Software Requirements Specification

for

RandomChatApp

**Version 1.1**

**Prepared by**

| Abhinav Kumar Singh | 200017 | CSE | abhinav555k@gmail.com |
| --- | --- | --- | --- |
| Boddu Lakshmi Pravallika | 200282 | CSE | boddulakshmipravallika@gmail.com |
| Gargi Naladkar | 200371 | CSE | garginaladkar@gmail.com |
| Kartavya Damor | 200492 | CSE | betudamor@gmail.com |
| Krishan Kumar | 200521 | CSE | krishkumar9352@gmail.com |
| Manish Meena | 200560 | CSE | manishlaha2002@gmail.com |
| Motupalli Sana Chaitanya | 200599 | CSE | sanachaitanya486@gmail.com |
| Ronit Mittal | 200820 | CSE | mittalronit897@gmail.com |
| Vasanthapu Poojitha Lakshmi | 201094 | CSE | vpoojithalakshmi@gmail.com |
| Vikash Kumar | 201119 | CSE | vikashpamg1999@gmail.com |

| **Course:** | **CS253** |
| --- | --- |
| **Mentor TA:** | **Sri Madhan** |
| **Date:** | **31-01-2022** |

Content

**Contents**

**Contents ii**

**Revisions iii**

**1** **Introduction 1**

1.1 Product Scope 1

1.2 Intended Audience and Document Overview 1

1.3 Definitions, Acronyms and Abbreviations 1

1.4 Document Conventions 1

1.5 References and Acknowledgments 1

**2** **Overall Description 2**

2.1 Product Overview 2

2.2 Product Functionality 3

2.3 Design and Implementation Constraints 3

2.4 Assumptions and Dependencies 3

**3** **Specific Requirements 4**

3.1 External Interface Requirements 4

3.2 Functional Requirements 5

3.3 Use Case Model 6-8

**4** **Other Non-functional Requirements 9**

4.1 Performance Requirements 9

4.2 Safety and Security Requirements 9

4.3 Software Quality Attributes 9

**5** **Other Requirements 10**

**Appendix A – Data Dictionary 11**

**Appendix B - Group Log 12**

**Revisions**

| **Version** | **Primary Author(s)** | **Description of Version** | **Date Completed** |
| --- | --- | --- | --- |
| 1.0 | Abhinav Kumar Singh, Boddu Lakshmi Pravallika, Gargi Naladkar, Kartavya Damor, Krishan Kumar, Manish Meena, Motupalli Sana Chaitanya, Ronit Mittal, Vasanthapu Poojitha Lakshmi, Vikash Kumar. | The app supports only messaging over text and users are verified by their Email through an OTP. | 01/02/2022 |
| 1.1 | Abhinav Kumar Singh, Boddu Lakshmi Pravallika, Gargi Naladkar, Kartavya Damor, Krishan Kumar, Manish Meena, Motupalli Sana Chaitanya, Ronit Mittal, Vasanthapu Poojitha Lakshmi, Vikash Kumar. | Changed the documentation as per the currentversion of our chat app. | 28/04/2022 |

# 

# Introduction

## Product Scope

* ***WebChatApp*** *is a web-based chatting App on which two users (from IITK) can ~~match~~ interact together and chat.*
* *The app only supports text messages(as of now).*
* *It will not support any type of calling facility( such as voice or video call).*
* *It might support sharing .jpg, .pdf and some other file formats in the coming future, if the texting thing goes on as expected.*
* *Users can send friend requests to each other and upon accepting the request, can start chatting.*

## Intended Audience and Document Overview

*This document is intended for the users (students of IITK), developers (the project team), course instructor (Dr. Indranil Saha) and to all the TA’s.*

## Definitions, Acronyms and Abbreviations

*TA - Technical Assistants*

*IITK- IIT Kanpur*

*app - Application*

*ytb - yet to be*

## Document Conventions

***Bold:*** *methods and objects.*

*Highlighted (yellow): important*

*Highlighted(red): security related*

*Highlighted(red): final changes*

## References and Acknowledgments

*<we will keep on updating this, with every website/ documentation which we will go through for help regarding the project>*

* ***Figma*** *<*[*https://www.figma.com*](https://www.figma.com)*> : We are using this website to design the UI of our application.*

# Overall Description

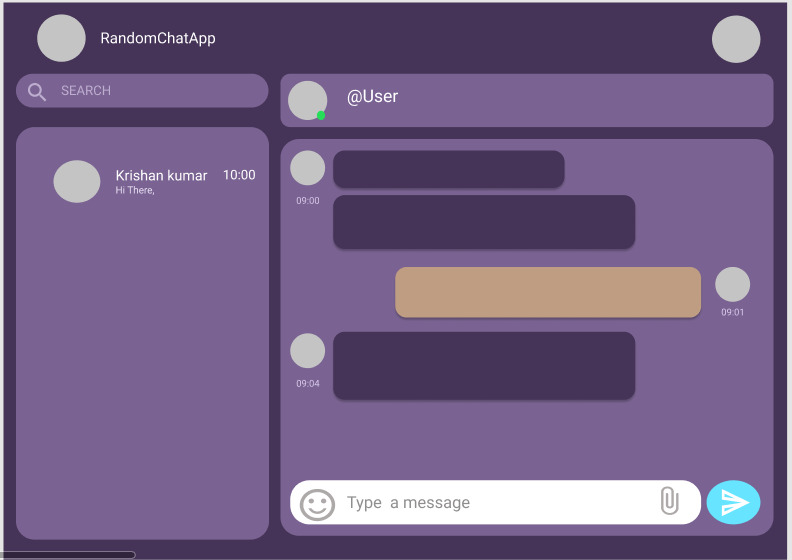
## Product Overview

*This app is developed for increasing social interactions among the IITK junta.*

*Even after shifting to the offline mode, many students are accustomed to the old lifestyle. We provide a window of opportunity to interact with the rest of the people and maybe even find the ones with similar interests.*

*The app allows you to chat with another student from IITK allowing you to send texts to each other.*

*The web app is somewhat similar to Omegle, but only with a text-chat feature (as of now).*

**

*System Interface*

*source:* [*https://www.figma.com/file/53ivCKdRZq3rfJhhPRCh5g/chat-application?node-id=0%3A1*](https://www.figma.com/file/53ivCKdRZq3rfJhhPRCh5g/chat-application?node-id=0%3A1)

## Product Functionality

* *Users can send friend requests to anyone.*
* *This app will allow people to* ***chat*** *with each other.*
* *Users can see each other’s profile before accepting a request .*
* *The app will have a* ***report link****, if a person reports someone then the admins will be able to ban him/her, if that person has done something against rules and regulations (-will be updated once the beta version goes live).*

## Design and Implementation Constraints

***Language Requirements:*** *The application will use*

***backend*** *-* ***NodeJS expressjs, NPM***

***frontend - HTML****,* ***CSS****,* ***JS, ejs, bootstrap***

***testing - Typescript,***

***database - mongoDB****,*

***messaging server*** *-* ***socket.io***

***Tools: VScode, VSCode, BurpSuite, Nmap,******Figma****(for designing interface)*

## Assumptions and Dependencies

**Assumptions:**

1. *The client/user will have a stable internet connection, to connect to the server and start chatting with other users.*
2. *The client/user will have a device with a web browser installed, to access the web-app.*
3. *The host server will never crash (although impossible), and will always be able to handle the load of online users and their data at every point of time.*
4. *There are no security breaches of any kind.*
5. *The web browser which the user is using will be able to render the web-app correctly.*

**Dependencies;**

*The web-app depends fully on the host server all the time, to elaborate some of them are:*

1. *The host server will be capable of handling the maximum number of users that can be online at the same time.*
2. *The database will securely save the passwords and other sensitive information of the users.*
3. *The Internet connection is fast enough to send/receive messages efficiently and quickly.*

# Specific Requirements

## External Interface Requirements

### User Interfaces

*First time users are required to sign-up and create a new account by verifying their email and creating a password for the same. Old users can simply sign-in with their pre-existing account. They also need to provide a user ID which would be visible to other clients.*

*~~Upon logging in, users would be asked to mention their interests. Based on these interests, the app would make a match with another user. The user can choose to click on the~~* ***~~Match Button~~*** *~~and is matched with that user (the identity of users is kept anonymous and only their user-name is visible.)~~. Upon logging in, the user can search for a user, via username or name, and send a request to them. After the request is accepted, a new chat-room is created and the users can send text-messages to each other.*

*~~Clicking again on the match button allows the user to chat with another user different from the previous one.~~ The chats and the list of friends are saved in the database .This would create a user’s personalised friends list.*

*~~The user can also report any inappropriate behaviour and strict actions would be taken.~~*

### Hardware Interfaces

*The app requires the following hardware:*

* *PC / Mobile (to access the web-app through a browser)*
* *Internet connection*

### Software Interfaces

*The app directly connects to the database “MongoDB” which stores login information, sensitive info like passwords, list of friends, chats, etc. This information will remain totally out of sight to the users but the admins can be able to access this information* ***(except passwords).***

## Functional Requirements

* + 1. *User Registration : New users will be able to register through a valid email Id, they will then be needed to create a unique username, and a password for their account.*
    2. *Login/ Sign In : Once a user has registered successfully he/ she will be able to login to the chat app using their email Id and password.*
    3. *Chat-Area : After successful login the users will be redirected to the Chat-Area where they will be able to interact with their FRIENDS. \*\**
       - 1. *\*\* FRIENDS : To ensure the privacy of all users, a user will only be able to chat/ interact with his/her friends.*
    4. *Friend-Request : To add a user as a FRIEND and start chatting, one is required to send a friend request to the user that he/she wants to add as a friend, that user will be added to the dashboard once the request is accepted, but if the request gets rejected/ declined then the user will not be able to resend the request to the same user again.*
    5. *The app would also generate a personalized friends list for every user, consisting of the users they have ever interacted with.*

…

* + 1. *Profile : Users will be able to see the Profile of their friends. The users will also be able to view and edit their own profile.*
    2. *Logout : For the safety of personal information of the user, it is advised that the users log-out of the app once they are done chatting with their friends.*
    3. *Chat:chats will be saved .*
    4. *search : users can search for a particular user and send a request in order to chat with them.*

…

## Use Case Model

### Use Case #1 Authentication System

**Author –** Kartavya Damor

**Purpose** - Registration and login to the system

**Requirements Traceability –** Chat bot with internet

**Priority** - High

**Preconditions** - Have a device with a web browser installed

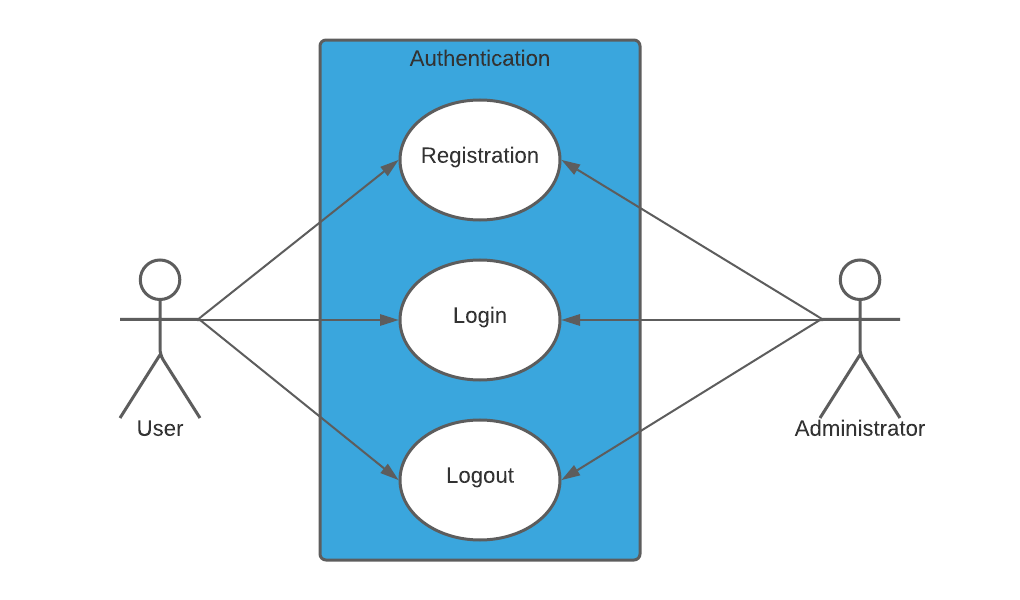
**Post conditions** - Access to the person’s account

**Actors** – Users and administrator

**Exceptions** - Error authenticating with database

**Includes** (other use case IDs)

**Notes/Issues** - Any relevant notes or issues that need to be resolved



### Use Case #2 Chat Box User interface

**Author –** Kartavya Damor

**Purpose** - User Interface after login

**Requirements Traceability –** Chatbot with internet

**Priority** - High

**Preconditions** - Successful login into the app

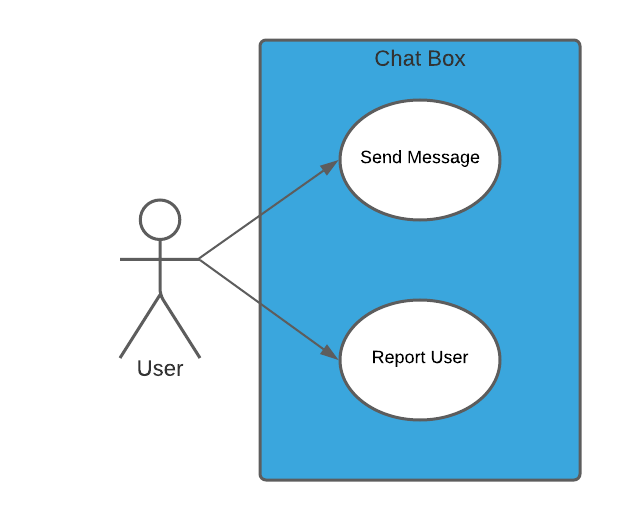
**Post conditions** - Chat with other users in accordance with Rules & Regulations

**Actors** – Users

**Exceptions** -

**Includes** (other use case IDs)

**Notes/Issues** - Any relevant notes or issues that need to be resolved



### Use Case #3 Chat Box Administrator interface (\*\*Future Development\*\*)

**Author –** Kartavya Damor

**Purpose** - Administrator Interface after login

**Requirements Traceability –** Chatbot with internet

**Priority** - Medium

**Preconditions** - Successful login into the app

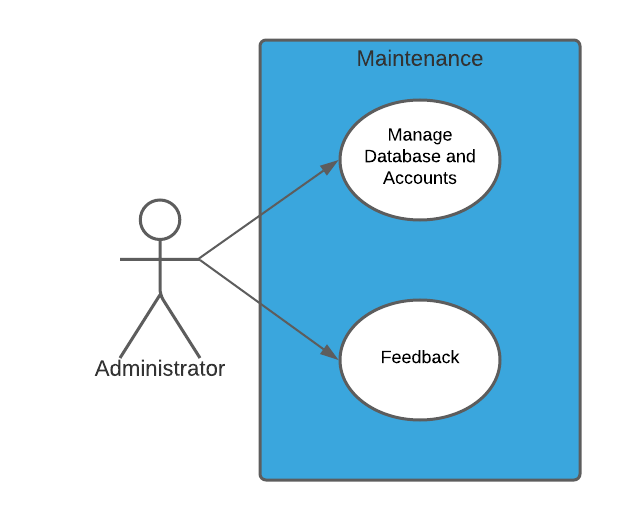
**Post conditions** -

**Actors** – Administrators

**Exceptions** -

**Includes** (other use case IDs)

**Notes/Issues** - Any relevant notes or issues that need to be resolved



# Other Non-functional Requirements

## Performance Requirements

*The back-end should not take more than 1s to deliver a message from one end to another. The UI should be “clean” and easy to use/understand.*

## Safety and Security Requirements

*The system should encrypt the password and then store it in the database. The logging info should be securely stored and the system should not be vulnerable to Cross-Site-Scripting. The peer-to-peer connection should be made securely.*

## Software Quality Attributes

**4.3.1 Security**

The system will save passwords and other sensitive data in an encrypted manner.

It will have non-repudiation ,i.e., authorized users can access or modify their username and interests.

All the data taken will be confidential and integrity will be maintained by encrypting user to user chats as their identities will be anonymous.

**4.3.2 Usability**

The system will have a clean user-friendly UI, similar to that of other chatting apps like “WhatsApp” and would be easy to understand. Also to give the user confidence that nothing inappropriate happens, there will be an option to report the other user. Strict action will be taken if a user receives more than a specific number of reports.

**4.3.3 Adaptability**

The app will run cross platform and will adapt according to the screen size, as we are making the front-end of this web-app using **HTML, CSS** we can make the web-app adaptable for all screen sizes (i.e. smartphone, laptop/desktop etc.) very easily using the **media query** feature of **CSS3.**

**Appendix A - Group Log**

* Group formation – 9 jan
* Idea discussion – 14 jan (4 - 6 pm on discord)
* Idea finalization – 15 Jan ( 5 - 6 pm )
* Meet with TA – 23 Jan ( 4 - 5 pm via zoom meet)
* SRS document meet – 25 Jan (3 -5 pm on discord)
* 28 Jan (4-6 pm)
* 30 Jan ( 7 -9.30 pm)
* Discussion with TA – 30 jan
* SRS document discussion – 01 Feb (7 -8 pm)
* Frequent discussion on discord and whatsapp groups about the project.