

Company Communications System

Software Requirements Specification

Revision History

[illegible]

Table of Contents

1. PURPOSE	4
1.1. SCOPE	4
1.2. DEFINITIONS, ACRONYMS, ABBREVIATIONS	4
1.3. REFERENCES	4
1.4. OVERVIEW	4
2. OVERALL DESCRIPTION	5
2.1. PRODUCT PERSPECTIVE	5
2.2. PRODUCT ARCHITECTURE	5
2.3. PRODUCT FUNCTIONALITY/FEATURES	5
2.4. CONSTRAINTS	5
2.5. ASSUMPTIONS AND DEPENDENCIES	5
3. SPECIFIC REQUIREMENTS	6
3.1. FUNCTIONAL REQUIREMENTS	6
3.2. EXTERNAL INTERFACE REQUIREMENTS	6
3.3. INTERNAL INTERFACE REQUIREMENTS	7
4. NON-FUNCTIONAL REQUIREMENTS	8
4.1. SECURITY AND PRIVACY REQUIREMENTS	8
4.2. ENVIRONMENTAL REQUIREMENTS	8
4.3. Performance Requirements	8

1. Purpose

This document outlines the requirements for the Company Communications System.

1.1. Scope

This document will catalog the user, system, and hardware requirements for the Company Communications System. It will not, however, document how these requirements will be implemented.

1.2. Definitions, Acronyms, Abbreviations

CCS: Company Communications System

1.3. References

[Use Case Specification Document](#)

[UML Use Case Diagrams Document](#)

[Class Diagrams](#)

[Sequence Diagrams](#)

1.4. Overview

The Company Communications System is designed to allow text communications for a very large company or organization. Due to its purpose, the Company Communications System does not allow anyone to self-register their own accounts. Employees will be furnished with an account by an authorized Administrator.

2. Overall Description

2.1. Product Perspective

The Company Communications System is developed for any large company seeking a robust application for sending and receiving text communications between employees. The System allows for the creation of private direct messages, as well as group messages.

2.2. Product Architecture

The system will be organized into 3 major modules: the User module, the Messaging module, Server module and the Record 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):

- 2.3.1. User module handles log-in, Admin/User rights
- 2.3.2. Messaging module handles sending and receiving messages from Server, group chats/private chats
- 2.3.3. Logging module handles logging of all messages (and actions taken by admins)
- 2.3.4. Server module handles sending messages between users

2.4. Constraints

- 2.4.1. The system is made entirely in Java.
- 2.4.2. No database system. Data will be managed using file input/output
- 2.4.3. The system must operate over TCP/IP.

2.5. Assumptions and Dependencies

- 2.5.1. All communication will be in text format
- 2.5.2. No external databases or third-party libraries will be part of its implementation
- 2.5.3. The CCS must be able to scale to the size of the organization as needed.

3. Specific Requirements

3.1. Functional Requirements

3.1.1. Common Requirements:

- 3.1.1.1. The system should authenticate users based on their ID and PIN.
- 3.1.1.2. The system should differentiate between IT and Employee roles, granting appropriate access rights.
- 3.1.1.3. All user accounts, messages, and logs should be stored persistently in files on the computer.
- 3.1.1.4. The system should provide real-time updates for messaging and notifications without manual refresh.
- 3.1.1.5. The system should have an intuitive interface for both IT and Employee roles.
- 3.1.1.6. The system should support multiple concurrent users and group chats.

3.1.2. User Module Requirements:

- 3.1.2.1. System should know if User is I.T. or Employee (have list of authorized I.T.) and correctly gives extra access upon log-in
- 3.1.2.2. ID and pin must be unique and cannot be change
- 3.1.2.3. Check ID and Pin with data stored in file
- 3.1.2.4. Allow log-in at any time (Logged-in user does not prevent more users from logging in)
- 3.1.2.5. I.T. can create new employee account
- 3.1.2.6. I.T. can view chat history of all employees
- 3.1.2.7. I.T can delete an employee account.

3.1.3. Messaging Module Requirements:

- 3.1.3.1. Enable private and group chat creation
- 3.1.3.2. Allow asynchronous messaging between users
- 3.1.3.3. Synchronous messaging between users.
- 3.1.3.4. Users are notify of new messages
- 3.1.3.5. Automatically update chat without having to manually refresh
- 3.1.3.6. Message sending and receiving are instantaneous
- 3.1.3.7. Have a chat history that display previously sent messages
- 3.1.3.8. There can be multiple group chats with the same Users
- 3.1.3.9. There can only be one chat for a pair of users.
- 3.1.3.10. Group chat members can add more users to the group chat
- 3.1.3.11. Group Chats should be nameable .

3.1.4. Record Module Requirements:

- 3.1.4.1. System will log and archive all messages
- 3.1.4.2. System record message timestamp
- 3.1.4.3. System record message sender's name and id
- 3.1.4.4. Store records in a file stored in the computer.
- 3.1.4.5. System will record when a user joins and leaves the chat with the timestamp.
- 3.1.4.6. Logged messages can not be edited or deleted.

3.2. External Interface Requirements

3.2.1 The System must provide a GUI interface such that a user can access any chats that they may have instantiated. Chats to the user must be presented as options Private chats and Group chats. Then they can access any chats that fall under that category. From here the user has the ability to enter the chat, leave the chat, or delete the chat.

3.2.2 The System's provided GUI interface must also provide a section where messages are presented chronologically so that the user can see the chat history of previously sent messages by other users or themselves. This must also provide the user the ability to enter text and send messages updating the chat history with newly sent messages. This will work for private and group messages.

3.2.4 The System must provide the a GUI interface option for a I.T User to access any account in the system and view that accounts chat logs. The GUI must also provide the I.T User the ability to access all chat logs registered by the server.

3.3. Internal Interface Requirements

- 3.3.1. The User Module must provide a secure interface for other modules to verify user credentials and roles.
- 3.3.2. The Messaging Module should offer an interface for sending messages between users, which can be called by other modules when necessary.
- 3.3.3. The system should use a standardized logging interface.

4. Non-Functional Requirements

4.1. Security and Privacy Requirements

- 4.1.1 No encryption.
- 4.1.2 Only the I.T user should have access to everyone's message log

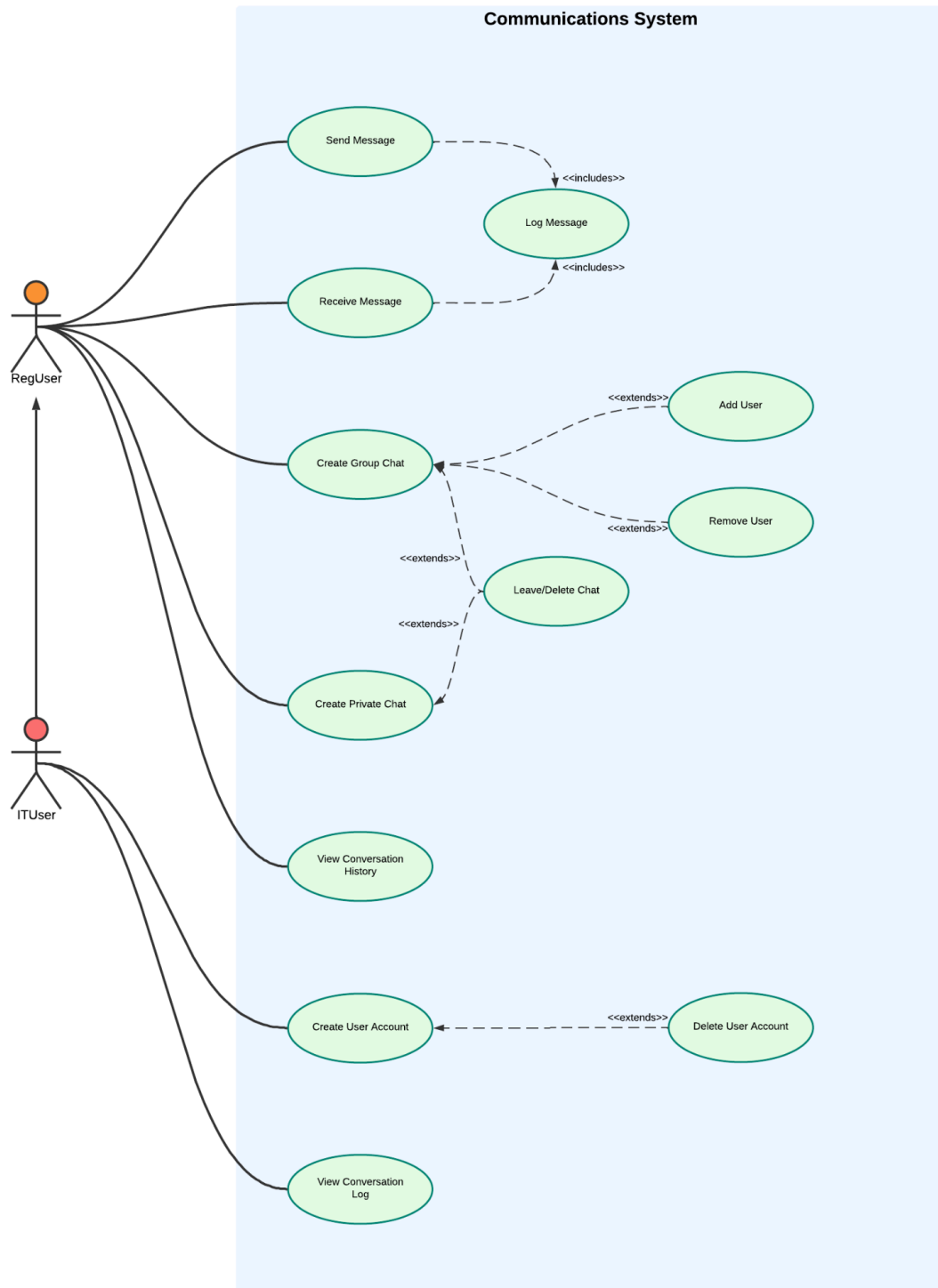
4.2. Environmental Requirements

- 4.2.1 System must have access to the internet, or servers.
- 4.2.2 Any computer should be able to access the application.
- 4.2.3 A useable and functional GUI that users and admins can use, not necessarily something that looks good
- 4.2.4 The system must utilize the TCP/IP protocol for communication between client and server.

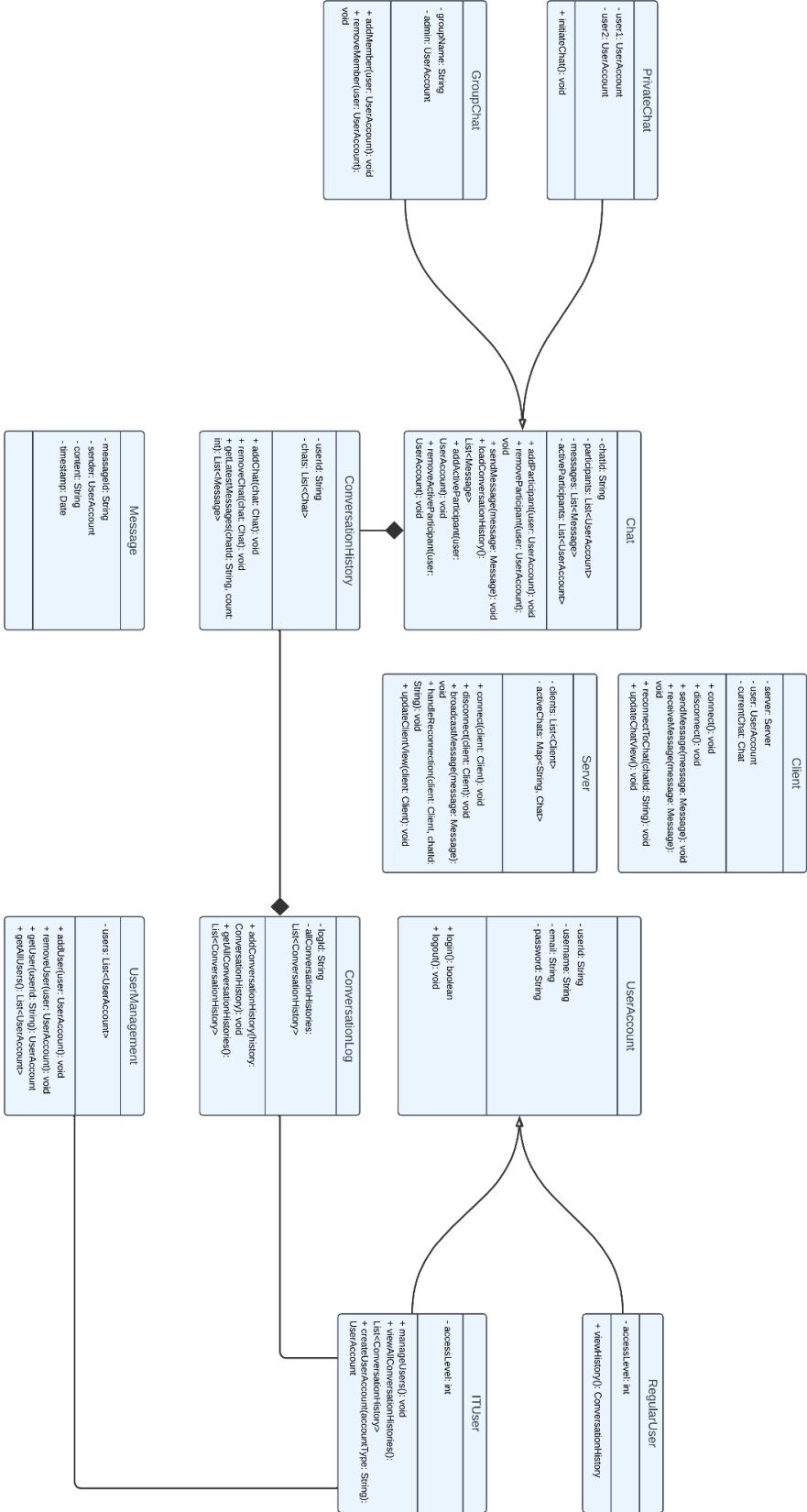
4.3. Performance Requirements

- 4.3.1 Messages should be sent and received in a timely manner, with as little delay as possible
- 4.3.2 The system should handle as many users as needed for the company, with room to expand. OR We want to handle as many users as possible in the system. Relative to the amount of resources and funds available.
- 4.3.3 System should be Reliable as possible with minimum downtime.
- 4.3.4 System will log and archive all messages.
- 4.3.5 Store all user login information and passwords.

UML USE CASE DIAGRAM



Communications System Class Diagram



SEQUENCE DIAGRAMS

