VIVEKANAND EDUCATION SOCIETY'S INSTITUTE OF TECHNOLOGY

Department of Computer Engineering

Project Report on

Secure Multimedia Communication

In partial fulfillment of the Fourth Year, Bachelor of Engineering (B.E.) Degree in Computer Engineering at the University of Mumbai Academic Year 2022-23

Project Mentor
Dr. Prashant Kanade
Mrs. Yugchhaya Galphat

Submitted by

Aashish Raheja (D17 - B , Roll no - 51)
Ashwin Pansare(D17 - B , Roll no - 43)
Kartikey Verma (D17 - B , Roll no - 65)
Karan Punjabi (D17 - B , Roll no - 49)

(2022-23)

VIVEKANAND EDUCATION SOCIETY'S INSTITUTE OF TECHNOLOGY

Department of Computer Engineering

Certificate

This is to certify that ***Aashish Raheja, Ashwin Pansare, Kartikey Verma, Karan Punjabi*** of Fourth Year Computer Engineering studying under the University of Mumbai have satisfactorily completed the project on ***"Secure Multimedia Communication"*** as a part of their coursework of PROJECT-II for Semester-VIII under the guidance of their mentor Dr./Mr. Prashant Kanade and Mrs Yugchhaya Dhote in the year 2022-23 .

This thesis/dissertation/project report entitled ***'SECURE MULTIMEDIA COMMUNICATIONS*** by ***Aashish Raheja, Ashwin Pansare, Kartikey Verma, Karan Punjabi*** is approved for the degree of B.E in Computer Engineering.

Programme Outcomes	Grade
PO1,PO2,PO3,PO4,PO5,PO6,PO7, PO8, PO9, PO10, PO11, PO12 PSO1, PSO2	

Date:

Project Guide:

Project Report Approval
For
B. E (Computer Engineering)

This thesis/dissertation/project report entitled ***SECURE MULTIMEDIA COMMUNICATIONS*** by ***Aashish Raheja, Ashwin Pansare, Kartikey Verma, Karan Punjabi*** is approved for the degree of B.E in Computer Engineering.

Internal Examiner

External Examiner

Head of the Department

Principal

Date:
Place:

Declaration

We declare that this written submission represents our ideas in our own words and where others' ideas or words have been included, we have adequately cited and referenced the original sources. We also declare that we have adhered to all principles of academic honesty and integrity and have not misrepresented or fabricated or falsified any idea/data/fact/source in our submission. We understand that any violation of the above will be cause for disciplinary action by the Institute and can also evoke penal action from the sources which have thus not been properly cited or from whom proper permission has not been taken when needed.

---	---
(Signature)	(Signature)
Aashish Raheja, 51	Ashwin Pansare, 43
---	---
(Name of student and Roll No.)	(Name of student and Roll No.)
---	---
(Signature)	(Signature)
Kartikey Verma, 65	Karan Punjabi, 49
---	---
(Name of student and Roll No.)	(Name of student and Roll No.)

Date:

ACKNOWLEDGEMENT

We are thankful to our college Vivekanand Education Society's Institute of Technology for considering our project and extending help at all stages needed during our work of collecting information regarding the project.

It gives us immense pleasure to express our deep and sincere gratitude to **Dr. Prashant Kanade and Mrs. Yugchhaya Galphat** (Project Guides) for her kind help and valuable advice during the development of project synopsis and for her guidance and suggestions.

We are deeply indebted to Head of the Computer Department **Dr.(Mrs.) Nupur Giri** and our Principal **Dr. (Mrs.) J.M. Nair,** for giving us this valuable opportunity to do this project.

We express our hearty thanks to them for their assistance without which it would have been difficult in finishing this project synopsis and project review successfully.

We convey our deep sense of gratitude to all teaching and non-teaching staff for their constant encouragement, support and selfless help throughout the project work. It is a great pleasure to acknowledge the help and suggestion, which we received from the Department of Computer Engineering.

We wish to express our profound thanks to all those who helped us in gathering information about the project. Our families too have provided moral support and encouragement several times.

Computer Engineering Department
COURSE OUTCOMES FOR B.E PROJECT

Learners will be to,

Course Outcome	Description of the Course Outcome
CO 1	Able to apply the relevant engineering concepts, knowledge and skills towards the project.
CO2	Able to identify, formulate and interpret the various relevant research papers and to determine the problem.
CO 3	Able to apply the engineering concepts towards designing solution for the problem.
CO 4	Able to interpret the data and datasets to be utilized.
CO 5	Able to create, select and apply appropriate technologies, techniques, resources and tools for the project.
CO 6	Able to apply ethical, professional policies and principles towards societal, environmental, safety and cultural benefit.
CO 7	Able to function effectively as an individual, and as a member of a team, allocating roles with clear lines of responsibility and accountability.
CO 8	Able to write effective reports, design documents and make effective presentations.
CO 9	Able to apply engineering and management principles to the project as a team member.
CO 10	Able to apply the project domain knowledge to sharpen one's competency.
CO 11	Able to develop professional, presentational, balanced and structured approach towards project development.
CO 12	Able to adopt skills, languages, environment and platforms for creating innovative solutions for the project.

INDEX

Abstract

An end-to-end encrypted chat application is a secure messaging platform that allows users to exchange messages without any third-party interception. Such an application ensures that only the sender and receiver of the messages can access the contents of their conversation.The System under consideration aims to provide the users a real time off the grid secure and private multimedia communication system with the user interface designed to be simple and easy to use. The system will fulfill the main requirement of security claims by not incorporating any third party APIs or other services. Users will be required to provide a username and password to gain access to the platform. This application is a cross platform messaging application with an End to End encryption service. The major goal is to enable data transfer of any kind of files including and not limited to images, videos, documents, while ensuring that the content remains encrypted and secure. The most unique feature of this system would be the use of our very own cloud server and database system without inclusion of any third party cloud servers.

1. Introduction

1.1 Introduction

In today's digital age, the importance of privacy and security has become increasingly paramount. With the rise of data breaches and cyber attacks, individuals and organizations alike are more aware than ever of the risks associated with storing and sharing sensitive information online. This has led to a growing demand for secure communication solutions that prioritize privacy and protect against unauthorized access and data theft.

One of the biggest concerns facing users today is the practice of data monetization, where companies collect and sell user data to third parties without their knowledge or consent. This is a particularly pressing issue for those who rely on free messaging apps or other online services, which often use personal data to target advertisements and generate revenue.

To address these concerns, anonymous communication software that prioritizes user privacy above all else is being developed. By eliminating the need for third-party services, it can be ensured that user data remains secure and confidential at all times. The goal is to provide a platform that allows individuals and organizations to exchange sensitive information without fear of data theft or unauthorized access.

In addition to addressing privacy concerns, secure messaging also has a wide range of applications in various industries. For example, in the healthcare sector, secure messaging can be used to exchange patient data between providers without compromising patient privacy. Similarly, businesses can use secure messaging to exchange confidential documents and information with their customers or partners.

Overall, this anonymous communication software represents a major step forward in the field of secure messaging. By providing a platform that prioritizes privacy and eliminates the risks associated with data monetization, the aim of the project is to provide a more secure and reliable solution for users across various industries.

1.2 Motivation

Currently all other messaging services are hosted on Third party cloud platforms mainly Google and AWS cloud services. This raises concerns about privacy as the third party has access to the user's private data. Let us consider whatsapp for example, which has been recently acquired by The Facebook team. After this acquisition whatsapp updated it’s privacy policy which gave access to facebook to collect private information on their users causing many controversies and heated discussions amongst the IT industry around the globe. This problem will be resolved by not allowing any third party cloud servers. The private cloud is recommended to avoid the sharing of resources with any third party which allows users to exchange vital information with other users and groups of users without being concerned about any kind of data leak.

1.3 Problem Definition

Aatmanirbhar Sanchar is a cross platform messaging application providing End to End encryption service.

The application will include the following functionalities:

- A user will be able to send any type of message(text,video,documents)
- Whole Transfer of the message will be encrypted from both sender and receiver sides.
- Transfer of the message will not include any third party servers like AWS and will be deployed on servers in the college premises.

1.4 Lacuna of the existing systems

1. Lack of end-to-end encryption: Many popular messaging apps, including WhatsApp and Facebook Messenger, do not offer true end-to-end encryption, meaning that messages are vulnerable to interception and snooping by third parties.
2. Data monetization: As mentioned earlier, many free messaging apps use personal data to generate revenue, which can compromise user privacy and security.
3. Centralized servers: Most messaging apps rely on centralized servers to store and transmit data, which can be vulnerable to hacking or other forms of unauthorized access.
4. Limited control over data: In many cases, users have little control over how their data is collected, stored, and used by messaging apps, which can lead to concerns about privacy and security.

5. Limited interoperability: Many messaging apps are not interoperable, meaning that users are limited in their ability to communicate with those who use different platforms.

1.5 Relevance of the Project

There is a dire need to focus on these privacy problems faced by users using chat applications as the company owning these applications don't take utmost care in handling the user data but they drive their marketing agenda through the data provided by the user in the application. So a cross-platform messaging application for secure communication is being built, wherein a user can exchange vital information with other users and groups of users without being concerned about any kind of data leak or data monetization.

2. Literature Survey

2.1 Research Papers Referred

1. "Blockchain-enabled End-to-End Encryption for Instant Messaging Applications" by Raman Singh et al.

   It introduces a new framework that utilizes blockchain technology to enhance the security of messaging applications. The framework employs end-to-end encryption (E2EE) and generates a public/private key pair during application installation. The mobile network operator (MNO) issues a digital certificate which is stored on a public blockchain. The users can obtain each other's certificates from the application server and communicate securely using a ratchet forward encryption mechanism. By adopting this framework, many of the vulnerabilities present in contemporary messaging applications can be avoided.

2. "Design and implementation of web based real time chat interfacing server" by Diotra Henriyan et al.

   A new chat application is being developed, and it has been designed to start with the collection of relevant data. This data will then be displayed in both the web and mobile versions of the application, making it accessible to users on multiple platforms. In order to build the server-side of the application, the programming language Node.js has been chosen along with the express framework to facilitate the development process. Additionally, a MongoDB database has been selected to store the collected data in a structured manner. This approach will ensure that the application is scalable and can handle a large number of users. Moreover, using these technologies will allow for faster development and maintenance of the application. The end result will be a chat application that is both user-friendly and robust, providing users with a seamless messaging experience.

3. "ChatterBox- A Real Time Chat Application" by Nidhi Zala et al.

   The proposed concept is a chat platform that enables individuals to connect with one another regardless of distance. This web-based application is designed to provide users with real-time and multiplatform communication capabilities, making it accessible to a wide range of users. To ensure that the application is efficient and

user-friendly, the development process begins with the collection of relevant data that will be integrated into the web application. With the help of this online application, people can communicate effectively with one another regardless of their location. This innovative concept is expected to have a positive impact on the way people interact online, allowing for increased connectivity and more meaningful conversations. By providing a seamless user experience and incorporating the latest technology, this chat platform has the potential to revolutionize the way people communicate online.

4. "The Tor Instant Messaging Bundle: Encrypted and Anonymous Instant Messaging" by Colin Childs, et al.

   This paper describes the design and implementation of the Tor Instant Messaging Bundle, which provides a secure and anonymous messaging solution using the Tor network. The paper discusses the benefits and drawbacks of various messaging protocols, and explains how the Tor network can be used to provide anonymity and privacy for messaging.
   The authors also discuss the technical details of the Tor Instant Messaging Bundle, including the use of the Off-the-Record messaging protocol for end-to-end encryption, and the use of Tor's onion routing system to provide anonymity. The paper concludes with a discussion of the limitations of the system and potential future developments.

5. "Signal: Private Group Messaging" by Moxie Marlinspike and Trevor Perrin.

   This paper describes the design and implementation of the Signal messaging app, which provides end-to-end encryption and other privacy-focused features. The authors discuss the challenges of designing a messaging system that is both secure and user-friendly, and explain how they addressed these challenges in Signal.
   The paper discusses various technical aspects of the Signal protocol, including the use of the Double Ratchet algorithm for encryption, the use of a prekey system for key exchange, and the use of a centralized server for message routing. The authors also discuss the user interface and user experience design of the app, and the challenges of making a secure messaging app that is easy to use and understand.

6. "TextSecure: Private SMS/MMS Messaging" by Moxie Marlinspike and Stuart Anderson.

   This paper describes the design and implementation of TextSecure, a messaging app that provides end-to-end encryption for SMS and MMS messages. The authors discuss the challenges of providing secure messaging over a legacy communication protocol like SMS, and explain how they addressed these challenges in TextSecure.
   The paper discusses various technical aspects of the TextSecure protocol, including the use of the Axolotl ratchet algorithm for encryption, the use of a centralized server for message routing, and the use of a key management system to manage user keys. The authors also discuss the user interface and user experience design of the app, and the challenges of making a secure messaging app that is compatible with existing messaging infrastructure.

7. "Matrix: An Open Standard for Decentralized Communication" by Matthew Hodgson and Amandine Le Pape.

   This paper describes the design and implementation of the Matrix protocol, which provides a decentralized and secure messaging solution. The authors discuss the limitations of existing messaging solutions, and explain how the Matrix protocol provides a more flexible and privacy-focused alternative.
   The paper discusses various technical aspects of the Matrix protocol, including the use of end-to-end encryption for message security, the use of a decentralized identity system for user authentication, and the use of a federated network for message routing. The authors also discuss the potential use cases for the Matrix protocol, including in the healthcare and financial sectors.

8. "Towards Secure Instant Messaging" by Jan Seidl and Christian Wawrzinek.

   This paper examines various aspects of secure instant messaging, including encryption, authentication, and key management. The authors discuss the challenges of providing secure messaging over a public communication infrastructure, and explain how various messaging protocols address these challenges.
   The paper provides a comprehensive overview of various messaging protocols, including Signal, TextSecure, and the Extensible Messaging and Presence Protocol (XMPP). The authors also discuss the challenges of providing secure messaging in a mobile context, and explain how various apps address these challenges. The paper

concludes with a discussion of the potential future developments in secure instant messaging, and the challenges that must be addressed in order to make secure messaging more widely adopted.

2.2 Patent search

2.2.1 Secure instant messaging system (US7739508B2)

A secure instant messaging system integrates secure text instant messaging and secure file transfers into existing instant messaging systems. At least one certificate authority (CA) is provided that issues a security certificate to a user that binds the user's instant messaging screen name to a public key which is used by other users to encrypt messages and files sent to the user and by the user to decrypt the received messages and files. A subscriber database is used by the CA to keep track of valid users and their associated information, such as: user screen names, user subscription expiration dates, and enrollment agent information. A user sends his certificate to the invention's instant messaging server which publishes the user's certificate to other users by creating a hash value of the user's certificate and sending it to the other users which allows the recipients to decide if they need to update their caches with a new copy of the user's certificate. Instant messages and files are encrypted by a sending user using an encryption algorithm and the recipient's certificate. The sending user can sign instant messages using his private signing key. The security status of each received instant message is displayed to the user.

2.2.2 System and method for secure end-to-end chat system(US9432340B1)

The present invention provides an efficient secure end-to-end messaging system utilizing encrypted ephemeral messages. The method comprises the steps of using a combination of HTTPS for transport security, using symmetric key cryptography with rotating temporary keys for individual message security, and using elliptic curve cryptography for key derivation and message authentication. The key rotation scheme used provides forward secrecy even between messages and perfect forward secrecy between sessions.

2.3.3 Secure messaging (US9648001B2)

Systems and methods are disclosed for secure messaging and content sharing. In one implementation, a processor receives a message associated with a recipient, provides, to the recipient, a notification pertaining to the message, and, based on a determination of a performance of one or more authentication actions with respect to the message, provides the recipient with access to the message. In another implementation, a processor receives a message including one or more content segments, receives inputs in relation to at least one of the content segments, processes the inputs to determine that an authentication action is being performed with respect to the one of the one or more content segments, and based on a determination that the authentication action is being performed with respect to the one of the one or more content segments, presents the at least one of the one or more content segments.

2.2 Inference drawn

Based on the literature survey of "secure messaging applications", it can be inferred that there is a growing interest in developing messaging solutions that prioritize security and privacy for users. These solutions use various technical approaches, such as end-to-end encryption, decentralized messaging infrastructure, and anonymous communication networks like Tor, to provide users with a high level of protection against data theft and privacy invasion.

Additionally, the papers highlight the challenges of making secure messaging systems that are user-friendly, compatible with existing communication infrastructure, and accessible to a wide range of users. Many of the solutions discussed are designed to work on mobile devices, which introduces additional challenges due to the limited screen real estate and processing power of these devices.

Overall, the research papers suggest that there is a need for messaging solutions that balance security and usability, and that there is ongoing research and development in this area. As privacy concerns continue to grow, it is likely that messaging solutions that prioritize security and privacy will become increasingly important in the years to come.

3. Requirement of the Proposed System

3.1 Functional Requirements

- **User Authentication:** Users must be authenticated before they can access the chat platform.
- **Text Message communication:** The user should be able to send and receive messages directly from the application.
- **Group Messaging:** Users should be able to create or join groups to exchange messages with multiple users.
- **End-to-End Encryption:** The application should ensure end-to-end encryption for all messages exchanged between users to ensure privacy and security.
- **Image/Video Files:** The user should be able to send his/her clicked photos/videos easily to another person without loss in quality of the image/video for better communication.
- **Document Files:** The user should be able to send his/her stored documents easily to another person without loss in encryption of the document for better privacy.

3.2 Non-Functional Requirements

- **Secure Communication** : Application will provide end to end encryption which will ensure that messages are securely transferred from one user to another without involvement of third parties.
- **Compatibility**: The application should be compatible with multiple devices and platforms, including desktops, laptops, smartphones, and tablets.
- **Usability**: The application should be user-friendly, with a simple and intuitive interface that is easy to use and navigate.
- **Availability**: The application should be available, providing uninterrupted service to users at all times.
- **Independency**: Application will use its own server and not depend on third party cloud servers like GCP and AWS
- **Maintainability**: The application should be maintainable, allowing for easy updates, upgrades, and bug fixes.

3.3 Constraint

- Processing speed regarding encryption process
- Size limit on files sent
- Static IP server issues

3.4.Hardware, Software , Technology and tools utilized

The following tools and technologies have been used in the development of Aatmanirbhar Sanchar:

- Front-End Development:
    - HTML
    - CSS, Bootstrap
    - JavaScript
- Backend Development (server-side):
    - Libraries used:
        - NodeJS
        - Socket.IO
- Compatible Operating System:
    - Ubuntu (16.04)
    - Windows 10
- Requirements:
    - A Public Static IP to host the messaging application

4. Proposed Design

4.1 Block diagram and Modular representation of the proposed system

The block diagram of the proposed system is given below, which is divided into two parts. The first module of the system is the client aspect which interacts with the user and enables the respective user to utilize the system functionalities like joining the room, sending the data, encrypting and decrypting the data and exiting the chat room. The second module of the system is the server part of the system that performs all the backend operations as required by the user like creating chat rooms, connecting the users, broadcasting the message to all the users in the chat room and disconnecting the user from the chat room.

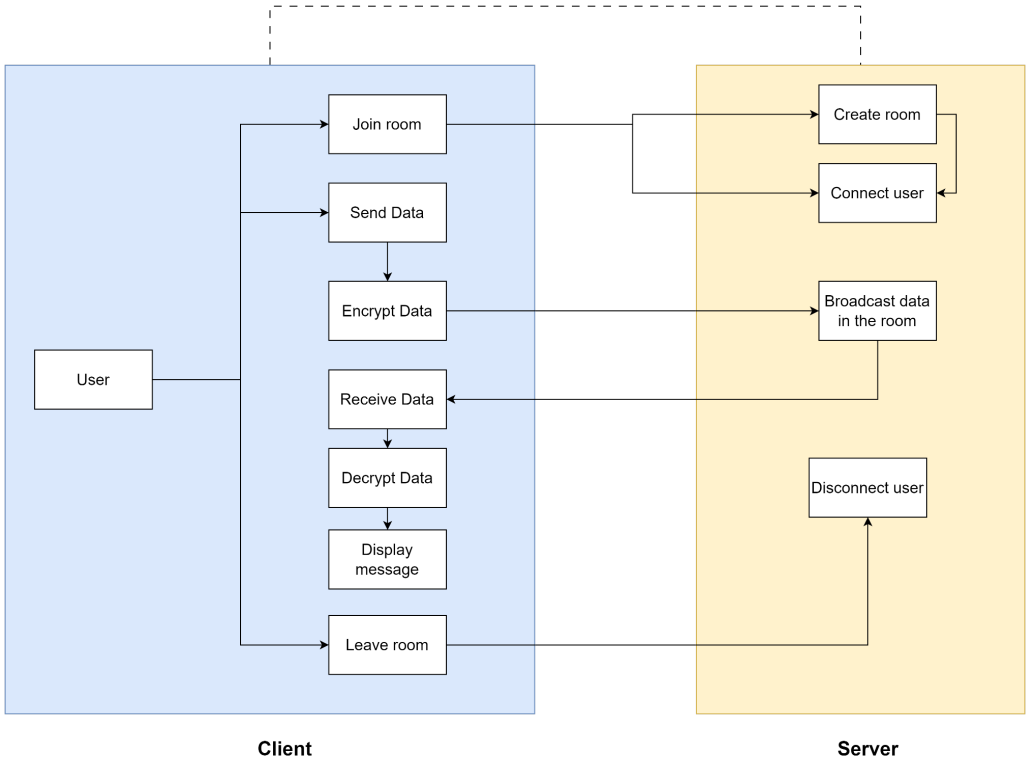


Fig 1: Block Diagram

4.2 Design of the proposed system

The design of the proposed system includes 3 layers, the first layer is the input layer which is initiated by the user through entering the room with other users of the system.

Through the application the user sends the different types of data using the user interface of the application. The second layer is the encryption layer, and this layer is responsible for end-to-end encryption of the data sent by the users to all others in the chat room.

This encrypted data packet is then sent to the other users present in the chat room where the data packets are decrypted and then presented to the recipients of the messages.

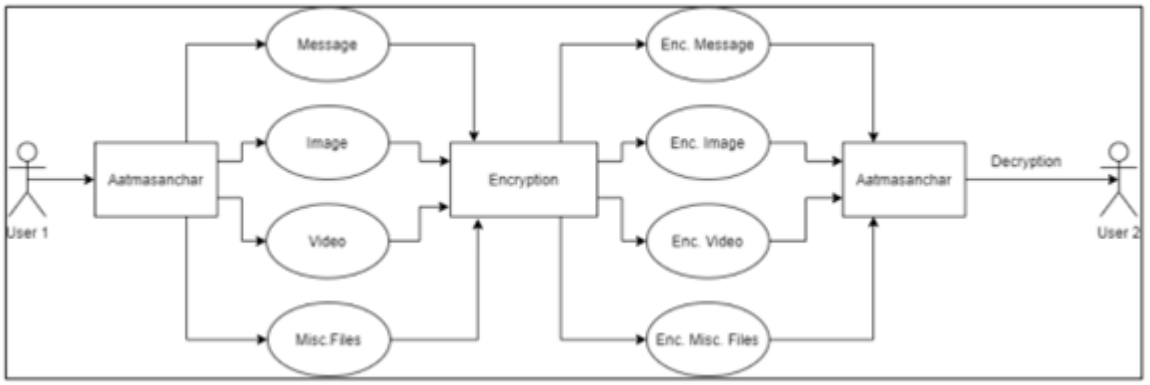


Fig 2: System Design

4.3 Gantt Chart

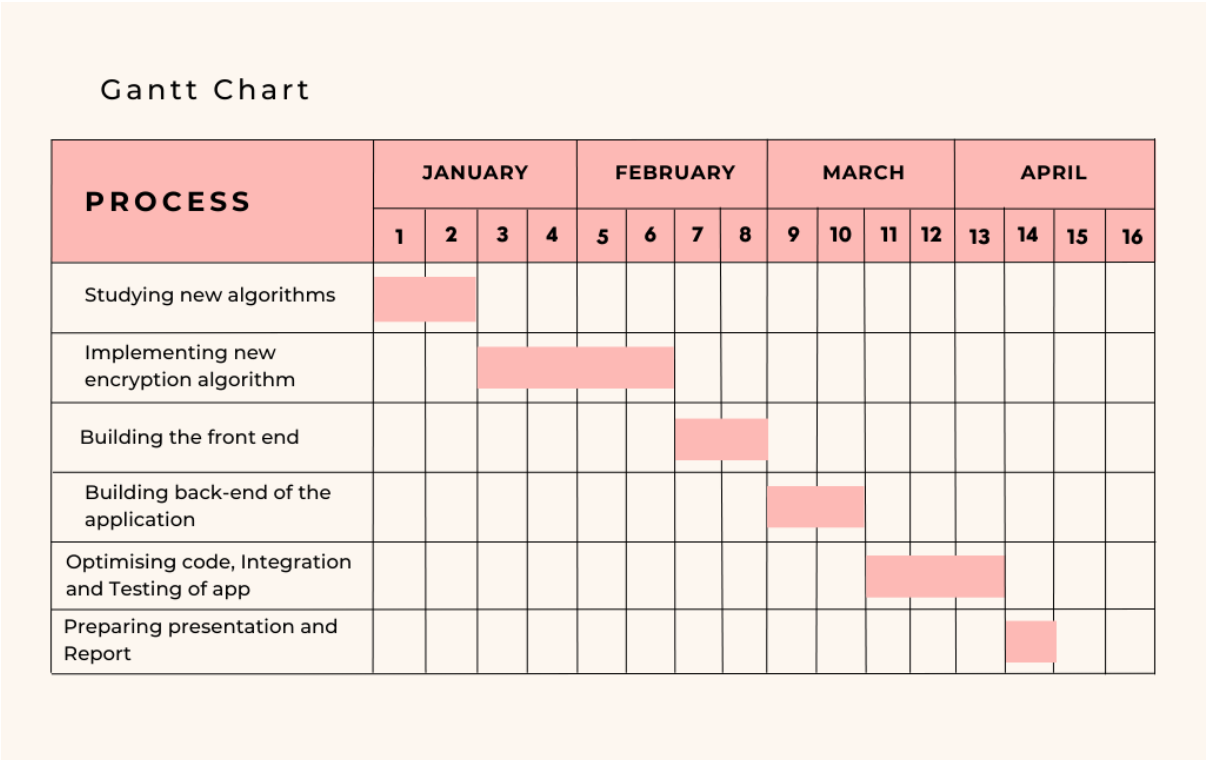


Fig 3: Gantt Chart

5. Implementation of the Proposed System

5.1. Methodology employed for development

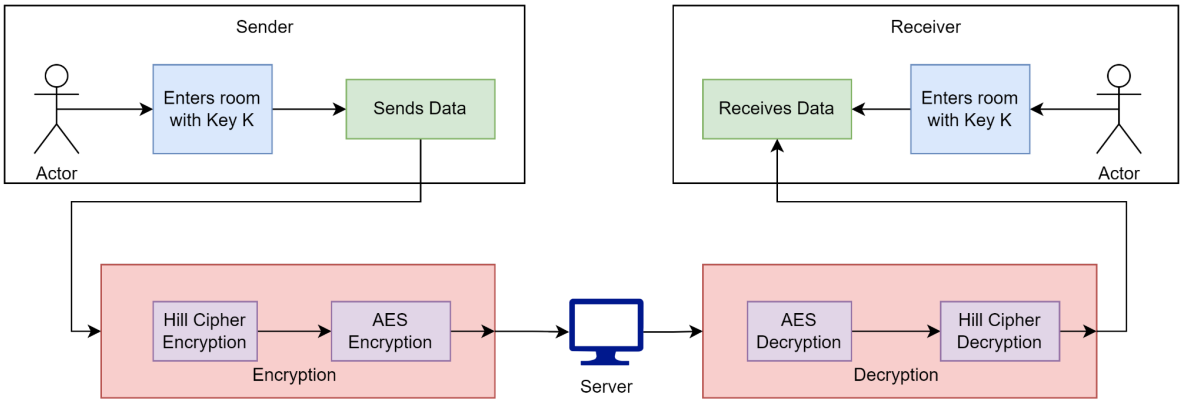


Fig 4: Methodology overview

- A user needs a username and room key to enter a chat room.
- To be able to communicate with each other the users need to enter the same room by entering the same room key. Once the users have entered the room, the messages and files that they send will be sent to all the users in that room.
- All data will be End-to-End encrypted. The room key will be used as a key for encryption and decryption. When a user sends some data, it will be first encrypted on the user's device then this encrypted data will be sent to the server, the server will send it to all other users in the room finally when the other user receives this data it will be decrypted on their device and the final message will be displayed to them.
- A double layered encryption process is used for achieving enhanced security. The first layer consists of the hill cipher encryption using the room key.
- This encryption layer is followed by the Advanced Encryption Standard (AES-256) algorithm to ensure privacy.

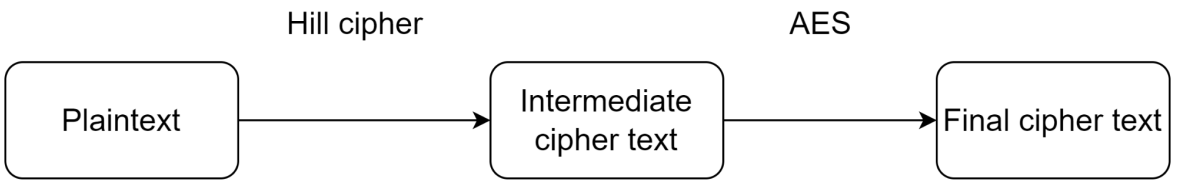


Fig 5: Encryption process

- Transmission of Messages:

  The messages that can be transmitted by the users are broadly classified into two types:

  - Text Message: Any text message sent by the user is first converted into its unicode values. Following this, the message is sent to the encryption layer where the encryption algorithm developed is applied. The first aspect is Hill cipher and the second part is AES. This step ensures that the messages are end-to-end encrypted. The encrypted data packet is then sent to the server, where the server redirects and broadcasts the data packer to the recipients present in the chat room.
  - Files: The file attached by the user is first converted into its base 64 format. Now that the base 64 format of the file is ready, the characters present in the text are transformed into their respective unicode values. Following this, the message is sent to the encryption layer where the encryption algorithm developed is applied. The first aspect is Hill cipher and the second part is AES. This step ensures that the messages are end-to-end encrypted. The encrypted data packet is then sent to the server, where the server redirects and broadcasts the data packer to the recipients present in the chat room.

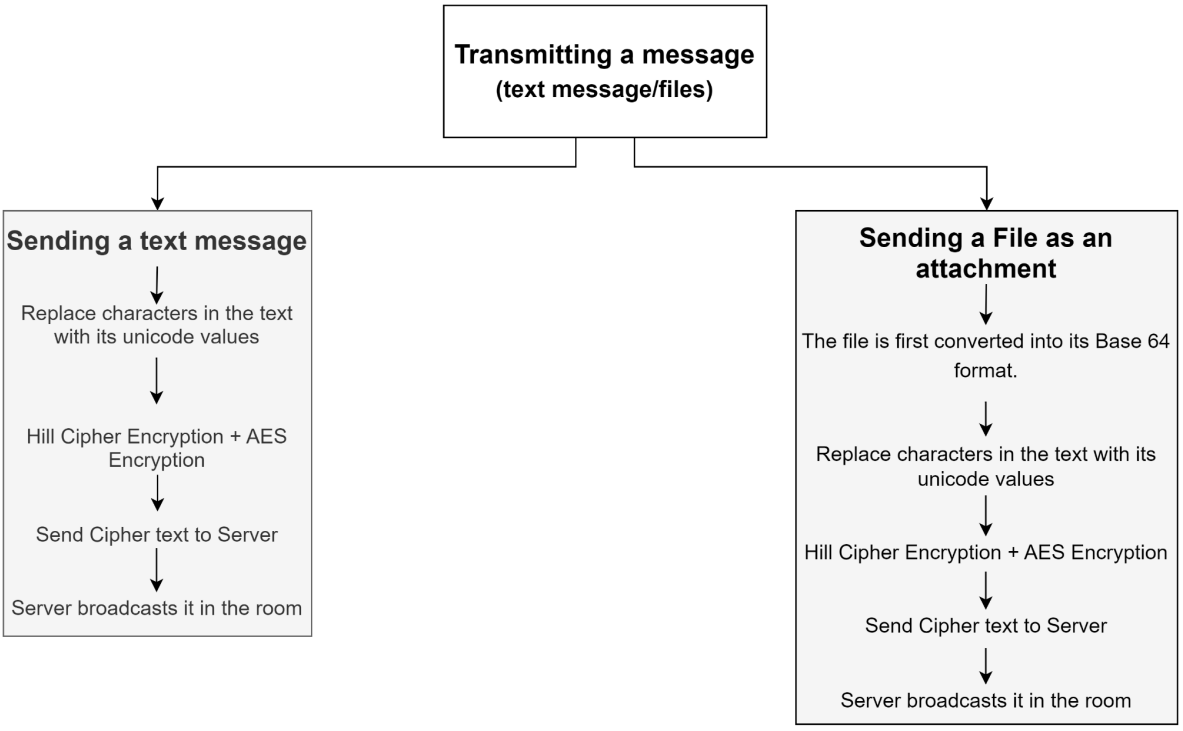


Fig 6: Message Transmission

- Receiving of Messages:

  After the transmission of the messages, the two types of messages received by the users follow the following steps.

  - Text Message: The encrypted cipher text is received by all the recipients in the chat room, where it follows a two step decryption process. The first level is the AES decryption followed by the second level which is the hill cipher algorithm. After this step the decrypted text is broadcasted across the chat room.
  - File: The encrypted file received is first passed through the decryption level which is the same as that for text data. Decryption generates the base 64 format for the given file which is then converted into its original file format according to the MIME type.

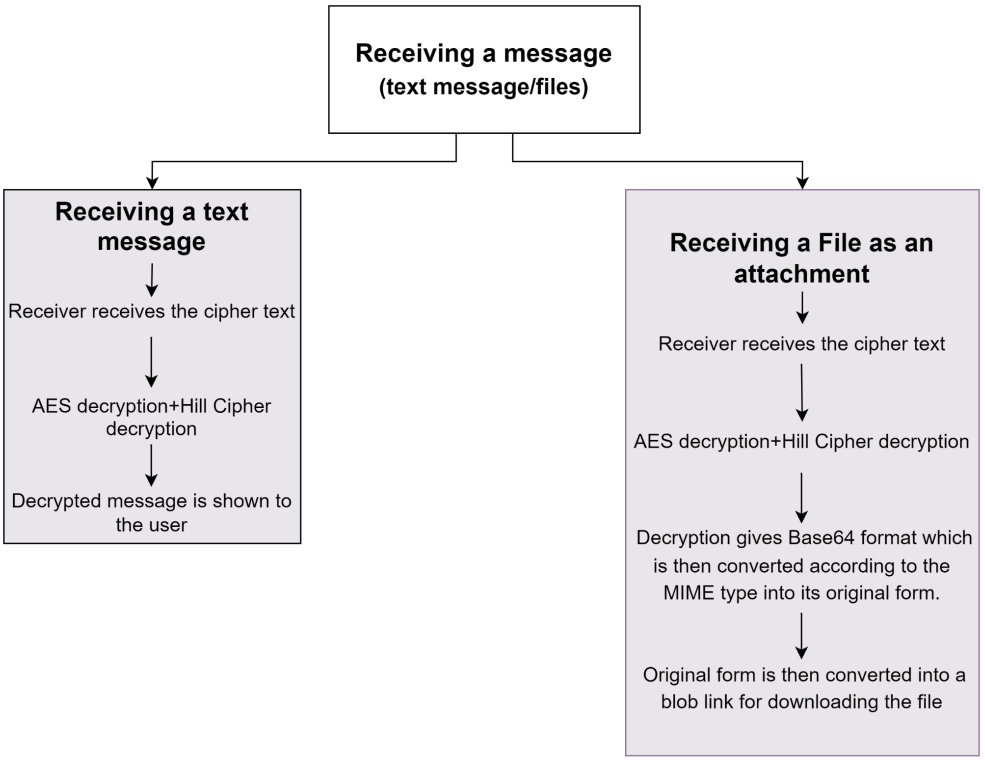


Fig 7: Receiving a Message.

5.2 Algorithms and flowcharts for the respective modules developed

Hill Cipher:

Hill cipher is a polygraphic substitution cipher. To encrypt a message, each block of n letters is multiplied by an invertible n × n matrix. To decrypt the message, each block is multiplied

by the inverse of the matrix used for encryption. The matrix used for encryption is the cipher key, and it should be chosen randomly from the set of invertible n × n matrices.

Key generation:

- Finding the matrices generated from passphrase which are invertible
  Eg. Passphrase - aaaaaababbbbbtterDominoesCurable_Pasture

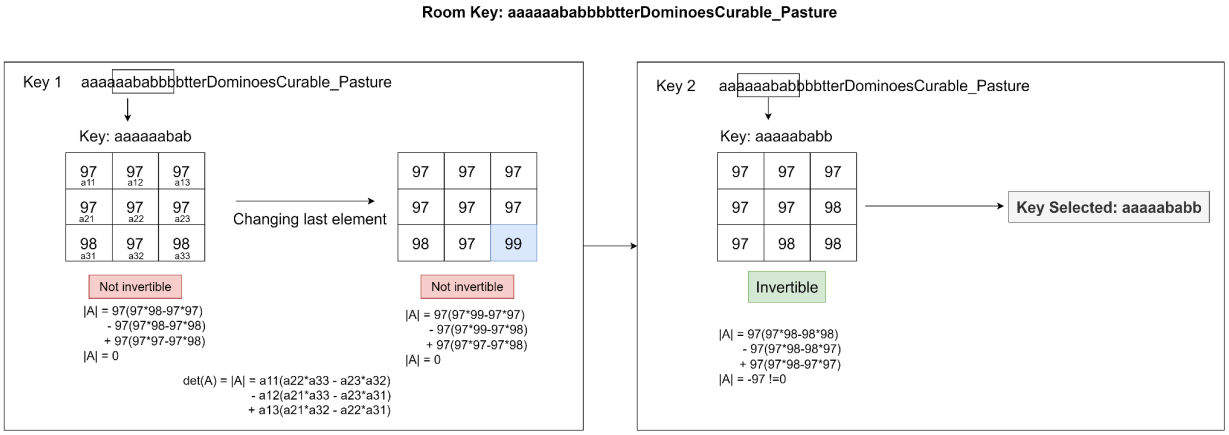


Fig 8: Key Matrix Generation

- Find inverse of selected key matrix
- The key matrix and the inverse will be then used in the Hill cipher algorithm for encryption and decryption processes respectively.

Steps for calculating inverse of a 3X3 matrix:

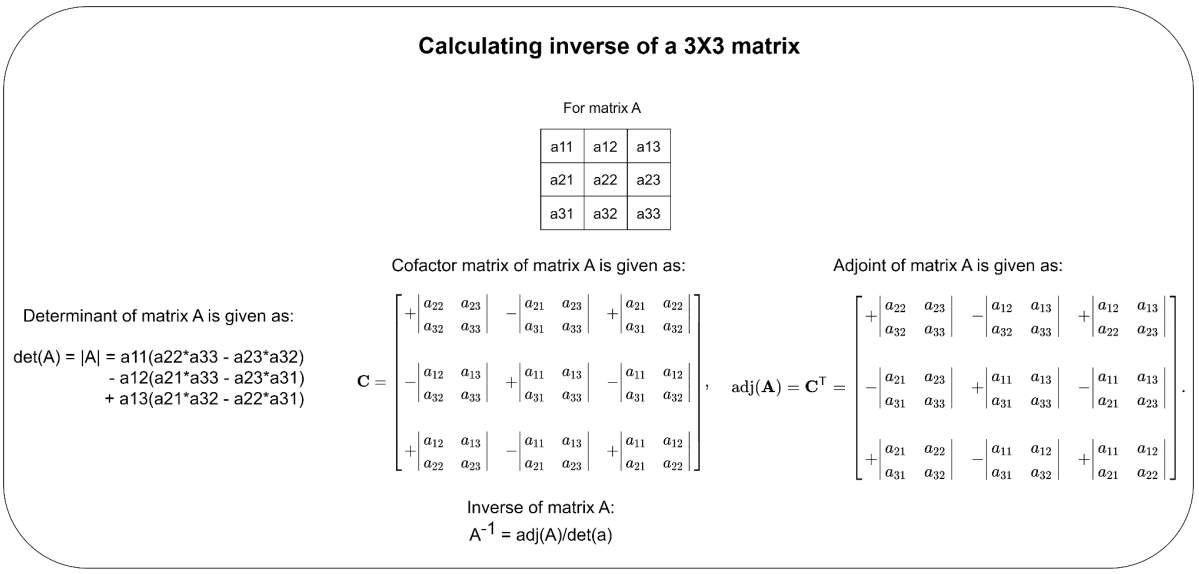


Fig 9: Inverse of a 3x3 matrix

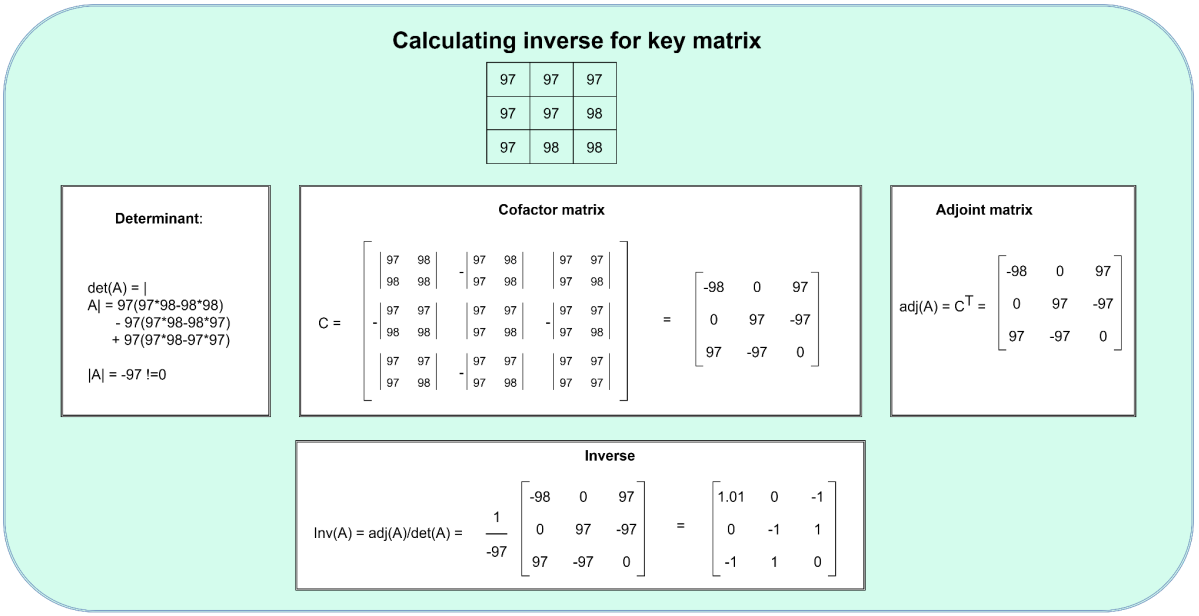


Fig 10: Inverse of the key matrix

Encryption:

1. The message is divided into blocks of size 3
2. Every character in the plaintext message is replaced by its Unicode value which will be a number between (0-65535)
3. Organize the plaintext message as a matrix of numbers based on the above conversion.
4. Now, the plaintext matrix is multiplied by the key matrix
5. Each element of the resultant matrix is concatenated with "+" symbol between them and a string is formed which will be the cipher text
6. The result of the above operation is sent as input for the AES encryption algorithm.

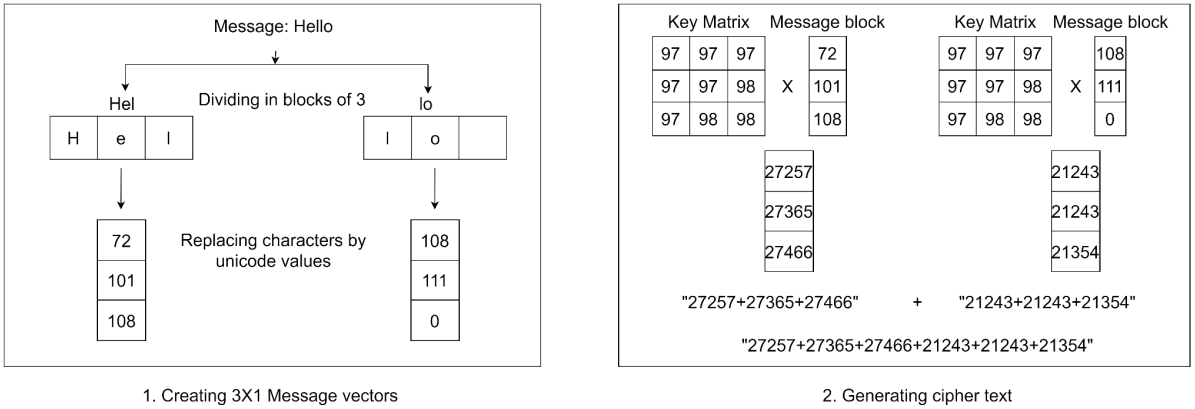


Fig 11: Generating cipher text

Decryption:

1. The string obtained from the result of AES decryption will be split according to the

"+" sign

2. Blocks of three numbers will form a 3x1 cipher matrix
3. This cipher matrix is multiplied by the inverse of the key matrix
4. It will give a 3x1 resultant matrix which will have the Character codes
5. The char codes will then be replaced by the corresponding characters and the final text will be generated.
6. This process will be continued for all the blocks and the results will be concatenated

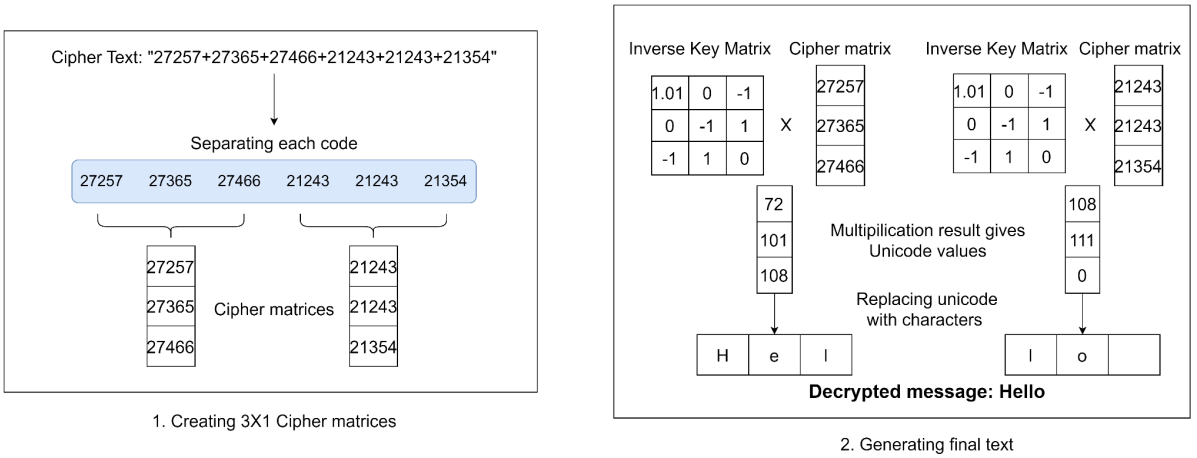


Fig 12:Decryption of cipher text.

AES

- AES is safe, fast, and flexible. AES is a much quicker algorithm compared to DES.
- It is the most widely used encryption algorithm today
- Many government agencies, including the National Security Agency (NSA), rely on the AES encryption algorithm to protect their sensitive information.
- AES is a block cipher.
- Encrypts data in blocks of 128 bits each.
- AES considers each block as a 16 byte (4 byte x 4 byte = 128) grid in a column major arrangement.

  [b0 | b4 | b8 | b12 |
  | b1 | b5 | b9 | b13 |
  | b2 | b6 | b10| b14 |
  | b3 | b7 | b11| b15]

- Each round comprises of 4 steps :
  - **SubBytes** :
    Each byte in the 4x4 matrix is substituted with another byte by using a S-box which is 16x16 table
    So the elements of the matrix are now hexadecimal numbers
    before:
    [b0 | b4 | b8 | b12 |
    | b1 | b5 | b9 | b13 |
    | b2 | b6 | b10| b14 |
    | b3 | b7 | b11| b15]
    after:
    [h0 | h4 | h8 | h12 |

| h1 | h5 | h9 | h13 |
| h2 | h6 | h10| h14 |
| h3 | h7 | h11| h15]

- **ShiftRows:**
  This step is just as it sounds. Each row is shifted a particular number of times. The first row is not shifted, the second row is shifted once to the left. The third row is shifted twice to the left. The fourth row is shifted thrice to the left.
  before:
  [h0 | h4 | h8 | h12 |
  | h1 | h5 | h9 | h13 |
  | h2 | h6 | h10| h14 |
  | h3 | h7 | h11| h15]

  after:
  [h0 | h4 | h8 | h12 |
  | h5 | h9 | h13 | h1
  | h10 | h14 | h2 | h6
  | h15 | h3 | h7 | h11]

- **MixColumns:**
  Each column is multiplied with a specific matrix
  [c0] = [2 3 1 1] [b0]
  | c1 | = | 1 2 3 1 | | b1 |
  | c2 | = | 1 1 2 3 | | b2 |
  [c3] = [3 1 1 2] [b3]

- **Add Round Key:**
  Now the resultant output of the previous stage is XOR-ed with the corresponding round key. Here, the 16 bytes is not considered as a grid but just as 128 bits of data.

6. Testing of the Proposed System

The given application can be utilized by the user through two ways:

- Web Application: The user can access and utilize the chat application which is hosted on the given link http://103.197.221.163:3478/ .
- Mobile Application: The users can also utilize the application either through the browser using the above given link or through an apk which can be downloaded.

6.1 . Introduction to testing:

The main objective of a chat application is to facilitate safe and secure communication between any nodes using the chat application. The application design needs to ensure the following aspects are taken into consideration and applied successfully.

1. The first aspect is to ensure that the application enables secure communication through broadcasting the message only to the selected personnel in the conversation and not to everyone using the application.
2. The second major aspect to consider is that the application supports end-to-end encryption of all the transmitted messages.
3. Transmission of all the major types of files should be supported by the application.

6.2. Types of tests considered:

1. Unit tests: These tests focus on verifying that individual components of the application work correctly. For example, unit tests are used to ensure that encryption and decryption functions are working properly.
2. Integration tests: These tests verify that different components of the application can work together seamlessly. For example, integration tests ensure that the encryption and decryption functions work together as expected.
3. Functional tests: These tests verify that the application functions as expected from a user perspective. For example, functional tests can ensure that messages are sent and received correctly.
4. Usability tests: These tests focus on how user-friendly the application is. For example, usability tests can check how easy it is to send and receive messages, or how intuitive the user interface is.
5. Performance tests: These tests focus on the performance of the application under various conditions. For example, performance tests can check how the application performs when sending large files or when there are many users on the system.

6.3. Various test case scenarios considered:

1. The application was tested by creating several rooms simultaneously and ensuring that messages from one room aren't transmitted to the other rooms even when the same user is present in several rooms
2. The encryption was tested for different types of media like text, images, videos, audios, pdf, word document and it was ensured that the receiver received the proper message.
3. The application was tested for different platforms including Android, Windows and Ubuntu as well as for mobile devices and desktop devices that are present on the same network or on different networks

Mobile-Mobile (Different networks):

File type - mp4 video, Size - 2.2 MB, Video length - 11 Seconds

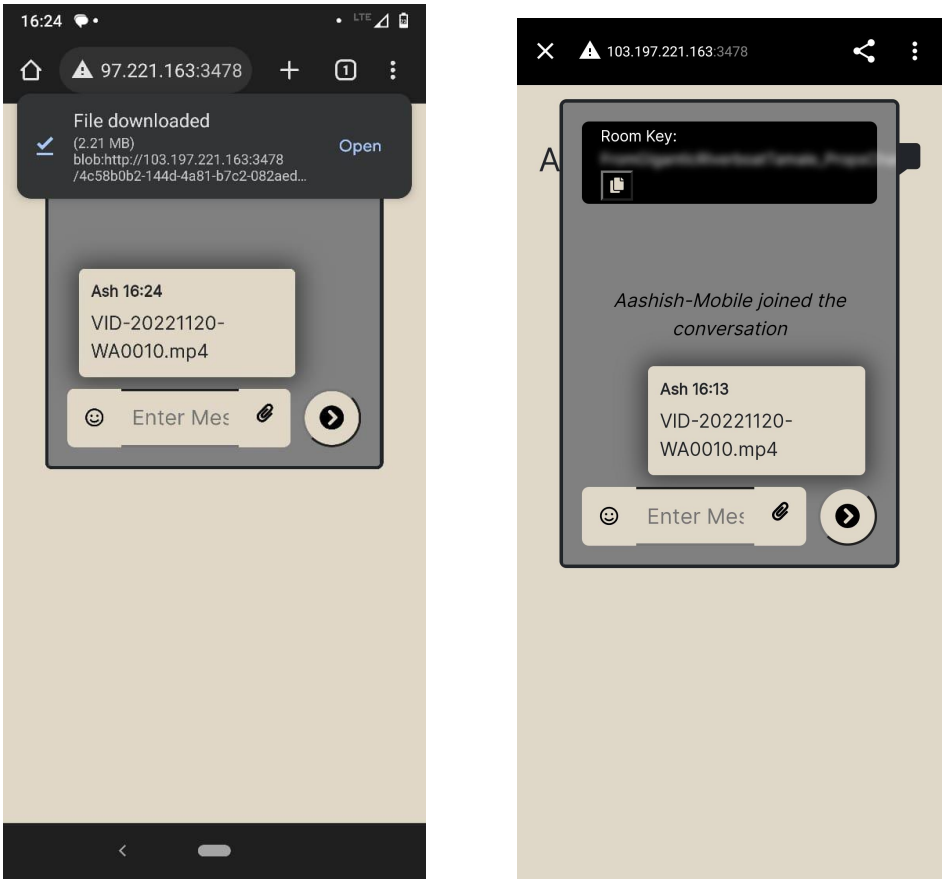


Fig 13: Mobile-to-Mobile(Different Networks)

Mobile-Desktop (Different networks) :

File type - mp3 audio, File size - 2.42 MB, Audio length - 2 minutes 38 seconds

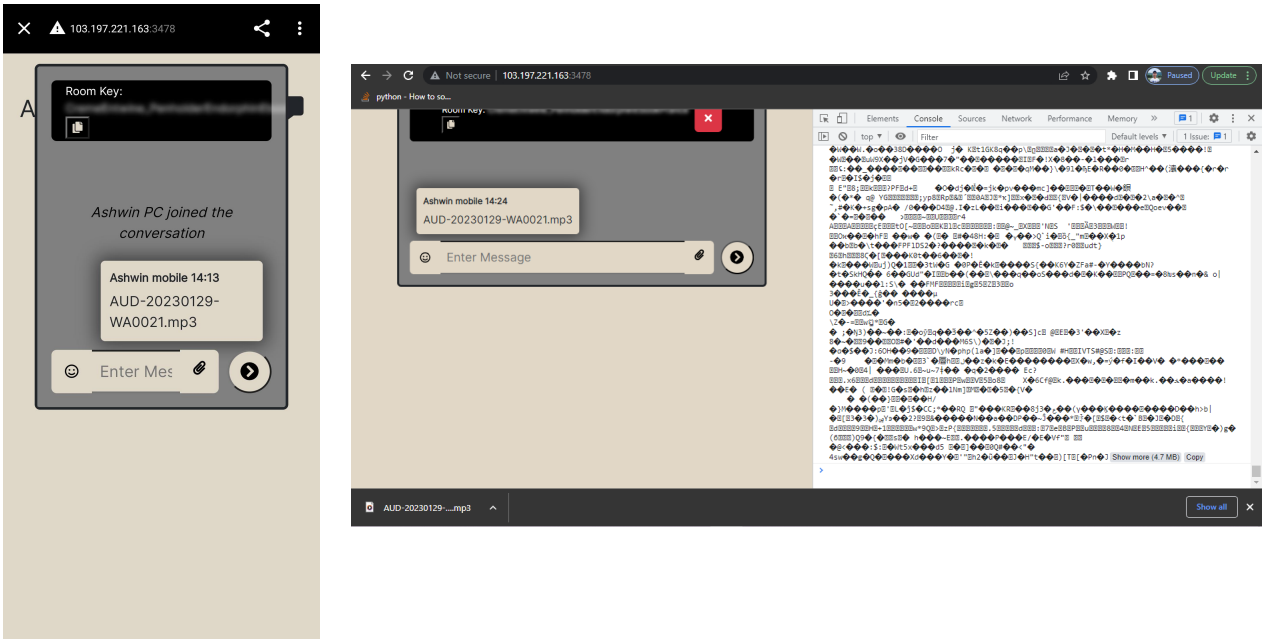


Fig 14: Mobile-to-Desktop(Different Networks)

Desktop-Mobile (Different networks) :

File size: 2.2MB, File type: pdf

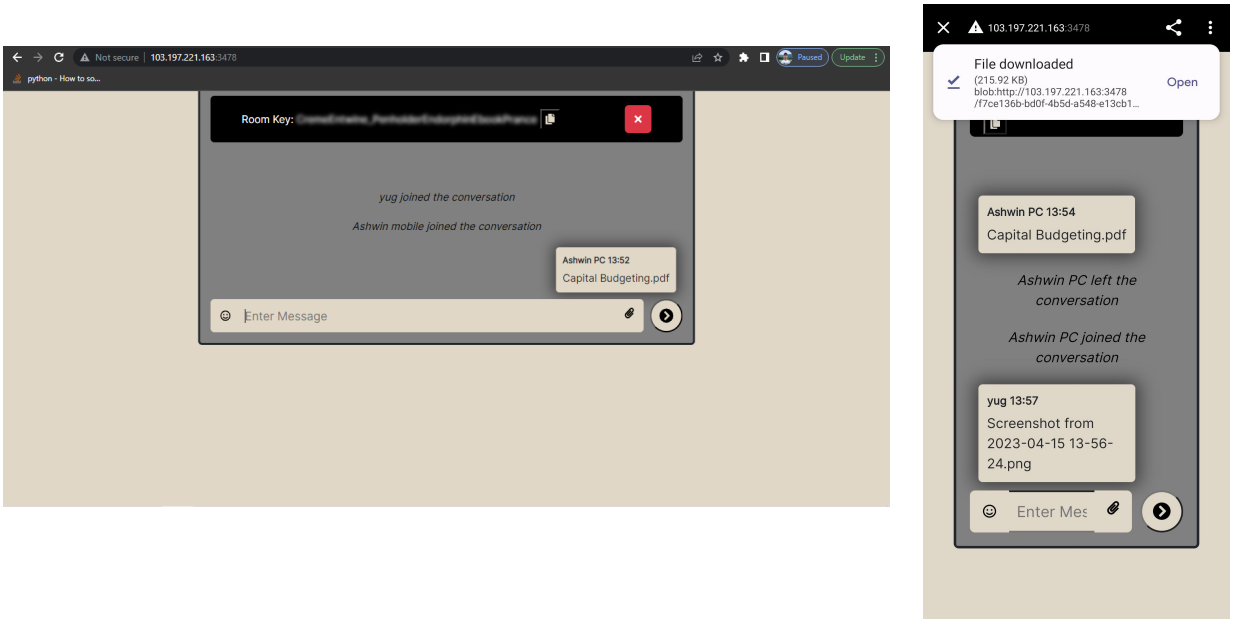


Fig 15: Desktop-Mobile((Different Networks)

Desktop-Desktop (Different networks):

File size: 2.2MB, File type: pdf

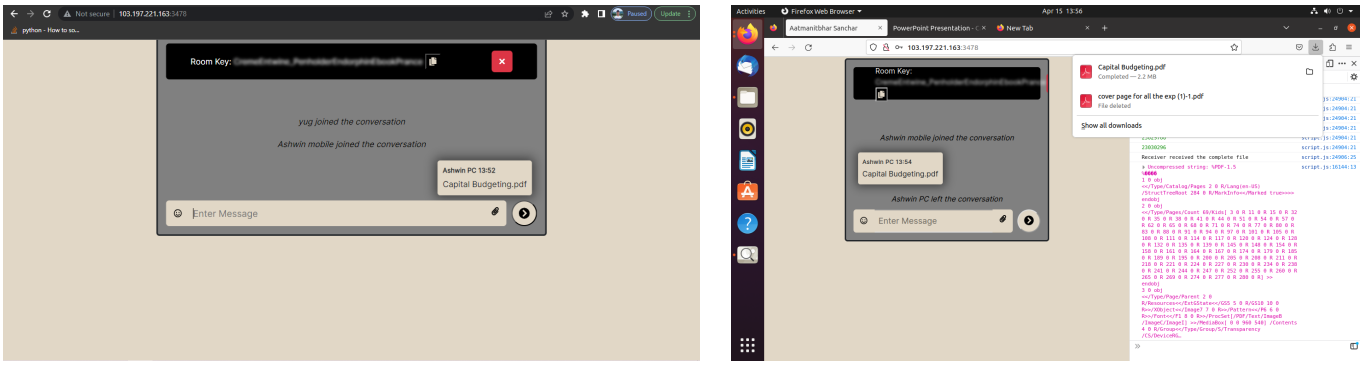


Fig 16: Desktop-Desktop(Different Networks)

6.4. Inference drawn:

The application works properly on mobile as well as desktop devices. The application also works when the users communicating are connected to different networks and different types of devices.

All the users in a chat room receive a message sent by any user in that room. The users on a chat room don't receive messages from other rooms. Users cannot access the messages without knowing the room key.

The data is encrypted before leaving the sender's device and is decrypted on the receiver's device and the receiver receives the original message after decryption. The encryption works perfectly for different types of media(text, image, video, audio, pdf, word doc).

The application can send text messages instantly. For other file types the transmission is slower and has been successfully tested only for file sizes up to 3MB. When sending files larger than that the users get disconnected as the encryption process and then transmission of the large encrypted files cause heavy load on the client device. The application also faced issues when sending files in rooms with more than 2 users

7. Results and Discussion

7.1. Screenshots of User Interface (UI) for the respective module

1. Main UI, Enter Username and Generate Roomkey:

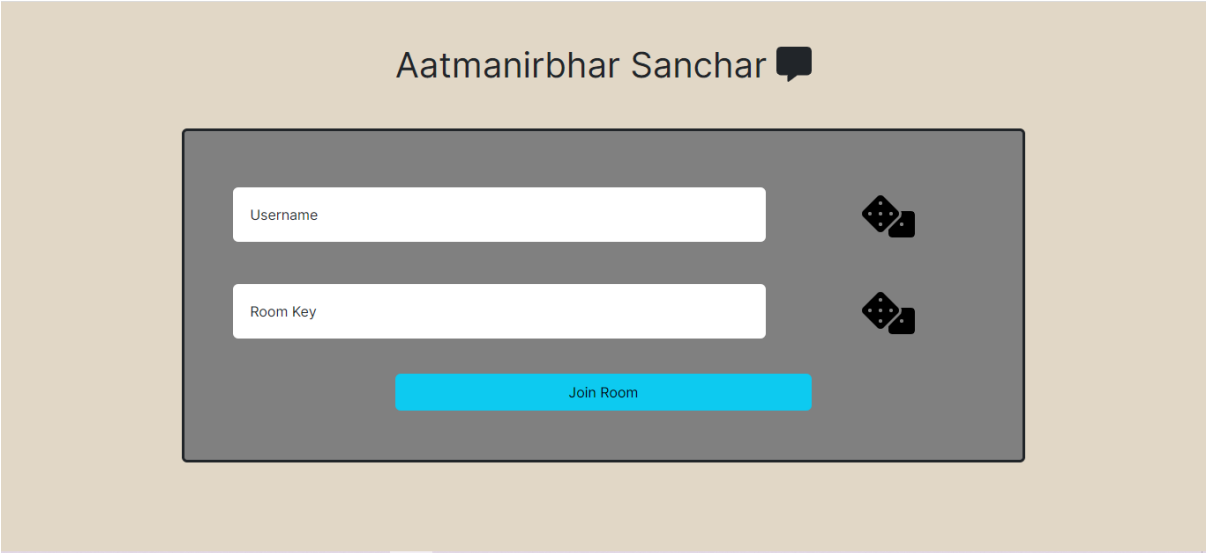


Fig 17: UI of join room page

2. Message interface :

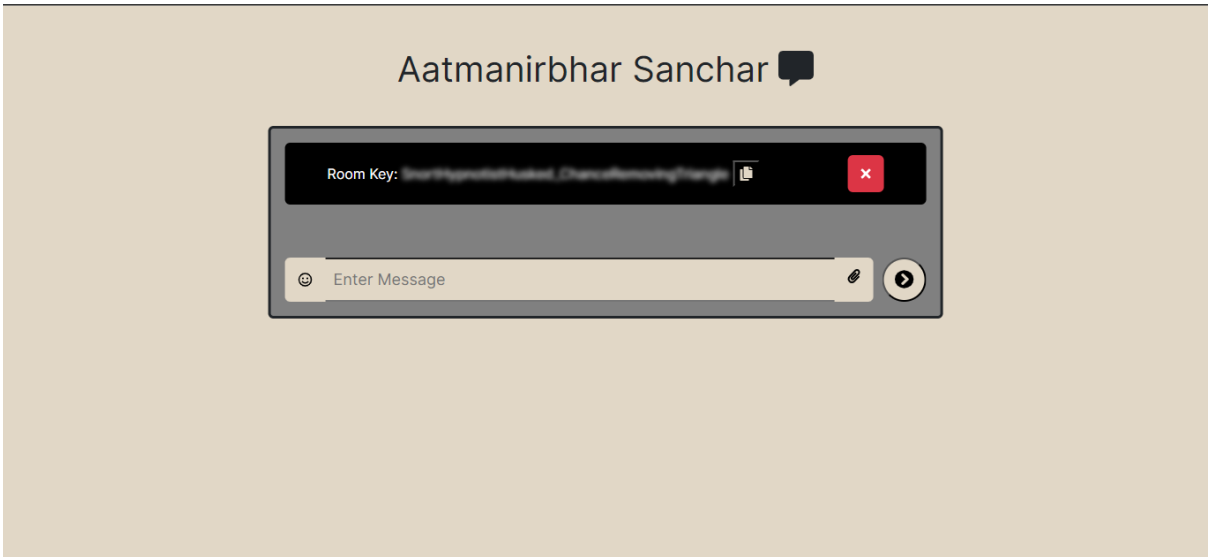


Fig 18: UI of chat room

7.2 Performance Evaluation measures :

Timely delivery of messages and files: The algorithm design ensures that the text messages are delivered instantly after double layered encryption.

Also, the files under 4 MB are delivered between 3-7 minutes after sending the files depending upon the internet speed of the device.

The file quality is not degraded at the receiver's end

7.3 Comparison of results with existing systems :

The new system follows a more enhanced double layered encryption system which is more secure than that of the existing where the first layer of encryption was just based on the XOR operation, rather than the more complex matrix based encryption.

The file transfer is slower than the existing chat applications.

The application doesn't use any compression techniques for files, as a result the file received is exactly the same as the file sent unlike other applications where the quality is degraded.

7.4. Inference drawn

Thus, the new double layered encryption algorithm developed ensures that the messages transmitted are less susceptible to attacks through enabling encryption of the messages through a matrix based hill cipher algorithm rather than simply shifting the bits of the message as incorporated in the earlier version of the application.

Also, the application now works efficiently for all the files under 4MB which was missing in the earlier version as the user was frequently getting disconnected from the chat room in the application.

8. Conclusion

8.1 Limitations

1. The application currently doesn't work for files larger than 4 MB
2. The User Interface of the system is not properly oriented for some mobile devices.
3. The User Interface in some mobile devices is congested and hence the application becomes difficult to operate for the user.
4. The file transmission is slower for devices connected on different networks

8.2 Conclusion

The development of an end-to-end encrypted chat application without the use of any third-party services has been successfully completed. The application has been designed to provide users with a high level of privacy and security when communicating with others, without sacrificing usability or functionality.

The project has been successful in achieving all the functional and non-functional requirements that were set out at the beginning of the project. The application provides users with a secure and private messaging platform that protects their personal data and communications from unauthorized access.

The research conducted on this project has revealed the increasing demand for messaging solutions that prioritize security and privacy. The development of this application is a step forward in providing users with a secure and reliable messaging platform that meets their privacy needs.

8.3 Future Scope

1. Work with larger files : One of the potential improvements that can be made is that the application works seamlessly with files larger than 4 MB, without compromising the security of the application.
2. Strengthen the security : The encryption of the messages can be strengthened in addition to ensuring that the algorithm is efficient in terms of limiting the increase of file size due to encryption which in turn increases the transmission time.

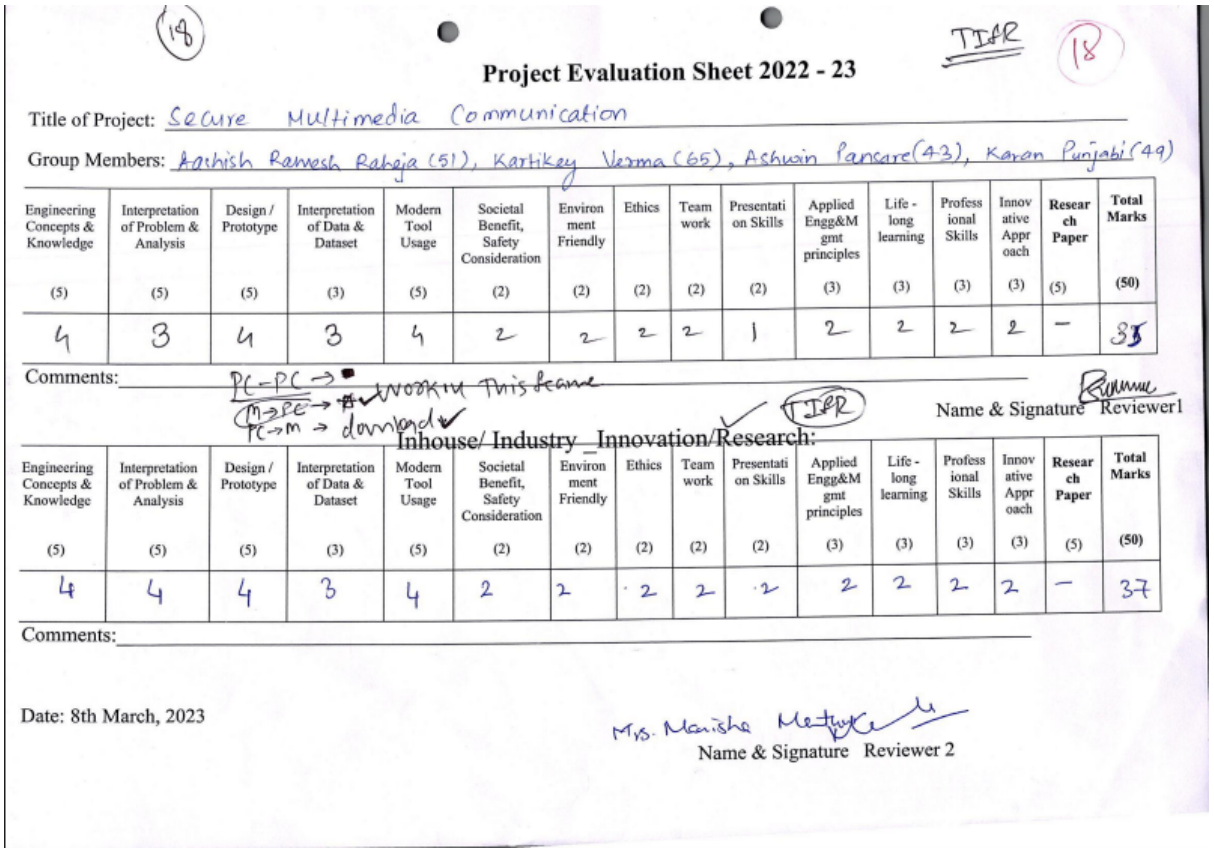
References

1. Cohn-Gordon, K., Cremers, C., Dowling, B., Garratt, L., & Stebila, D. (2020) “A Formal Security Analysis of the Signal Messaging Protocol.” Journal of Cryptology 33 (4): 1914– 1983.
2. Raman Singh, Hitesh Tewari. (2021) “Blockchain-Enabled End-to-End Encryption for Instant Messaging Applications.”
3. Botha, J.G., Van ‘t Wout, M.C. and Leenen, L. (2019) “A comparison of chat applications in terms of security and privacy.” 18th European Conference on Cyber Warfare and Security, University of Coimbra, Portugal.
4. Sabah, Noor & Mohamad, Jamal & Dhannoon, Ban N. (2017) “Developing an End-to-End Secure Chat Application.” International Journal of Computer Science and Network Security.
5. M. B. Kılıç , "Encryption Methods and Comparison of Popular Chat Applications", Advances in Artificial Intelligence Research, vol. 1, no. 2, pp. 52-59, Sep. 2021
6. Kseniia Ermoshina, Francesca Musiani, Harry Halpin. “End-to-End Encrypted Messaging Protocols: An Overview”. Internet Science. INSCI 2016. Lecture Notes in Computer Science(), vol 9934. Springer, Cham.
7. Discussion on projects using XMPP-based Instant Messaging on https://xmpp.org/uses/instant-messaging/
8. Document on the X3DH key agreement protocol on https://signal.org/docs/specifications/x3dh/
9. S. Blake-Wilson, D. Johnson, and A. Menezes, “Key agreement protocols and their security analysis,” in Cryptography and Coding: 6th IMA International Conference Cirencester, UK, December 17–19, 1997 Proceedings, 1997.
10. Xmpp.org. “Overview of XMPP” https://xmpp.org/about/technology-overview.html Accessed 26 Aug. 2022.
11. Signal.org. “The Double Ratchet Algorithm” on https://signal.org/docs/specifications/doubleratchet/ Accessed 26 Aug. 2022.
12. T. Perrin and M. Marlinspike, “The X3DH Key Agreement Protocol,” 2016. https://whispersystems.org/docs/specifications/x3dh/

Appendix

List of figures

Project Evaluation Sheet: Review 1



(18) TIAR (18)

Project Evaluation Sheet 2022 - 23

Title of Project: Secure Multimedia Communication

Group Members: Aashish Ramesh Raheja (51), Karthikey Verma (65), Ashwin Pansare (43), Karan Punjabi (49)

Engineering Concepts & Knowledge (5)	Interpretation of Problem & Analysis (5)	Design / Prototype (5)	Interpretation of Data & Dataset (3)	Modern Tool Usage (5)	Societal Benefit, Safety Consideration (2)	Environment Friendly (2)	Ethics (2)	Team work (2)	Presentation Skills (2)	Applied Engg&Mgmt principles (3)	Life - long learning (3)	Professional Skills (3)	Innovative Approach (3)	Research Paper (5)	Total Marks (50)
4	3	4	3	4	2	2	2	2	1	2	2	2	2	–	35

Comments: PC-PC → working this frame
M→PC
PC→M → download

Name & Signature Reviewer1

Inhouse/ Industry _Innovation/Research: TIAR

Engineering Concepts & Knowledge (5)	Interpretation of Problem & Analysis (5)	Design / Prototype (5)	Interpretation of Data & Dataset (3)	Modern Tool Usage (5)	Societal Benefit, Safety Consideration (2)	Environment Friendly (2)	Ethics (2)	Team work (2)	Presentation Skills (2)	Applied Engg&Mgmt principles (3)	Life - long learning (3)	Professional Skills (3)	Innovative Approach (3)	Research Paper (5)	Total Marks (50)
4	4	4	3	4	2	2	2	2	2	2	2	2	2	–	37

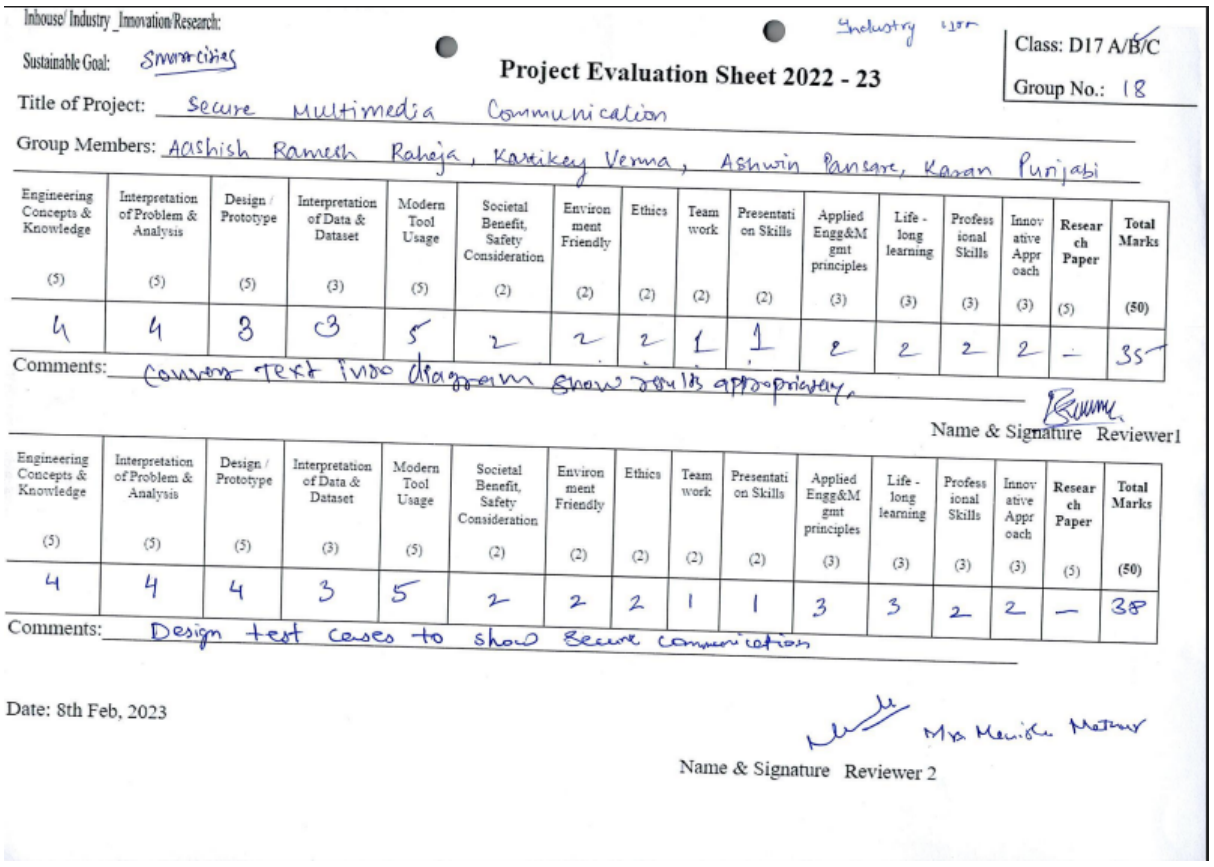
Comments:

Date: 8th March, 2023

Ms. Manisha Mehrotra

Name & Signature Reviewer 2

Project Evaluation Sheet: Review 2



Inhouse/ Industry _Innovation/Research: Industry

Sustainable Goal: Smart cities

Class: D17 A/B/C

Group No.: 18

Project Evaluation Sheet 2022 - 23

Title of Project: Secure Multimedia Communication

Group Members: Aashish Ramesh Raheja, Karthikey Verma, Ashwin Pansare, Karan Punjabi

Engineering Concepts & Knowledge (5)	Interpretation of Problem & Analysis (5)	Design / Prototype (5)	Interpretation of Data & Dataset (3)	Modern Tool Usage (5)	Societal Benefit, Safety Consideration (2)	Environment Friendly (2)	Ethics (2)	Team work (2)	Presentation Skills (2)	Applied Engg&Mgmt principles (3)	Life - long learning (3)	Professional Skills (3)	Innovative Approach (3)	Research Paper (5)	Total Marks (50)
4	4	3	3	5	2	2	2	1	1	2	2	2	2	–	35

Comments: Convert Text into diagram show results appropriately.

Name & Signature Reviewer1

Engineering Concepts & Knowledge (5)	Interpretation of Problem & Analysis (5)	Design / Prototype (5)	Interpretation of Data & Dataset (3)	Modern Tool Usage (5)	Societal Benefit, Safety Consideration (2)	Environment Friendly (2)	Ethics (2)	Team work (2)	Presentation Skills (2)	Applied Engg&Mgmt principles (3)	Life - long learning (3)	Professional Skills (3)	Innovative Approach (3)	Research Paper (5)	Total Marks (50)
4	4	4	3	5	2	2	2	1	1	3	3	2	2	–	38

Comments: Design test cases to show Secure communication

Date: 8th Feb, 2023

Ms Manisha Mehrotra

Name & Signature Reviewer 2