Software Requirement Specification

for

CO-convo-EP : A secure chat application for colleges

vesrion 1.0 approved

Prepared by:

1) Prajjwal Datir - 111803131
2) Ganesh Gitte - 111803139
3) Aditi Medhane – 111803177

College of Engineering, Pune.

Date : 21st February, 2021

1. Introduction 1

1.1 Purpose 1

The purpose of the Co-Convo-EP application is to allow Students And Teachers to be able to communicate in an effective manner to achieve maximum productivity. It'll have features such as users will be able to chat with each other, user to user as well as group communication will be possible.

This application will be focused to make the life of college students and teachers easy, productive and is available to be used on different devices whichever user prefers.

We will try to overcome different limitations of available chat applications and provide a secure and optimized chat application for college students and teachers. Providing features like real-time messaging, including text messages, images, documents, etc., creating and managing groups with group category folders and a tree like structure to effectively navigate and visualize the groups.
We'll be discussing SRS in depth in the following points.

1.2 Document Conventions 1

UI : User Interface
UX : User Design
MERN: MongoDB, Express.js, ReactJS, NodeJS Frameworks
PWA : Progressive Web App
User : Person who is using the web-app on client side
Font : Times
Signal Encryption : Encryption algorithm of the Signal app
Full Stack : Tech stack which includes Front-end, Back-end and Database.

1.3 Intended Audience and Reading Suggestions 1

- Developers Class : Developer willing to build projects in Javascriptstack.
- User Class : Students in Universities
- Faculty Class: Faculties in Universities

1.4 Product Scope 1

Our web app will be made specifically for Universities & Students, to make their life more organized and productive. We'll provide features unlike any other existing chat apps. Our key features include Tree View & Filtering of Official Groups, etc. It'll also have existing features of Chat apps like one-one chatting as well as Group chat but in an organized manner.

Our App will be Free and Open Source Software supporting Free software ideology. Teachers will have extra features to create an unofficial group by themselves and tag it to a year and branch. While students can only Join/View Official groups and can create(admin)/edit(admin)/view/join the Unofficial groups.

1.5 References 1

[1] Nikolaos G. Bardis Hellenic Army Academy,"Design of a Secure Chat Application based on AES Cryptographic Algorithm and KeyManagement" ,2008

[2] Jeeun Lee, Rakyong Cho, Sungsook Kim, Kwangjo Kim, "Security analysis of end-to-end encryption in Telegram", 2017

[3] Hayk Saribekyan, Akaki Margvelashvili, "Security Analysis of Telegram ", 2017

[4] Heidi Hemmer, Minnesota State University, Mankato, "Impact of Text Messaging on Communication", 2009

[5] Naveen Kumar, Sudhansh Sharma, "Survey Analysis on the usage & Impact of Whatsapp Messenger", 2016

[6] Jamal Mohamad, Noor Sabah Jamal Mohamad,

"Developing an End-to-End Secure Chat Application", 2017

2. Overall Description

2.1 Product Perspective

Co-Convo-EP is a secure chat application designed for Colleges, Schools and other Educational Institutes to increase simplicity and productivity of users. It could be useful when a user needs to exchange information as quickly as possible in an efficient and smart way.

2.2 Product Functions

- Personal and Group chatting – Users will be able to chat with people in their contact list personally as well as in a group of people created by the user.
- Media sharing – Users will be able to share different media like documents, images, etc. through personal or group chat.
- Tree View – Users will get a tree-like interface to easily navigate to the desired

group saving a lot of time and increasing productivity.
- Group Categories – The user can divide the groups into different categories for different teams or occasions or to create a workflow.
- Group Locking - The user can lock the specific set of groups to avoid accidental messaging in Official Groups.

2.3 User Classes and Characteristics
- Typical users who will be benefited from this project will be from educational background. Ex. Faculties and Students
- Programmers who are exploring about Socket.io & Javascript framework

2.4 Operating Environment
The application is designed to work on any Android or IOS supporting Internet Explorer 9 and above(inclusive of all parallel versions of Chrome, Firefox, etc.)

2.5 Design and Implementation Constraints
Co-Convo-EP is a progressive web application developed using MERN stack. Real-time communication required for communication of users will be done using socket.io.
Frontend is done by using the open source ReactJS library from facebook with Bootstrap for some of the components and CSS to style the interface.For backend ExpressJS framework will be used which provides simple API's for building backends along with NodeJS which is event-driven JavaScript runtime. Signal Encryption Protocol is used for encrypting messages providing data security.

2.6 User Documentation
- User : Will be provided on the Home Page
- Developer : An elaborative README will be provided which will describe system architecture and other docs. Along with that GitHub Wiki will be provided for reference.

2.7 Assumptions and Dependencies
- Users must have an active internet connection.
- Users must have a latest version of Web Browser on their computer/android/ios.

3. External Interface Requirements 3

3.1 User Interfaces 3
- Web Browser :
  As Co-Convo-EP is a Progressive Web application, there is no need of dependencies except for a web browser

3.2 Hardware Interfaces 3
- Android/ IOS : As the web application can be useful as a mobile application as well.The general system that can run a web-browser is enough.
- Keyboard

3.3 Software Interfaces 3

- **React :** Front-end JS library to be used for frontend work
- **MongoDB-Atlas :** To be used to store databases
- **Socket.io :** To be used to integrate live communication and to provide a environment along with peers
- **Express :** Back-end JS library to design web application and APIs

3.4 Communications Interfaces 3

- **HTTP** : A secure HTTP will be used to communicate between client and server which will ensure that the data is encrypted. Encryption will be done with the help of Signal encryption Algorithm.
- **SMTP** : This protocol will be used for user email verification while registering.

4. System features

Priority 1 is lowest and 10 is highest

4.1 Chatting

4.1.1 Description and Priority

Chatting is the main feature of our application. Our app being a progressive web app will allow users to use chatting features from any smartphone, tablet, laptop or desktop on any operating system MacOS, Windows, Linux, Android, ios, etc. Sharing of text messages, different media will be done through this feature.

Priority : 10

4.1.2 Stimulus / Response sequences

1. First user logs in or creates an account
2. Select the group / person they want to chat with
3. Type message in the message box or select media to share
4. Click on send button
5. The message will be encrypted and sent from sender to server and then to the receiver

4.1.3 Functional Requirements

Users will be able to chat with one another by using the chatting feature

4.2 Tree view

4.2.1 Description and Priority

While using different chatting applications we often find them not organized as well as not optimized for college / school use. This feature organizes the official college groups in a tree-like structure increasing productivity for students as well as teachers.

Priority : 9

4.2.2 Stimulus / Response sequences

1. Logged in user will see a button at the bottom of the home page
2. After clicking on the home button user will be able to see tree-like arrangement of buttons of group folders
3. Users will be able to click on these buttons to reveal containing group buttons and the groups themselves
4. The tree-like structure will take users to their desired group/chat

4.2.3 Functional Requirements

The software will provide users with a tree-like interface to easily navigate to the desired group saving a lot of time and increasing productivity.

4.3 Group category folders

4.3.1 Description and Priority

Usually the groups, personal chats are all in the same list in many chat applications. The group category folders feature classifies different groups into an official group, unofficial group or any other user defined group folder. This helps to maintain a productive workflow.

Priority : 8

4.3.2 Stimulus / Response sequences

1. Logged in users will create group folders with name of choice
2. Users will add all the related groups in these folders
3. The group folder button will appear in tab navigation
4. On clicking the group folder all groups in that folder will appear as a list
5. User will be able to switch between different group folders

4.3.3 Functional Requirements

The user can divide the groups into different categories for different teams or occasions or to create a productive workflow.

4.4 Security features

4.4.1 Description and Priority

Privacy is a major concern to many people using any application, especially social apps. So, there are security features to protect user privacy like the signal encryption, locking a group or personal chat so unintentional and unwanted sharing can be avoided.

Priority : 10

4.4.2 Stimulus / Response sequences

1. When users are using the application for chatting the sent text and media are end to end encrypted
2. They are decrypted at the user side only to protect privacy.
3. To share something on the locked groups users will be asked their password and confirmation to avoid unwanted sharing.

4.4.3 Functional Requirements

The user need not worry about the privacy and data concerns and will be assured about any messages sent to official / locked groups.

5. Other Nonfunctional Requirements 5

1. Scalability :

Webapp should be able to provide instant messaging services to all users at any given time.

2. Privacy :

Messages shared between users should be encrypted to maintain privacy.

3. Performance:

Applications must be lightweight and must send messages instantly.A bug tracker will be available where users can report any bugs they have encountered so that can be fixed in the next release.

4. Security Requirements :

Users shall create a username that stores their personal progress profiles. A unique username and password should be assigned to the user. The password should be hashed using the particular technique to ensure security

6. Other Requirements 6

1. **Internet** - User needs to have internet connection to be able to send text messages and share media to other users and groups

Appendix A: Glossary 5

MERN - technology stack MongoDB, Express, React, NodeJS
MongoDB - NOSQL database
Express - Web application framework
React - a javascript library for building user interfaces
NodeJS - backend javascript runtime environment
PWA - Progressive Web Apps a type of application software delivered through web

Appendix B: Analysis Models 5

Appendix C: To Be Determined List 6