Company Communications System

Software Requirements Specification

Revision History

Date	Revision	Description	Author
09/12/2024	1.0	Document Creation	Kai Abillar
9/24/2024	1.1	Updated Sections 1.0 and 1.4	Rowwel Ponesto
9/27/2024	1.2	Added a few bullet points in 2.4 & 2.5	Kai Abillar
9/30/2024	1.3	Added 2.4.3, 2.1, 2.5.3 Begin 1.2 Definitions, added first entry Formatting	Rowwel Ponesto
10/01/2024	1.4	Added some	Johnny Hoang
10/01/2024	1.5	4.X	John Huynh
10/02/2024	1.6	3.1.3 Requirements added (4) 3.1.2 Requirements added(4) + moved requirements around from non-functional to functional Section 4: Modified Requirements Added Link to: *Use Case Specification Document.* 3.2 External Interface requirements added (3)	Bryan Madrigal
10/02/2024	1.7	3.1.2.5	Rowwel Ponesto
10/02/2024	1.8	Added in the group's diagrams Added interface requirements	Kai Abillar
10/29/2024	1.9	Revamped Modules Section	Kai Abillar
10/31/2024	2.0	Appended Project GitHub link under references	Rowwel Ponesto

Table of Contents

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

Design Document

Gantt Chart

Project GitHub

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. User Authentication and Access

- 3.1.1.1.1. The System should authenticate users based on their ID and PIN
- 3.1.1.1.2. The system should differentiate between IT and Regular users , granting appropriate access rights
- 3.1.1.1.3. Allow log-in at any time (Logged-in user does not prevent more users from logging in)

3.1.1.2. Data Persistence

- 3.1.1.2.1. All user accounts, messages, and logs should be stored persistently in files on the computer
- 3.1.1.2.2. Store records in a file stored in the computer

3.1.1.3. Real-time Operations

- 3.1.1.3.1. The system should provide real-time updates for messaging and notifications without manual refresh
- 3.1.1.3.2. The system should support multiple concurrent users and group chats
- 3.1.1.3.3. Message sending and receiving are instantaneous
- 3.1.1.3.4. Automatically update chat without having to manually refresh

3.1.1.4. Interface

- 3.1.1.4.1. The system should have an intuitive interface for both IT and Regular users.

3.1.2. Communications Module Requirements:

3.1.2.1. Allow asynchronous messaging between users

3.1.2.2. Synchronous messaging between users

3.1.3. User Management Module Requirements:

3.1.3.1. System should know if User Account is IT or Regular (have list of authorized IT) and correctly gives extra access upon log-in

3.1.3.2. ID and pin must be unique and cannot be changed

3.1.3.3. Check ID and pin with data stored in file

3.1.3.4. IT can create new employee account

3.1.3.5. IT can delete an employee account

3.1.4. Chat Module Requirements:

3.1.4.1. Enable private and group chat creation

3.1.4.2. Users are notified of new messages

3.1.4.3. There can be multiple group chats with the same users

3.1.4.4. There can only be one chat for a pair of users

3.1.4.5. Group chat members can add more users to the group chat

3.1.4.6. Group Chats should be nameable

3.1.5. Record Module Requirements:

3.1.5.1. System will log and archive all messages

3.1.5.2. System record message timestamp

3.1.5.3. System record message sender's name and ID

3.1.5.4. System will record when a user joins and leaves the chat with the timestamp

3.1.5.5. Logged messages cannot be edited or deleted

3.1.5.6. IT can view chat history of all employees

3.1.5.7. Have a chat history that displays previously sent messages

3.1.6. UI Module Requirements

3.1.6.1. See Common Interface Requirements (3.1.1.4)

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 password