Instant Messaging Application to share messages using Image Steganography

Prudhvinath Reddy Katha
Computer Science
Kent State University
Kent OH
pkatha@kent.edu

Madan Kumar Sutapalli
Computer Science
Kent State University
Kent OH
Msutapal@kent.edu

Vijay Durgesam
Computer Science
Kent State University
Kent OH
vdurgesa@kent.edu

Supervised By:
Maha Ali Allouzi, PhD
Computer Science
Kent State University
Kent OH
mallouzu@kent.edu

Abstract— Secure chat is a new way communication application that focuses on storage, security along with Image Steganography. We as a team had a good understanding that we should not create another chat application but make it stand out of the existing one. We first focused on building a neat interface that would make the users feel free and usable to chat. We took some inspiration for the designing of UI from different existing applications and able to get a powerful and effective UI using Angular, JS, Firebase. Each of them has their own working, we tried to combine each of them to get a web Secure Chat application. One of the best features included in the project is Image Steganography function, it gives a new concept of hiding text under simple and ordinary images. It is a best way to secretly send messages to your friend or colleague in an image. After developing this Secure Chat, we tested it with some real time users like friends, and the feedback received was very much positive, we had an almost 85% acceptance rate for the smooth flow of application, no lags in message transfers. The UI was attractive and was clear with instructions, text encoding decoding in images stands out in this project.

Keywords – AES (Advanced Encryption Standard), Image Steganography, Encryption, Decryption, security, Angular, Firebase, storage, Node,js

I. INTRODUCTION

SECURE CHAT is a social networking platform that uses technological advancements to enable text, audio, and image sharing and connection between users. Present-day initiatives provide several features, including voice and text messaging, audio and video chats, location sharing, the ability to share updates and images, pidgin English local networking, games, and financial transactions, that make it simple to remain in contact with people you know. In conclusion, Secure Chat provides a comprehensive solution for communicating with others and preserving relationships.

Mobile devices running iOS and Android may easily share files thanks to the SECURE CHAT app's design. Because the mobile device acts as a host for file downloads, the app eliminates the necessity for file uploading and downloading. The goal of SECURE CHAT's Pidgin English function is to foster friendship among people of all socioeconomic classes by dismantling obstacles to social interaction. Regardless of religion or nationality, the app encourages equality and togetherness among all people.

Secure communication is now essential in a world where connections are growing. Data privacy and protection are essential, whether it is while sending private communications, sensitive information, or secret papers. A potent method that fulfills this need is picture steganography. Steganography, as it relates to digital photographs, encodes data or hidden messages into individual pixels. The objective is to maintain the overall visual look of the image while making the adjustments undetectable to human observers.

A. PRIMARY AIM – SECURE CHAT & STORAGE

Instant messaging applications are widely used, but the security of message transmission often faces threats. Although end-to-end encryption provides a solution, its implementation in Android or web-based messaging apps always poses challenges. This project aims to develop a comprehensive end-to-end encryption framework for messaging web applications. It will incorporate features such as End-to-End encryption, and AES cryptography algorithm to ensure security and efficiency. Furthermore, the project will enhance security by integrating Image Steganography to share confidential messages securely.

In chat applications, storage is crucial for the preservation of message history, media files, user data and backups. In order to maintain the performance and scale, efficient database management is essential. Storage space may be quickly exhausted by large amounts of data, in particular media files and message history. A smooth user experience and secure data storage is ensured by effective management of the database. In order to prevent the loss of data, it is also essential to provide redundancy and redundancies. In the current project we were able to work on storage more effectively by using cloud google firebase as a database for storage of logins, messages, images etc.

II. LITERATURE REVIEW

In today's world of digital things, having secure communication with others has been a challenging task. Using this algorithm, they encrypted messages by hiding the original text. You are chatting with your friend in a message application; you will think that the chat between you and your friend is secure and can be read by only you people. This means the third party should not have access to your messages, that's where this end-to-end encryption comes into play. In recent years (2022) Nurhayati, Kastari, and Fahrianto few researchers have been working to see that the messages are encrypted. They used algorithms like AES cryptology for encryption and decryption. Using this algorithm, they were able to encrypt messages by hiding the original text. That text can be seen by the intended person who has the decipher key. Why use only text to hide text, researchers like Subramanian in 2021 came up with the concept of Image steganography which it acts as an extra layer of protection to read a message or log in to a system. Let's say you send an image to a user; others can think that you sent an image but if you embedded a message, they would never guess it. It has a secret message code, only you can get the message by decrypting the image. In 2023 Chandani and Sharma focused on data transmission between users. Even if we encrypt a message, it cannot entirely give security to the application and the data we transfer. So, they developed cryptography where everything is encrypted, like the messages and users, for secure data transfer. By focusing on all these things, we can ensure that the communication between you and your friend is secure and can chat peacefully in this digital world.

III. REQUIREMENT ANALYSIS AND DESIGN

A. *Overview*

This paper provides standards for planning and creating a chat application, covering both functional and non-functional needs. The project aims to produce a user-friendly chat program. Allows users to converse using instant messaging technologies. While creating the program, we also focused on security options. Security is essential in every social media platform. The program will have a angular based user interactions and provide security as needed.

B. *Requirement Specification*

Requirement	Description	Priority	Actor
Privacy	Ensure User privacy	High	Admin
Robustness	Handle error robustly	High	Admin
Performance	Maintain application performance	High	Admin
Usability	Ensure easy access	High	Admin
Reliability	Build trust with users	High	Admin
Supportability	Provide support	High	Admin

Non-Functional Requirements

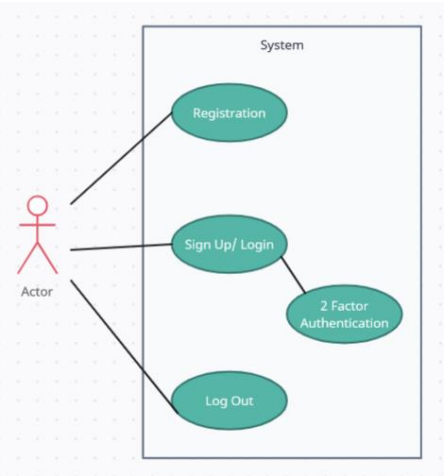
Requirement	Description	Priority	Actor
User Registration	Allow users to create an account	High	User
Login/Logout	Enable access to user accounts	High	User
Add Friend	Provide functionality to add friends	Medium	User
Friend List	Display list of friends in chat	High	User
Searching	Search/find friends or messages	High	User
Profile	User profile with name, picture, DOB, etc.	Medium	User
Send Message	Ability to send messages	High	User
Message Status	View the status of sent messages	Medium	User
History	View past messages	High	User

Functional Requirements

C. *Use case Diagram*

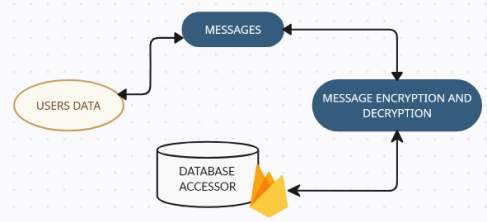
The user who is able to login or log out of the chat application is the primary actor in the use case diagram. The user can see the chat, send messages to other users as well as modify or delete any message they've sent after logging in. You can also receive messages from another user and see the details of those messages using a chat application. Some use case diagrams can be found below:

Authentication Service: In the picture that shows how the chat app works, the main person is the User. They can log in or out of the app. After logging in, the User can see their chats, send messages to others, and change or delete the messages they've sent. The app also lets the User get messages from others and see the details of those messages.



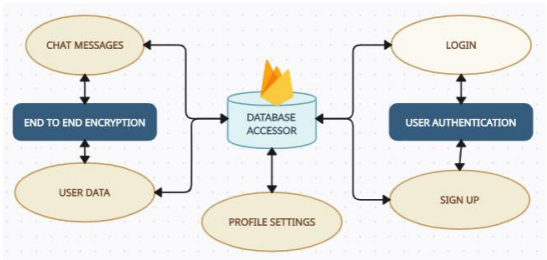
Authentication Service

End to End Encryption: By scrambling your messages on the device before sending them, EndToend encryption will keep you private. They can only be unscrambled and read by the person you're talking to. That means that when they travel, no one else will be able to see your messages except the app developer or anyone in between. You and the person you're communicating need a specific secret code, which is only known to you two. One of the most important security features that helps keep your conversations and conversations private is Endtoend encryption. It is therefore widely used by chat applications and the rest of the platforms for communication.



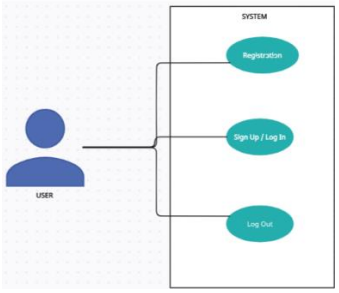
End to End Encryption

Database Connection: The database is like a big storage room where we keep all the important information in our application. We use Firebase, a cloud service that stores this data. A simple diagram showing how we're handling information from the User Interface UI, and the steps we're taking to ensure that the information is kept safe by encrypting it, and that only authorised persons are allowed to access it.



Different component connection to database

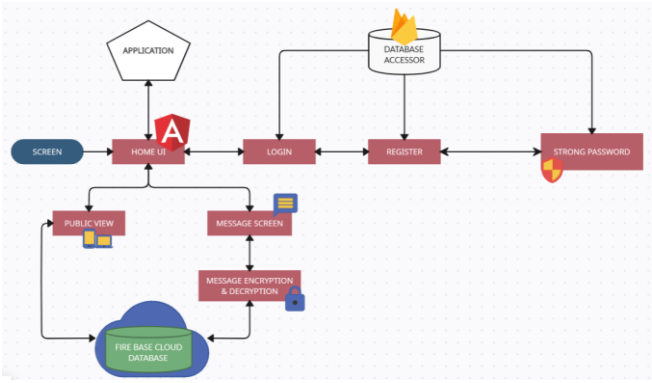
User workflow: The next Use Case Diagram gives us a good idea of what the chat is like, The services that you can enable such as searching user, sending user, Get the message history when you remove the user.



User workflow

D. System Design

Chat Application Architecture: The architecture diagram shows how the chat app's different components work together. Angular, which communicates with Firebase for real time messaging and authentication, is used to create the user interface. Firebase's NoSQL database stores the app's data for quick and easy retrieval and manipulation. Generally speaking, this architecture produces a scalable and secure chat application that is ideal for organizations with high security needs.



Sample Chat Application architecture

Image Steganography: We need an image and text to make this Image Steganography work, we first encrypt the image with the text and then decrypt it using secret key.

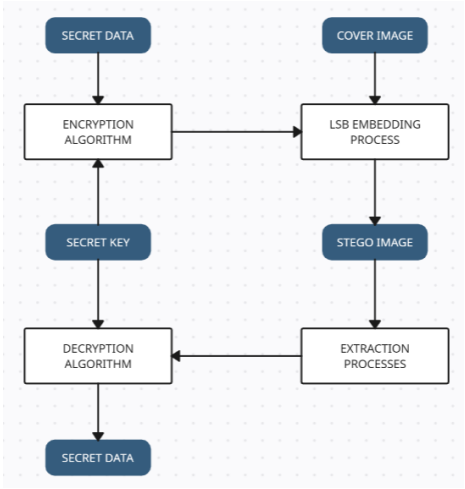


Image Steganography architecture flow

IV. IMPLEMENTATION

The specifications needed for the design and development of a chat application, including both functional and not functioning requirements, are set out in this document. The primary objective of the project is to build a user-friendly chat application that allows you to communicate with each other through instant messenger technology. In developing the application, security options are also taken into consideration. In any social media application, safety plays a major role. The

application shall consist of a server which facilitates communication between users and, where necessary, provides security. Communication through point to point link between two users will be facilitated by the chat server application, which facilitates text communications. By allowing features such as End to end encryption, user anonymity, user authentication etc. the application will focus more on data security for users. Some external features are to search messages, upload photos, profile updates, Image Steganography, third party encryption. App to be designed in such a way where system never learns anything from user messages. And also planned to implement a secure messaging through images where we try to include text in an images binary code and get the text by intended recipient by decrypting the image.

A. *Methodology*

It can be challenging to develop a chat application that is both secure and efficient and also a focus on storage. However, by using Angular and Firebase, developers can create a chat app that is not only fast and responsive but also highly secure. Angular is an appropriate platform for building modular, dynamic web applications and Firebase provides a cloud backend service that offers real time data synchronization, authentication and storage. The user interacts with the tool using GUI. There are two ways in which the GUI works, User forms and chat forms. The User forms contain the list of all added chat users and the chat form will be used to chat with added user (friend). Also, a new function which uses regular concepts for image steganography.

We have divided this project into two major steps

Frontend – GUI:

Developed the user interface by using angular where we had multiple components like home-page component, sign-up component, profile-page component, landing-page component, environments etc.

Backend – database:

Firebase is the backend support. We have successfully integrated the database to connect with the components to use some services like authentication, storage, database querying many other features.

B. *Focusing on Storage and Security in Secure Chat Application*

Storage is a factor to be consider because we will be working with large amounts of user data which needs security, Cost of local storage devices is costly, it also effects the performance. Here we achieved it by using Firebase which stands as the backend database and also large storage block to store images, files, user chats, user logins etc.

Many **security** safeguards are often included in any chat program to protect the user's data. To protect against threats, it is possible to achieve several of them with the present CHATTER application. Here we achieved it by using firebase authentication service, and data encryption by using AES algorithms for encryption and decryption, Image Steganography for image encode and decoding process etc.

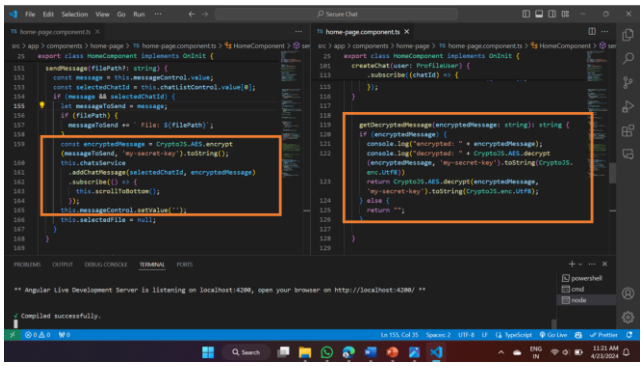
The application provides the following basic security features:

- End-to-end encryption: This process encrypts messages before they are sent and only decrypts them on the recipient's device. Consequently, during the transmission or reception of messages, no one can see them other than the chat application's provider.
- Two factor authentication: In addition to the password, users are required to provide a second form of identity, such as a code sent to their email or an authentication app, in order to use this additional security measure.
- Data security: The application uses encryption, which can prevent unauthorized access or data breaches, to store user information safely.
- Safe connections: Chat programs should use safe connections, such as HTTPS, in order to avoid eavesdropping and more threats.
- Secure log-in and log-out processes are made for the app to prevent unwanted access to user accounts.
- In order to protect user information and communications from unauthorized access and attacks, these security elements work together. In order to ensure that they adapt to the latest threats and vulnerabilities, it is important to focus on security and to upgrade security patches with technology.

C. *AES Algorithm – Encryption & Decryption*

The well-known JavaScript package CryptoJS offers a variety of cryptographic methods, including Advanced Encryption Standard AES encryption. Using the AES symmetric key encryption technique, sensitive data are often encrypted and decrypted. The strength and security of AES encryption is one of its major features. AES employs a block cipher that separates the data into fixed-size blocks and then encrypts each block independently. In addition, it uses a secret key to encrypt and decrypt the data in order to guarantee that authorized persons are able to access this information. With CryptoJS, AES encryption can be easily implemented in JavaScript applications. To encrypt and decode data, the library provides a simple interface to use AES with various keys sizes, modes or padding properties. This enables developers to integrate strong encryption into their applications, thus helping prevent unauthorized access and the interception of critical data. As a matter of fact, AES encryption with CryptoJS is a strong security feature that can help protect sensitive data in JavaScript applications. By introducing AES encryption into their

applications, developers can create an additional layer of security to help guarantee the confidentiality and integrity of user data.



AES - Encryption & Decryption logics

D. *Image Steganography*

The steganography technique is an essential tool for digital communication, improving the security of data transmission processes. It uses the human eye's inability to detect microscopic changes in pixels that are of less significant importance so as to hide sensitive data from unauthorized access. In digital communication, where RGB images contain a number of color channels like Red, Green and Blue, it is especially useful to use steganography. Data protection professionals can mitigate the risk of data leakage and unauthorized access, ensuring that digital communication is secure by using cryptography and encryption technique.

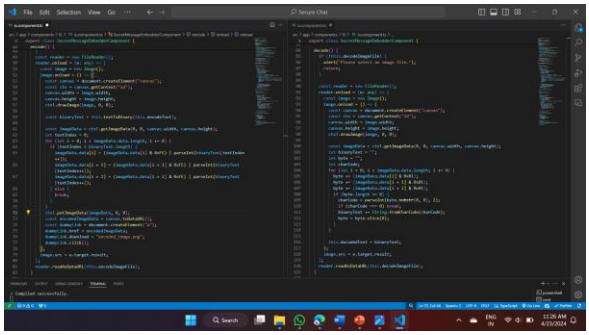


Image Steganography – encode & decode logic

The LSB steganography is a method for improving data security by combining AES cryptography with minimal encryption. AES converts plain text secret messages into cipher text, ensuring confidentiality and protection from illegal access. Encryption of data and hiding in image formats are improved by the Quick Response code. LSB steganography discreetly inserts encrypted data into cover picture pixel data, increasing obscurity and enhancing encryption.

E. *Used Tools & Technologies*

Database Used: Firebase

Firebase is like a big electronic filing cabinet where you can store all sorts of information. It's like a digital warehouse where you can keep your data safe and organized. You can think of Firebase as a cloud-based storage system that allows you to store your data securely on the internet.

Programming Language: Angular, JavaScript, HTML and CSS, NoSQL

Angular, JavaScript, HTML, and CSS are like the building blocks of a website or web application. They are the languages and tools developers use to create and design the look and functionality of web pages. It's similar to using different tools to create a painting – each language or tool has its role in making the final product.

Tools & Libraries: Angular, VSCode, Firebase, Browser, Crypto JS

Angular and VSCode are like the toolbox and workspace for developers, helping them write and organize their code efficiently. Firebase is a toolkit for storing and managing data, like a digital Swiss Army knife. The browser is like a window into the digital world, allowing users to access and interact with websites and applications. Crypto JS is like a secret decoder ring for encrypting and decrypting data, keeping it safe from prying eyes.

Build & development: Node.js

Angular does not require Node.js directly, and it is not mandatory to use Node.js. But for all building and development tools, you'll need Node.js.

In our project we used node.js for the same to build and develop application. We have installed angular command line interface using the same.

F. *Challenges & Limitation*

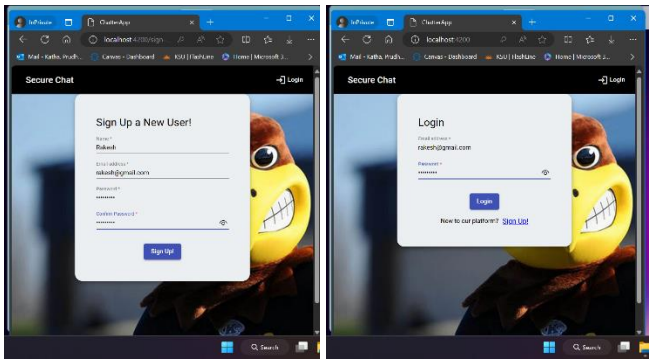
Main challenges faced during the project was connecting angular and Firebase since there were most of the version dependencies, API connection errors, routing module errors. Working with large amount of code often brings confusion, even simple things are changed it effects application running. This application is limited to some features only, these are some of the limitations of this secure chat application

- The application needs internet to run (to send message, store in online database). which may be a barrier for those with limited internet access.
- There are lot of dependencies which needs an additional support of Node.js
- Even provided with most security features, there is always room for some security concerns like having weak password, unauthorized login, defected systems etc.
- This is sample web application is not included with any audio video calling functionality

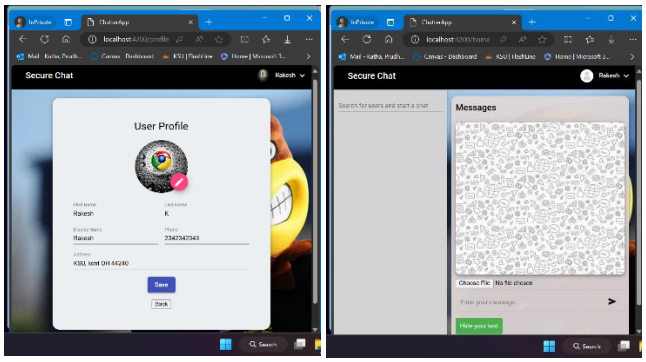
- Secure Chat may not provide the same functionality as email or video conferencing. File sharing is also not accepted.
- Secure Chat application lacks emotional indicators, such as voice tone, facial expressions, and body language, that are present in face-to-face interactions. This might make understanding the message difficult and lead to misunderstanding.

V. OUTPUTS

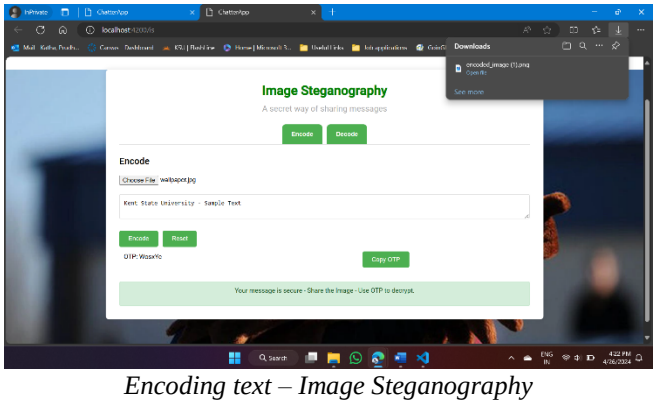
GUI of different screens is provided here, each of them is implemented as components



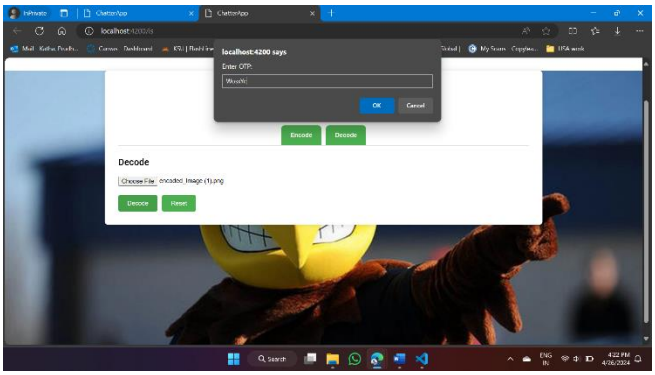
Sign up and login screens



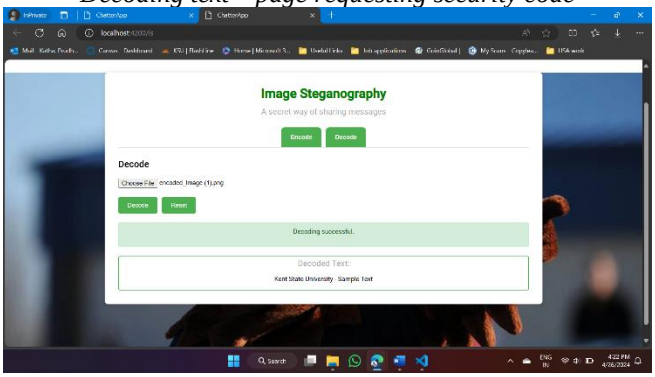
User profile screen and chat screen



Encoding text – Image Steganography

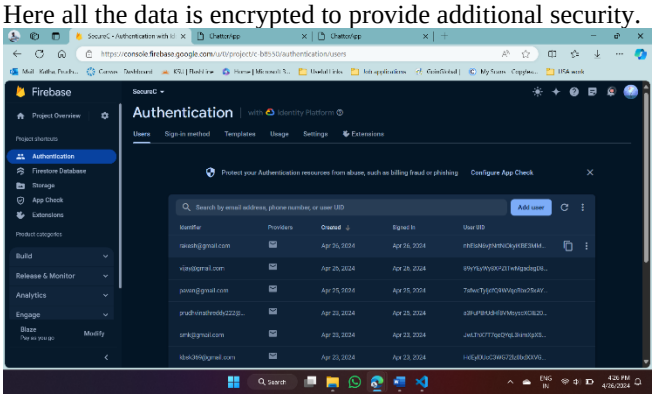


Decoding text – page requesting security code



Successfully decoding text

Database: We have used google firebase for our database support where all the images, chats, user data are stored. Here all the data is encrypted to provide additional security.



Authentication – signed up users

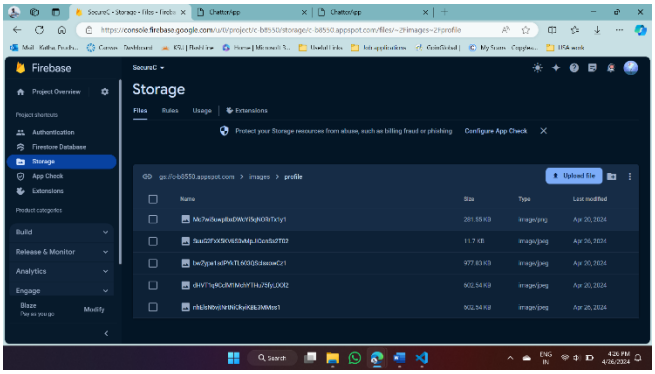
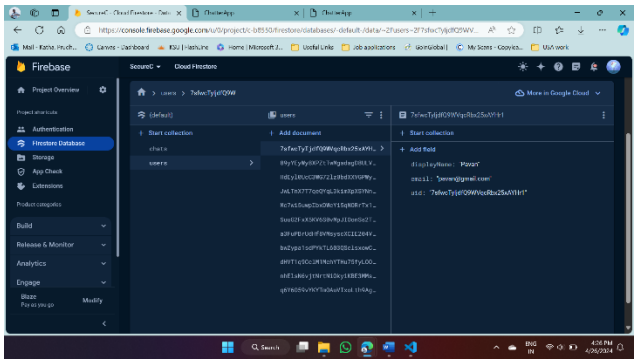
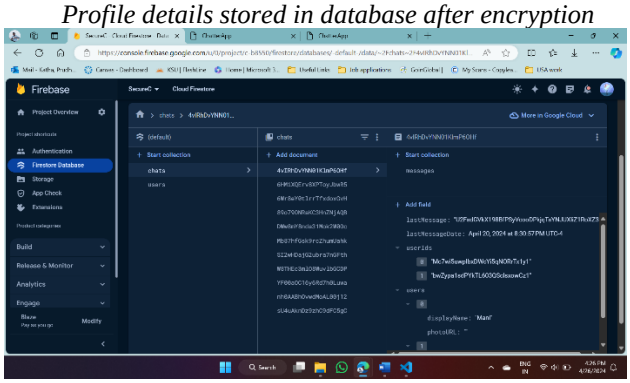


Image Storage – profile and chat images



Profile details stored in database after encryption



Chat data stored in database after encryption

VI. FUTURE PLANS

We can make additional functionalities in future for this Secure chat web application. Some of them are adding voice/ video calling functionality, image editing function, sending messages in different font styles, emojis etc. Animation to GUI, allow users to share large size data files with ease. Creating groups, Work on Image steganography like adding extra layer of protection to decrypt a message (like giving a QR code to scan the image verify the user and then decrypt the image to get the hidden text). Allow access to share multiple images at once. Camera access to take instant pictures and send in chat, news feed option to get more access to the world.

CONCLUSION

During the design phase, the project has successfully fulfilled all user requirements and prioritised data integrity and avoidance of redundancies. Team successfully developed a chat application along with an added feature of Image steganography. The interface is user friendly, with technical details and interactions aimed at making users feel comfortable using the system. A demo is provided, although this may not be necessary in view of the user oriented approach. The system is also flexible, which allows for changes to be made without having an impact on the functionality. The app is compatible with any version of the android operating system, making it available to users that have a different level of device capability. Our main focus was on storage and security. We achieved it by using firebase as our storage, authentication service and AES for security of messages from both ends.

REFERENCES

[1] *Nurhayati, Kastari and F. Fahrianto, "**End-To-End Encryption on the Instant Messaging Application Based Android using AES Cryptography Algorithm to a Text Message**," 2022 IEEE, 10th International Conference on Cyber and IT Service Management (CITSM), Yogyakarta, Indonesia, 2022, pp. 01-06, doi: 10.1109/CITSM56380.2022.9935963.* e2ee_documentation.pdf

[2] *Neogi, Pinaki Prasad Guha. "**A Dive into WhatsApp's End-to-End Encryption.**" arXiv preprint arXiv:2209.11198 (2022).* WhatsApp's E2EE.pdf (arxiv.org)

[3] *Williamson, J. a. (2021). **The Role of Multi-factor Authentication for Modern Day Security.** Semiconductor Science and Information Devices, 03(01). Retrieved from* Multifactor_authentication.pdf (researchgate.net)

[4] *Maliheh Shirvanian and Shashank Agrawal. 2021. **2D-2FA: A New Dimension in Two-Factor Authentication**. In Proceedings of the 37th Annual Computer Security Applications Conference (ACSAC '21). Association for Computing Machinery, New York, NY, USA, 482–496. Doi:* https://doi.org/10.1145/3485832.3485910 2FA_ACM_documentation.pdf

[5] *N. Subramanian, O. Elharrouss, S. Al-Maadeed and A. Bouridane, "**Image Steganography: A Review of the Recent Advances**," in IEEE Access, vol. 9, pp. 23409-23423, 2021, doi: 10.1109/ACCESS.2021.3053998.* Image_steganography_IEE

[6] *P. Chandani and M. Sharma, "**Secure Data Transmission using Cryptography for Internet of Things and Sensor Networks Applications**," 2023 2nd International Conference for Innovation in Technology (INOCON), Bangalore, India, 2023, pp. 1-7, doi: 10.1109/INOCON57975.2023.10101069.* Secure Data transmission

ADDITIONAL IMAGE REFERENCES

- *Chat Application System UML Diagram | FreeProjectz*
- *Instant Messaging System UML Diagram*
- *Google messages prepares to end-to-end encryption of RCS messages - AndroidNox*