



PES UNIVERSITY, BANGALORE

Department of Computer Science and Engineering

# Software Requirements Specification

## Real-Time Chat Application using SocketIO

Prepared by –

- Y Srinivas: PES1UG20CS517
- Vrushank G: PES1UG20CS516
- Akash S: PES1UG20CS534
- Venkatesh BR : PES1UG20CS494

Prepared on : 9/09/2022



PES UNIVERSITY, BANGALORE

Department of Computer Science and Engineering

## Table of Contents

<b>Table of Contents .....</b>	<b>.....</b>
<b>Revision History .....</b>	<b>.....</b>
<b>1. Introduction .....</b>	<b>.....</b>
1.1 Purpose.....	.....
1.2 Intended Audience and Reading Suggestions .....	.....
1.3 Product Scope .....	.....
1.4 References.....	.....
<b>2. Overall Description .....</b>	<b>.....</b>
2.1 Product Perspective.....	.....
2.2 Product Functions .....	.....
2.3 User Classes and Characteristics .....	.....
2.4 Operating Environment.....	.....
2.5 Design and Implementation Constraints .....	.....
2.6 Assumptions and Dependencies .....	.....
<b>3. External Interface Requirements .....</b>	<b>.....</b>
3.1 User Interfaces .....	.....
3.2 Software Interfaces .....	.....
3.3 Communications Interfaces .....	.....
<b>4. Analysis Models .....</b>	<b>.....</b>
<b>5. System Features .....</b>	<b>.....</b>
5.1 System Feature 1 .....	.....
5.2 System Feature 2 (and so on).....	.....
<b>6. Other Nonfunctional Requirements .....</b>	<b>.....</b>
6.1 Performance Requirements .....	.....
6.2 Safety Requirements .....	.....
6.3 Security Requirements .....	.....
6.4 Software Quality Attributes .....	.....
6.5 Business Rules .....	.....
<b>Appendix A: Glossary.....</b>	<b>.....</b>
<b>Appendix B: Field Layouts .....</b>	<b>.....</b>
<b>Appendix C: Requirement Traceability matrix.....</b>	<b>.....</b>



PES UNIVERSITY, BANGALORE

Department of Computer Science and Engineering

## Revision History

Name	Date	Reason For Changes	Version



## PES UNIVERSITY, BANGALORE

### Department of Computer Science and Engineering

## Introduction

### Purpose

A chat application **makes it easy to communicate with people anywhere in the world by sending and receiving messages in real time**. With a chat app, users are able to receive the same engaging and lively interactions through custom messaging features, just as they would in person. Socket.IO enables real-time bidirectional event-based communication. It works on browser, focusing equally on reliability and speed. Socket.IO is built on top of the WebSockets API (Client side) and Node.js. It is one of the most depended upon library on **npm** (Node Package Manager).

We will be using express to build the web server that Socket.IO will work with. Any other node-server-side framework or even node HTTP server can be used. Express JS makes it easy to define routes and other things.

### Intended Audience

Messaging is more widely and frequently used than even social media. Online chats have become an eminent part of both our daily life and business environment. Instant messaging is faster and more convenient than email and less stressful than a phone call, so there is no wonder why we use chat apps so often.

Three-quarters of your customers would rather use web chat and text instead of calling a service provider.

Half of customers would rather communicate with a business using messaging than email. Our chat application is available on browsers. Computers offer lots of screen space where users might be more accepting of information-dense messages with lots of buttons and options. When building a chat app, another key factor to consider is concurrency. Whether a private chat, group chat, or large scale chat experience, being able to build without worrying about user fluctuations and concurrency limits on your platform is crucial. And as your user base grows, choosing a solution that scales with you will ultimately be a benefit as your developers can focus on delivering engaging in-app chat experiences to your users.

### Product Scope

The aim of this project is to build a real-time messaging developer messaging app using modern web technologies.

Unlike most chat apps available in the market, this one will focus on developers and will try to increase their productivity. While we do not expect it to have a lot of resources due to limited time, coding and archive view will be our main features.



## **PES UNIVERSITY, BANGALORE**

### **Department of Computer Science and Engineering**

It will be a completely open source. Everyone will be able to dig into the code to learn what happens after the scenes, or contribute to the source code. It was therefore within our intent to write pure codes, which could be measured by following the most popular patterns and principles of each language and relevant libraries.

#### References

[Introduction](#) | [Socket.IO](#)

[MongoDB Atlas Database](#) | [Multi-Cloud Database Service](#) | [MongoDB](#)

[Documentation](#) | [Node.js \(nodejs.org\)](#)

[Express routing \(expressjs.com\)](#)

### Overall Description

#### Product Perspective

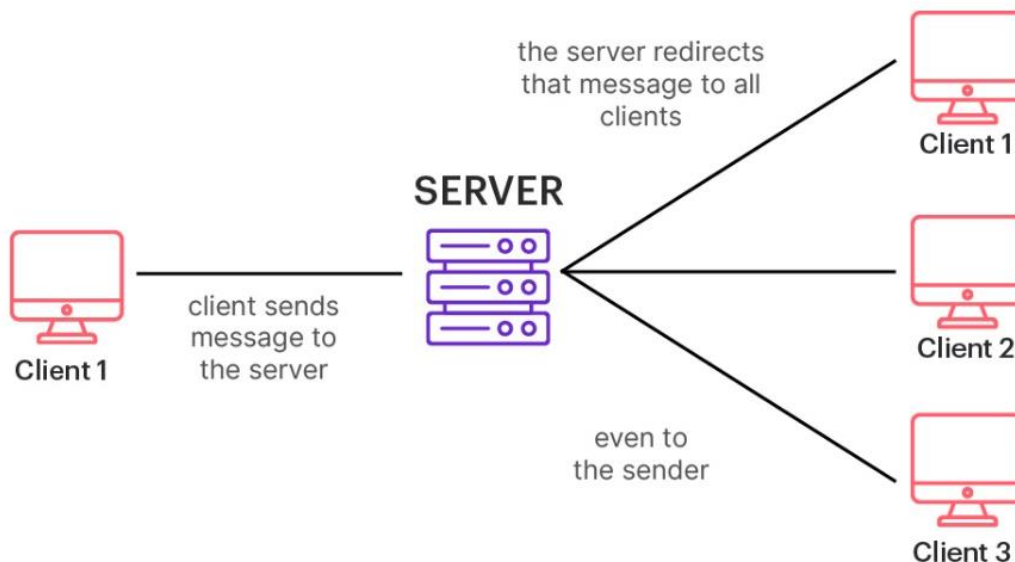
The reason for this application is simple—people love to chat. It's the preferred method of communication in a multitude of different scenarios from collaborating with a colleague to checking in on a loved one. Chat and messaging applications help foster a sense of community and connection that other forms of communication can't reproduce.

System design of a chat app is unique in how it deals with the idiosyncratic business needs, you can always break it down to two major components: the chat client and the chat server.



**PES UNIVERSITY, BANGALORE**

**Department of Computer Science and Engineering**



### Product Functions

People around the world are used to chatting online about everything, including discussing the latest episode with a friend, organizing a team-building meeting, or asking your favourite coffee shop's chatbot about new flavours.

Numbers and statistics are also on the side of chat apps. The overwhelming majority of studies show that instant messaging is becoming more and more popular among users worldwide.

### Chat Features

- Simple Login
- Cloud storage
- Dark and light modes
- Message broadcasting
- **User settings & profile customization**
- **Sharing of media files**
- Cross-platform solution

### User Classes and Characteristics

Each user who logs onto our application with the essential credentials would be able to perform all the functions mentioned above



## **PES UNIVERSITY, BANGALORE**

### **Department of Computer Science and Engineering**

The following aspects of the discussion will be needed to support any basic understanding of the instruction.

1. Application registration page and user verification method
2. Toad message editing field with keyboard
3. Communication window with messages sent and received clearly separated and sorted by chronological
4. List contact list for easy contact import and editing functionality
5. Ability to package and message
6. Ability to receive, translate, and convey a message
7. Notifications, statistics for unread messages, and / or message status (read / unread)
8. Saving past messages
9. User presence indicator (available, available, offline, uptime)
10. Direct messaging: two users can chat with each other
11. Group chat: users can participate in group conversations
12. Join/leave groups
13. Typing indicator: when typing, the recipient gets notified

#### **Operating Environment**

Since we are the Developers and also the Implementers, The Chat application shall function on Normal Laptops[quad-core,i5-processor]. This entails the system to operate on the Windows CE platform. On the Software Side , we will be using MongoDB for database management and using React JS for Front-end experience and using Express JS, Node JS for implementing the Back-end of the application

#### **Design and Implementation Constraints**

The chat client is what the user experiences. The chat client is responsible for interacting with the OS here it's the browser . Interactions include sending push notifications, displaying data to the user and storing messages and files. When you type a message and



**PES UNIVERSITY, BANGALORE**

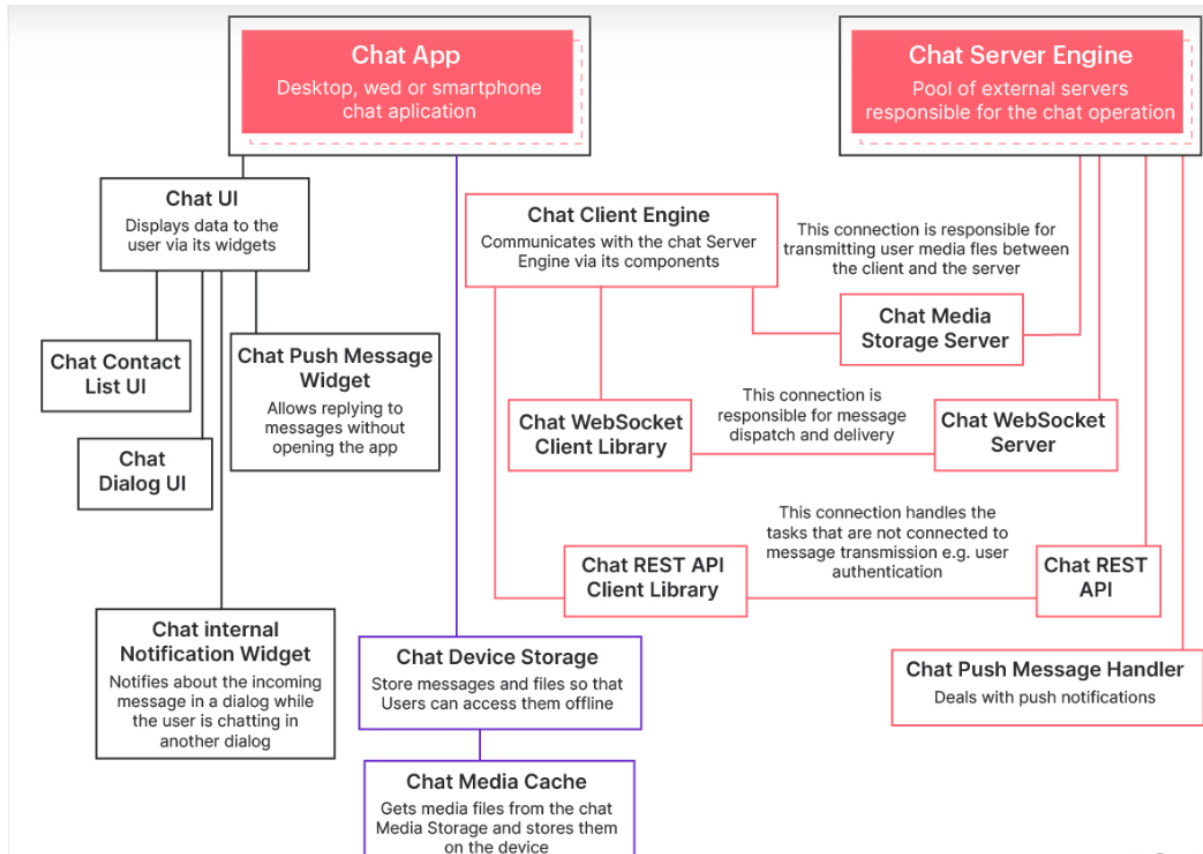
**Department of Computer Science and Engineering**

hit send, the chat client transmits that message to the other major component: the chat server.

The chat server is just that, a server that hosts all the software, frameworks and databases necessary for the chat app to operate. This server, or pool of servers, is responsible for securely receiving a message, identifying the correct recipient, queuing the message and then forwarding the message to the recipient's chat client. The chat server's resources can include a REST API, a WebSocket server, MongoDB Atlas for media storage and cloud platform, etc.

WebSockets are one of many different tools for building web applications that provide instant, real-time updates and communication. The WebSocket Protocol establishes full-duplex, bidirectional communication between a client and server. This two-way flow is unique to WebSocket connections, and it means they can transfer data very quickly and efficiently. While there are many great uses for WebSockets, there are also environments where it will work better to use a different approach, like long polling.





## 2.6 Assumptions and Dependencies

### Assumptions:

1. Using a private network chat system or organizations.
2. Ensuring the security of message and confidential data to be shared over the network.
3. Keeping data confidential in a secure way.
4. Creating a two-way communication system.
5. Allow both group chat and private chat.
6. To allow for easier and faster communication between people.



## **PES UNIVERSITY, BANGALORE**

### **Department of Computer Science and Engineering**

7. Ensure unlimited data transfer without any size limit.
8. Making people connect with others anytime, anywhere.

Dependencies: The following challenges were identified after the development of the program

1. Only registered users can use the program
2. The Internet must be available to use the system
3. No message ordering: If two messages are sent in sequence, it is possible that the first one arrives super late. It will seem to the user that an unread message pops out above the latest message you just read, which is very confusing.
4. Large number of request per second: If each message is sent with a separate HTTP call, then the ingress traffic could overwhelm web socket Servers.
5. If the consumer (web socket server) is down, we don't want the messages to accumulate in the queue since the users will reconnect to a different server and initiate history catch-up. Servers come and go all the time, it is laborous to create/purge queues with them.
6. The failure of databases which are implemented in the project may occur at times due to technical glitches. That's why the assumptions made earlier must be taken care of to ensure smooth working of the software.

## External Interface Requirements

### User Interfaces

*There are few pages in the chat bot website:*

- Login page is used by clients to login
- All members in the database can be viewed by client.
- Log out page is used by clients to log out
- Clients can view previously sent messages
- Home page is used to view menu by clients



## PES UNIVERSITY, BANGALORE

### Department of Computer Science and Engineering

#### Software Interfaces

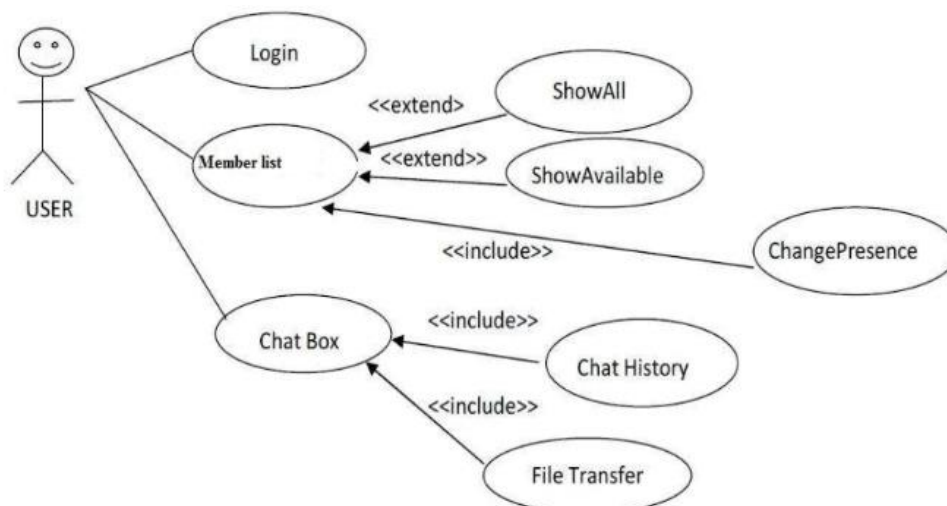
- User can view text sent to members.
- User can send text messages to members via internet.
- Text messages sent to a particular member will be stored in database and can be viewed by client at any time.
- Client can send text messages to multiple members present in database.
- Client can set his application in dark or light mode.
- Client can delete messages stored in database.

#### Communications Interfaces

- In Our Project we are implementing socketio using java script.
- Extensible Messaging and Presence Protocol (XMPP) to exchange data between the users.
- The protocol is decentralized, secure, and flexible
- We use xampp server

#### Analysis Models

Use case diagram



#### System Features



## **PES UNIVERSITY, BANGALORE**

### **Department of Computer Science and Engineering**

Why do people use messaging apps ? Simply because of the following reasons

.

- Real-time text transmission
- Conveniency
- Records of a texting history
- Easy for multitasking
- Operating at any time and from any location using WiFi or cellular networks
- Stickers

The more the features offered , the more people prefer it .

Some basic features include Application must be lightweight and must send messages instantly. Application must work on all mobile and tablet devices. User interface must be consistent on all devices.

### **Functional Requirements :**

#### **1. User Registration**

User must be able to register for the application through a valid phone number. On installing the application, user must be prompted to register their phone number. If user skips this step, application should close. This is of the highest priority as this will help us in knowing about the user and its detail will be stored in the database .

#### **2. Adding New Contacts**

The application should detect all contacts from the user's phone book.

#### **3. Send Message**

User should be able to send instant message . User should be notified when message is successfully delivered to the recipient .

#### **4. Send Attachments**

User should be able to send audio, video and images as attachments. Specifying of what file type can be attached or not should be made clear .

#### **5. Broadcast Message**

User should be able to create groups of contacts. User should be able to broadcast messages to these groups.

#### **6. Message Status**



## **PES UNIVERSITY, BANGALORE**

### **Department of Computer Science and Engineering**

User must be able to get information on whether the message sent has been read by the intended recipient.

The above are some of the basic functionalities of any Chat-Application and hence are of the highest priority in our “ Real Time Chat Application Using SocketIO “ .

Some other Features include :

#### **1. ENCRYPTION & ONLINE PRIVACY**

Starting from terabytes of data sent via your messenger, and ending with confidential data of the large corporations which are loaded, transferred and stored every single day thus, respect for privacy and adequate data protection are critical. Here the End-To-End encryption appears. The initial requirements of the app define Encryption & Online Privacy. Chat programs frequently change their cryptographic keys for further protection. A true backdoor for hackers might be created by a messenger for a bank or cryptocurrency messenger.

#### **2. CLOUD & DATA SYNCHRONIZATION**

Cloud Services are hardly new features which enables storing your files in different places up-to-date. All other devices are updated automatically when you make a modification to a document on one device. As a result, conversation history and data supplied by users may be retrieved at any time and from any location.

#### **3. BOTS**

These are small special features of instant messaging software that are embedded in chats or public channels to perform a specific function.

What is its purpose?

- Self-destructing statuses may be used to manage communities and blogs.
- Automated assistance with registration and other concerns.
- Custom created chat bots using the open-source frameworks, etc.

#### **4. LIGHT MODE AND DARK MODE**



## **PES UNIVERSITY, BANGALORE**

### **Department of Computer Science and Engineering**

These are not only desirable features of good message passing system. Additionally, real improvements in battery life may be gained. Google claims that accessing YouTube in Dark Mode saves batteries by 14 percent when the screen brightness is set to 50 percent. With the screen brightness set to 100 percent, the saving jumps to 60 percent. The harsh bright light of a smartphone can cause discomfort when viewed in a dark room, and the muted aesthetic of a dark UI will not show up as well in a brightly lit area. Even yet, designing the user interface is a creative and aesthetic endeavor in and of itself.

#### **5. DATA PERSISTENCE FOR OFFLINE USAGE**

WiFi and mobile Internet eat your battery. Application with full support of offline mode is escalating at a higher rate. Creating a safe and secure environment that can be used by individuals of all ages and backgrounds is at the heart of the mission statement of these applications. Developing an application that will work effectively even in offline mode or adopt the offline capability will allow people to have smooth experience when the connection is low, slow, flickering or not working. TYour app's offline mode offers several benefits:

- No roaming cost when you are traveling as all the files can be cached
- No monthly data usage on maps
- Quick loading time

Due to time complexity, we will try our best to cover on all the features .

### **Other Nonfunctional Requirements**

#### **Performance Requirements**

PR-1: RESPONSE TIME: The software system shall likely show no deterioration in accessing the required information about the user information and his/her previous records. Any major delay might be possible because of some incompatibility in the socket implementation which might take the software longer than usual time to respond.

PR-2: LOADNIG SPEED : The software has a fairly less probability of taking a unusually longer time to run . There might be some productivity tools or file issues or some problems related to the environment on which the system is running, which may lead to some lags and loading issues.



## PES UNIVERSITY, BANGALORE

### Department of Computer Science and Engineering

#### Safety Requirements

Real-time message synchronization between a user's apps, persistent chats, reliable push notifications for all apps.

Customer data sovereignty, data economy and minimal data usage, no storage of address books, anonymized user IDs (with secure algorithms), data is only processed and stored in the customer's country, data retention guidelines,

Direct integrations to third-party software and services, secured access under administrative control, automation of communication

#### Security Requirements

- **End-to-end encryption:**-have a secure messaging via end-to-end encryption in communication, ensures the encryption of data that has been transferred during the communication. This is from the moment a message has been typed to the time it reaches the receiver, here no one else can be able to view the data, either the app maker or internet service providers or government organizations ensuring security.
- **Multi-mode communication:**-Getting bored by communicating with text messages, you can also share media. Whenever a platform allows the users to send across messages through multiple modes, it automatically permits the user to delete the messages if they wish to, enhancing the security.
- **Supportive to Multi-platform:**-You can be able to synchronize the messages across any platform like web app. However, it allows the user to store their private messages in any of their desired locations and also permits them to delete from other locations, if they are not feeling comfortable. Thus, this adds up a great impact to one's data privacy ensuring security.

#### Software Quality Attributes

**Availability:** The system shall be available to all users who registers and logs in to view the chat history.



## **PES UNIVERSITY, BANGALORE**

### **Department of Computer Science and Engineering**

**Reliability:** Due to the use of a wireless network, reliability of the system at all times is not guaranteed. However, overall reliability of the system and roll information shall be achieved through the process of database manipulation.

**Reusability:** The system must be reused for each new user.

**Updatability:** The system shall allow for addition or deletion of users on the basis of account manipulation

#### **Business Rules**

- Over the last few years, customers have started using chat apps and other messaging tools to request information from businesses
- Many companies across industries can improve customer experience by using chat software. E-Commerce startups love using chat software to keep customers informed of the latest deals and help them with their requests.
- B2B SaaS enterprises also can benefit from using a chat app to boost communication with businesses that purchase or deal with them.
- The primary purpose of this technology is to provide prospects or customers with direct one-on-one support without disturbing their browsing flow. Research has shown that many clients are now turning to live chat and social messaging apps for customer support.

## **Appendix A: Glossary**

SRS: Software Requirement Specification

RAM: Random Access Memory

OS: operating system

DB: Databases

HTTP: Hyper Text Transfer Protocol





**PES UNIVERSITY, BANGALORE**

**Department of Computer Science and Engineering**

## Appendix B: Field Layouts

An Excel sheet containing field layouts and properties/attributes and report requirements.

**Sample sheet with information required to register the customer**

Field	Length	Data Type	Description	Is Mandatory
Account Number	16	Numeric		Y
ISFC code	11	Alphanumeric		Y
Card Amount	20	Numeric		Y
Mandate Start Date	8	Date	Date of Mandate Registration	N
Mandate End Date	8	Date	Date of Mandate Expiry	N
Status	25	Alphanumeric	Status of Registration	Y
Customer Name	60	String		Y
Reject Reason Code	4	String	Reject Reason code in case mandate is rejected	N

**Sample Report Requirements: Include the fields to be included in the report**

Registration Report	Transaction Report
Bank Account Number	Transaction Reference Number
ISFC Code	Bank Account Number
Bank Name	IFSC Code
Account Status	Bank Name
Account Type	Customer Name



**PES UNIVERSITY, BANGALORE**

**Department of Computer Science and Engineering**

Customer Name	Card Number
Card Number	Debit Transaction Amount
SI Start Date	Transaction Date
Status	Status
Remarks	Debit Attempt Number
	Remarks

## Appendix C: Requirement Traceability Matrix

Sl. No	Requirement ID	Brief Description of Requirement	Architecture Reference	Design Reference	Code File Reference	Test Case ID	System Test Case ID