

# *Software Requirements Specification*

## Revision History

Date	Revision	Description	Author
02/14/22	1.0	Initial Version	Daniel Morales
<u>02/21/22</u>	1.1	Added requirements from meeting <u>02/21/22</u>	Harrison Fisher

# Table of Contents

<b>1. PURPOSE</b>	<b>4</b>
1.1. SCOPE	4
1.2. DEFINITIONS, ACRONYMS, ABBREVIATIONS	4
1.3. REFERENCES	4
1.4. OVERVIEW	11
<b>2. OVERALL DESCRIPTION</b>	<b>12</b>
2.1. PRODUCT PERSPECTIVE	12
2.2. PRODUCT ARCHITECTURE	12
2.3. PRODUCT FUNCTIONALITY/FEATURES	12
2.4. CONSTRAINTS	12
2.5. ASSUMPTIONS AND DEPENDENCIES	12
<b>3. SPECIFIC REQUIREMENTS</b>	<b>13</b>
3.1. FUNCTIONAL REQUIREMENTS	13
3.2. EXTERNAL INTERFACE REQUIREMENTS	13
3.3. INTERNAL INTERFACE REQUIREMENTS	13
<b>4. NON-FUNCTIONAL REQUIREMENTS</b>	<b>14</b>
4.1. SECURITY AND PRIVACY REQUIREMENTS	14
4.2. ENVIRONMENTAL REQUIREMENTS	14
4.3. Performance Requirements	14

# 1. Purpose

This document outlines the requirements for the Chat System (CH).

## 1.1. Scope

Use ip address and port number

UI for server is diff than client

Add names and etc from UI for server for admin and IT

Leveled users (employee, IT, employer, admin, etc)

Separate chat log files (either by room or user) to keep past conversations

Every message should be time stamped in the GUI and the Old past logs

## 1.2. Definitions, Acronyms, Abbreviations

None.

## 1.3. References

### Use Case Specification

**Use Case ID:** 100

**Use Case Name:** Login account

**Relevant Requirements:** srs\_document.docx.

**Primary Actor:** Users

**Pre-conditions:** User attempts to login an account

**Post-conditions:** User has logged in, the UI displayed is appropriate for the type of user, chat history and contacts are available.

#### Basic Flow or Main Scenario:

1. User enters a username.
2. User enters a password.
3. Username and password are validated.
4. Type of user is determined.
5. Chat history and contacts are available.
6. User UI is displayed.

#### Extensions or Alternate Flows:

- 3a. Username or password are invalid.
- 4a. User is IT.
- 4b. User is not IT.
- 6a. IT UI is displayed, with ability to view chat logs, create and delete user.
- 6b. non-IT UI is displayed.

#### Exceptions:

1. Invalid username or password.
2. No chat history.

**Related Use Cases:** Send message, Get chatlog, Get conversation, Create User, Delete User.

**Use Case ID:** 101

**Use Case Name:** Send Message

**Relevant Requirements:** srs\_document.docx.

**Primary Actor:** Users

**Pre-conditions:** Sender requests a message to be delivered.

**Post-conditions:** Receiver(s) gets message from sender. Chat log is updated.

**Basic Flow or Main Scenario:**

1. Sender is logged in.
2. Sender finds a contact to send a message to.
3. Sender requests a message to be delivered.
4. Server accepts request and attempts to deliver to receiver(s).
5. Server updates chat log.
6. Receiver is logged in.
7. Receiver gets a message.

**Extensions or Alternate Flows:**

- 2a. Contact cannot be found.
- 4a. Message attempted to be sent to invalid receiver.
- 4b. There is one receiver.
- 4c. There are multiple receivers.
- 6a. Receiver is not logged in, server waits...

**Exceptions:**

1. Receiver does not exist.
2. Message sent is not text.
3. Receiver is not logged in.

**Related Use Cases:** Get conversation, Login account

**Use Case ID:** 102

**Use Case Name:** Get conversation

**Relevant Requirements:** srs\_document.docx.

**Primary Actor:** Users

**Pre-conditions:** User requests to view conversation

**Post-conditions:** Conversation results are displayed as well as the involved receiver(s).

**Basic Flow or Main Scenario:**

1. User is logged in.
2. User requests to view a conversation.
3. Conversation history and receiver(s) are displayed.

**Extensions or Alternate Flows:**

**Exceptions:**

1. Conversation history is empty.

**Related Use Cases:** Send message, Login Account

**Use Case ID:** 200

**Use Case Name:** Get Chatlog

**Relevant Requirements:** srs\_document.docx.

**Primary Actor:** IT Users

**Pre-conditions:** IT user requests to view chat logs of employees.

**Post-conditions:** It gets a chat log for employees.

**Basic Flow or Main Scenario:**

1. IT user logs in.
2. IT user requests to view the chat logs of employees.
3. Server attempts to find employees' chat logs.
4. Server sends back a chat log of requested employees.
5. Chatlog results are displayed.

**Extensions or Alternate Flows:**

- 3a. Employees cannot be found.
- 4a. Server sends back a message of an error.
- 5a. Error results are displayed.

**Exceptions:**

1. Chatlog for employees does not exist.
2. Employee does not exist.

**Related Use Cases:** Login Account

**Use Case ID:** 201

**Use Case Name:** Create user

**Relevant Requirements:** srs\_document.docx.

**Primary Actor:** IT user

**Pre-conditions:** IT has entered necessary information to create a user account.

**Post-conditions:** A user account is created

**Basic Flow or Main Scenario:**

1. IT is logged in.
2. IT chooses to create a new user option.
3. IT is prompted for information to create a user.
4. IT enters information to create the new user.
5. IT requests for the new user to be created.
6. Server stores information for the new user.
7. Server displays that user has been created.

**Extensions or Alternate Flows:**

- 5a. Not all necessary information is provided.
- 5b. Invalid information is provided.
- 6a. The user already exists, and returns an error.
- 7a. Display error message.

**Exceptions:**

1. The user already exists.
2. Invalid password for new user.
3. Invalid username for new user.

**Related Use Cases:** Login, Delete User

**Use Case ID:** 202

**Use Case Name:** Delete User

**Relevant Requirements:** srs\_document.docx.

**Primary Actor:** IT User

**Pre-conditions:** It has entered id of user to delete

**Post-conditions:** User is deleted

**Basic Flow or Main Scenario:**

1. IT is logged in.
2. IT chooses the delete-user option.
3. IT is prompted for information to delete a user.
4. IT enters information to delete the new user.
5. IT requests for the new user to be deleted.
6. Server finds and removes a user.
7. Server displays that the user has been deleted.

**Extensions or Alternate Flows:**

- 5a. Not all necessary information is provided.
- 5b. Invalid information is provided.
- 6a. Server cannot find user, return error.
- 7a. Display error message.

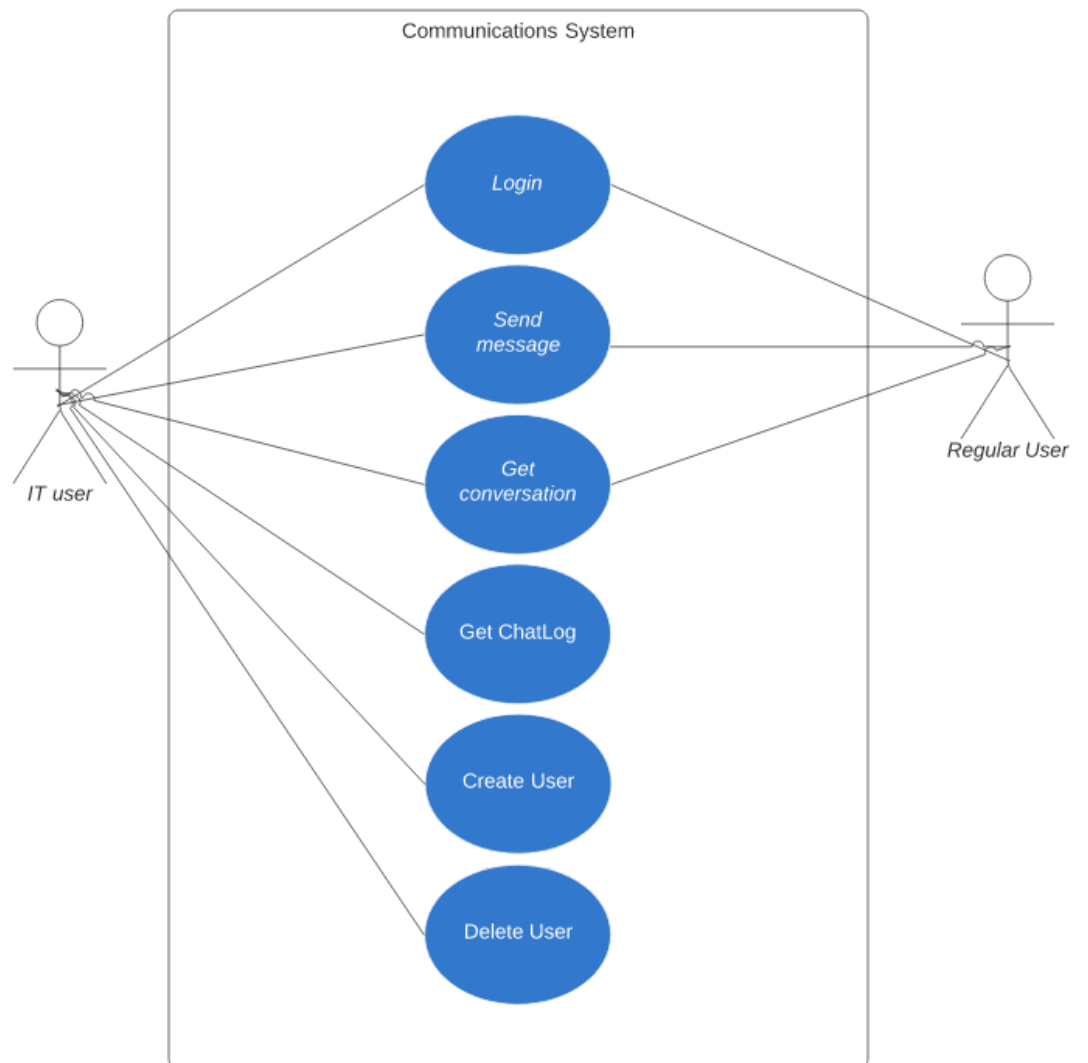
**Exceptions:**

1. User does not exist

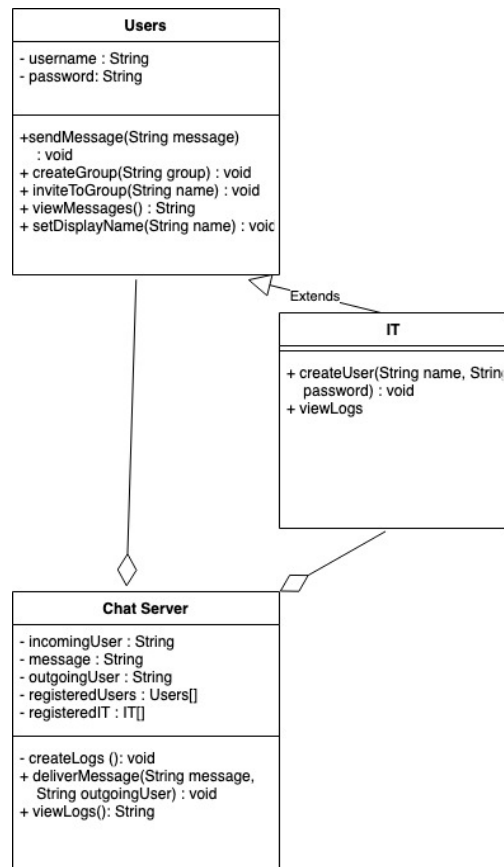
**Related Use Cases:** Login, Create User.



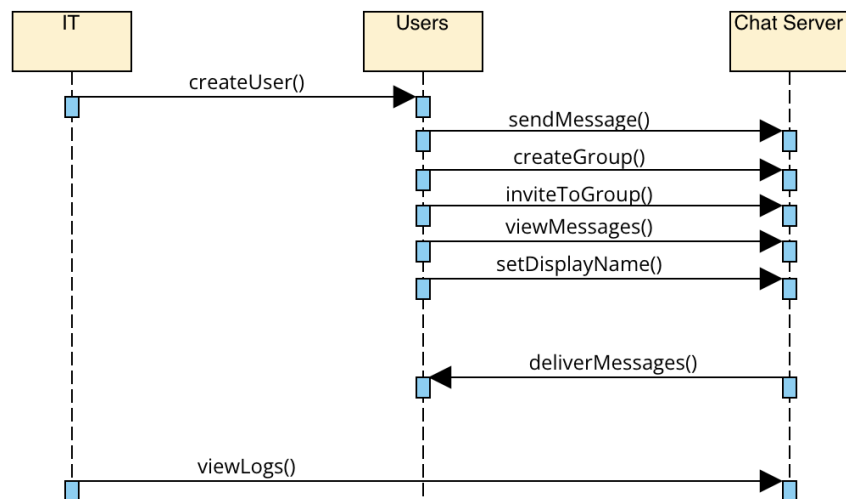
## Use Case Diagram



## Class Diagrams



## Sequence Diagrams



#### **1.4. Overview**

The Chat System (CH), is designed for corporate employees to communicate with each other over various facilities.

## **2. Overall Description**

### **2.1. Product Perspective**

### **2.2. Product Architecture**

The system will be organized into 3 major modules: the server module, the user module, and the IT module.

### **2.3. Product Functionality/Features**

The high-level features of the system are as follows (see section 3 of this document for more detailed requirements that address these features):

### **2.4. Constraints**

Users may not access it through a web browser.

Only available through company use.

### **2.5. Assumptions and Dependencies**

There are thousands of employees.

The company internet is fast and reliable.

## 3. Specific Requirements

### 3.1. Functional Requirements

#### 3.1.1. Common Requirements:

- 3.1.1.1 All communication must be logged
- 3.1.1.2 Both private and group communication
- 3.1.1.3 If using library code keep it minimal, and keep things simple in general

#### 3.1.2. Server Module Requirements:

- 3.1.2.1 Must be able to handle multiple client requests
- 3.1.2.2 Must direct messages from client to another client
- 3.1.2.3 Save messages for when client when offline
- 3.1.2.4 No data bases

#### 3.1.3. User Module Requirements:

- 3.1.3.1 Different UI for users
- 3.1.3.2 Username and password
- 3.1.3.3 A gui for the user

#### 3.1.4. IT Module Requirements:

- 3.1.4.1 Different UI for IT
- 3.1.4.2 Username and password
- 3.1.4.3 Can create and delete users

### 3.2. External Interface Requirements

- 3.2.1 The system must provide an interface to users so that they can log into a profile.
- 3.2.2 The system must provide an interface for users to message other users
- 3.2.3 The system must provide an interface for users to view messages with others users
- 3.2.4. The system must provide an interface for IT to create other users

### 3.3. Internal Interface Requirements

- 3.3.1 The system must process client to server communication.
- 3.3.2 The system must save logs of messages between clients

## **4. Non-Functional Requirements**

### **4.1. Security and Privacy Requirements**

- 4.1.1 Only users can see their communications with others
- 4.1.2 Users must be required to log in into their profiles
- 4.1.3 No encryption, store user info in plain text
- 4.1.4 Chat logs are done big brother style so users aren't aware of chat logs (only admin staff can see it)
- 4.1.5 Users can't see other peoples chat logs

### **4.2. Environmental Requirements**

- 4.2.1 System must be java based
- 4.2.2 Deadline at week 05/02/22
- 4.2.3 Deadline for phase 1 at 03/02/22
- 4.2.4 Deadline for phase 2 at 04/03/22
- 4.2.5 Deadline for phase 3 at 05/04/22

### **4.3. Performance Requirements**

- 4.3.1 None