GALGOTIAS UNIVERSITY

(Established under Galgotias University Uttar Pradesh Act No. 14 of 2011)

FILE TRANSFER AND CHAT SYSTEM

A Report for the Evaluation 3 of Project 2

Submitted by

APARAJEET BANERJEE

(1713122001 / 17SCSE122001)

in partial fulfillment for the award of the degree

of

BACHELOR OF COMPUTER APPLICATIONS

IN

Computer Science and Engineering with Specialization of Multimedia & Animation

SCHOOL OF COMPUTING SCIENCE AND ENGINEERING

Under the Supervision of

Dr. ANVESHA KATTI

APRIL/MAY – 2020

TABLE OF CONTENTS

Abstract

This File Transfer and Chat (Message Sending) system has been developed in Java 1.3 which is based on Object Oriented Methodology. There are several packages in Java, but mainly swing packages and networking has been utilized in developing this project.

In order to chat with the respective language, the user must install the editor of that language.

This application works on LAN communication when user login with application request is sent to the server for authentication and gets a response. The server-side database will store information of each user data.

Chat system is very common communication tools that have been used in human in this modern cutting edge technology world. This chat system has become one of the important intermediate tools for everyone to share knowledge and materials via network. So this Chat system for Computer Science Student OGITEC.

The Environment is developed in order to enhance the communication among the students via network so that this will be easy for the discussions related to their studies. This chat system is developed by using peer-t-peer concept and multicast technique and also developed by using the Rapid Application Development (RAD) methodology.

This system is built by using PHP server scripting language. The text messaging in this system does explain the peer-peer concept and multicast technique very well. The output from this system enables the student to chat among themselves in internal chatting System.

Modules Overview:

This project works under two modules, namely Active and Passive. Only passive clients can receive files, but active clients can send as well as receive files. Upon successful transfer of a file, an acknowledgement of successful file transfer is received, and then the passive clients can be disconnected. Any kind of files, including *.fmx* files, *.exe* files and more, can be sent using this system.

Introduction

Online chatting system with display chat history application works on wired networks like LAN. This chat system project has the option to chat with different languages.

Chat application works for multiple users but each user must register with the application. After login user can enter into a group for a chat.

After register and login part is done user can view available user online and offline. User can chat with a person who is logged to the system. In order to chat with the respective language, the user must install the editor of that language.

This application works on LAN communication when user login with application request is sent to the server for authentication and gets a response. The server-side database will store information of each user data.

User Registration Form:

The first step for using this application is the registration form where the user will ask for a login or signup option. User can select register if he is a new user and enter all these details to get a unique id and password.

Using these details he can log in an application. The registration form will have all validations checks for each field.

User Module:

After registration is done user will get user id and password using these details he can log in to the application and use all features.

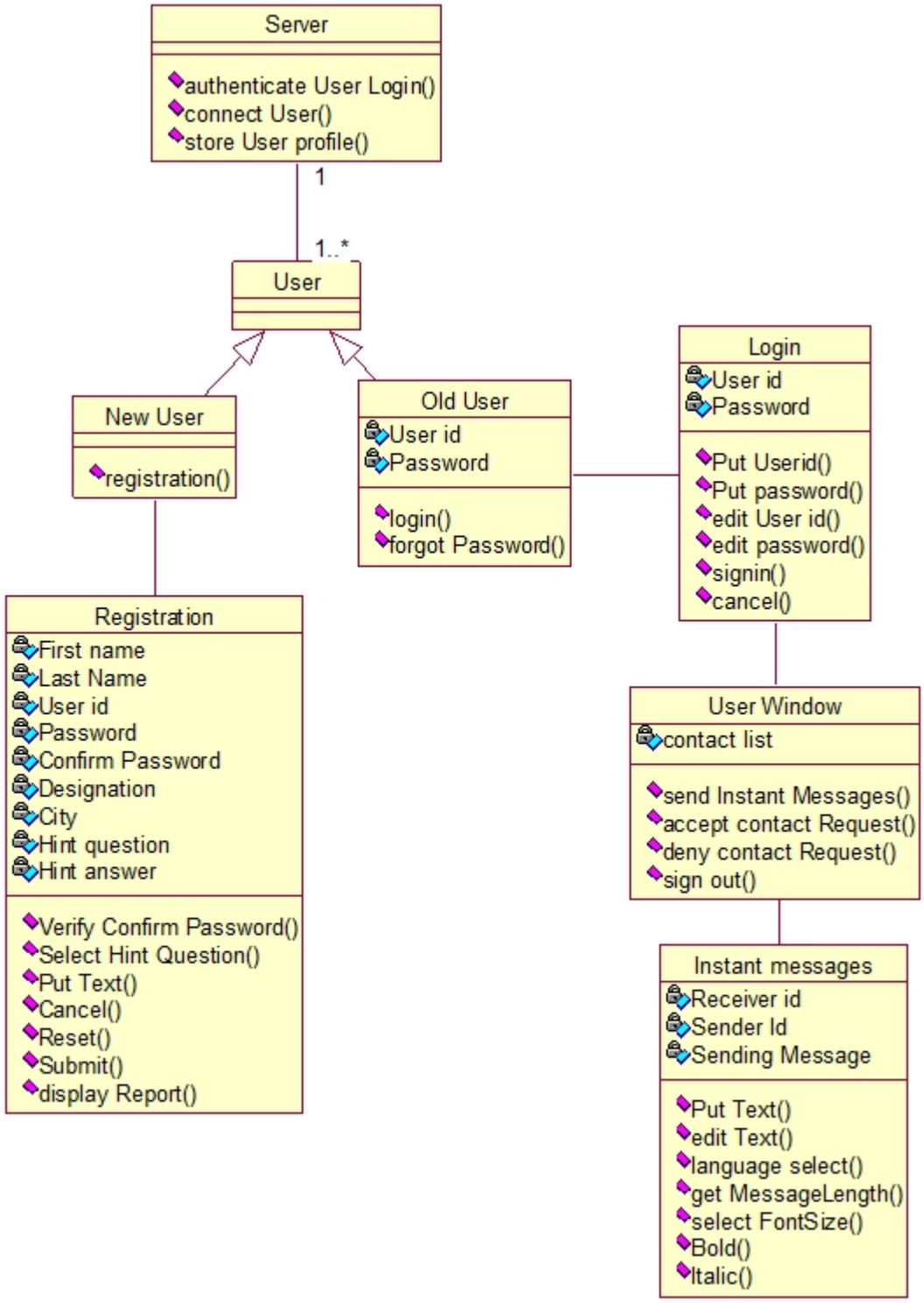
There are options for users to change password. User can view the list of available users online and offline and perform chat operations.

Chatting Module:

Under chatting module user can view a list of available users with online and offline status. He can select the user and send request after acceptance he can chat with the user.

User will have setting options for changing font and color. User can enter a message and use the send button to send a message which is displayed under the upper window.

User can chat with multiple users at a time he can view list previous chats and delete messages.



Features of Java:

- **Simple: Java is a language which is based on Object Oriented Methodology, so it is very easy to learn and can be used effectively.**
- **Robust: – Java Programs are said to be robust because they will take care of memory management and will never crash under any circumstances.**
- **Secure: – Even though Java is developed using Object Oriented Principles, it eliminated the Pointers Concept. So it is not possible to access memory directly, that's why Java is said to be Secure and is applicable for Internet, for that Applet is designed which can be understandable by the browsers.**
- **Portable: – Java Programs are Portable that those can be run under any kind of environment irrespective of the hardware used. This is known as platform independent.**
- **Compiled & Interpreted: – Unlike other Programming languages, Java code is both Compiled and Interpreted. The output after compilation is 'Byte Code' which is interpreted to produce output. This Byte Code is a new evolution, which makes Java a Platform Independent Language.**

Since the File Transfer and Chat system is built using Objected Oriented Paradigm, the requirements of the clients are shown as Use cases and Scenarios. The detailed information for this can be found in the project report

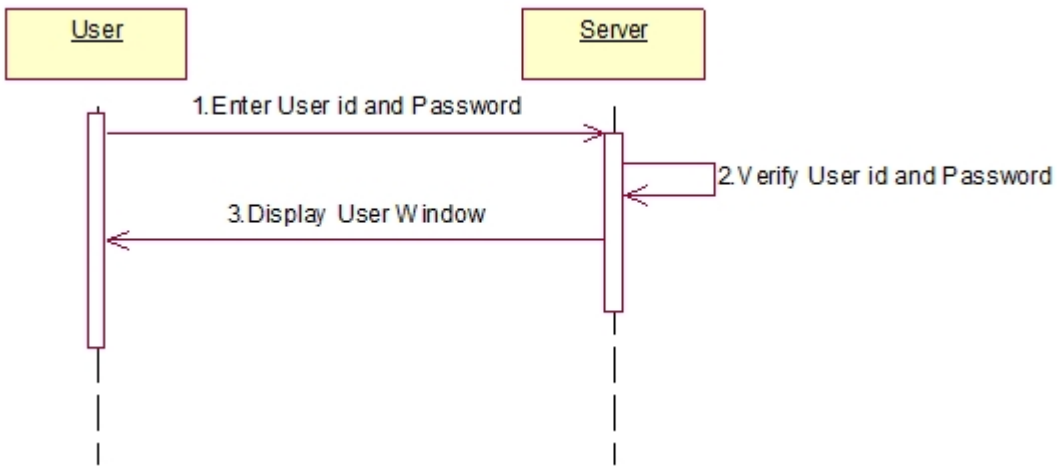
System Requirements:

1. Functional Requirements:

- **It is only possible to connect to the server whenever it is under running. If the server is running, then any number of clients can get connected. Once a client connected to the server, it is not possible to connect again to the server.**

- **In order to send a file, the Source & Destination paths must be entered correctly. If the Source path is entered wrong or not entered, then an error should be displayed. If the Destination path is wrong then the client has to be intimated by the server that the path doesn't exist in the remote client.**

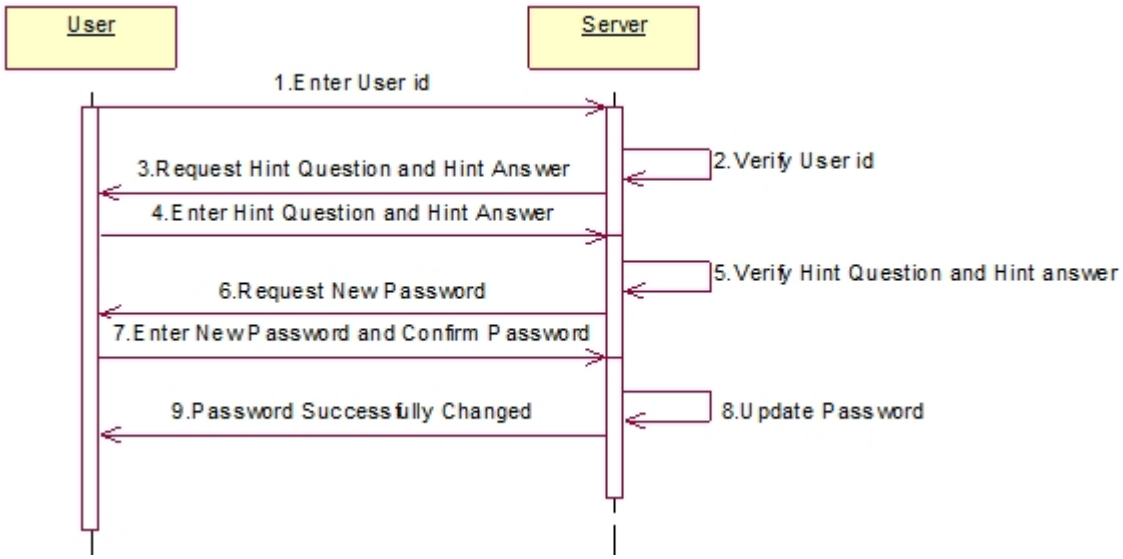
- **To disconnect a single client or multiple clients, choose the clients and disconnect. After disconnecting, observe whether the clients are in the connected clients list or not.**



Performance Requirements:

- **The Performance of the Server is not constant. It is depend upon the network constraints, the hardware configuration of the Server. If less than 50 clients are connected to the Server then the burden on server is less. If more than 50 clients are connected then the burden will increase.**

- **The remedy to the problem is simply disconnecting some of the clients in passive state. The response time will also be considered to assess performance, which is also depending upon the network constraints, and the hardware configuration of Server.**

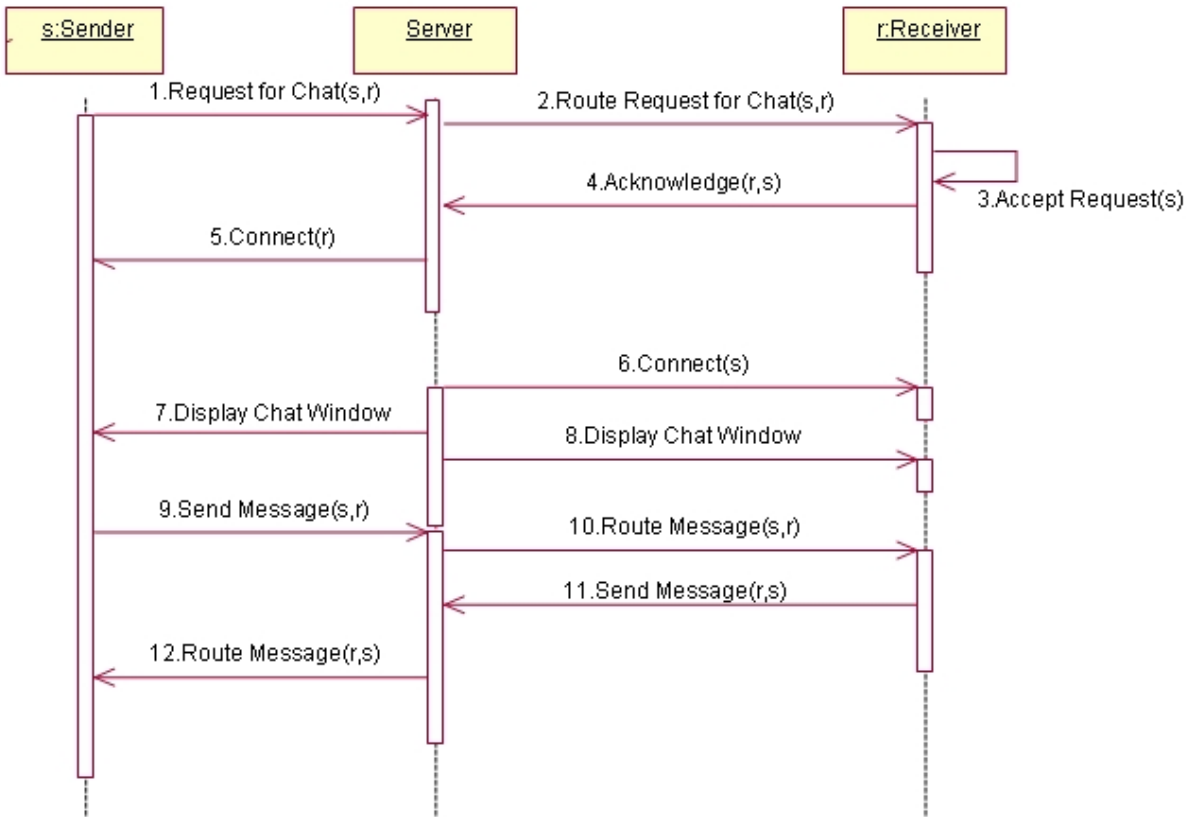


Server-side Hardware Requirements:

- **Console: –** **Mouse, Monitor, 104keys Keyboard**
- **Processor: –** **Intel Pentium III processor, 1.1 GHz speed.**
- **Memory: –** **128MB RAM, 20GB Hard Disk.**
- **N/w Components: –** **Network Adapter, RJ-45 Connector, HUB.**

Client-side Hardware Requirements:

- **Console: –** **Mouse, Monitor, 104keys Keyboard**
- **Processor: –** **Intel Pentium III processor.**
- **Memory: –** **64 MB RAM, 20GB Hard Disk.**
- **N/w Components: –** **Network Adapter, RJ-45 Connector, UTP Cable, HUB.**

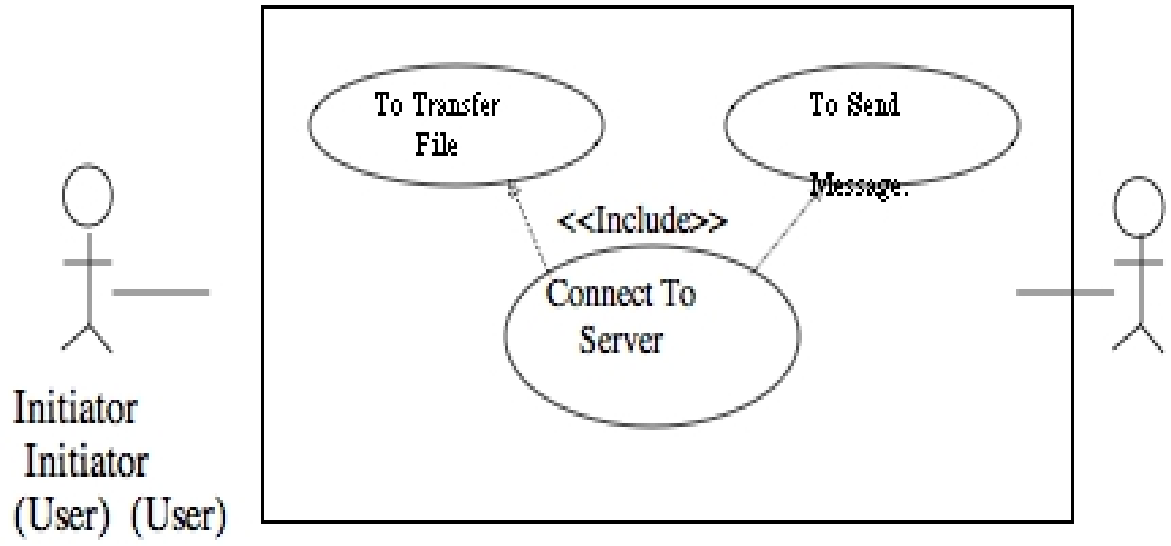


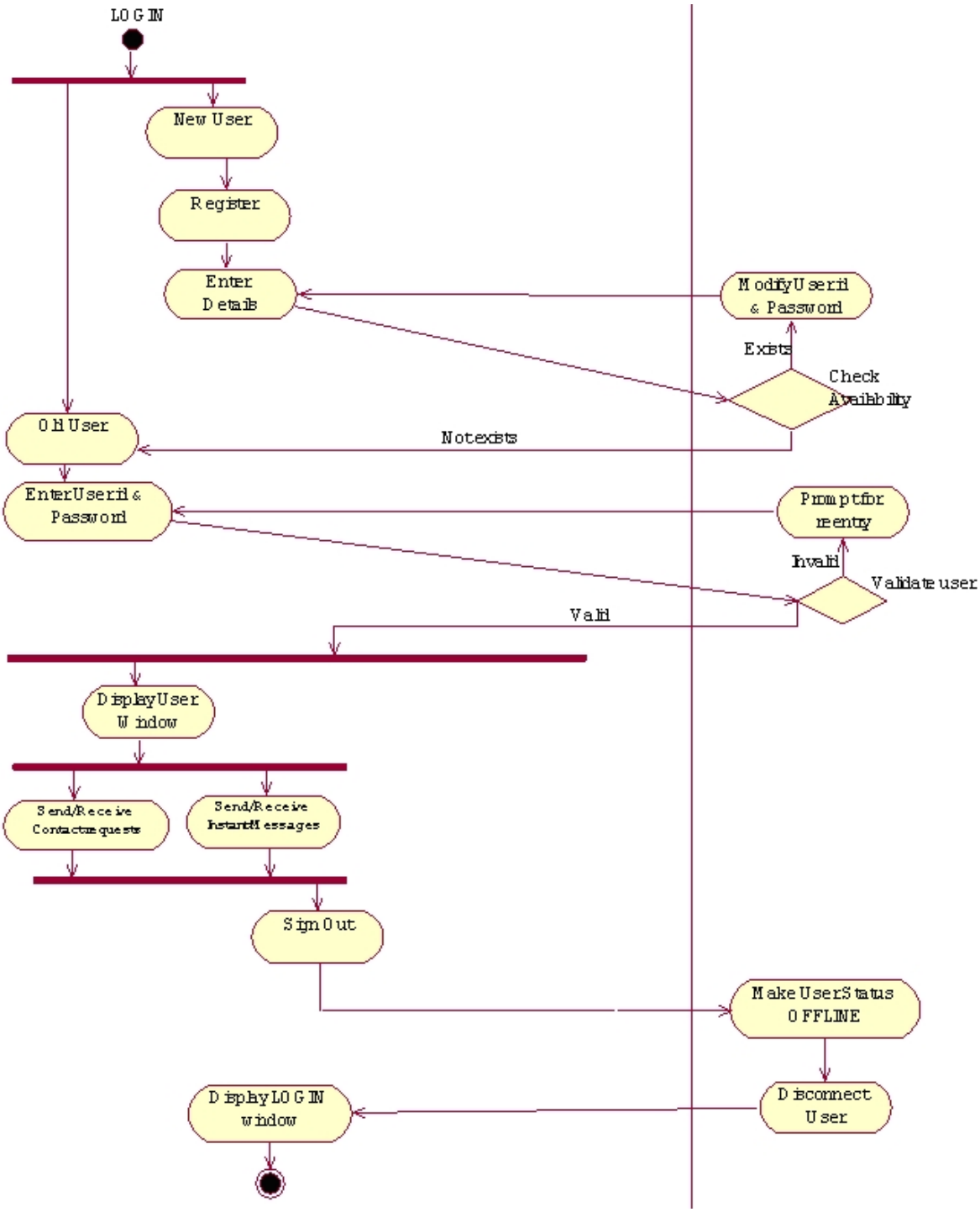
Server-side Software Requirements:

- **Operating System: – MS-DOS, Windows 9x/XP (or) Windows NT.**
- **System Software : – JDK 1.2.2 (or) JDK 1.3 (or) JDK 1.4.**
- **Application Software: – Ftpser.class**

Client-side Software Requirements:

- **Operating System: – MS-DOS, Windows 9x.**
- **Database: – Oracle.**
- **System Software : – JDK 1.2.2 (or) JDK 1.3 (or) JDK 1.4.**
- **Application Software: – ActiveForm.class, ChatForm.class, Sendall.class, Killall.class, Form.class, Filter.class, Chat.class**




LOGIN
New User
Register
Enter Details
Modify Userid & Password
Exists
Check Availability
Old User
Notexists
EnterUserid & Password
Prompt for reentry
Invalid
Validate user
Valid
DisplayUser Window
Send/Receive Contactrequests
Send/Receive InstantMessages
Sign Out
MakeUserStatus OFFLINE
Disconnect User
DisplayLOGIN window

Code Procedure

1). Open Source Code folder and open ChatC.java program and change IPAddress(192.168.0.5) in main method.

2). Here by we proceed to go to Oracle SQLPLUS with username Arj and password tiger

then create table with below syntax

"create table reg(uname varchar2(20),password varchar2(20),fname varchar2(20),lname varchar2(20),design varchar2(20),city varchar2(20),hintQues varchar2(50),hintans varchar2(30))"

3). for creating DSN to connect the Oracle database:

1.ControlPanel>Performance And Maintenance>Administrative tools>Data Sources>UserDSN>

2. Click Add to add dsn

3. select *Oracle Driver* option

4. Enter Data Source name as "chat" and username Arj (stay consistent with the user name as it functions as the variable id, in case of my software, here ARJ is acronym for aparajeet banerjee essentially, but can vary from user to, not that it will matter in encapsulation)

5. then click ok.

6. chat DSN was created succesfully.

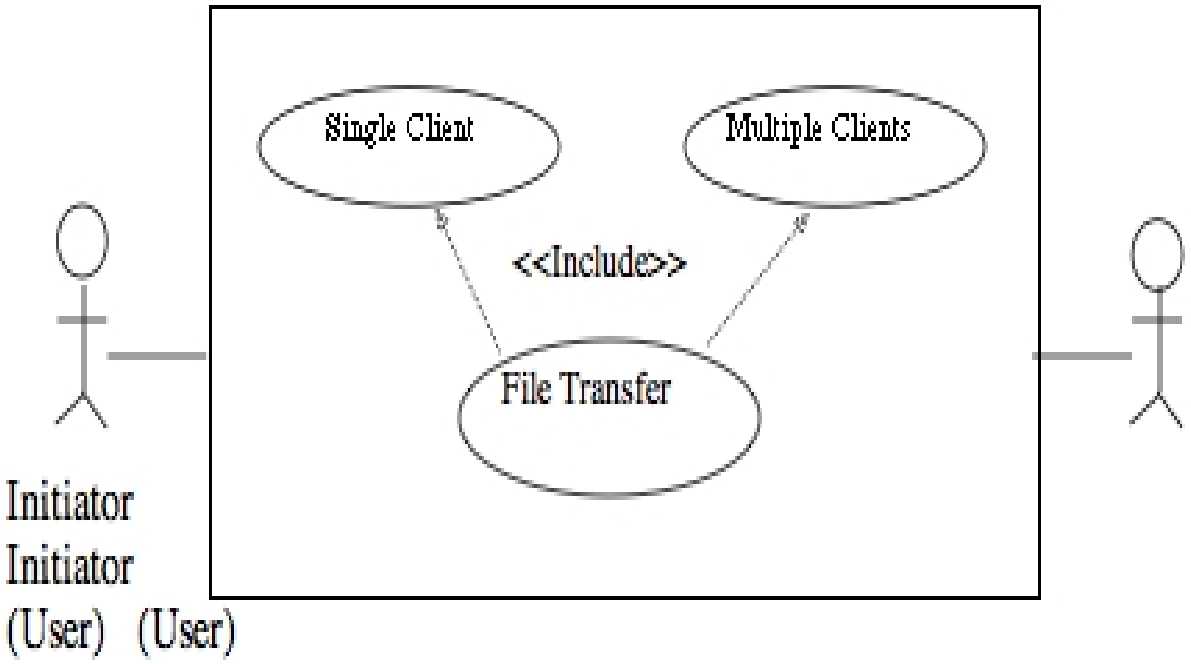
4). install java and set the classpath to the java bin folder

4). from source code folder compile the files as "javac ChatC.java" and "javac ChatS.java"

5). Run the ChatS file as "java ChatS" from another command prompt run the client application as "java ChatS"

6. <u>Client-side Software Requirements</u>:

- **Operating System: – MS-DOS, Windows 9x.**
- **Database: – Oracle.**
- **System Software : – JDK 1.2.2 (or) JDK 1.3 (or) JDK 1.4.**
- **Application Software: – ActiveForm.class, ChatForm.class, Sendall.class, Killall.class, Form.class, Filter.class, Chat.class**



Bibliography

[1], Arpita Bhatacharyajee; FTP vs SFTP – Easy Methods for File Transfer

[2], Introduction to WebRTC by Infinite Skills

[3], proceedings

[4] Galgotia Library and Faculties, GBU Library

[5], shown in References below.

Acknowledgments

The information proceeding is based on various sites, such as udemy and boobs written from real life work issues existing from current system and softwares, and the remainder is seen for by having conversed with faculties who expertised in the subject in past.

References

[1] https://www.udemy.com/tutorial/step-by-step-html-and-css-for-absolute-beginners/understanding-ftp-file-transfer-protocol/

[2] https://www.udemy.com/course/java-network-programming/

[3] https://www.udemy.com/course/zoom-course/?referralCode=D04BF39A3125813502F4

[4] https://www.udemy.com/course/introduction-to-webrtc/

[5] https://www.udemy.com/blog/ftp-vs-sftp/