CS 401-01

Summer 2022

Software Engineering

Group 7: Communication Systems

Design Specification Document

Authors: Ayumi Toki

Emanuel Baca

Revision History

Date	Revision	Description	Author
7/11/2022	1.0	Rough Draft	Group 7
7/13/2022	2.0	Filling in Sections 2, 3 and 4	Group 7
7/18/2022	3.0	Finalizing and spell checking	Group 7

Table of Contents

1.0 Introduction

1.1 Goals and Objectives

This document describes the important aspects of the implementation of the #create a group, #send a chat and #view log file.

1.2 Statement of Scope

Decisions in this document are made based on the following priorities in the following order of importance: Maintainability, Usability, Portability, Efficiency

1.3 Software Contents

Chat information will be maintained in the client object. The various commands will update and query the data using writeToLog and retrieveLog.

1.4 Major Constraints

Issue 1: Where should we store the information regarding the establishment of the chat log?

Option 1.1: Store the information in the Client object logging into the system.

Option 1.2: Store the information on the Server object and retrieve chats upon a successful connection to the server.

Decision: Use option 1.2 as we do not want the client code to hold any data.

Issue 2: Where should the “Employee Records” be stored, such as a text file or a hard coded in as list, array, or a queue?

Option 2.1: Store Employee Records in a list, this will mean that the data will be in the source code.

Option 2.2: Store Employee Records in a text file and read them into a data structure so we can have records viewable without running the code.

Decision: Use option 2.1 that way we do not need to maintain the text file for Employee Records. The server is always on and listening so this should not be a problem.

2.0 Data Design

2.1 Client side

- The seven new commands will be accepted by the **clientInput** and passed to the server.
- Responses from the server will be displayed on the UI, this way the Client is able to understand what buttons they press and the result they get from the server.

2.2 Server side

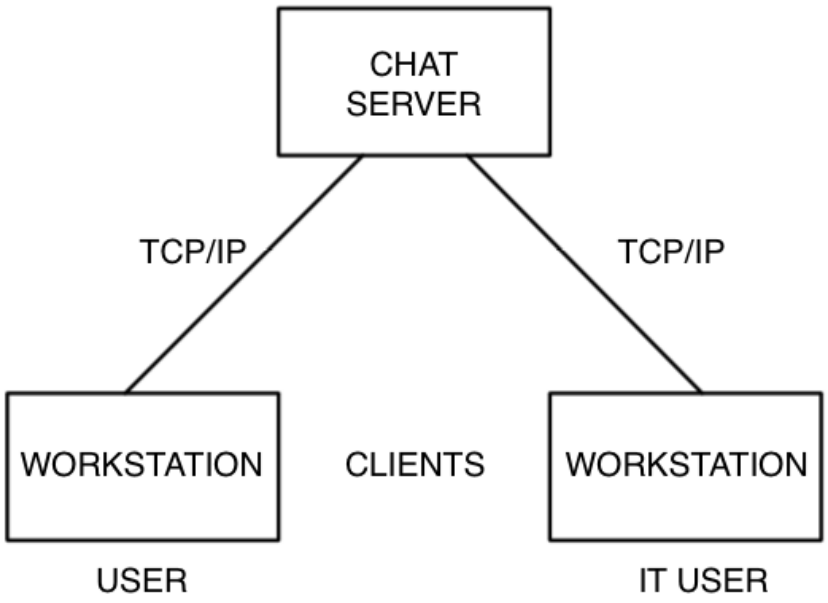
- Method **clientInput** will interpret what command was passed through to the server from the client, then it will be dealt with either by sending a message to user(s), creating a new group, logging off the system, returning client chats and logging all client chat history.
- The Employee Records will be kept in **empRecords** which is a **list of Type User** in order to have all attributes: firstName, lastName, empID, password, status, role.
- Method **clientMessage** will interpret the Message object being passed through by the client that allows us to log and send the message to corresponding receivers
- Method **checkEmployeeStatus** will access our empRecords list and see if the user has logged into the server or is offline. This will determine the message type whether it will be Asynchronous or Synchronous.
- Method **newClientGroupChat** will return the client the employees of desired chat along with their statuses(Online or Offline).
- Method **returnClientChats** will return all chat history of the client based on the chat history log file, this is run upon the client successfully logging into the system.
- Method **ClientLoggingOff** will set the Client status to offline and log all of the chat history in the our chat log file.

3.0 Architectural and Component-level Design

3.1 Program Structure

The communication system is structured for clients to message each other. The structure is different between a regular client and an IT client. The client must connect to the server with credentials before having access to chats and other actions. When the client is no longer an employee of the organization, the IT client will update the information to the server and terminate the terminated employee's credentials.

3.1.1 Architecture diagram



3.2 Description of Client

3.2.1 Client processing narrative.

The client is an employee. The server holds information of the employee, such as first name, last name, login id, password, and status.

The Client passes the login id and password to the server. The server will send a message back to the client if the credentials are valid or invalid.

3.2.2 Client interface description

After the client opens the program, the client will see the login screen. The client put in their login id, into the user input box. And password into the password box.

After successful login, the client will see options.

The User client will see buttons for sending a message, view messages, create group, leave group, and logout.

For IT clients, they will see the following buttons: send message, view messages, create group, leave group, logout, create new employee, delete employee, view chat log file.

3.2.3 Client processing details

The server will take in the client credentials and compare the client with the stored information.

If the client's credentials are valid, then they are an active employee. Therefore, the client will be able to access the communication system. If the client's credentials do not exist on the server, then the client is a terminated employee or not an employee and cannot access the system.

3.3 Description of Server

The Server waits to get the login id and password. After that, the server gets the message. The server sends the logged chat to the log file.

3.4 Software Interface Description

3.4.1 External Interface

This external interface will provide a login screen and later the action of the client.

The system must provide the interface where clients can send messages, view messages, create groups, leave groups, and log out. When the client views personal messages the maximum number of users in the chat is two.

When the client views groups messages, the client will see messages from themselves and at least two other users in the group chat. The client will see the button to leave the group.

After the client uses the log-out button, they no longer have access to the interface.

3.4.2 Internal Interfaces

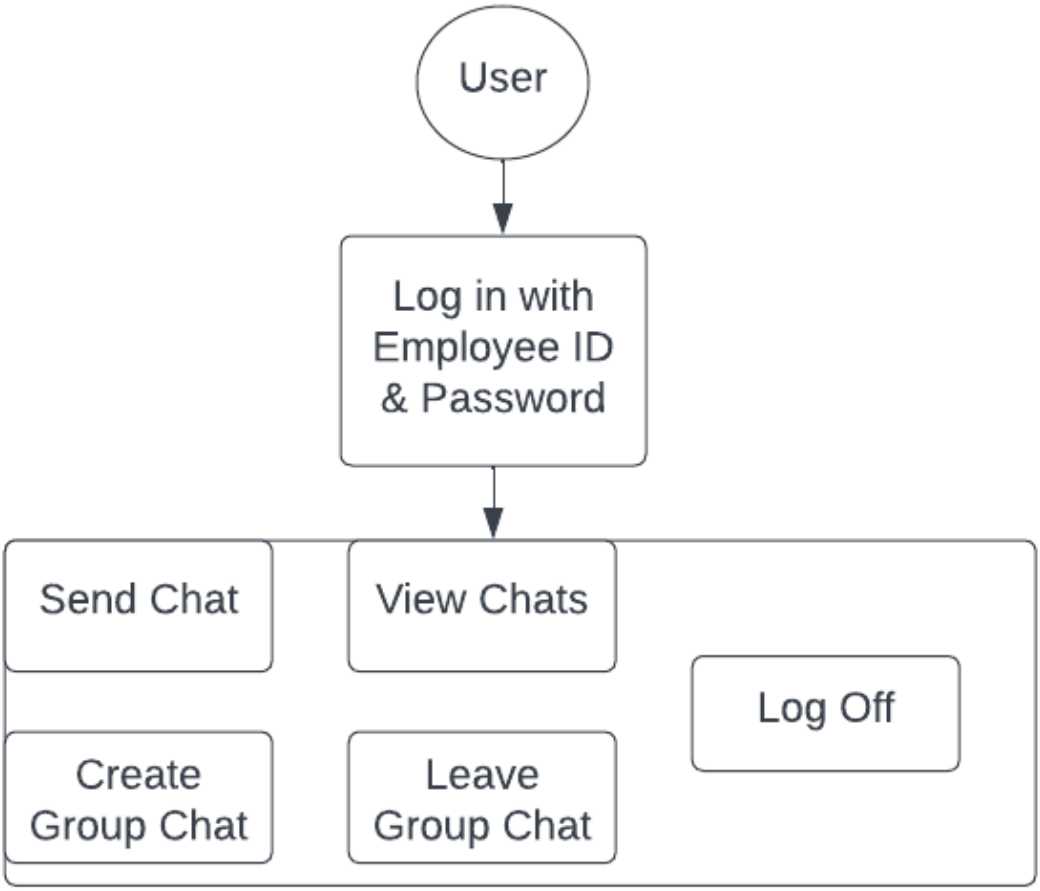
The server holds the chat log and information of each employee.

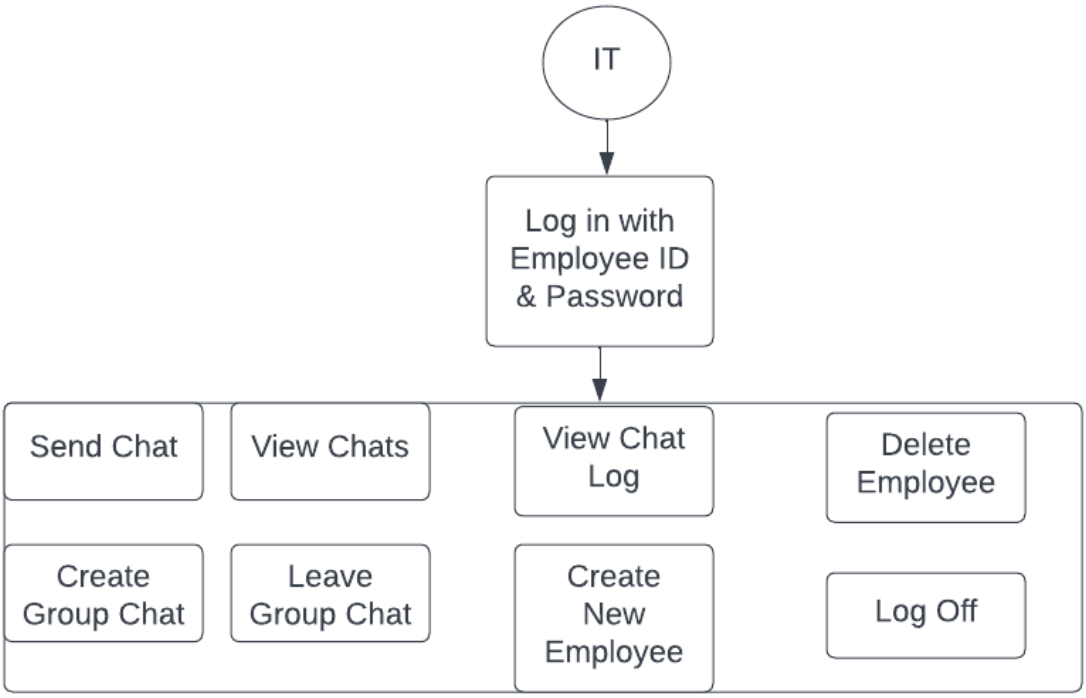
The chat log will be saved in a text file on the server side. After new group chats are created or old ones removed, the file will be overwritten with newer updates.

The system must process the chat log where conversations between employees are held. The IT client has access to the chat log.

3.4.3 Human Interfaces. The type of user interface is Graphical user interface (GUI).

4.0 User Interface Design





- The Client User interface will differ depending on the User or IT role of the Client logging in, both are displayed above along with what the Client Interface would appear on their end.

5.0 Restrictions, Limitations, and constraints

5.1 Restrictions

5.1.1 Non-IT users cannot access the chat log. And cannot create new employees.

5.2 Limitations

5.2.1 Non-IT clients can send messages, view personal messages, view group messages, create group chat, and leave group chat.

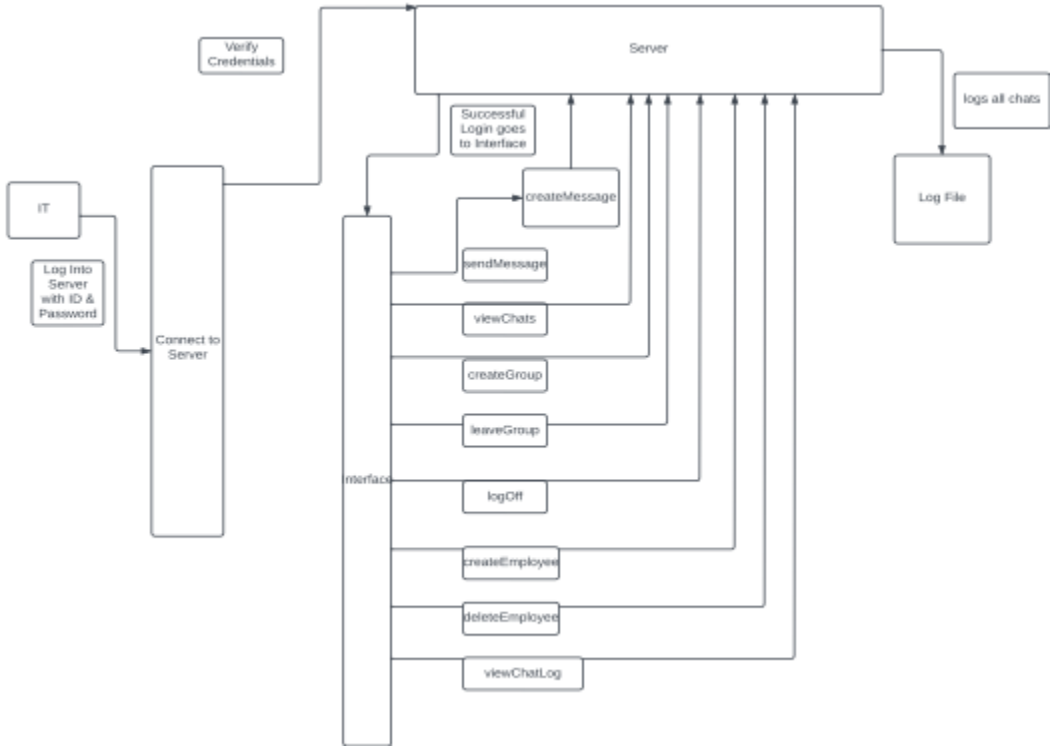
6.0 Testing Issues

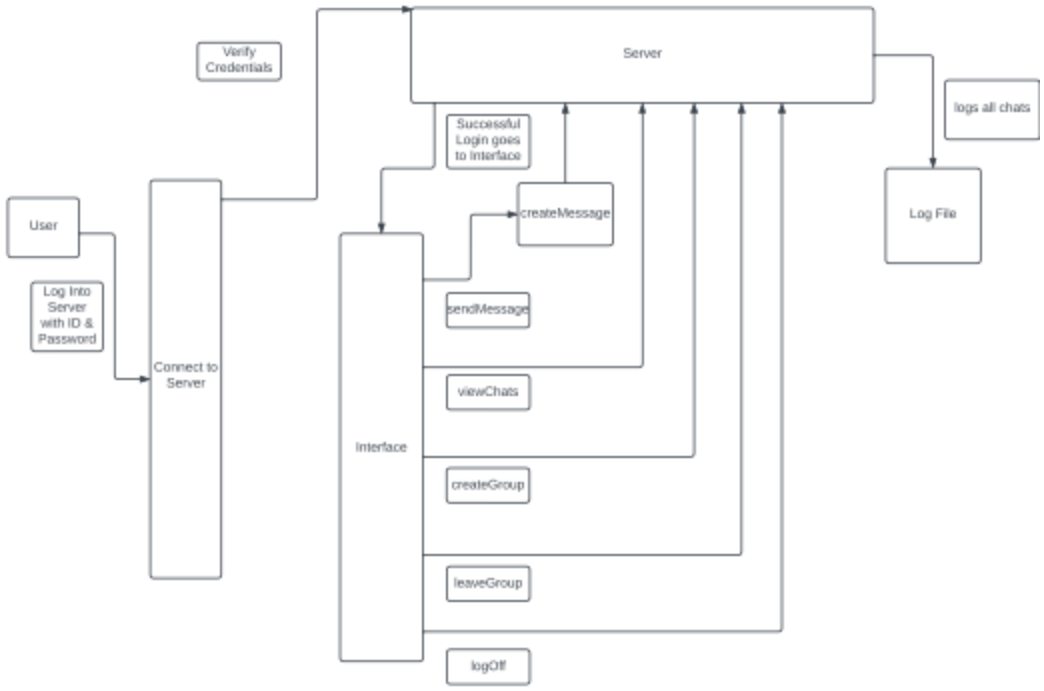
6.1 Testing connectivity between Client and Server
6.2 Unable to test the maximum size of Chat Log file and Employee Records

7.0 Appendices

N/A

8.0 UML diagrams




Server
Verify Credentials
logs all chats
Successful Login goes to interface
User
createMessage
Log File
Log Into Server with ID & Password
sendMessage
Connect to Server
viewChats
Interface
createGroup
leaveGroup
logOff