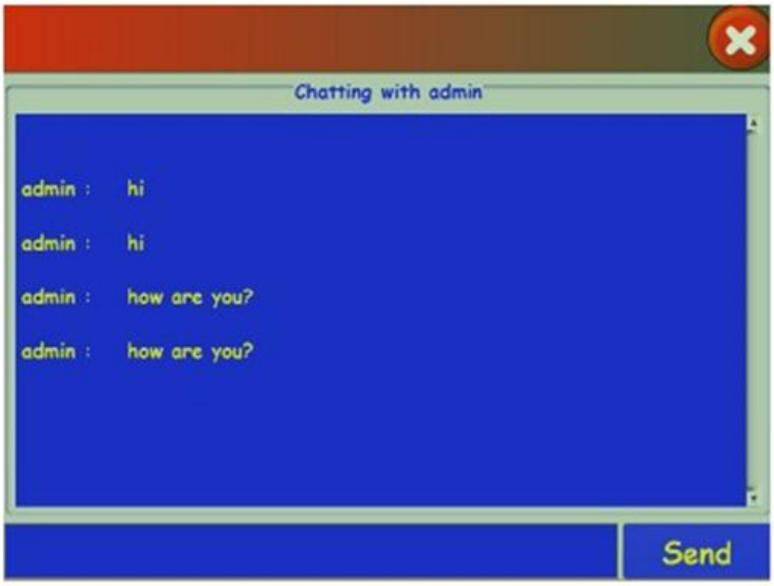
Type of screen : Full Colors

Resolotion 1920-1080

Changing Resolution

8.4 Chatting

- ➢ You can chat to you peer client easily with authenticated connection.



Chatting with admin

admin : hi

admin : hi

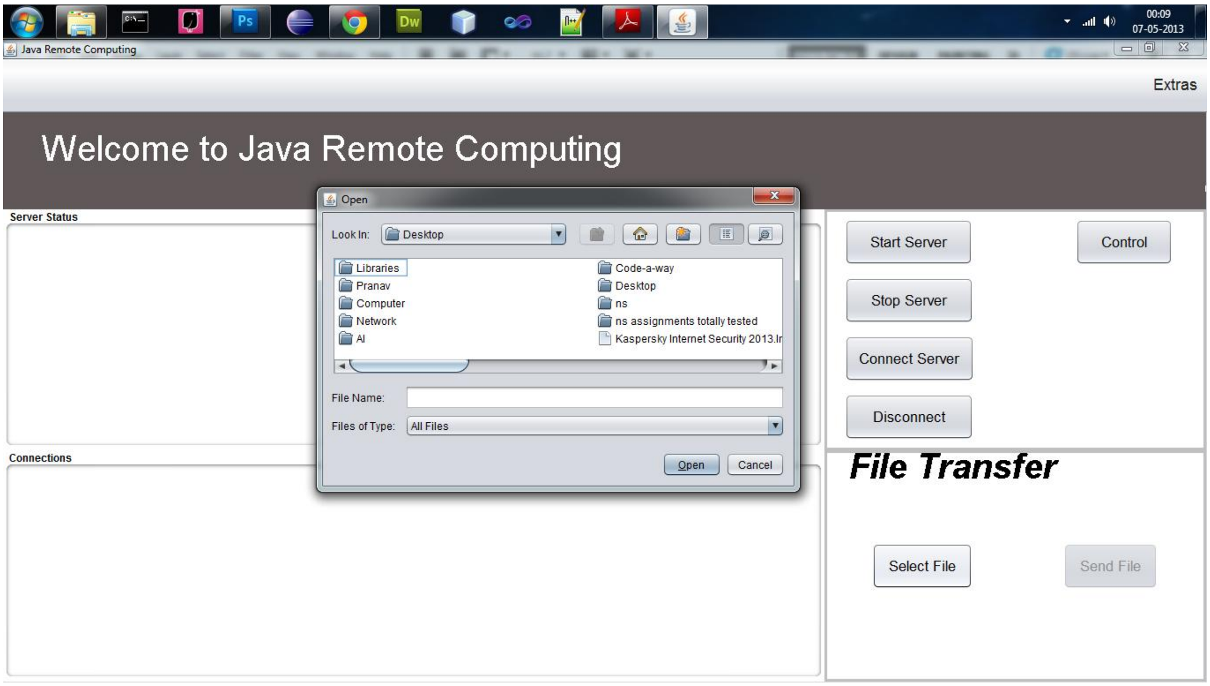
admin : how are you?

admin : how are you?

Chat Interface

8.5 File Transfer

- ➢ You can send files of GBs data to your peer client easily in seconds.



Look In: Desktop

Libraries
Pranav
Computer
Network
AI
Code-a-way
Desktop
ns
ns assignments totally tested
Kaspersky Internet Security 2013.lr

File Name:

Files of Type: All Files

File-Transfer

- ➢ Help option is provided in the software for any further assistance.

9.0 Limitation and Future Enhancement

Limitations:

- Application will not allow transfer of ALT+Ctrl+Delete key combination.
- Using this application we cannot control remote pc connected on internet.

Future Enhancements:

- Future versions will be faster.
- Will be updating audio-video conferencing.
- Future versions will work on internet.
- Access controls for files can be implemented in future versions.

10.0 Summary and Conclusion

Summary of Project work

Following are the various phases that have been experienced while development of project:

- Find the Definition.
- Get the requirement.
- Do the analysis on requirement.
- According to requirement do the designing phase.
- According to design do the coding.
- Found the various problems in coding & got the solution step by step.
- Do the testing & find the limitations.
- Completion of the Project.

Conclusion

The project “Remote Computing Using Java” is working successfully and thus can be further enhanced for the higher level versions.

11.0 Bibliography

11.1 Websites

www.sun.oracle.com

www.teamviewer.com

www.sourceforge.com

www.google.com

11.2 References

The Complete Reference J2SE By Herbert Schillt

Core-Java Head First By Cathay Seira

Core-Java By Cay Horstman