Coimbatore Institute of Technology

Department of AI and DS



**Multiple Chat User System**

**Software Requirements Specification**

**Version: 1.0d**

Document Id: SEP1-SRS-V 1.0 d-20231021

Author(s): GOKUL, RANJITH, DINESH KUMAR, VEL MURUGAN

Reviewed By:, RANJITH, DINESH KUMAR, VEL MURUGAN

Date: 22-10-2023

**Modification History**

|  |  |  |  |
| --- | --- | --- | --- |
| **Date** | **Modifications** | **Reason** | **Version** |
| 02/10/2023 | Create a draft version of the document. | Adapted from various IEEE standards on software requirements specification | 0.7d |
| 15/10/2023 | Add guidelines and faked examples as italics. | For students to follow | 0.8d |
| 20/10/2023 | Minor amendments | As recommended by reviewer | 0.9d |
| 22/10/2023 | Modify document content for the MCS | Adapted for Assignment 1 of Semester 3 | 1.0d |

Table 1.0 Modification History

**Contents**

[1. Introduction](#_Toc21329102) 4

[2. Product Overview 7](#_Toc21329109)

[3. Functional Requirements 9](#_Toc21329116)

[4. User Interface Requirements 12](#_Toc21329120)

[5. Non-Functional requirements 12](#_Toc21329123)

[6. Operating Environment 13](#_Toc21329130)

[7. Acceptance Criteria 13](#_Toc21329135)

[References 14](#_Toc21329136)

[Glossary 14](#_Toc21329137)

[Index 15](#_Toc21329138)

[Appendix A: External Context Model 16](#_Toc21329140)

[Appendix B: Internal Context Model 16](#_Toc21329141)

[Appendix C: Use Case Analysis Notes 16](#_Toc21329142)

[Appendix D: Development Process 16](#_Toc21329143)

[Appendix E: Contributions 17](#_Toc21329144)

[Appendix F: Meeting Agendas/Minutes 17](#_Toc21329145)

[Appendix G: Problem Investigation Reports 18](#_Toc21329146)

# 1. Introduction

## Document Purpose

This document presents the software requirements, as determined during initial analysis by creating multiple chat user system (MCS) which is more helpful for discussion and forums. Specifically, the document details the functional requirements and constraints for the computer software to be developed by CIT AI firm for the Client.

This document will be subject to formal/informal review by the CIT AI firm development team and the Client, and will form the basis for ongoing development of software, by CIT AI firm, to meet the requirements of the Client.

This document is the main deliverable output from the Requirements Elicitation Activity. It is intended to be a baseline, to supply sufficient requirements information to the Client to provide a foundation for subsequent software assessment and approval. Further, it also provides the development team with a basis for on-going software design.

## About the Project

The aim of this project is to perform the ‘Requirements Elicitation and Specification’ activity. All tasks associated with this activity shall be performed based on the theoretical support provided as part of ‘Software Engineering/Introduction to Software Engineering’ course. It is assumed that other related/required activities involved in the development of the system shall be completed under different project(s).

## Document Scope

The system must ensure secure user authentication methods, allowing users to reset passwords securely. It should support the creation and joining of chat rooms, one-on-one and group messaging, with support for text, multimedia files, and emojis. Real-time message delivery and updates are essential features. Moreover, the system must prioritize security and privacy, encrypting messages, allowing users to block/report inappropriate behavior, and ensuring compliance with data protection regulations for user privacy.

User

using sections 3 to 7 give a ten lines of document scope

ChatGPT

Document Scope:

This document outlines the comprehensive specifications for a Multiple Chat User System, detailing functional, user interface, and non-functional requirements. The system's primary goal is to facilitate seamless communication between users through text, multimedia, and emojis. Users can create, join, and manage chat rooms, engage in one-on-one and group messaging, and access features like message formatting and attachment capabilities. The system prioritizes security by implementing secure user authentication methods, encryption for messages, and user privacy compliance with data protection regulations. Its user-friendly interface encompasses a dashboard displaying discussion groups, profile management options, and notification settings. The system's performance parameters include rapid response times, scalability to handle increasing users and messages, and fault tolerance to ensure uninterrupted service. Accessibility, compatibility across platforms and browsers, and adherence to industry standards further enhance its usability. Additionally, the document details hardware and software requirements, acceptance criteria for user authentication, chat features, and security and privacy aspects, ensuring a comprehensive and robust chat system tailored to user needs.

## Terminology Used

|  |  |
| --- | --- |
| **Term/Acronym** | **Definition/Description** |
| Developers | The team responsible for the complete development of the software system. |
| Process | A process can be defined as a series of interrelated and interdependent activities or steps that are undertaken to achieve a specific goal or produce a specific output |
| Requirement | A condition or capability needed by a user to solve a problem or achieve an objective. |
| Specification | A document that prescribes, in a complete, precise, verifiable manner, the requirements, design, behaviours or other characteristic of a system or system components. |
| User | The person operating and/or using the software system. |
| Use Case | Descriptions of interactions between the system and external entities, outlining specific scenarios of system usage. |

Table 2.0 Definitions, acronyms and abbreviations

## Related Documents

List any and all applicable documents that have been used in the development of this document. This section should ideally be completed towards the end of this project.

## Document Overview

This document, the Software Requirements Specification (SRS), identifies the software requirements for the project System Name.

This document has x major sections and y appendices:

1. **Introduction** (Section 1) provides an overview of the entire SRS document, the project and the product being specified.
2. **Product Overview** (Section 2) describes the product in overview, addressing the hardware and software requirements, user characteristics, constraints, assumptions and dependencies, expressed in natural language.
3. **Functional Requirements** (Section 3) gives Detailed specifications of system functions, including user interactions, message handling, and group management.
4. **User Interface Requirements** (Section 4) Describes the user experience, including registration, dashboard, discussion group features, messaging, and security protocols
5. **Non-Functional Requirements** (Section 5) Specifies performance, reliability, security, usability, compatibility, compliance, and documentation aspects for the chat system
6. **Operating Environment** (Section 6) The system operates on Pentium-200MHz or higher hardware with 128 MB RAM and 20MB free disk space, compatible with Windows 2000 OS. It requires Java v1.3.1 or above and a compatible printer for text files. Data storage methods and external access are specified.
7. **Acceptance Criteria** (Section 7) The system ensures secure user authentication, supports chat features including one-on-one and group messaging, multimedia sharing, and real-time updates. It encrypts messages, enables user blocking/reporting, and complies with data protection regulations.

# Product Overview

## Introduction

This section provides a high-level description of the requirements that have been identified to be essential to the software system to be developed. All the characteristics and limitations of the proposed software system have been provided below.

**Example:**

1. **Slack:** A professional messaging platform with group discussions, file sharing, and integration capabilities.
2. **Discord:** A voice, video, and text communication platform designed for gamers, supporting chat rooms and private messaging.
3. **Microsoft Teams:** Collaboration platform with chat, video meetings, and file sharing, integrated into the Microsoft 365 suite.

## Business Services Supported

The system shall provide software support for the following operations:

1. Registration
2. Statistical Report of client usage
3. Real-time collaboration
4. Project management discussions
5. Customer support chats
6. Online education forums
7. Virtual event discussions
8. Team coordination
9. Community building
10. Technical support chats
11. Sales and marketing discussions
12. Interest-based communities
13. Professional networking
14. Research collaboration
15. Remote team communication
16. Group brainstorming
17. Language learning communities
18. Health and wellness support groups
19. Coding and programming help
20. Content creators collaboration
21. Travel planning discussions
22. Hobbyist forums

## Product Characteristics

The proposed chat system is a robust communication platform facilitating real-time messaging, multimedia sharing, and group collaboration. It ensures secure user authentication, customizable interfaces, and adherence to data protection regulations. The system offers seamless integration with various operating systems and web browsers, including mobile devices, and features comprehensive logging and monitoring capabilities

### User Interface

The system shall provide Graphical User Interfaces.

### Operating Environment

The deployment platform requirements are a PC equipped with a Pentium-200Mhz (equivalent or above) CPU with 128Mb of RAM and 5 Mb of free disk space, running Windows 2000 Operating System. However, the system specified in this document shall also run efficiently on any operating environment (including the hardware and operating system) that supports the Java Virtual Machine.

### Hardware Interfaces

For effective operation the software system will need:

The chat system primarily operates as a web application, accessible through standard computer systems, laptops, tablets, and smartphones. It requires internet connectivity and is compatible with widely used web browsers such as Chrome, Firefox, Safari, and Edge

## User Characteristics

This section shall describe the various types of users that are expected to use this system and what the minimum requirements are in terms of their knowledge and skills.

The users of this system shall be:

* Multimedia Group employee
* Novice Users
* Experienced Users
* Administrators
* Collaborative Teams
* Individual Contributors

### General User Characteristics

All users can be assumed to have the following characteristics:

1. Ability to read and understand English.

* Prefer an intuitive and visually appealing interface.
* Appreciate easy navigation and seamless user experience.
* Use emojis, stickers, and multimedia for expressive communication.
* Engage in both public and private discussions based on interests and work requirements.
* Expect a responsive and fast system to enhance productivity.

### Multimedia group Admin Characteristics

* Manage multimedia content sharing within groups.
* Ensure compliance with copyright and content guidelines.
* Monitor multimedia file sizes and formats for compatibility.
* Have the ability to moderate multimedia content.
* Set permissions for uploading, downloading, and deleting multimedia files

### Novice Users Characteristics

* Require intuitive interface.
* Benefit from guided tutorials.
* Prefer predefined themes for customization.
* Need straightforward messaging and group joining process.
* Rely on clear error messages for troubleshooting.

### Experienced Users Characteristics

* Seek advanced message formatting options.
* Use keyboard shortcuts for efficiency.
* Prefer quick access to recent chats/groups.
* Appreciate in-depth notification settings.
* Utilize search and filter functions extensively

### Administrators Characteristics

* Require user management tools.
* Need access to detailed usage analytics.
* Manage group permissions and discussions.
* Monitor system health and performance.
* Administer security settings and access controls

## General Constraints

General design/implementation constraints include:

* The software system will be developed on and run under Windows 2000.

1. All code shall be developed to run on a Java Virtual Machine supporting Java 1.3.1 or above.
2. Plain text files shall be used for all text-based data storage and reports.
3. The documentation and code shall be in accord with relevant CIT AI firm documentation and style standards.

## Priority of requirements

(Students to list the high-level requirements in the order from the most critical to the least important ones.)

The following requirements are listed in the order from the most critical to the least important:

* **Critical Priority:**
* User Authentication and Authorization.
* Message Posting and Delivery.
* System Availability and Reliability.
* **High Priority:**
* User Registration and Login.
* Dashboard Design and Functionality.
* Discussion Group List and Management.
* Message Formatting and Multimedia Support.
* **Medium Priority:**
* User Profile Management.
* Search and Filters Functionality.
* Security Measures and Data Encryption.
* Real-time Monitoring and Issue Resolution.
* **Low Priority:**
* User Preferences and Customization.
* Accessibility Compliance.
* Documentation for Users and Administrators.
* Third-Party Integrations and Compatibility.

# Functional Requirements

The generic guidelines that are applicable for all the various functions specified in this document are:

* Users shall be informed of all Format and Rule Validation failures with an appropriate message, which shall not be documented in the Exceptions section.
* In case of Format or Rule Validation failures with system generated data, system shall abort, whilst in case of failures with user-input data, system shall allow user to retry.
* It shall be assumed that exceptions shall include any failure of a pre-condition and/or a business rule.
* The description of each of the functions shall describe the normal processing procedure, not the exceptional event flows.
* The data that is to be manipulated by a particular functional requirement shall be described by
  + A unique name
  + Whether the data is input or output to that requirement or provided by the system (eg date or time).
  + Whether the data is alphanumeric, numeric or of another format
  + Any validation rules, such as maximum size or specific ranges of values
  + How many times the data occurs (eg once or multiple times)
  + Whether the data is optional or not

## Registering to the chat system

The system shall support registration with all information’s

**Function Name:** Chat\_Register

### Trigger

* A person wants to register to entering to the chat system

### Pre-Conditions

* The person is a employee of multimedia group
* Registration need to be selected

### Post-Conditions

Either

* Entering personal information’s for creating account

or

* Use existing account information

### Business Rules Applicable

* Registration can be made at anytime
* Multiple accounts can be created

### Data Manipulated

|  |  |  |  |
| --- | --- | --- | --- |
| **Type** | **Name** | **Format** | **Validation Rules** |
| Input | Start time | hh:mm | Can be anytime |
| Input | Duration | Numeric | From 1 to 9 |
| Input | Chat ID | Numeric | From 1 to 9 |
| System | Password | Alpha-numeric | Max 8 characters with some special characters |
| System | System date | dd/mm/yyyy | dd<32,mm<13,  yyyy<2024 |
| System | System time | Hh:mm:ss | Hh<25,mm<60,ss<60 |

Table 3.1 Inputs required for Chat \_Register

### Normal Processing Procedure

* User enters to chat system
* User creates account
* User enters user id and password
* System validates login information
* User establishes the connection to server
* If user is available, system displays all available users
* User selects one of the available users.
* User start chatting by sending message
* System exits while user is offline

### Variations

* If user is idle, system displays “Sorry, user is idle for long time”.
* System prompts user to quit
* Reports Generated

No reports are generated for this function.

### Exceptions

|  |  |
| --- | --- |
| **Exception** | **Error Message (Displayed/Generated/Logged)** |
| Creating account with same name | Already name exists |
| Creating password with undefined symbols | Type your password correctly |

Table 3.2 Exceptions during Chat\_ register

## Login Into The System

**Function Name:** Login

**3.2.1 Trigger:**

* A user enters their username and password into the login form and clicks the "Login" button.

**3.2.2 Pre-Condition:**

* The user has a valid username and password.

**3.2.3 Post-Condition:**

* The user is logged into the chat system.
* The user is redirected to the chat lobby.

**3.2.4 Business Rules Applicable:**

* Only registered users can log in to the chat system.
* Usernames and passwords must be unique and case-sensitive.
* Passwords must be at least 8 characters long and contain at least one uppercase letter, one lowercase letter, and one number.

**3.2.5 Data Manipulated:**

* User information (username, password, ID)

|  |  |  |  |
| --- | --- | --- | --- |
| **Type** | **Name** | **Format** | **Validation Rules** |
| Input | Username | String | Should not contain symbols |
| Input | Password | String | Min. 8 characters with special characters |
| System | User ID | Integer | From 1 to 9 |

Table 4.1 Inputs required for Login

**3.2.6 Normal Processing Procedure:**

1. The user enters their username and password into the login form and clicks the "Login" button.
2. The system validates the user's input (username and password).
3. The system checks if the username and password match a registered user account.
4. If the username and password are valid, the system logs the user in and redirects them to the chat lobby.

**3.2.7 Variations:**

* If the username or password is invalid, the system displays an error message informing the user that their login credentials are incorrect.

**3.2.8 Exceptions:**

|  |  |
| --- | --- |
| **Exception** | **Error Message (Displayed/Generated/Logged)** |
| Invalid username | "Invalid username. Please enter a valid username." |
| Invalid password | "Invalid password. Please enter a valid password." |
| User account not found | "User account not found. Please check your username and password." |
| Unable to log in | "Unable to log in. Please try again later." |

Table 4.2 Exceptions during Login

**3.3 Validation**

**Function Name:** Validate\_Chat\_Input

**3.3.1 Trigger:**

* A user submits a chat message.

**3.3.2 Pre-Condition:**

* The user is logged in to the chat system.
* The user is currently in a chat session (public or private).

**3.3.3 Post-Condition:**

* The chat message is validated against the applicable rules.
* If the message is valid, it is processed and displayed in the chat session.
* If the message is invalid, an error message is displayed to the user.

**3.3.4 Business Rules Applicable:**

* Message content must not be empty.
* Message content must not exceed the maximum character limit.
* Message content must not contain any prohibited words or phrases.

**3.3.5 Data Manipulated :**

|  |  |  |  |
| --- | --- | --- | --- |
| **Type** | **Name** | **Format** | **Validation Rules** |
| Input | Message Content | Plain text | Must not be empty. Must not exceed the maximum character limit. Must not contain any prohibited words or phrases. |

Table 5.1 Inputs for Validate\_Chat\_Input

**3.3.6 Normal Processing Procedure:**

1. The user submits a chat message.
2. The system validates the message content against the applicable rules.
3. If the message is valid, it is processed and displayed in the chat session.
4. If the message is invalid, an error message is displayed to the user.

**3.3.7 Variations:**

* If the message content exceeds the maximum character limit, the system truncates the message to the maximum length and displays a warning message to the user.

**3.3.8 Exceptions:**

|  |  |
| --- | --- |
| **Exception** | **Error Message (Displayed/Generated/Logged)** |
| Empty message content | "Message content cannot be empty." |
| Message exceeds maximum length | "Message content exceeds the maximum length of 500 characters." |
| Message contains prohibited words | "Message content contains prohibited words or phrases." |

Table 5.2 Exceptions for Validate\_Chat\_Input

**3.4 Public And Private Chat**

**3.4.1Function Name:** Chat\_Public

**3.4.1.1 Trigger:**

* A user joins the chat system.

**3.4.1.2 Pre-Condition:**

* The user is logged in to the chat system.

**3.4.1.3 Post-Condition:**

* The user is added to the public chat session.
* The user can view and send messages in the public chat session.

**3.4.1.4 Business Rules Applicable:**

* All logged-in users are automatically added to the public chat session.
* Users can send messages to all other users in the public chat session.

**3.4.1.5 Data Manipulated:**

|  |  |  |  |
| --- | --- | --- | --- |
| **Type** | **Name** | **Format** | **Validation Rules** |
| Input | Username | Non-empty alphanumeric string | Unique, non-empty |
| System | Session ID | Positive integer | Unique positive integer |
| System | User IDs | Array of positive integers | Valid User IDs |
| Input | Message Content | String | Non-empty |
| Input | Datetime | Dd/mm/yyyy | Valid datetime format |

Table 6.1 Inputs for Chat\_Public

**3.4.1.6 Normal Processing Procedure:**

1. The user joins the chat system.
2. The system validates the user's login credentials.
3. If the user is successfully logged in, the system adds the user to the public chat session.
4. The user can view and send messages in the public chat session.

**3.4.1.7 Variations:**

* If none of the receivers is not online, the system displays a message informing the user that the receivers is unavailable.

**3.4.1.8 Exceptions:**

|  |  |
| --- | --- |
| **Exception** | **Error Message (Displayed/Generated/Logged)** |
| Unable to join public chat session | "Unable to join public chat session. Please try again later." |

Table 6.2 exceptions for Chat\_Public

**3.4.2Function Name:** Chat\_Private

**3.4.2.1 Trigger:**

* A user clicks on the "Create Private Session" button.

**3.4.2.2 Pre-Condition:**

* The user is logged in to the chat system.
* The user is currently in a public chat session.

**3.4.2.3 Post-Condition:**

* A new private chat session is created between the user and the selected recipient.
* The user is redirected to the private chat session.

**3.4.2.4 Business Rules Applicable:**

* Only logged-in users can create private chat sessions.
* A private chat session can only be created with one other user.
* Users cannot create private chat sessions with themselves.

**3.4.2.5 Data Manipulated:**

|  |  |  |  |
| --- | --- | --- | --- |
| **Type** | **Name** | **Format** | **Validation Rules** |
| Input | Username | Non-empty alphanumeric string | Unique, non-empty |
| System | Session ID | Positive integer | Unique positive integer |
| System | User IDs | Array of positive integers | Valid User IDs |
| Input | Message Content | String | Non-empty |
| Input | Datetime | Dd/mm/yyyy | Valid datetime format |

Table 6.3 Inputs for Chat\_Private

**3.4.2.6 Normal Processing Procedure:**

1. The user clicks on the "Create Private Session" button.
2. The system validates the user's input (username and recipient username).
3. The system checks if the recipient user is currently online.
4. If the recipient user is online, the system creates a new private chat session between the user and the recipient.
5. The system redirects the user to the private chat session.

**3.4.2.7 Variations:**

* If the recipient user is not online, the system displays a message informing the user that the recipient is unavailable.

**3.4.2.8 Exceptions:**

|  |  |
| --- | --- |
| **Exception** | **Error Message (Displayed/Generated/Logged)** |
| Invalid username | "Invalid username. Please enter a valid username." |
| Recipient user not found | "Recipient user not found. Please enter a valid username." |
| Unable to create private chat session | "Unable to create private chat session. Please try again later." |

Table 6.4 Exceptions for Chat\_Private

**3.5 Voice Chat**

**Function Name:** Manage\_Voice\_Chat\_Preferences

**3.5.1 Trigger:**

* A user clicks on the "Voice Chat Preferences" button or accesses the voice chat settings menu.

**3.5.2 Pre-Condition:**

* The user is logged in to the chat system.
* The user has a microphone and speakers/headphones connected to their device.

**3.5.3 Post-Condition:**

* The user's voice chat preferences are updated.
* The user's voice chat experience is adjusted according to their preferences.

**3.5.4 Business Rules Applicable:**

* Only logged-in users can manage their voice chat preferences.
* Users can only modify their own voice chat preferences.
* Voice chat preferences should adhere to system limitations and privacy guidelines.

**3.5.5 Data Manipulated:**

|  |  |  |  |
| --- | --- | --- | --- |
| **Data Type** | **Name** | **Format** | **Validation Rules** |
| User Information | Username | String | Unique and non-empty |
| User Information | User ID | Integer | Unique positive integer |
| Voice Chat Preferences | Input Device | String | Valid audio input device |
| Voice Chat Preferences | Output Device | String | Valid audio output device |
| Voice Chat Preferences | Voice Activation Mode | Boolean | True or False |
| Voice Chat Preferences | Push-to-Talk Key bind | String | Valid keyboard key or mouse button |
| Voice Chat Preferences | Microphone Sensitivity | Integer | Range: 0-100 |
| Voice Chat Preferences | Voice Chat Notification | Boolean | True or False |

Table 7.1 Inputs for Manage\_Voice\_Chat\_Preferences

**3.5.6 Normal Processing Procedure:**

1. The user accesses the voice chat preferences menu.
2. The system displays the user's current voice chat preferences.
3. The user modifies their voice chat preferences as desired.
4. The user saves their updated voice chat preferences.
5. The system validates the user's input and updates their preferences accordingly.
6. The system applies the updated preferences to the user's voice chat experience.

**3.5.7 Variations:**

* If the user's input is invalid, the system displays an error message and prompts for correct input.
* If the user's device settings change, the system may prompt for reconfiguration of voice chat preferences.

**3.5.8 Exceptions:**

|  |  |
| --- | --- |
| **Exception** | **Error Message (Displayed/Generated/Logged)** |
| Invalid input device | "Invalid input device selected. Please choose a valid microphone." |
| Invalid output device | "Invalid output device selected. Please choose valid speakers or headphones." |
| Invalid keybind | "Invalid keybind selected. Please choose a valid key or button." |
| Invalid microphone sensitivity | "Microphone sensitivity must be between 0 and 100." |

Table 7.2 Exceptions for Manage\_Voice\_Chat\_Preferences

**3.6 Video Chat**

**Function Name:** Manage\_Video\_Chat\_Preferences

**3.6.1 Trigger:**

* A user clicks on the "Video Chat Preferences" button or accesses the video chat settings menu.

**3.6.2 Pre-Condition:**

* The user is logged in to the chat system.
* The user has a webcam connected to their device.

**3.6.3 Post-Condition:**

* The user's video chat preferences are updated.
* The user's video chat experience is adjusted according to their preferences.

**3.6.4 Business Rules Applicable:**

* Only logged-in users can manage their video chat preferences.
* Users can only modify their own video chat preferences.
* Video chat preferences should adhere to system limitations and privacy guidelines.

**3.6.5 Data Manipulated:**

|  |  |  |  |
| --- | --- | --- | --- |
| **Data Type** | **Name** | **Format** | **Validation Rules** |
| User Information | Username | String | Unique and non-empty |
| User Information | User ID | Integer | Unique positive integer |
| Video Chat Preferences | Video Input Device | String | Valid video input device (webcam) |
| Video Chat Preferences | Video Resolution | String | Valid video resolution (e.g., 720p, 1080p) |
| Video Chat Preferences | Video Framerate | Integer | Valid framerate (e.g., 30fps, 60fps) |
| Video Chat Preferences | Video Quality | String | Valid video quality setting (e.g., Low, Medium, High) |
| Video Chat Preferences | Video Background | String | Valid video background option (e.g., None, Blur, Virtual Background) |

Table 8.1 Inputs for Manage\_Video\_Chat\_Preferences

**3.6.6 Normal Processing Procedure:**

1. The user accesses the video chat preferences menu.
2. The system displays the user's current video chat preferences.
3. The user modifies their video chat preferences as desired.
4. The user saves their updated video chat preferences.
5. The system validates the user's input and updates their preferences accordingly.
6. The system applies the updated preferences to the user's video chat experience.

**3.6.7 Variations:**

* If the user's input is invalid, the system displays an error message and prompts for correct input.
* If the user's device settings change, the system may prompt for reconfiguration of video chat preferences.

**3.6.8 Exceptions:**

|  |  |
| --- | --- |
| **Exception** | **Error Message (Displayed/Generated/Logged)** |
| Invalid input device | "Invalid input device selected. Please choose a valid webcam." |
| Invalid resolution | "Invalid video resolution selected. Please choose a supported resolution." |
| Invalid framerate | "Invalid framerate selected. Please choose a supported framerate." |
| Invalid quality setting | "Invalid video quality setting selected. Please choose a valid quality option." |

Table 8.2 Exceptions for Manage\_Video\_Chat\_Preferences

**3.7 Add User**

**Function Name:** Add\_User

**3.7.1 Trigger:**

* An administrator or authorized user initiates the process of adding a new user to the chat system.

**3.7.2 Pre-Condition:**

* The administrator or authorized user is logged in to the chat system.
* They have the necessary permissions to add new users.

**3.7.3 Post-Condition:**

* A new user account is created with the provided information.
* The new user is added to the chat system's user database.

**3.7.4 Business Rules Applicable:**

* Only authorized users can add new users to the chat system.
* Usernames must be unique and non-empty.
* Passwords must meet minimum complexity requirements.
* User information must be accurate and complete.

**3.7.5 Data Manipulated:**

|  |  |  |  |
| --- | --- | --- | --- |
| **Data Type** | **Name** | **Format** | **Validation Rules** |
| User Information | Username | String | Unique and non-empty |
| User Information | Password | String | Minimum length, mix of characters |
| User Information | Email Address | String | Valid email format |
| User Information | First Name | String | Non-empty |
| User Information | Last Name | String | Non-empty |

Table 9.1 Inputs for Add\_User

**3.7.6 Normal Processing Procedure:**

1. The administrator or authorized user accesses the user management interface.
2. They initiate the process of adding a new user.
3. The system prompts for the new user's information (username, password, email, first name, last name).
4. The administrator or authorized user enters the required information.
5. The system validates the input and checks for uniqueness and adherence to business rules.
6. If the input is valid, the system creates a new user account and adds it to the user database.
7. The system generates a confirmation message indicating successful user creation.

**3.7.7 Variations:**

* If the username is already taken, the system prompts for a different username.
* If the password does not meet complexity requirements, the system prompts for a stronger password.
* If the email address is invalid, the system prompts for a valid email address.

**3.7.8 Exceptions:**

|  |  |
| --- | --- |
| **Exception** | **Error Message (Displayed/Generated/Logged)** |
| Duplicate username | "Username already exists. Please choose a different username." |
| Invalid password | "Password does not meet complexity requirements. Please choose a stronger password." |
| Invalid email address | "Invalid email address. Please enter a valid email address." |
| Missing information | "Please enter all required information." |

Table 9.2 Exceptions for Add\_User

**3.8 Remove User**

**Function Name:** Remove\_User

**3.8.1 Trigger:**

* An admin clicks on the "Remove User" button or selects the option from a user management menu.

**3.8.2Pre-Condition:**

* The admin is logged in to the chat system with administrative privileges.
* The user to be removed exists in the chat system.

**3.8.3 Post-Condition:**

* The selected user is removed from the chat system.
* The removed user can no longer access the chat system or participate in chat sessions.

**3.8.4 Business Rules Applicable:**

* Only admins with appropriate permissions can remove users.
* Admins cannot remove themselves from the chat system.
* Removing a user should adhere to data retention policies and audit trails.

**3.8.5 Data Manipulated:**

|  |  |  |  |
| --- | --- | --- | --- |
| **Data Type** | **Name** | **Format** | **Validation Rules** |
| User Information | Username | String | Unique and non-empty |
| User Information | User ID | Integer | Unique positive integer |
| User Status | Active | Boolean | True or False |
| User Access | Admin | Boolean | True or False |

Table 10.1 Inputs for Remove\_User

**3.8.6 Normal Processing Procedure:**

1. The admin selects the user to be removed.
2. The system confirms the admin's intention to remove the user.
3. The admin confirms the removal action.
4. The system validates the admin's input and verifies the user's existence.
5. The system deactivates the user's account and removes them from the active user list.
6. The system logs the removal action for auditing purposes.

**3.8.7 Variations:**

* If the admin cancels the removal action, the system aborts the process and no changes are made.
* If the user to be removed is currently in a chat session, the system may notify the participants of their removal.

**3.8.8 Exceptions:**

|  |  |
| --- | --- |
| **Exception** | **Error Message (Displayed/Generated/Logged)** |
| Invalid admin credentials | "Invalid admin credentials. Please check your username and password." |
| Insufficient admin privileges | "Insufficient admin privileges to remove users. Please contact a higher-level administrator." |
| Invalid user username | "Invalid user username. Please enter a valid username." |
| User not found | "User not found. Please enter a valid username." |

Table 10.2 Exceptions for Remove\_User

**3.9 Alert Notification**

**Function Name:** Manage\_Alert\_Notifications

**3.9.1 Trigger:**

* A user clicks on the "Alert Notifications" button or accesses the notification settings menu.

**3.9.2 Pre-Condition:**

* The user is logged in to the chat system.

**3.9.3 Post-Condition:**

* The user's alert notification preferences are updated.
* The user receives notifications according to their preferences.

**3.9.4 Business Rules Applicable:**

* Only logged-in users can manage their alert notification preferences.
* Users can only modify their own alert notification preferences.
* Alert notification preferences should adhere to system limitations and privacy guidelines.

**3.9.5 Data Manipulated:**

|  |  |  |  |
| --- | --- | --- | --- |
| **Data Type** | **Name** | **Format** | **Validation Rules** |
| User Information | Username | String | Unique and non-empty |
| User Information | User ID | Integer | Unique positive integer |
| Alert Notification Preferences | New Message Notification | Boolean | True or False |
| Alert Notification Preferences | Mention Notification | Boolean | True or False |
| Alert Notification Preferences | Sound Alert | Boolean | True or False |
| Alert Notification Preferences | Desktop Alert | Boolean | True or False |
| Alert Notification Preferences | Mobile Push Notification | Boolean | True or False |
| Alert Notification Preferences | Notification Timeout | Integer | Range: 0-3600 seconds |

Table 11.1 Inputs for Manage\_Alert\_Notifications

**3.9.6 Normal Processing Procedure:**

1. The user accesses the alert notification preferences menu.
2. The system displays the user's current alert notification preferences.
3. The user modifies their alert notification preferences as desired.
4. The user saves their updated alert notification preferences.
5. The system validates the user's input and updates their preferences accordingly.
6. The system applies the updated preferences to the user's notification experience.

**3.9.7 Variations:**

* If the user's input is invalid, the system displays an error message and prompts for correct input.
* If the user's device settings change, the system may prompt for reconfiguration of alert notification preferences.

**3.9.8 Exceptions:**

|  |  |
| --- | --- |
| **Exception** | **Error Message (Displayed/Generated/Logged)** |
| Invalid notification timeout | "Notification timeout must be between 0 and 3600 seconds." |

Table 11.2 Exceptions for Manage\_Alert\_Notifications

**3.10 FILE PERMISSION**

**Function Name:** Manage\_File\_Permissions

**3.10.1 Trigger:**

* A user attempts to upload, download, or share a file in the chat system.

**3.10.2 Pre-Condition:**

* The user is logged in to the chat system.
* The user has access to the file they want to manage permissions for.

**3.10.3 Post-Condition:**

* The file permissions are updated according to the user's actions.
* Users with appropriate permissions can access or modify the file as allowed.

**3.10.4 Business Rules Applicable:**

* Only authorized users can manage file permissions.
* File permissions should adhere to the chat system's security policies.
* File permissions should not conflict with user roles and access levels.

**3.10.5 Data Manipulated:**

|  |  |  |  |
| --- | --- | --- | --- |
| **Data Type** | **Name** | **Format** | **Validation Rules** |
| File Information | File ID | Integer | Unique positive integer |
| File Information | File Name | String | Non-empty, valid file name |
| File Information | File Type | String | Valid file extension |
| File Information | File Size | Integer | Positive integer |
| File Information | File Owner | User ID | Valid User ID |
| File Permissions | Access Level | String | Read, Write, or None |
| File Permissions | User/Group | User ID or Group ID | Valid User ID or Group ID |

Table 12.1 Inputs for Manage\_File\_Permissions

**3.10.6 Normal Processing Procedure:**

1. The user initiates an action involving file permissions (upload, download, share).
2. The system checks the user's permissions for the file.
3. If the user has sufficient permissions, the action is allowed.
4. If the user lacks necessary permissions, the system prompts for permission management.
5. The user specifies the desired access level and the user or group to apply it to.
6. The system validates the user's input and updates the file permissions accordingly.

**3.10.7 Variations:**

* If the user's input is invalid, the system displays an error message and prompts for correct input.
* If the user attempts to grant permissions beyond their own access level, the system denies the request.

**3.10.8 Exceptions:**

|  |  |
| --- | --- |
| **Exception** | **Error Message (Displayed/Generated/Logged)** |
| Invalid file ID | "Invalid file ID specified. Please select a valid file." |
| Invalid access level | "Invalid access level specified. Please choose Read, Write, or None." |
| Invalid user/group | "Invalid user or group specified. Please enter a valid User ID or Group ID." |
| Permission conflict | "Cannot grant permissions beyond your own access level." |

Table 12.2 Exceptions for Manage\_File\_Permissions

**3.11 Chat Search Filter**

- Search functionality for finding specific messages, users, or chat rooms.

- Filters for sorting conversations based on criteria like date, user, or topic.

**Function Name:** Chat\_Search\_Filter

**3.11.1 Trigger:**

* A user enters a search term or applies a filter in the chat history.

**3.11.2 Pre-Condition:**

* The user is logged in to the chat system.
* The user has access to the chat history.

**3.11.3 Post-Condition:**

* The chat history is filtered based on the user's search term or filter criteria.
* The filtered chat messages are displayed to the user.

**3.11.4 Business Rules Applicable:**

* Only logged-in users can search and filter chat history.
* Users can only search and filter chat history that they have access to.
* Search terms and filter criteria must be valid.

**3.11.5 Data Manipulated: Data Manipulated:**

|  |  |  |  |
| --- | --- | --- | --- |
| **Data Type** | **Name** | **Format** | **Validation Rules** |
| Input | Chat messages | String | Should be sent or received messages only |

Table 13.1 Inputs for Chat\_Search\_Filter

**3.11.6 Normal Processing Procedure:**

1. The user enters a search term or applies a filter in the chat history.
2. The system validates the user's input (search term or filter criteria).
3. The system retrieves the chat history from the database.
4. The system filters the chat history based on the user's search term or filter criteria.
5. The system displays the filtered chat messages to the user.

**3.11.7 Variations:**

* If no search term or filter criteria are provided, the system displays the entire chat history.

**3.11.8 Exceptions:**

|  |  |
| --- | --- |
| **Exception** | **Error Message (Displayed/Generated/Logged)** |
| Invalid search term | "Invalid search term. Please enter a valid search term." |
| Invalid filter criteria | "Invalid filter criteria. Please enter valid filter criteria." |
| Unable to retrieve chat history | "Unable to retrieve chat history. Please try again later." |

Table 13.2 Exceptions for Chat\_Search\_Filter

# User Interface Requirements

**4.1 User Registration and Login:**

* Users should be able to create an account with a unique username and password.
* There should be a login page for registered users to access the system.

**4.2 Dashboard:**

* Upon logging in, users should be greeted with a user-friendly dashboard.
* The dashboard should display a list of discussion groups the user is a part of.
* There should be options to manage their profile, settings, and notifications.

**4.3 Discussion Group List:**

* Users should see a list of available discussion groups.
* They should be able to search and filter discussion groups by name, topic, or category.
* Each group should display its name, description, and the number of members.

**4.4 Joining/Leaving Groups:**

* Users should have the option to join or leave discussion groups they are interested in.
* There should be a clear and easy-to-use interface for joining and leaving groups.

**4.5 Discussion Group Interface:**

* Clicking on a group should lead to a discussion group interface.
* Users should be able to view the group's messages, member list, and group details.
* The discussion interface should display messages in a chronological order, with the latest messages at the top.
* Users should be able to post messages within the group.

**4.6 Message Posting:**

* There should be a text input field for users to type and send messages.
* Users should have the ability to format their messages, including text styles and attachments.
* There should be an option to preview the message before posting it.

**4.7 Notifications:**

* Users should receive notifications for new messages in the groups they are part of.
* Notification preferences (e.g., email, in-app, push notifications) should be customizable.

**4.8 Profile Management:**

* Users should be able to view and edit their profile information.
* They should have the option to upload a profile picture and set a status message.

**4.9 Search and Filters:**

* Users should be able to search for specific messages, users, or discussion groups.
* Filters such as date, topic, and user should be available to refine search results.

**4.10 User Interactions:**

* Users should be able to like, reply to, and report messages.
* There should be options to send private messages to other users within a group.

**4.11 Accessibility and Responsiveness:**

* The user interface should be accessible to users with disabilities.
* The application should be responsive, ensuring a consistent experience across different devices and screen sizes.

**4.12 Security:**

* User data, including messages and personal information, should be encrypted and secured.
* Secure authentication and authorization mechanisms should be in place to protect user accounts.

**4.13 Help and Support:**

* Users should have access to a help center or support feature for assistance.
* There should be a FAQ section and contact options for user support.

**4.14 Feedback and Reporting:**

* Users should have a way to provide feedback and report issues or inappropriate content.
* There should be a mechanism to address and resolve user-reported problems.

**4.15 User Preferences:**

* Users should have options to customize the interface's appearance, such as theme selection and font size.

**4.16 Logout:**

* Users should be able to log out of the system securely.

**4.17 Privacy Settings:**

* Users should have control over their privacy settings, including who can see their profile and messages.

**4.18 User Documentation:**

* Provide user documentation or a user manual to help users navigate and understand the system's features.

# Non-Functional requirements

Non-functional requirements for a Multiple Chat User System Software Requirements Specification (SRS) describe aspects of the system's performance, reliability, security, and other characteristics that are not directly related to its functional capabilities. These requirements are essential for ensuring that the system meets user expectations and performs effectively. Here are some non-functional requirements that should be considered for a multiple chat user system:

**5.1 Performance:**

- **Response Time:** The system should respond to user actions, such as sending or receiving messages, within a specified time frame, e.g., less than 2 seconds.

- **Scalability:** The system should handle an increasing number of users and messages without significant degradation in performance.

- **Throughput**: Define the number of messages or users the system can handle per unit of time.

- **Resource Utilization:** Monitor and optimize the utilization of server resources, such as CPU and memory.

**5.2 Reliability:**

- **Availability:** The system should be available 24/7 with minimal downtime for maintenance or upgrades.

- **Fault Tolerance:** The system should continue to operate in the presence of failures, ensuring data integrity and message delivery.

- **Data Backup and Recovery:** Regularly backup chat data and provide a mechanism for data recovery in case of data loss or corruption.

**5.3 Security:**

- **Authentication:** Users should authenticate securely, and access should be restricted to authorized users only.

- **Authorization:** Define roles and permissions to restrict access to specific features and data.

- **Data Encryption:** Ensure that messages and user data are encrypted during transmission and storage.

- **Protection Against Attacks:** Implement security measures to prevent common attacks like cross-site scripting (XSS), SQL injection, and denial of service (DoS).

**5.4 Scalability:**

- **Load Balancing:** Implement load balancing mechanisms to distribute user requests evenly across multiple servers or instances.

- **Database Scalability:** The database should be scalable to accommodate increasing data volumes and user accounts.

**5.5 Usability:**

- **User Interface (UI) Design**: Design an intuitive and user-friendly interface to facilitate ease of use.

- **Accessibility:** Ensure the system is accessible to users with disabilities, complying with relevant accessibility standards.

**5.6 Compatibility:**

- **Cross-Platform Compatibility:** Ensure the system is compatible with a variety of operating systems and web browsers.

- **Mobile Compatibility:** If applicable, ensure that the system functions well on mobile devices and provides a mobile app or responsive web design.

**5.7 Compliance:**

- **Regulatory Compliance:** Ensure the system complies with data protection regulations, such as GDPR, as well as any industry-specific regulations.

- **Industry Standards:** Adhere to industry standards and best practices for chat system security and privacy.

**5.8 Performance Testing:**

- **Stress Testing:** Assess the system's performance under heavy loads and determine its breaking point.

- **Load Testing**: Measure how the system performs under typical and peak loads to identify and resolve performance bottlenecks.

**5.9 Monitoring and Logging:**

- **Logging:** Implement comprehensive logging to track system events, user activities, and errors for auditing and troubleshooting.

- **Real-time Monitoring:** Use monitoring tools to detect and resolve issues in real time, ensuring system health.

**5.10 Data Privacy and Retention:**

- Define data retention policies and ensure compliance with privacy regulations regarding the storage and deletion of user data.

**5.11 Disaster Recovery:**

- Define a disaster recovery plan to recover the system in case of catastrophic events, including data center failures or natural disasters.

**5.12 Documentation:**

- Provide clear and comprehensive documentation for system administrators, users, and developers.

**5.13 Third-Party Integrations:**

- Ensure compatibility with third-party APIs and services, if applicable, such as authentication providers and message notification services.

**6.Operating Environment**

The system specified shall operate on the hardware, software requirements outlined in

this section.

**6.1 Hardware**

The system is expected to have the following minimum configuration:

 Pentium-200MHz (equivalent or above).

 128 MB RAM.

 20MB free hard disk space.

 VGA Monitor.

**6.2 Software**

The basic software required for effective operation shall be:

 Windows 2000 operating system.

 Java Virtual Machine capable of running Java v1.3.1 applications or above.

**6.3 Printer**

The printer must be compatible with the computer system and must be capable of

printing text files.

**6.4 External Data Storage**

All data that needs to be stored on external storage devices for security and protection

reasons shall be clearly identified. Descriptions of all data files and methods of access

to archive these data items shall be provided as part of the User Manual. It is

expected that the technical operator of this system shall have the required expertise to

transfer all relevant data files on to an external archive (e.g. Floppy Disk, DAT, CD

and/or ZIP disk).

**7. Acceptance Criteria**

The system must be able to, at least

• User Authentication:

The chat system must provide secure user authentication methods, such as username/password and multi-factor authentication.

Users should be able to reset their passwords securely in case of a forgotten password.

• Chat Features:

Users must be able to create new chat rooms or join existing ones.

The chat system should support one-on-one messaging as well as group messaging.

Users should be able to send text messages, multimedia files, and emojis in chats.

The system should provide real-time message delivery and updates.

• Security and Privacy:

Messages sent and received in the chat system should be encrypted to ensure data privacy.

Users should have the option to block/report other users for inappropriate behavior.

The system should comply with data protection regulations and ensure user privacy.

# 

# Glossary

|  |  |
| --- | --- |
| **Term** | **Description** |
| User Authentication | The process by which users confirm their identity before accessing the chat system. |
| User Registration | The process allowing users to create accounts, providing necessary details like username and password. |
| Message Sending | The functionality enabling users to send text, images, or multimedia messages to other users or groups. |
| Message Reception | The system's ability to receive and display messages sent by other users in real-time. |
| User Presence Status | Indication of whether a user is online, offline, or away to facilitate communication decisions. |
| Group Chat Creation | Capability for users to create chat groups, adding multiple users for collaborative conversations. |
| File Sharing | Functionality allowing users to share files (documents, images, videos) within chat conversations. |
| Notifications | Alerts sent to users for new messages, ensuring users are aware of incoming communications |
| User Profile Management | Users can edit their profiles, change profile pictures, and update personal information. |
| Intuitive User Interface | An easy-to-navigate interface ensuring users can quickly access chats, contacts, and features |
| User Help and Support | Access to help resources, FAQs, and customer support to assist users in case of issues. |
| Emojis and Stickers | Capability to use emojis and stickers to enhance the expression within chat messages. |
| Performance | The chat system should provide swift message delivery, even under high user loads, ensuring real-time communication |
| Scalability | Ability to handle a growing number of users and messages without compromising performance |
| Security | End-to-end encryption ensuring that messages and shared files are secure and private. |
| Reliability | The system should be stable and available, minimizing downtime and ensuring continuous service |
| Compatibility | The chat system should be compatible with various devices (desktop, mobile) and operating systems |
| Data Backup and Recovery | Regular backup of chat data and mechanisms for data recovery in case of system failures |
| Compliance | Adherence to data protection laws and regulations ensuring user privacy and legal compliance |

Table G1.0 Glossary

# Appendix A: External Context Model

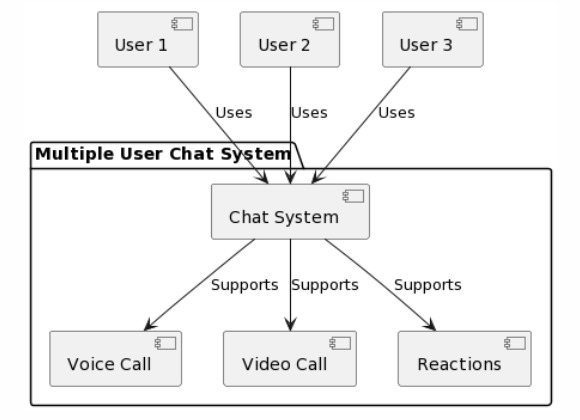


Figure 1.0 External Context Model

# Appendix B: Internal Context Model

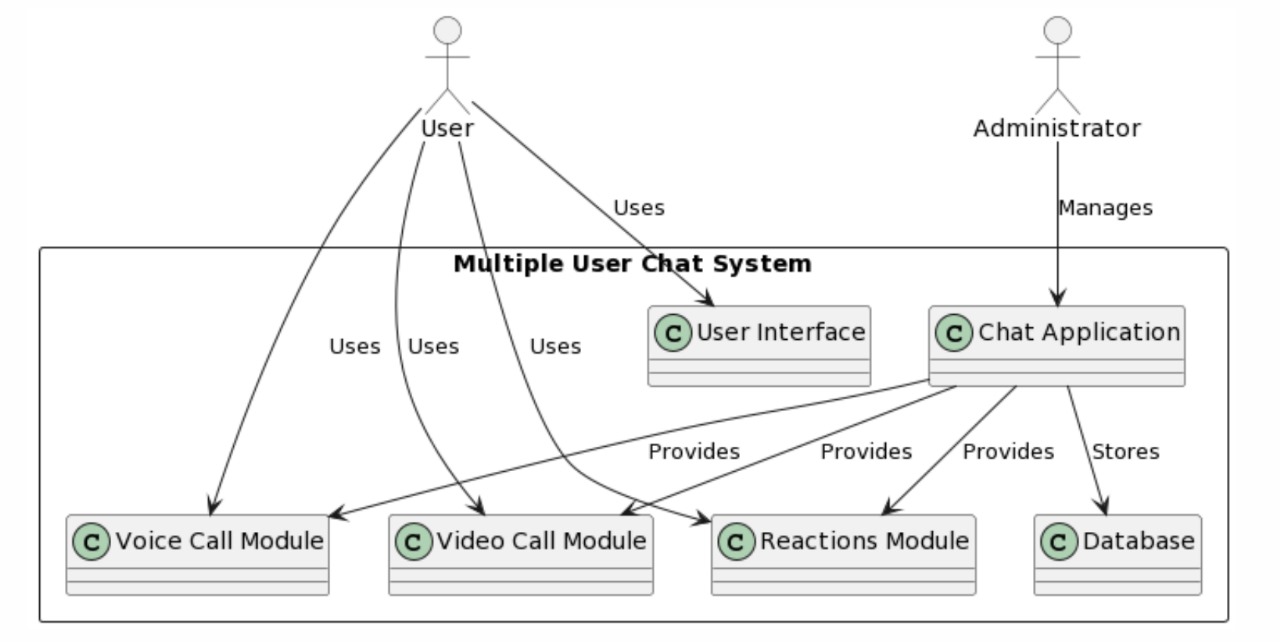


Figure 2.0 Internal Context Diagram

# Appendix C: Use Case Analysis Notes

**Use Case: User Registration and Login**

**Actor:** Unregistered User

**Description:** Unregistered users can create an account (registration) and log in to access the system.

**Preconditions:** The user is not registered.

**Basic Flow:**

* User selects the "Register" option.
* User provides a unique username and password.
* User submits the registration form.
* The system validates the information and creates a new user account.
* User logs in with the newly created credentials.

**Alternate Flows:**

1. If the chosen username is already taken, the system prompts the user to select a different one.

**Use Case: Dashboard**

**Actor:** Registered User

**Description:** After logging in, users are presented with a dashboard displaying their discussion groups and account management options.

**Preconditions:** The user is logged in.

**Basic Flow:**

* User logs in.
* The system presents a dashboard with discussion groups and account options.
* User can access and interact with their groups or manage their account.

**Alternate Flows:**

1. Users can navigate to discussion groups or settings.

**Use Case: Joining/Leaving Groups**

**Actor:** Registered User

**Description:** Users can join or leave discussion groups.

**Preconditions:** The user is logged in.

**Basic Flow:**

* User navigates to the group they want to join/leave.
* User selects the "Join" or "Leave" option.
* The system updates the user's group membership accordingly**.**

**Alternate Flows:**

1. Users can be prompted to confirm their choice if leaving a group to prevent accidental exits.

**Use Case: Discussion Group Interface**

**Actor:** Registered User

**Description:** Users can access discussion groups, view messages, and post messages.

**Preconditions:** The user is logged in and part of the group.

* Basic Flow:
* User selects a discussion group.
* The system presents the discussion group interface.
* User can view group messages, member list, and post messages.

**Alternate Flows:**

1. Users can interact with messages, such as liking, replying, or reporting.
2. Users can send private messages to other group members.

**Use Case: Message Posting**

**Actor:** Registered User

**Description:** Users can compose and send messages within a discussion group.

**Preconditions:** The user is logged in and is in a discussion group.

**Basic Flow:**

* User navigates to the discussion group interface.
* User types a message in the text input field.
* User can format the message (e.g., add attachments).
* User previews the message (optional).
* User posts the message.

**Alternate Flows:**

1. Users can attach files, images, or links to messages.
2. There may be character limits for messages.

**Use Case: Notifications**

**Actor:** Registered User

**Description:** Users receive notifications for new messages and can customize their notification preferences.

**Preconditions:** The user is logged in.

**Basic Flow:**

* User receives notifications for new messages in their groups.
* User can access notification settings and customize notification preferences (e.g., email, in-app, push notifications).

**Use Case: Profile Management**

**Actor:** Registered User

**Description:** Users can view and edit their profile information.

**Preconditions:** The user is logged in.

**Basic Flow:**

* User accesses their profile.

User can view and edit profile details, including profile picture and status message

# Appendix D: Development Process

Tasks Undertaken

|  |  |  |
| --- | --- | --- |
| **Task Name** | **Description** | **Technique(s) Used** |
| Problem Investigation | Understand the project goal | User interviews, Requirement Documents |
| Problem Analysis | Defining Scope, resources of the project | Project plan, Team splitting |
| Use Case Modelling | System architecture creation | DBMS, UI |
| Deployment | Deploying the system to users | Apps, DBMS (servers) |
| Requirements Review | Users testing the system | Feedbacks, Corrections |
| Documentation | Document system features and uses | Creating User manuals |

Table D1.0 Tasks undertaken in the development of this document

**Task Name:** Problem Investigation

**Description:** Understand the project goals and user needs.

**Techniques Used:** Conduct user interviews and surveys, Create a requirements document.

**Task Name:** Problem Analysis

**Description:** Define project scope, schedule, and resources.

**Techniques Used:** Create a project plan, Identify team members and roles, Allocate resources.

**Task Name:** Use Case Modelling

**Description:** Create the system's architecture and user interface.

**Techniques Used:** Design the database schema, Design the user interface (UI/UX).

**Task Name:** Deployment

**Description:** Make the system available to users.

**Techniques Used:** Deploy the application on servers, Configure databases and server environments.

**Task Name:** Requirements Review

**Description:** Let a group of users test the system.

**Techniques Used:** Gather user feedback, Identify areas for improvement.

**Task Name:** Documentation

**Description:** Document system features and usage.

**Techniques Used:** Create user manuals and help documentation, Develop system documentation for maintenance

Task Outputs

|  |  |  |
| --- | --- | --- |
| **Task Name** | **Outputs** | **Section(s)** |
| Problem Investigation | Understanding of project goals | 3,4,5 |
| Problem Analysis | Defined project scope and allocated resources | 2,6,7 |
| Use Case Modelling | Creation of system architecture and use cases | 3,4,6 |
| Deployment | System deployed to users for real-world usage | 5,6,7 |
| Requirements Review | Refined system based on user feedback and corrections made | 3,4,7 |
| Documentation | Comprehensive user manuals detailing system features and usage | 3,4,6 |

Table 52.0 Sections affected by the various tasks.

**a) Task Name:** Problem Investigation

**b) Outputs:** Understanding of project goals

**c) Sections:**

Section 3 (Functional Requirements): Understanding user needs and expectations.

Section 4 (User Interface Requirements): Gathering insights on user interaction preferences.

Section 5 (Non-Functional Requirements): Identifying performance, security, and other non-functional aspects.

**a) Task Name:** Problem Analysis

**b) Outputs:** Defined project scope and allocated resources

**c) Sections:**

Section 2 (System Overview): Defining the scope of the system and its overall goals.

Section 6 (Operating Environment): Specifying hardware and software resources required for the project.

Section 7 (Acceptance Criteria): Ensuring alignment with project goals and defining user expectations.

**a) Task Name:** Use Case Modelling

**b) Outputs:** Creation of system architecture and use cases

**c) Sections:**

Section 3 (Functional Requirements): Creating detailed functional requirements based on use cases.

Section 4 (User Interface Requirements): Designing user interactions and interfaces based on identified use cases.

Section 6 (Operating Environment): Considering system architecture and software compatibility for deployment

**a) Task Name:** Deployment

**b) Outputs:** System deployed to users for real-world usage

**c) Sections:**

Section 6 (Operating Environment): Ensuring compatibility with specified hardware and software.

Section 7 (Acceptance Criteria): Validation against user acceptance criteria post-deployment.

Section 5 (Non-Functional Requirements): Verifying performance, security, and other non-functional aspects during deployment.

**a) Task Name:** Requirements Review

**b) Outputs:** Refined system based on user feedback and corrections made

**c) Sections:**

Section 3 (Functional Requirements): Adjusting functional requirements based on user testing and feedback.

Section 4 (User Interface Requirements): Modifying interface elements and user interactions based on user testing.

Section 7 (Acceptance Criteria): Ensuring that the system aligns with user expectations and acceptance criteria.

**a) Task Name:** Documentation

**b) Outputs:** Comprehensive user manuals detailing system features and usage

**c) Sections:**

Section 3 (Functional Requirements): Documenting functional aspects and features.

Section 4 (User Interface Requirements): Detailing user interactions and interface elements for users.

Section 6 (Operating Environment): Providing technical documentation regarding hardware and software requirements.

# Appendix E: Contributions

This section shall contain the names of all the contributors to this document. The sections that each contributor has worked on shall be described in the table below.

|  |  |
| --- | --- |
| **Contributor Name** | **Sections Worked On** |
| Gokul | Document Template Development, Document Preparation(1 – Appendices), Formatting all tables,  Introduction, Product Overview, Acceptance Criteria, Glossary, Development Process, Task Outputs |
| Dinesh Kumar | Functional Requirements, External Context Diagram |
| Ranjith | User – Interface Requirements, Use Case Analysis Notes |
| Velmurugan | Non – functional requirements, Internal Context Diagram |
| Ranjith | Document Review (1 – Appendices) |
| Dinesh Kumar | Document Review (1 – Appendices) |
| Velmurugan | Document Review (1 – Appendices) |

Table 52.0 Document Contributions

# Appendix F: Meeting Agendas/Minutes

This section shall include agendas and minutes from all the various meetings that team members have attended as part of this project.

|  |  |  |
| --- | --- | --- |
| **ID** | **Meeting Date/Time** | **Apologies** |
| 1 | October 20th 2023, 2:30 – 5:30 pm | For not finishing tables |
| 2 | October 21st 2023, 3:00 – 6:00 pm | For not finishing footers |
| 3 | Oct6ober 22nd 2023, 5:00 – 7:30 pm | -None- |

Table 53.0 Meeting dates/times

# Appendix G: Problem Investigation Reports

* During interviews, users expressed a strong preference for a simple and intuitive user interface, emphasizing the importance of easy navigation and clear instructions.
* Users highlighted the need for robust security features, emphasizing secure authentication methods and end-to-end encryption for messages, ensuring data privacy.
* The document inspection revealed well-structured functional requirements, detailing user interactions and system behaviour comprehensively.
* User interface requirements were found to be thorough, specifying elements like dashboard layout, message formatting options, and search functionalities, ensuring a user-friendly experience.