
Software Requirements Specification

for

ALAP_CHARITA

Prepared by

**Mithun Chandra Sarkar
MUH2025029M**

**Institute of Information Technology
Noakhali Science and Technology University**

Supervised By,

Shudeb Babu Sen Amit

Lecturer

Institute of Information Technology

Noakhali Science and Technology University

18.02.2024

Table of Contents

1. Introduction.....	5
1.1 Problem Statement.....	5
1.2 Purpose.....	6
1.3 Project Scope	7
1.4 Glossary	8
1.5 Overview.....	8
2. Stakeholders and Characteristics.....	9
2.1 Users:	9
2.2 Developers:	9
2.3 Cloud Service Providers.....	9
2.4 Regulatory Authorities.....	9
2.5 Advertisers:	9
2.6 Feedback Providers.....	9
3. Design and Implementation Constrains.....	9
4. Hardware and Software Requirements:.....	10
5. Requirement Specification	11
5.1 Functional Requirement.....	11
5.2 Non-Functional Requirements	12
6. Requirement Engineering Process	12
6.1 Requirement Elicitation Techniques.....	13
6.1.1 Hold Interviews.....	13
6.1.2 Perform Document Analysis.....	13
6.1.3 System Interface Analysis.....	13
6.1.4 Distribute Questionnaires.....	13
6.2 Requirement Validation	14
6.2.1 Review the Requirements	14
6.2.2 Test the Requirements.....	14
6.2.3 Simulate the requirements.....	14
7. Project Scenario	14
8. Use Case Diagram	16
9. Use Case Description	17
9.1 UC-01.....	17
9.2 UC-02.....	17
9.3 UC-03.....	18
9.4 UC-04.....	18
9.5 UC-05.....	19
9.6 UC-06.....	19
9.7 UC-07.....	20
9.8 UC-08.....	20
10. Sequence Diagram	21
11. Activity Diagram.....	22
11.1 Send Message.....	23
11.2 Create Chatroom	24
11.3 Join Chatroom.....	25
12. Requirement Traceability Matrix	26

13. Appendix.....	27
13.1 Prioritization of requirements	27
13.1.1 Three-level Scale	27
13.1.2 Prioritization of the requirements of ALAP_CHARITA.....	27

List of Figures

Figure 1: Usecase Diagram.....	16
Figure 2: Sequence Diagram.....	21
Figure 3: Send Message.....	23
Figure 4: Create Chatroom	24
Figure 5: Join Chatroom	25

List of Tables

Table 1: Create Account	17
Table 2: Access Control.....	17
Table 3: View Products.....	18
Table 4: Order Product.....	18
Table 5: Search Product	19
Table 6: View Recommended Product	19
Table 7: Search Category	20
Table 8: Show Products Availability	21

1. Introduction

The Software Specification & Requirement Report (SSRR) for the ALAP_CHARITA chatting application aims to provide a comprehensive overview of the project's policies, scope, references, and a summary of its requirements. This document serves as a guide for the development team, stakeholders, and end-users, facilitating a clear understanding of the objectives and functionalities of the ALAP_CHARITA application. In today's digital age, communication is paramount, and mobile applications have revolutionized the way people connect and interact. ALAP_CHARITA is an android-based chatting application designed to streamline group communication and facilitate seamless interactions among users across different networks. By leveraging socket programming and Heroku cloud server technology, ALAP_CHARITA offers a simple yet powerful platform for users to engage in group conversations in real-time.

The primary purpose of ALAP_CHARITA is to provide users with a convenient and efficient means of communication, enabling them to exchange messages and stay connected with friends, family, colleagues, or communities. Unlike traditional chatting applications, ALAP_CHARITA focuses on group chats rather than one-to-one conversations, fostering a sense of community and collaboration among users. Upon installation of the ALAP_CHARITA application, users are prompted to sign in by entering their name, granting them access to the chat room where they can participate in ongoing conversations. The cloud server infrastructure ensures that users from different networks can join the chat seamlessly, enhancing the inclusivity and accessibility of the platform. Through ALAP_CHARITA, users can engage in vibrant discussions, share information, collaborate on projects, or simply connect with like-minded individuals. By eliminating the need to scroll through endless message histories, ALAP_CHARITA offers a fresh and clutter-free chatting experience, focusing solely on the present conversation.

1.1 Problem Statement

In today's digital age, communication has become increasingly vital, with individuals seeking efficient and accessible means to connect with others. While numerous messaging applications exist, many lack the simplicity and accessibility desired by users. Moreover, there's a growing need for secure and seamless group communication platforms that transcend geographical boundaries.

ALAP_CHARITA, a proposed android-based chatting application, aims to address these challenges by providing a user-friendly platform for group communication. Traditional messaging apps often lack the ability to facilitate group conversations seamlessly, leading to fragmented discussions and difficulties in managing multiple contacts. Additionally, concerns over privacy and data security persist, with users often wary of sharing sensitive information on mainstream platforms. ALAP_CHARITA seeks to offer a solution by providing a straightforward, android-based application that allows users to quickly sign in and join group chats. By leveraging socket programming and Heroku cloud server technology, ALAP_CHARITA enables users from different networks to participate in real-time conversations, fostering a sense of inclusivity and connectivity.

1.2 Purpose

The purpose of this report is to assess the functionality and usability of the ALAP_CHARITA chatting application and to identify any potential areas for improvement in its design and implementation. This report will conduct a comprehensive evaluation of ALAP_CHARITA's sign-in process, group chat functionality, and integration with Heroku cloud server technology. Additionally, it will examine the user experience and satisfaction levels with the application, highlighting any issues or challenges faced by users during their interaction with the platform. Furthermore, the report will explore the broader landscape of messaging applications and analyze ALAP_CHARITA's competitive positioning within this market. By providing actionable insights and recommendations, this report aims to guide future development efforts and enhance the overall effectiveness and user satisfaction of the ALAP_CHARITA chatting application.

1.3 Project Scope

This report will focus on evaluating the functionality and effectiveness of the ALAP_CHARITA chat application, aiming to identify areas for improvement and refinement within its operational model. The analysis will encompass the following key aspects:

- **User Experience and Interface Design:** This section will delve into the usability and intuitiveness of the ALAP_CHARITA application's user interface. It will evaluate the ease of navigation, clarity of instructions, and overall user experience during the sign-in process and while engaging in group chats. Additionally, this section will assess the visual design elements and responsiveness of the application across different Android devices.
- **Security Measures:** The report will examine the robustness of the security features implemented within the ALAP_CHARITA application to safeguard user data and communication channels. It will assess encryption protocols, authentication mechanisms, and data protection measures to ensure user privacy and confidentiality.
- **Functionality and Performance:** This section will analyze the reliability and performance of the ALAP_CHARITA application in facilitating real-time group communication. It will evaluate the speed of message delivery, system responsiveness under varying network conditions, and overall stability of the application during peak usage periods. Additionally, the report will assess the scalability of the application to accommodate an increasing number of users without compromising performance.
- **User Feedback and Satisfaction:** The report will gather feedback from ALAP_CHARITA users to gauge their satisfaction levels and identify any pain points or areas for improvement. It will conduct surveys or interviews to capture user perceptions regarding the application's features, usability, and overall experience. Insights from user feedback will inform recommendations for enhancing the application's functionality and addressing user needs effectively.
- **Competitive Analysis:** This section will provide a comparative analysis of ALAP_CHARITA against other android-based messaging applications available in the market. It will assess ALAP_CHARITA's unique features, competitive advantages, and areas where it can differentiate itself to attract and retain users. Additionally, the report will identify emerging trends in the messaging app landscape and potential opportunities for innovation within the ALAP_CHARITA platform.

1.4 Glossary

This section provides definitions for all document names, acronyms, and abbreviations. The application domain's terms and concepts are defined.

GUI - Graphical User Interface

API – Application Programming Interface

SRS – Software Requirement Specification

UI – User Interface

SDLC – Software Development Life Cycle

MB – Megabytes

XML – Extensible Markup Language

RESTful – Representational State Transfer

HTML – Hyper Text Markup Language

1.5 Overview

This report evaluates the functionality and effectiveness of the ALAP_CHARITA chat application, a simple android-based messaging platform designed for group communication. ALAP_CHARITA aims to provide users with a seamless and secure means of connecting with others in real-time, irrespective of geographical constraints.

The report assesses various aspects of the ALAP_CHARITA application, including its user interface, security measures, functionality, performance, and user satisfaction. By examining these key areas, the report aims to identify opportunities for improvement and refinement within the application's operational model.

ALAP_CHARITA offers users a straightforward sign-in process, enabling them to quickly join group chats upon installation of the application. Leveraging socket programming and Heroku cloud server technology, ALAP_CHARITA facilitates real-time communication among users from different networks, fostering inclusivity and accessibility. The report will delve into the user experience provided by the ALAP_CHARITA application, evaluating factors such as ease of navigation, clarity of instructions, and overall usability. Additionally, it will assess the security features implemented within the application to safeguard user data and communication channels, ensuring privacy and confidentiality.

Furthermore, the report will analyze the functionality and performance of the ALAP_CHARITA application, including message delivery speed, system responsiveness, and scalability under varying network conditions. Insights from user feedback will be gathered to gauge satisfaction levels and identify areas for enhancement. The project overview outlines the structure of the report, which includes sections covering the project scope, introduction, purpose, methodology, findings, and recommendations. The report serves as a valuable resource for the development team behind ALAP_CHARITA, as well as stakeholders interested in exploring alternative messaging solutions that prioritize simplicity, security, and user satisfaction.

2. Stakeholders and Characteristics

2.1 Users: Users are the primary stakeholders of the ALAP_CHARITA chat application. They engage with the platform to communicate with others in group chats. The key characteristics of users are:

- They seek simplicity and ease of use in the application interface.
- They prioritize privacy and security in their communications.
- They may come from diverse backgrounds and technical expertise levels.
- They value real-time and seamless communication experiences.

2.2 Developers: Developers are responsible for designing, building, and maintaining the ALAP_CHARITA application. The key characteristics of developers are:

- They possess technical expertise in android development and socket programming.
- They prioritize reliability, scalability, and performance in the application architecture.
- They require access to relevant development tools and resources.
- They may collaborate with cloud service providers for server management and maintenance.

2.3 Cloud Service Providers: Cloud service providers host the ALAP_CHARITA application's server infrastructure. The key characteristics of cloud service providers are:

- They offer scalable and reliable cloud computing services.
- They prioritize security and compliance with data protection regulations.
- They provide technical support and maintenance for the server infrastructure.
- They ensure high availability and uptime for the ALAP_CHARITA application.

2.4 Regulatory Authorities: Regulatory authorities oversee compliance with legal and regulatory requirements related to data privacy and communication services. The key characteristics of regulatory authorities are:

- They enforce laws and regulations governing data protection and privacy.
- They conduct audits and inspections to ensure compliance with regulatory standards.
- They provide guidelines and recommendations for secure communication platforms.
- They may collaborate with stakeholders to address emerging regulatory challenges.

2.5 Advertisers: Advertisers may utilize the ALAP_CHARITA platform to reach users through targeted advertising campaigns. The key characteristics of advertisers are:

- They seek opportunities to promote their products or services to a relevant audience.
- They value insights and analytics to measure the effectiveness of advertising campaigns.
- They may require ad placement options and targeting features within the application.
- They prioritize transparency and accountability in advertising practices.

2.6 Feedback Providers: Feedback providers offer insights and suggestions for improving the ALAP_CHARITA application based on their user experience. The key characteristics of feedback providers are:

- They provide constructive feedback on usability, features, and overall experience.
- They may participate in beta testing and user research activities.
- They value responsiveness and acknowledgment from the development team.
- They contribute to the continuous improvement and refinement of the application.

3. Design and Implementation Constrains

Design and implementation constraints are crucial factors that shape the development and deployment of the ALAP_CHARITA chat application. These constraints ensure the successful

execution of the project and guide the development team throughout the process. The following constraints have been identified for the ALAP_CHARITA chat application:

1. **Scalability:** The ALAP_CHARITA chat application must be designed to handle a potentially large user base and high volumes of concurrent messages. It should be scalable to accommodate growth in user traffic without compromising performance or reliability.
2. **Security:** Security is paramount in the ALAP_CHARITA chat application to protect user data and communication channels from unauthorized access or breaches. Measures such as encryption, secure authentication, and data privacy protocols must be implemented to ensure the confidentiality and integrity of user information.
3. **Availability:** The ALAP_CHARITA chat application should be highly available and reliable, ensuring users can access the platform and engage in real-time communication without interruptions. Measures such as redundancy, load balancing, and disaster recovery plans should be in place to minimize downtime and ensure continuous service availability.
4. **User Experience:** The user experience is a critical aspect of the ALAP_CHARITA chat application, requiring intuitive design, responsive interfaces, and seamless navigation. The application should prioritize ease of use and accessibility for users from diverse backgrounds and technical expertise levels.
5. **Data Management:** Effective data management is essential for the ALAP_CHARITA chat application to store and manage user profiles, chat histories, and other relevant information securely. Robust database design, data encryption, and backup mechanisms should be implemented to ensure data integrity and availability.
6. **Integration:** The ALAP_CHARITA chat application may require integration with third-party services such as cloud platforms, payment gateways, and analytics tools. An API-based architecture should be adopted to facilitate seamless integration with external systems and services.
7. **Technology Stack:** The technology stack for the ALAP_CHARITA chat application includes Java for Android development, Node.js for server-side scripting, and Heroku cloud platform for hosting. These technologies offer robust frameworks and tools for building scalable and reliable applications.

4. Hardware and Software Requirements The hardware and software requirements for developing and deploying the ALAP_CHARITA chat application are as follows:

1. Hardware Requirements:

- Android device with a minimum version of Android 7 (Nougat) for testing and deployment.
- Development machine (e.g., laptop or desktop) with adequate processing power and memory for software development and testing.

2. Software Requirements:

- Android Studio: Integrated development environment (IDE) for Android app development using Java.

- Node.js: JavaScript runtime environment for server-side scripting and application logic.
- Heroku: Cloud platform-as-a-service (PaaS) for hosting and deploying the ALAP_CHARITA chat application.
- PostgreSQL: Advanced open-source relational database for storing user data and chat messages securely.
- React Native: JavaScript framework for building native mobile apps, if cross-platform development is desired.
- Git: Version control system for tracking changes to the application codebase and collaborating with team members.

3. Dependencies

```
"dependencies": {  
  
  "ejs": "^3.1.9",  
  
  "express": "^4.15.2",  
  
  "socket.io": "^1.7.2"  
  
},
```

5. Requirement Specification

All the requirements based on the elicitation process are described in this section.

5.1 Functional Requirement

Functional requirements are those requirements that are used to illustrate the internal working nature of the system, the description of the system, and explanation of each subsystem. It consists of what task the system should perform, the processes involved, which data the system should hold and the interfaces with the user.

Requirement ID	Description	Stakeholders	Priority
FR-1	User Sign In	Users	High
	Users should be able to sign in to their accounts upon launching the app.		
FR-2	Group Chat Creation	Users	High
	Users can create group chats and invite others to join the conversation.		
FR-3	Real-time Messaging	Users	High
	Users can send and receive messages in real-time within the group chat.		
FR-4	Cloud Server Integration	Development Team	Medium

	Integration with Heroku cloud server to facilitate real-time communication across different networks.		
FR-5	User Authentication	Users	High
	Users must authenticate their identity before accessing the chat application.		

5.2 Non-Functional Requirements

Non-functional requirements specify the system's qualities and constraints, such as performance, security, and usability.

Requirement ID	Description	Stakeholders	Priority
NFR-1	Security	Users, Development Team	High
	The application must implement robust security measures to protect user data and communication channels.		
NFR-2	Performance	Users, Development Team	High
	The application must be responsive and perform efficiently, even under high user loads and network congestion.		
NFR-3	Compatibility	Users, Development Team	Medium
	The application should be compatible with Android version 7 and above.		
NFR-4	Usability	Users	High
	The application should have an intuitive and user-friendly interface, allowing users to navigate and use the app effortlessly.		

6. Requirement Engineering Process

Requirements Engineering (RE) determines software requirements according to customer requirements or needs. Requirements engineering process includes requirements elicitation, needs modeling, requirements analysis, requirements assurance & validation, and requirements management.

6.1 Requirement Elicitation Techniques

Requirements elicitation is the practice of researching and finding system requirements for users, customers, and other stakeholders, also referred to as "requirement gathering". Requirement elicitation can be done by contacting participants directly or by doing some research, analysis and testing.

6.1.1 Hold Interviews

We hold discussions that can be held individually or with a small group of participants. They are an effective way to access services without spending a lot of time with participants because we meet with people to discuss only certain important requirements of this program. Negotiations are useful for obtaining individual requirements for members in organizing workshops where those members of the program come together to resolve any issues or conflicts. We mainly perform our interview based on some specific criteria.

- Short description about goals and objectives
- Registration process
- Searching Audio Files
- Storage system of each account
- Compression size of audio files

6.1.2 Perform Document Analysis

Existing documentation can help to show how systems are currently operating or what they are what I should do. Documents include written information about current programs, business processes, needs specifications, and competitor research. Review once textual analysis can help determine which performance should remain and functionality that isn't in use. After existing document

In analysis, we found several problems with the existing system.

- Existing systems cannot perform file compression.
- A user cannot share a file with others.
- No cloud storage system is provided by the existing systems.

6.1.3 System Interface Analysis

The first thing to do is to identify which systems the system-to-be shall communicate with. It could be a server on the Internet, a piece of software on the same host as the system-to-be, some hardware or something completely different.

6.1.4 Distribute Questionnaires

The questionnaire is a useful way to investigate styles, changes in attitudes and users' ideas, and user satisfaction with priorities and preferences. Our lists of questions were as short as possible. The respondent may be tired or frustrated. Had a basic reason for all the questions as well as group the topic areas together for the respondent to focus on. The main advantage of this survey responses

was that they were collected in the usual way. Information was summarized by a large number of people.

6.2 Requirement Validation

Requirement validation ensures that the requirements are correct and reflect the quality you want from this program. In the beginning, our requirements looked good but when we read them and tried to work with them they came out having ambiguities and gaps.

6.2.1 Review the Requirements

Negative peer review, especially the type of rigorous review called evaluation, is unique among the highest quality software processes available. We had a team of reviewers representing different perspectives and carefully examined written needs, analysis models, and related information on disability.

6.2.2 Test the Requirements

The test creates another view of the requirements. We also performed writing tests regarding assurance of whether the expected performance was found or not. Getting tested by the user needs to document the expected product behavior under specified conditions.

6.2.3 Simulate the requirements

To stimulate requirements, trading tools are available that we have used to simulate a proposed system in place or to add details of written requirements. The simulation takes prototyping to the next level.

7. Project Scenario

Mithun, a college student, recently downloaded the ALAP_CHARITA Chat Application to stay connected with his friends and classmates. Here's how he navigates through the application:

Sign In:

Mithun launches the ALAP_CHARITA app on his Android device. He enters his name in the provided field and clicks on the "Sign In" button. The application validates his credentials, and upon successful authentication, Mithun enters the main interface of the app.

Explore Group Chats:

In the main interface, Mithun sees a list of available group chats. He selects a chat group related to his college course and clicks on it to join the conversation. Upon entering the group chat, Mithun sees messages exchanged by other users in real-time.

Send Messages:

Excited to participate, Mithun types a message in the text input field. He hits the send button, and his message instantly appears in the group chat for others to see. Mithun engages in discussions with his peers, asking questions and sharing study resources seamlessly.

Receive Notifications:

While studying for an exam, Mithun receives a notification from the ALAP_CHARITA app. He checks the notification and sees that his friends have started a new discussion in the group chat. Mithun clicks on the notification and joins the conversation without delay.

Explore Additional Features:

Curious about other features, Mithun explores the menu options within the ALAP_CHARITA app. He discovers functionalities such as creating custom chat groups, inviting friends to join, and setting notification preferences. Mithun also explores the app settings to customize his profile, including adding a profile picture and updating his status.

Log Out:

After spending time chatting with friends, Mithun decides to log out of the ALAP_CHARITA app. He navigates to the settings menu and selects the "Log Out" option. The application confirms his action, and Mithun successfully logs out, ending his session.

8. Use Case Diagram

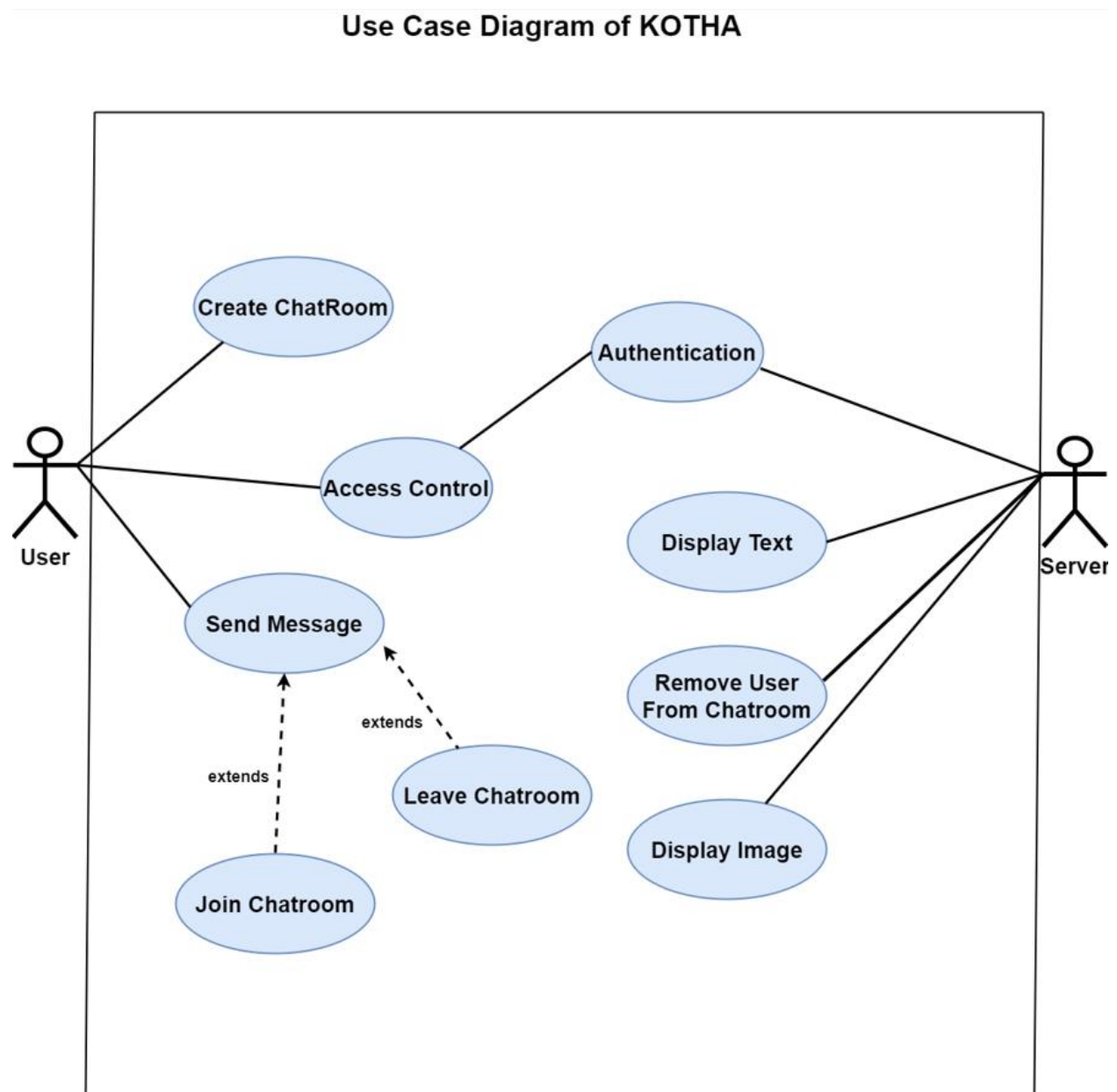


Figure 1: Usecase Diagram

9. Use Case Description

Table-01

9.1 UC-01

Use Case	Access Control
Goal	Allow users to sign in and access the chat application
Precondition	User has installed the ALAP_CHARITA Chat Application
Success End Condition	User successfully signs in and gains access to the app
Failed End Conditions	1) Incorrect username or password entered 2) User fails to authenticate due to network issues
Primary Actor	User
Secondary Actor	N/A
Trigger	User launches the ALAP_CHARITA Chat Application
Main Success Flow	1) User enters their username and password in the provided fields 2) System verifies the user's credentials and authenticates the user
Alternative Flow	1a) If the user enters incorrect credentials, the system displays an error message
Quality Requirements	1) The authentication process should be secure and resistant to brute force attacks 2) The app should provide appropriate error messages to guide users in case of authentication failures

Table-02

9.2 UC-02

Use Case	Create Chatroom
Goal	Allow users to create a new chatroom
Precondition	User is authenticated and logged in to the app
Success End Condition	New chatroom is successfully created
Failed End Conditions	1) User lacks necessary permissions to create a chatroom 2) System encounters an error while creating the chatroom
Primary Actor	User
Secondary Actor	N/A
Trigger	User selects the option to create a new chatroom
Main Success Flow	1) User provides a name for the new chatroom 2) User selects the members to invite to the chatroom 3) System creates the chatroom and adds the selected members

Alternative Flow	2a) If the user decides not to invite any members, the chatroom is created without any members
Quality Requirements	1) The creation process should be fast and seamless, even for users with slower internet connections
	2) The system should prevent the creation of duplicate chatroom names

Table-03

9.3 UC-03

Use Case	Send Message
Goal	Allow users to send messages in a chatroom
Precondition	User is authenticated and has joined a chatroom
Success End Condition	Message is successfully sent and displayed in the chatroom
Failed End Conditions	1) User loses network connection while sending the message
	2) System encounters an error while sending the message
Primary Actor	User
Secondary Actor	N/A
Trigger	User types a message and selects the send button
Main Success Flow	1) User types the message in the text input field
	2) User clicks on the send button to send the message
	3) System sends the message to the chatroom members
Alternative Flow	1a) If the user loses network connection, the message is queued for sending once connection is restored
Quality Requirements	1) Messages should be sent and displayed in real-time to provide a seamless chatting experience
	2) The system should handle message delivery failures gracefully and provide appropriate error messages

Table-04

9.4 UC-04

Use Case	Join Chatroom
Goal	Allow users to join an existing chatroom
Precondition	User is authenticated and has access to available chatrooms
Success End Condition	User successfully joins the selected chatroom
Failed End Conditions	1) User attempts to join a private chatroom without invitation
	2) System encounters an error while processing the join request
Primary Actor	User

Secondary Actor	N/A
Trigger	User selects a chatroom to join from the available options
Main Success Flow	1) User selects the desired chatroom from the list
	2) System adds the user to the selected chatroom
Alternative Flow	1a) If the chatroom is private and requires an invitation, the system prompts the user to request an invitation
Quality Requirements	1) The system should display a list of available chatrooms quickly and efficiently
	2) Joining a chatroom should be a straightforward process for users

Table-05

9.5 UC-05

Use Case	Leave Chatroom
Goal	Allow users to leave a chatroom they no longer want to participate in
Precondition	User is authenticated and has joined the chatroom
Success End Condition	User successfully leaves the chatroom
Failed End Conditions	1) User encounters an error while leaving the chatroom
Primary Actor	User
Secondary Actor	N/A
Trigger	User selects the option to leave the chatroom
Main Success Flow	1) User selects the option to leave the chatroom
	2) System removes the user from the chatroom
Alternative Flow	N/A
Quality Requirements	1) Leaving a chatroom should be a simple and intuitive process for users
	2) The system should update the chatroom member list promptly after a user leaves

Table-06

9.6 UC-06

Use Case	Authenticate User
Goal	Verify the identity of users accessing the app
Precondition	User has installed the ALAP_CHARITA Chat Application
Success End Condition	User successfully authenticates and gains access to the app
Failed End Conditions	1) User enters incorrect username or password
	2) System encounters an error while processing the authentication request

Primary Actor	User
Secondary Actor	N/A
Trigger	User launches the ALAP_CHARITA Chat Application
Main Success Flow	1) User enters their username and password in the provided fields 2) System verifies the user's credentials and authenticates the user
Alternative Flow	1a) If the user enters incorrect credentials, the system displays an error message
Quality Requirements	1) Authentication process should be secure and resistant to unauthorized access 2) The app should provide clear and informative error messages in case of authentication failures

Table-07

9.7 UC-07

Use Case	Display Text, Image
Goal	Display text and image messages in the chatroom
Precondition	User has joined a chatroom and received messages
Success End Condition	Text and image messages are successfully displayed
Failed End Conditions	1) System encounters an error while loading or displaying messages
Primary Actor	User
Secondary Actor	N/A
Trigger	User accesses the chatroom and scrolls through messages
Main Success Flow	1) System retrieves text and image messages from the server 2) System displays the messages in the chatroom
Alternative Flow	N/A
Quality Requirements	1) The system should load and display messages quickly and efficiently 2) Text and image messages should be rendered accurately and clearly for users

Table-08

9.8 UC-08

Use Case	Remove User from Chatroom
Goal	Allow chatroom administrators to remove users
Precondition	User is an administrator of the chatroom
Success End Condition	User is successfully removed from the chatroom
Failed End Conditions	1) User lacks necessary permissions to remove other users 2) System encounters an error while processing the removal request

Primary Actor	Administrator
Secondary Actor	N/A
Trigger	Administrator selects the option to remove a user
Main Success Flow	1) Administrator selects the user to be removed from the chatroom 2) System removes the selected user from the chatroom
Alternative Flow	N/A
Quality Requirements	1) The system should provide clear guidelines and restrictions for user removal to prevent misuse 2) Removal of users should be logged and audited for accountability

10. Sequence Diagram

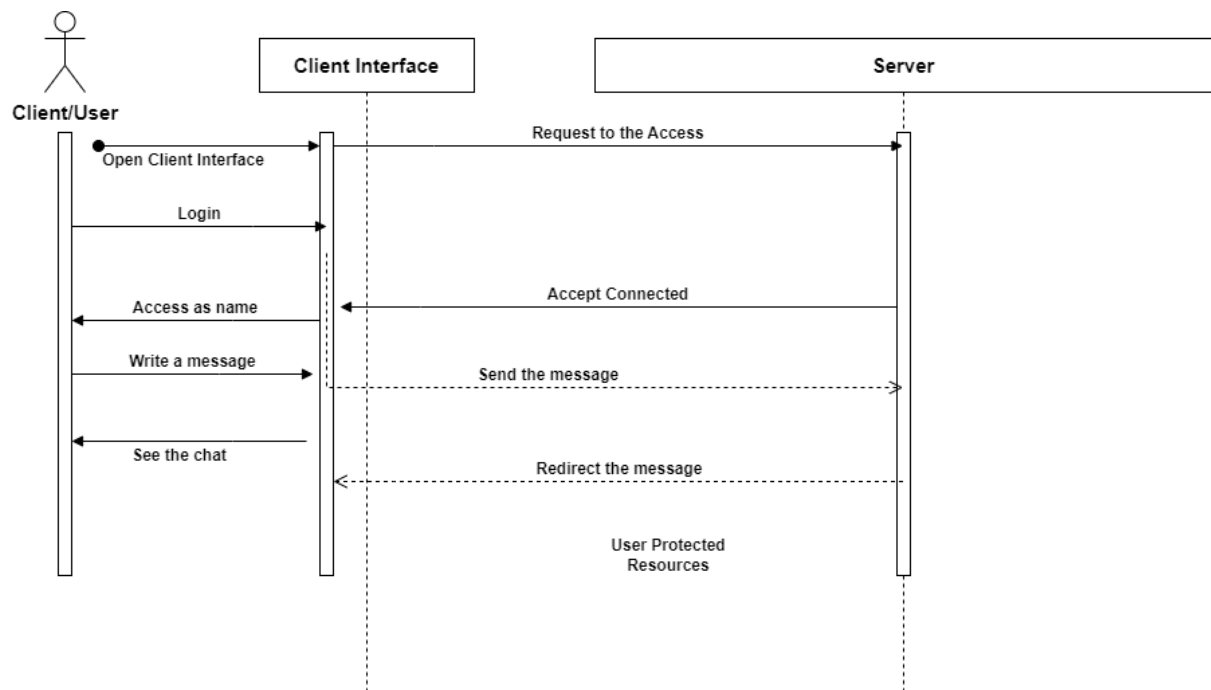


Figure 2: Sequence Diagram

11. Activity Diagram

11.1 Send Message

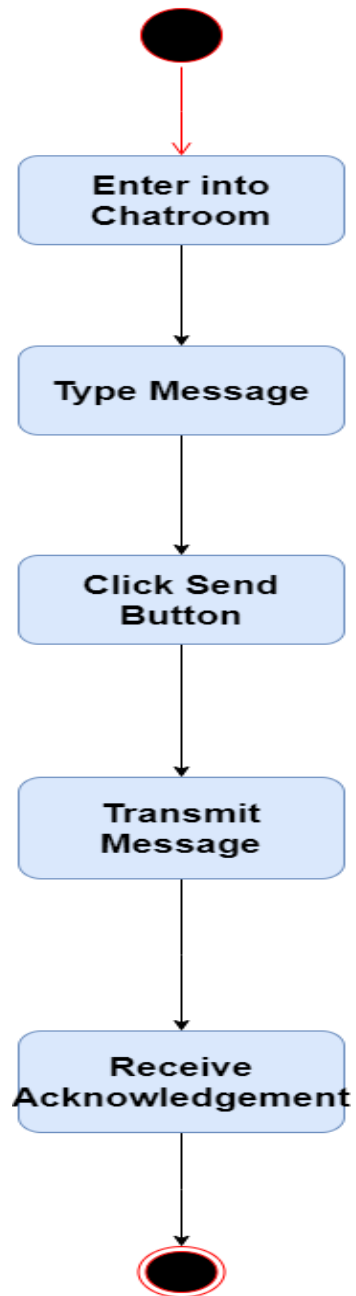


Figure 3:Send Message

11.2 Create Chatroom

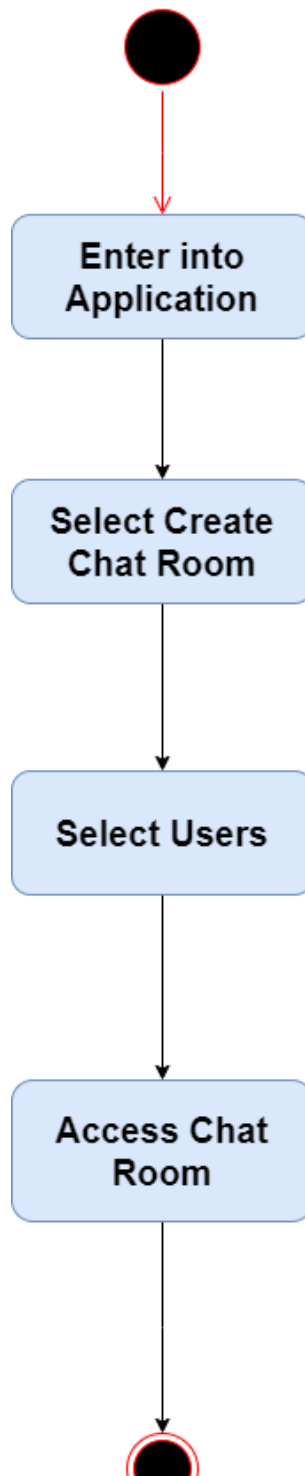


Figure 4:Create Chatroom

11.3 Join Chatroom

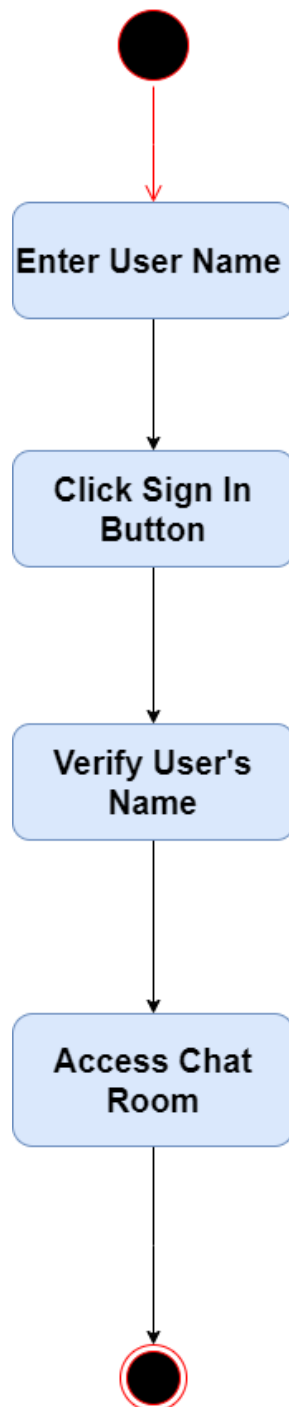


Figure 5:Join Chatroom

12. Requirement Traceability Matrix

A traceability matrix is a document, usually in the form of a table, used to assist in determining the completeness of a relationship by correlating any two baselined documents using a many-to-many relationship comparison. It is often used with high-level requirements (these often consist of marketing requirements) and detailed requirements of the product to the matching parts of high-level design, detailed design, test plan, and test cases.

Requirements Traceability Matrix					
Project Name	E-Farm	Business Area		Local	
Functional Activity	Use Case Reference	Design Document Reference	Code Module/Reference	User Acceptance Validation	Comments
FR1	UC1			Pass	
FR2	UC2			Pass	
FR3	UC3,UC5			Pass	
FR4	UC4			Verified	
FR5	UC6			Pass	
NFR6	UC7			Verified	
NFR7	UC8			Pass	
NFR8	UC9			Pass	

NFR9	UC5,UC7			Verified	
-------------	---------	--	--	----------	--

13. Appendix

13.1 Prioritization of requirements

We've prioritized the functional requirements by following Three-level Scale technique.

13.1.1 Three-level Scale

When a Business Analyst categorizes the requirements in any of the ordering or ranking scale, it is subject to the analyst's understanding of the business. Many analysts suggest that this method has some drawbacks and advocate methods that have more than one scale.

13.1.2 Prioritization of the requirements of ALAP_CHARITA

FR1 – High priority

FR2 – High priority

FR3 – High priority

FR4 – Medium priority

FR5 – High priority

NFR6 – High priority

NFR7 – High priority

NFR8 – Medium priority

NFR9 – High priority