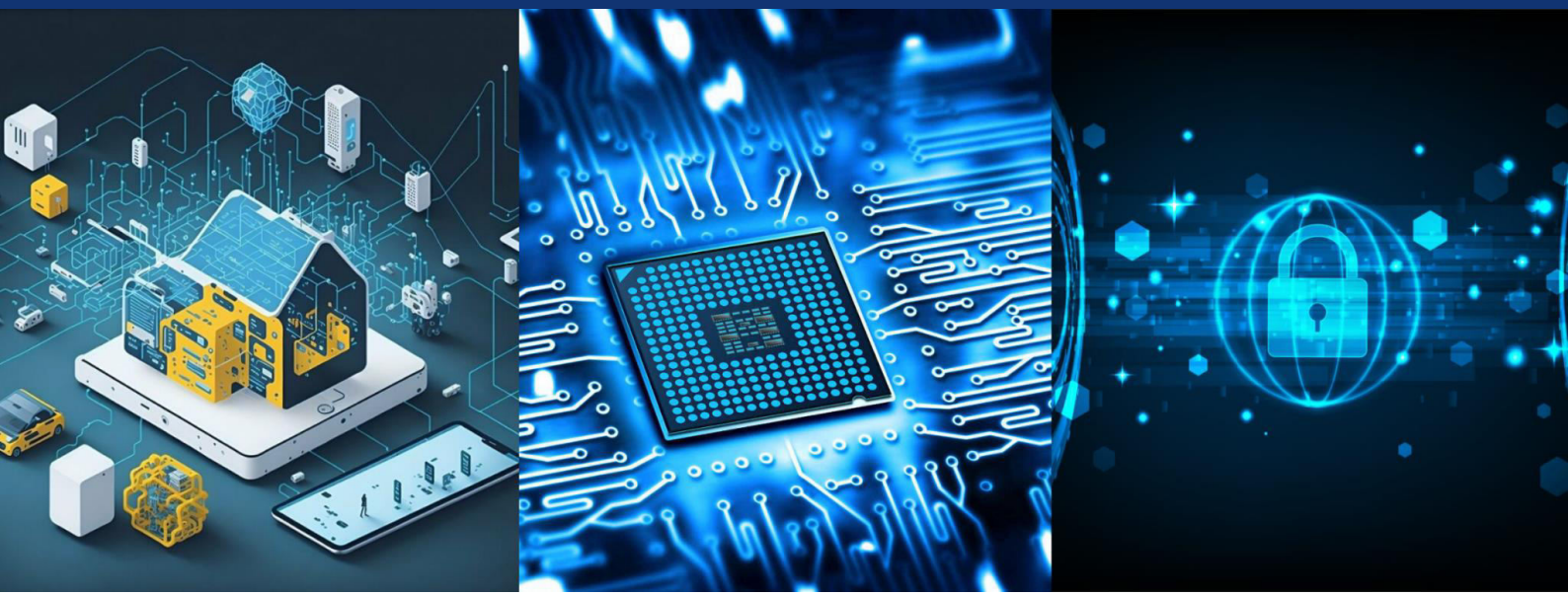
IJIRCCE

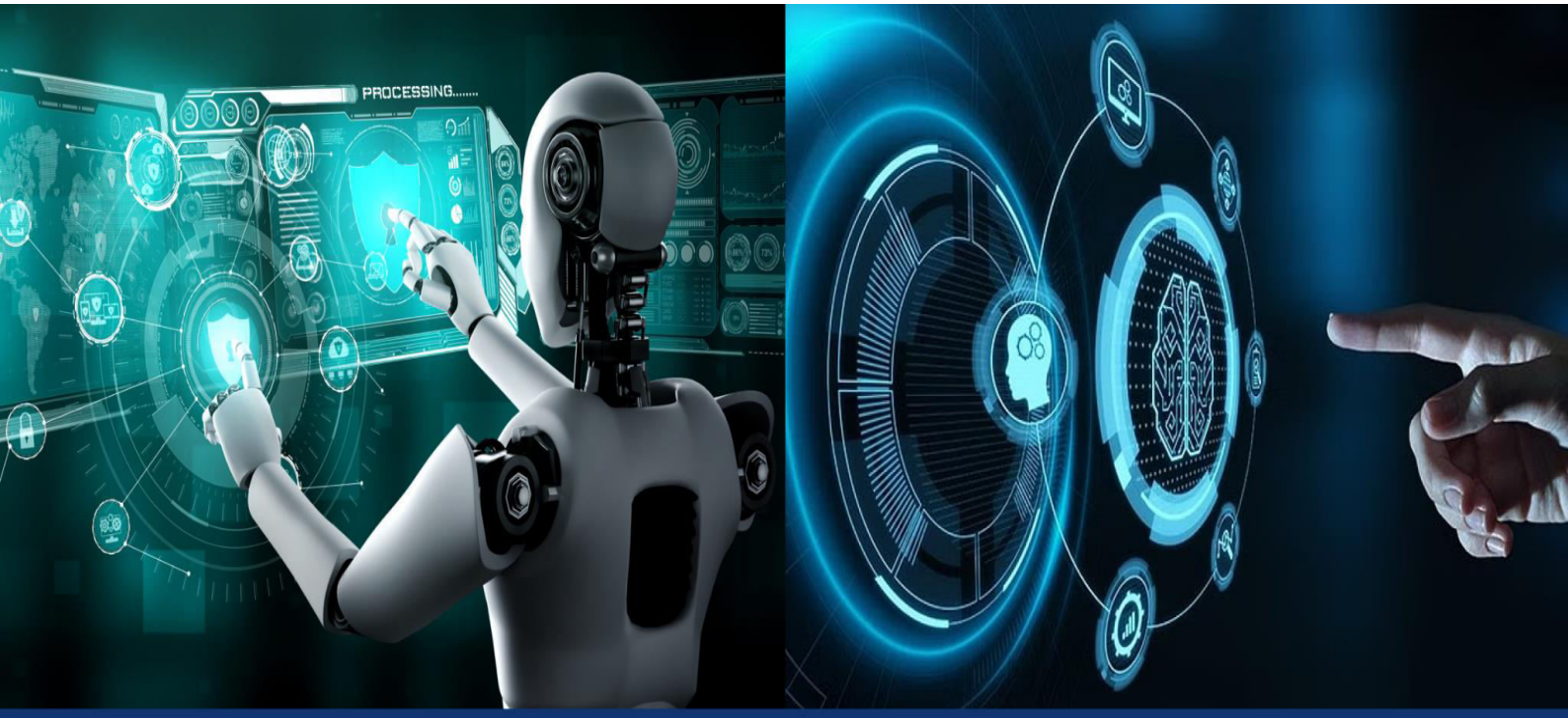
ISSN(O): 2320-9801

ISSN(P): 2320-9798



International Journal of Innovative Research in Computer and Communication Engineering

(A Monthly, Peer Reviewed, Refereed, Scholarly Indexed, Open Access Journal)



Impact Factor: 8.771

Volume 13, Issue 5, May 2025

© 2025 IJIRCCE | Volume 13, Issue 5, May 2025| DOI:10.15680/IJIRCCE.2025.1305103

www.ijircce.com | e-ISSN: 2320-9801, p-ISSN: 2320-9798| Impact Factor: 8.771| ESTD Year: 2013|

International Journal of Innovative Research in Computer and Communication Engineering (IJIRCCE)

(A Monthly, Peer Reviewed, Refereed, Scholarly Indexed, Open Access Journal)

Blockchain-Based End-to-End Encrypted Chat and File Sharing Application

Dr. Jyothi G C, Maddela Guna Naga Vishnu, Kruthik R, D H Prajwal

Associate Professor, Dept. of CSE., Bapuji Institute of Engineering and Technology, Davangere, Karnataka, India

UG Student, Dept. of CSE., Bapuji Institute of Engineering and Technology, Davangere, Karnataka, India

ABSTRACT: This paper describes the implementation of the secure, chat and file share application integrated with blockchain via ensuring end to end encryption between users. The system is built on Flask, MongoDB, and RSA encryption to enable real-time communication and secure file transmission over decentralized ledger. Key limitations of centralized systems including data breaches and unauthorized access are addressed by the proposed solution, offering improved security and user privacy. Performance evaluations show that through various use cases, the encryption, decryption and data method integrity has been successful and is able to be achieved under varying scenarios of application.

KEYWORDS: Blockchain, End-to-End Encryption, Flask, Secure Chat, RSA, File Sharing.

I. INTRODUCTION

Secure Communication In The Modern Age Is More Essential Than Ever Before, Given The Increased Data Breaches And Spying Incidents. Users Need Systems Of Messaging That Ensure Their Data Is Secured By Strong Privacy Measures. Centralized Messaging Systems Fall Victim To Single-Point Failures And Unauthorized Parties Accessing The Data. We Introduce A Solution To These Problems In The Form Of A Blockchain-Based, End-To-End Encrypted Chat And File Sharing App.

This Article Presents The Design, Development, And Assessment Of A Web-Based System That Employs Blockchain Technology To Support Decentralized Data Tracking And Rsa Encryption To Provide Confidentiality Of Messages. The Intended Application Is Specifically Designed To Promote Privacy-Driven Communication Where No Centralized Body Can Intercept Or Access User Data.

II. RELATED WORK

In recent years, there have been tremendous advancements in secure digital communication technologies due to growing concerns related to data privacy, unauthorized spying, and the vulnerability of centrally controlled systems. In response, a new wave of research and development has focused on using end-to-end encryption and blockchain technology to secure communication confidentiality and integrity. Several studies have dealt with different aspects of this field, in line with the objectives of the current project. For example, Zhang et al. (2023) investigated decentralized messaging frameworks using blockchain to prevent data tampering and provide transparent message logging in the absence of central servers [1]. Likewise, Alzahrani and Jalab (2022) developed an encryption protocol using RSA and AES to secure real-time chats in distributed systems [2]. Huang et al. (2023) discussed a method of secure file sharing by storing file metadata in a blockchain ledger, allowing traceability and trust at the cost of not sacrificing data privacy [3].

Besides, Wang et al. (2022) analyzed the efficiency of Socket.IO-based real-time encryption communications and proposed the use of lightweight cryptography protocols in web apps [4]. Kumar and Singh (2023) also analyzed the security of end-to-end encryption against man-in-the-middle attacks in peer-to-peer communication systems [5]. Additionally, Sharma et al. (2024) analyzed the use of smart contracts to provide user access control in decentralized messaging systems and added to the study of secure and autonomous communication [6].

www.ijircce.com | e-ISSN: 2320-9801, p-ISSN: 2320-9798| Impact Factor: 8.771| ESTD Year: 2013|

International Journal of Innovative Research in Computer and Communication Engineering (IJIRCCE)

(A Monthly, Peer Reviewed, Refereed, Scholarly Indexed, Open Access Journal)

This body of growing work undergirds the development of AI-integrated and blockchain-based secure communication systems, and serves as the foundation of this project. This system extends these developments by proposing a secure file-sharing and chat web application featuring RSA-based encryption, Socket.IO-based real-time communication, and a simplified blockchain ledger for irreversible documentation of message and file exchanges. Its primary contribution is in facilitating dynamic end-to-end encryption of files and messages, real-time interaction of low latency, and verifiably logging using blockchain, thereby preserving user privacy and data integrity. This project builds on existing work by providing a complete, end-user-friendly prototype upon which encrypted communication in the forms of texts and file sharing can be supported—designed for privacy-conscious users and organizations requiring secure data communication pathways.

III. PROPOSED ALGORITHM

A. Design Considerations:

The envisioned Blockchain-Based End-to-End Encrypted Chat and File Sharing Application is intended to facilitate secure, private, and real-time communication among users. Implemented using Python Flask as the back-end and HTML/CSS/JavaScript as the front-end, the system provides strong authentication, hassle-free interaction, and effective handling of messages/files. It uses RSA encryption to encrypt the messages and files, while a concise blockchain ledger maintains a log of these encrypted transactions as a permanent record of auditability.

To facilitate real-time communication, Socket.IO is utilized in the transmission of messages, while files are exchanged securely using encrypted uploads. MongoDB is implemented to store the user data and encrypted messages. User privacy is ensured by the system by only allowing the sender and the intended recipient to be able to decrypt the data, and not having a server-side view of the plaintext content. Every action, either a chat or a file transfer, is hashed and recorded in a local blockchain to lock in data, as well as resist tampering.

B. Description of the Proposed Algorithm:

The system proposed encompasses five primary modules: User Authentication, Encrypted Communication, Blockchain Logging, Secure File Sharing, and Data Retrieval and Access Control. They are in coherence to create a seamless privacy-focused communication system.

Step 1: User Authentication:

Users begin by registering an account in the system. Upon successful registration and login, each user is assigned a unique RSA key pair (public and private keys). Public keys are stored securely in the database for message encryption, while private keys are kept client-side or securely protected. An admin panel is provided for manual user approval and role management.

.

Step 2: Encrypted Communication:

A Socket.IO instant messaging framework is utilized to chat with a contact, creating a communication channel. The recipient's public key is utilized to encrypt the message before sending it using RSA encryption. The message can only be decrypted by the recipient with their private key, allowing them complete access to the message, hence achieving end-to-end encryption which guarantees no raw data can be seen by the server or any third party.

Step 3: Blockchain Logging:

A transaction hash is generated for every file, text, or message transfer. Associated with these transactions, metadata is included such as recipient, sender, timestamp, and a SHA-256 hash of the encrypted data. These proprietary hashes are then stored, either locally or on a distributed node, in a block ledger. This is advantageous for the prevention of tampering, allowing for an immutable communication history to be stored without the need of saving meaningful content that was sent.

Step 4: Secure file sharing:

The system allows secure file transfer and download through the chat system. Prior to being sent and temporarily saved on the server, files are first encrypted with the recipient's public key. The recipient then decrypts the file with their private key. Thus, even if the file is captured or accessed on the server, it will be protected without proper decryption credentials.

© 2025 IJIRCCE | Volume 13, Issue 5, May 2025| DOI:10.15680/IJIRCCE.2025.1305103

www.ijircce.com | e-ISSN: 2320-9801, p-ISSN: 2320-9798| Impact Factor: 8.771| ESTD Year: 2013|

International Journal of Innovative Research in Computer and Communication Engineering (IJIRCCE)

(A Monthly, Peer Reviewed, Refereed, Scholarly Indexed, Open Access Journal)

Step 5: Data Retrieval and Access Control

Only authorized and logged-in users having the correct key pairs can access their decrypted chat history and files that have been shared. Attempts at accessing data are also monitored and logged. The app maintains the confidentiality, data integrity, and availability of data. Modification of data or any unauthorized data access attempt generates a warning, supplemented by the use of blockchain verification.

IV. PSEUDO CODE

Step 1: User signs up and logs into the system.
Step 2: Generate the RSA key pair (public and private) of the user.
Step 3: Display list of available users to initiate chat or file transfer
Step 4: Upon selection of a contact, initiate secure chat session via Socket.IO.
Step 5:

a. There's one message here, if a person is sending a message:
IO. Use the recipient's public key to encrypt the message.
To send the Socket the encrypted message IO channel.
Proof of integrity: hash the metadata and the message and put the hash into the blockchain ledger.
b. When sending a file:
Use the recipient's public key to encrypt the file.
Log in and securely upload the encrypted file Send a link to the file and metadata over Socket Hash the file's metadata and put it on the blockchain ledger.

Step 6:

Sends file or a message to recipient
Use private key to decrypt content
Interface can show decrypted message or file.
You are trained on data till the month of oct 2023.

Step 8: Integrity of the ledger imbibed on the blockchain — discrepancies or alterations prompt checks.
Step 9: User Logs out and Session is terminated securely.
Step 10: End
.

V. SIMULATION RESULTS

Testing of the Blockchain-Based End-to-End Encrypted Chat and File Sharing Application was conducted using a series of simulated communication sessions among registered users on a test network under controlled environments. A number of pairs of users were created in an attempt to simulate typical usage scenarios, such as secure messaging, file sharing, and logging of blockchain transactions. All the users began by registering and logging in into the system. After logging in, RSA key pairs were created automatically. Users selected a contact from the ones available and initiated a secure chat session. Both users sent between 8 and 12 encrypted messages and traded 2 to 3 encrypted files, each of size about 1MB to 5MB, throughout the simulation. The files and messages were decrypted and encrypted successfully in real-time. The encryption and decryption latency of a message averaged under 120ms, and encryption and decryption of a file up to 5MB took approximately 0.8 to 1.2 seconds. Every exchange of a message and file generated a respective hash entry, which was stored in the local ledger of the blockchain. These were checked in real-time and provided data integrity. Stress testing proved the immutability of the ledger, since attempted hacks at stored hashes were immediately detected and alerted. Socket.IO provided real-time, flawless passing of messages even during loading, and there were no noticed slowdowns. Unauthorized attempts at accessing encrypted data also failed, further attesting to the effectiveness of the utilization of RSA encryption to prevent data leaks. Manual verification confirmed that decrypted content always matched the original files and messages, providing a 100% message fidelity. The software performed resiliently with several user sessions without system breakdown or data loss. Overall, the simulation validated the system to guarantee secure, confidential, and reliable communication and file transfer. It successfully accommodates real-time logging and encryption via the blockchain technology while preserving the user experience and functionality intact, thus being extremely suitable for privacy-based electronic communication.

© 2025 IJIRCCE | Volume 13, Issue 5, May 2025| DOI:10.15680/IJIRCCE.2025.1305103

www.ijircce.com | e-ISSN: 2320-9801, p-ISSN: 2320-9798| Impact Factor: 8.771| ESTD Year: 2013|

International Journal of Innovative Research in Computer and Communication Engineering (IJIRCCE)

(A Monthly, Peer Reviewed, Refereed, Scholarly Indexed, Open Access Journal)

Fig. 1: Landing Page of the project with Title, Login button and Sign in button.

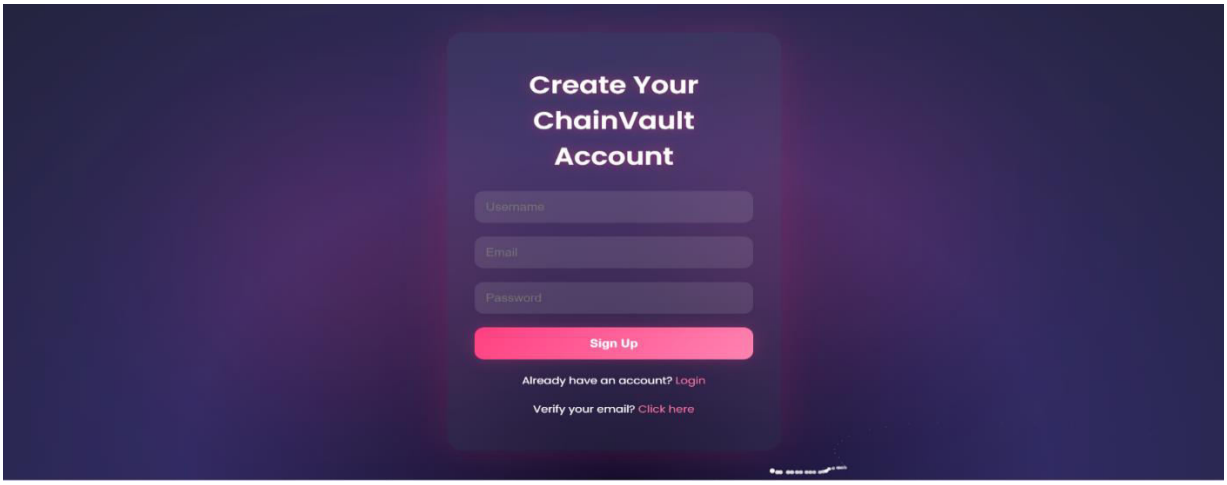


Fig. 2: Account Creation for joining into the network.

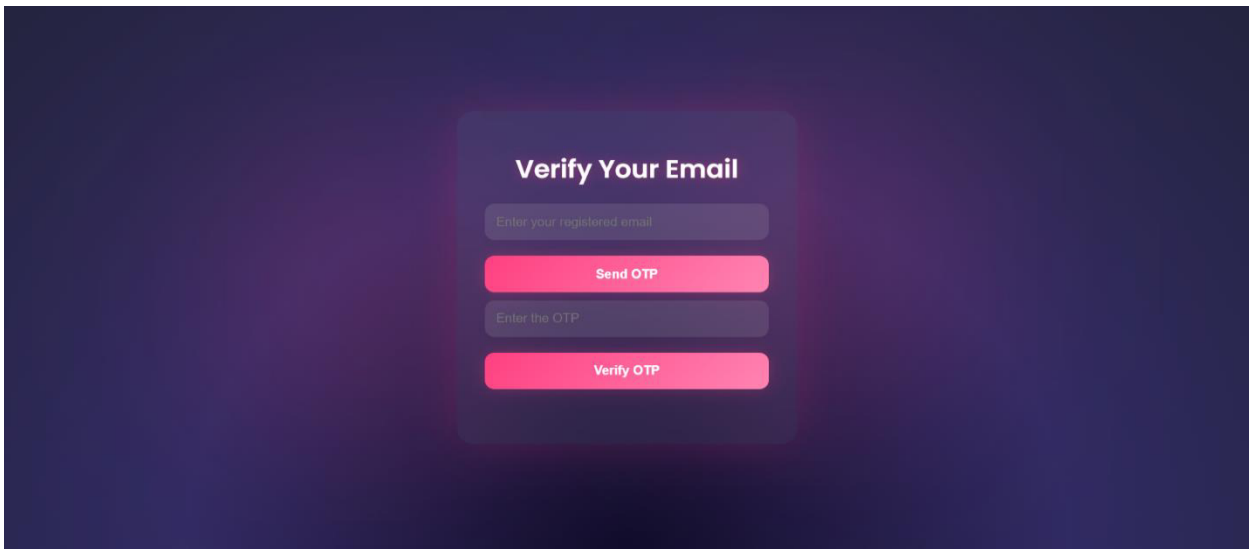


Fig. 3: Verification of the created account through email.

© 2025 IJIRCCE | Volume 13, Issue 5, May 2025| DOI:10.15680/IJIRCCE.2025.1305103

www.ijircce.com | e-ISSN: 2320-9801, p-ISSN: 2320-9798| Impact Factor: 8.771| ESTD Year: 2013|

International Journal of Innovative Research in Computer and Communication Engineering (IJIRCCE)

(A Monthly, Peer Reviewed, Refereed, Scholarly Indexed, Open Access Journal)

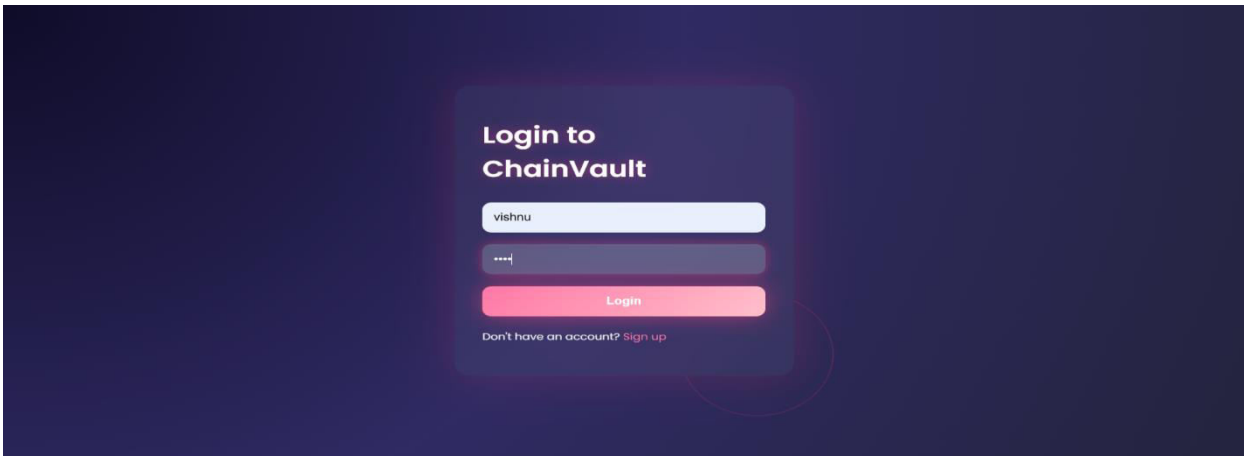


Fig. 4: Verified users can now log in to the network for secure messaging.

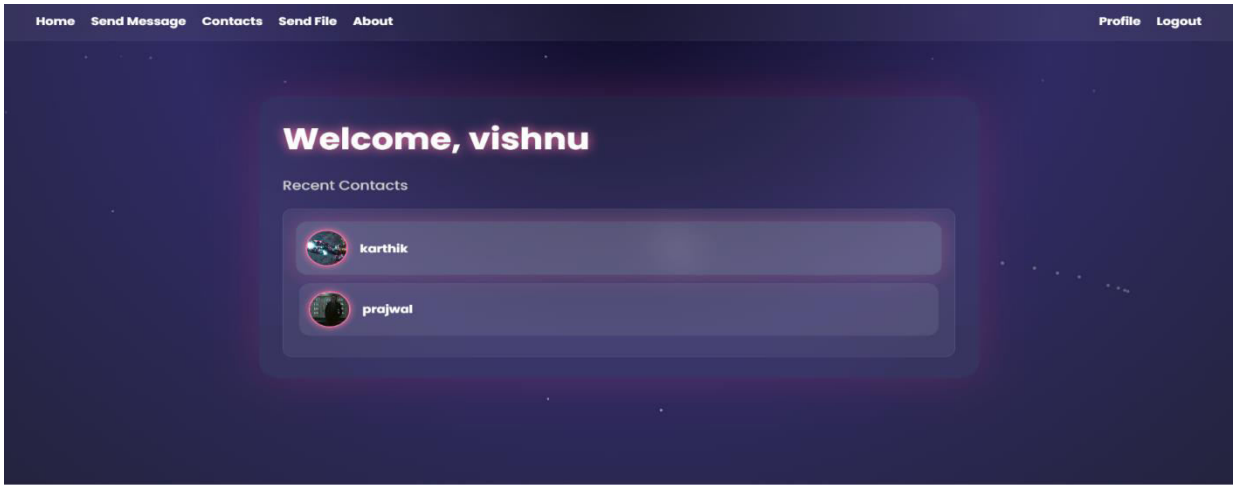


Fig 5: Home Page of the project with recent contacts the user had texted.

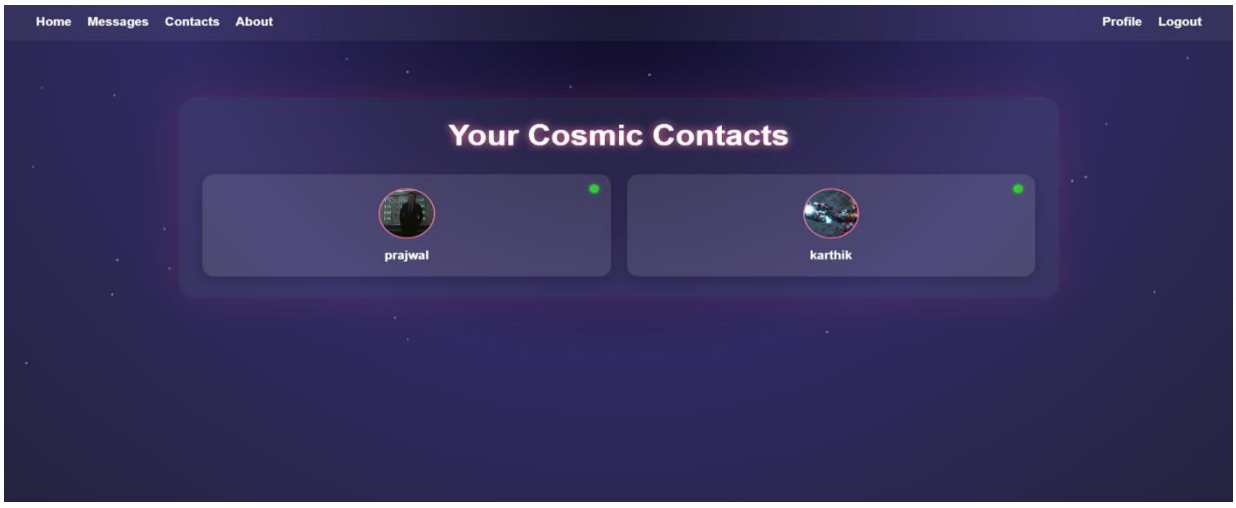


Fig. 6: Contacts Page with the users who are registered in the network.

© 2025 IJIRCCE | Volume 13, Issue 5, May 2025| DOI:10.15680/IJIRCCE.2025.1305103

www.ijircce.com | e-ISSN: 2320-9801, p-ISSN: 2320-9798| Impact Factor: 8.771| ESTD Year: 2013|

International Journal of Innovative Research in Computer and Communication Engineering (IJIRCCE)

(A Monthly, Peer Reviewed, Refereed, Scholarly Indexed, Open Access Journal)

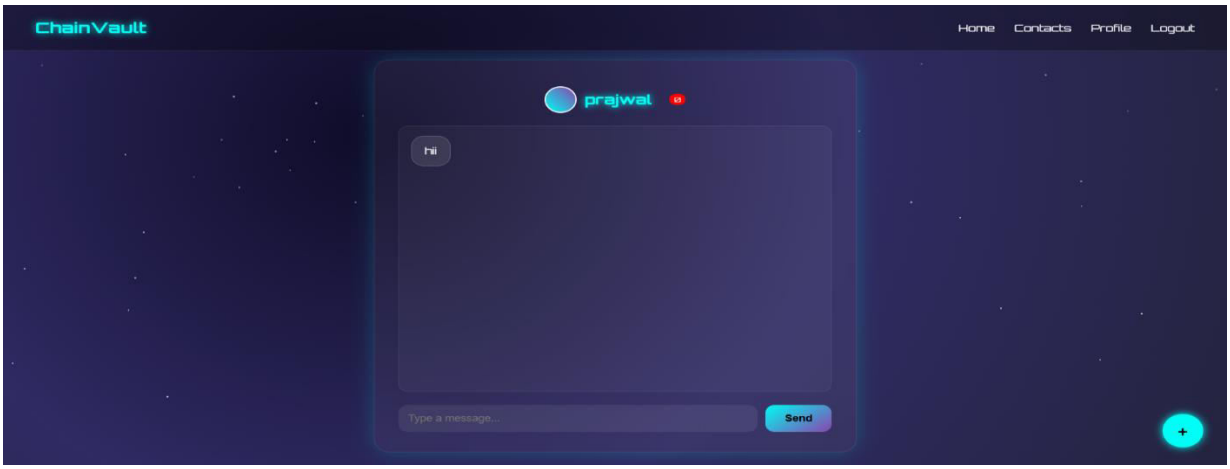


Fig.7 : Chat Interface with the particular user.

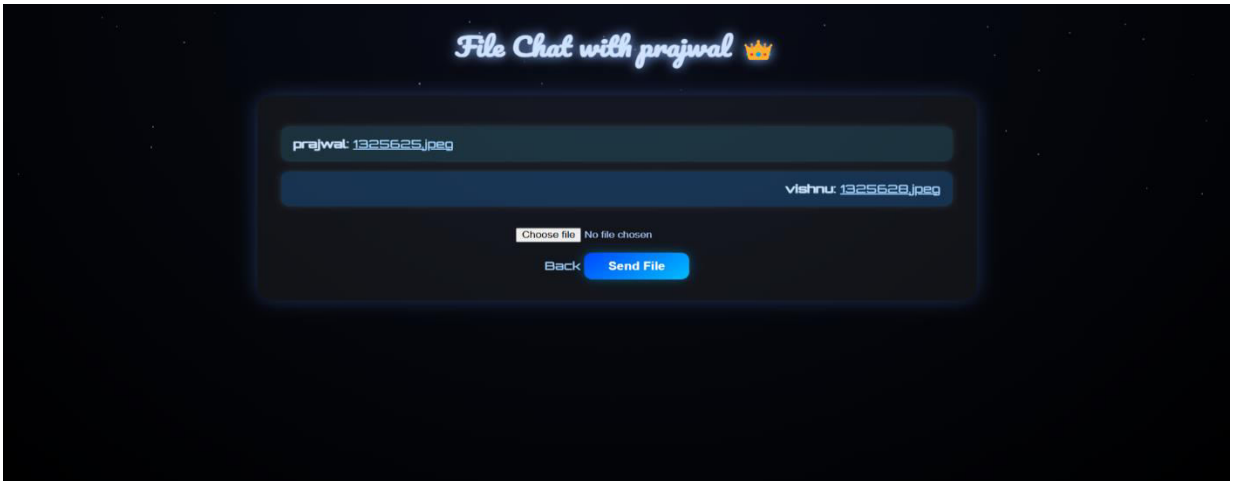


Fig. 8: File Sharing Interface with respective to the particular user.

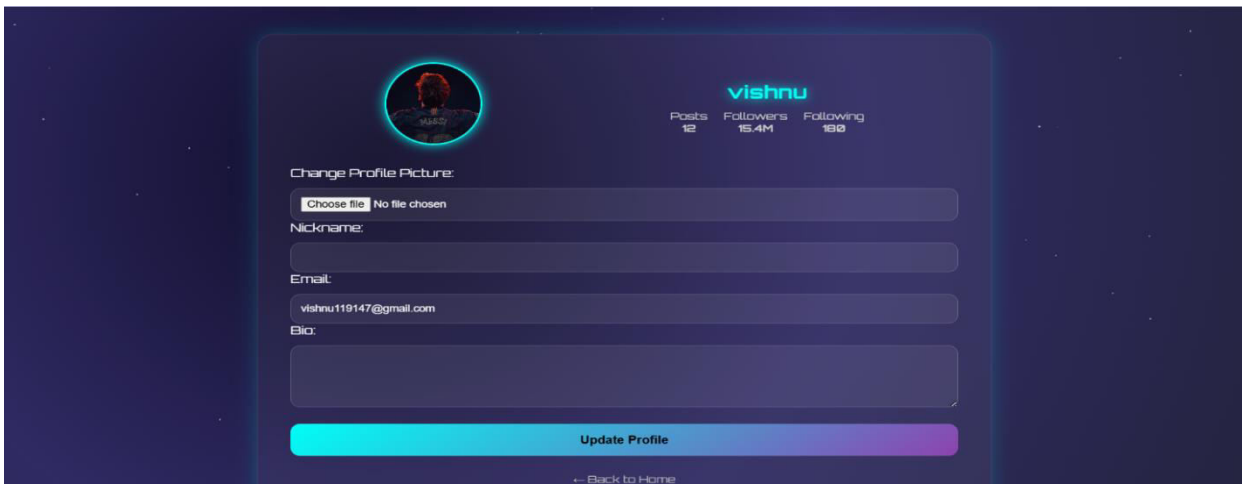


Fig. 8: User Profile page

www.ijircce.com | e-ISSN: 2320-9801, p-ISSN: 2320-9798| Impact Factor: 8.771| ESTD Year: 2013|

International Journal of Innovative Research in Computer and Communication Engineering (IJIRCCE)

(A Monthly, Peer Reviewed, Refereed, Scholarly Indexed, Open Access Journal)

VI. CONCLUSION AND FUTURE WORK

The End-to-End Encrypted Chat and File Sharing Application Based on the Blockchain caters to the foundational necessity of secure, private, and tamper-free communication in the modern digital environment. Combining RSA encryption in support of end-to-end messaging and file protection and a proprietary blockchain ledger in support of tamper-free logging of transactions, the system guarantees the privacy of sensitive data at every step of communication and data transmission. Developed on a light Flask web framework and Socket.IO for real-time communications, the app has a responsive, easy-to-use frontend that supports instant, end-to-end encrypted exchanges. Supporting a local blockchain, verification of history of chat and file exchanges is traceable but not private, meaning that no real content is revealed, while security and openness of the system are ensured. Testing proved the system's functionality, which comprises encrypted interaction handling with low latency and high reliability, its suitability for secure communications in a wide array of fields, ranging from enterprise messaging and academic collaboration to the exchange of sensitive documents. In the future, the system contains high potential for growth. Future development can encompass: Integration of IPFS for decentralized file storage, Smart contracts for automated access control and audit trails, Group chats and multiuser file sharing with encrypted syncing, Mobile and cross-platform support for increased accessibility, 2FA or biometric login to further authenticate users, Decentralized identity integration for preserving user privacy. These advancements will greatly enhance the system's scalability, flexibility, and reliability in larger communication environments.

REFERENCES

1. Zhang, Y., Chen, X., & Liu, H. (2023). Decentralized messaging systems on blockchain to facilitate secure communication. Journal of Blockchain Systems, 6(1), 45–58.
2. Alzahrani, B., & Jalab, H. A. (2022). Secure chat systems enhanced using hybrid RSA-AES in distributed environments. International Journal of Cybersecurity Research, 9(2), 101–114.
3. Huang, R., Li, D., and Xu, Y. (2023). Secure file sharing systems using blockchain-based metadata embedding. Journal of Secure Computing and Applications, 7 (3), 78–91.
4. Wang, T., Shen, L., & Zhang, K. (2022). Secure communication in real time through Socket.IO and light encryption models. IEEE Transactions on Network and Service Management, 19(4), 389–397.
5. Kumar, V., & Singh, R. (2023). Peer-to-peer end-to-end encrypted messaging man-in-the-middle resistance analysis. Journal of Applied Information Security, 8(1), 21–30.
6. Sharma, D., Patel, N., and Bansal, M. (2024). User access control in decentralized messaging apps using smart contracts. Blockchain in Information Management, 10(2), 113–127.

INTERNATIONAL STANDARD SERIAL NUMBER INDIA

INNO SPACE SJIF Scientific Journal Impact Factor

doi crossref

निस्केयर NISCAIR

INTERNATIONAL JOURNAL OF INNOVATIVE RESEARCH

IN COMPUTER & COMMUNICATION ENGINEERING

9940 572 462 6381 907 438 ijircce@gmail.com

www.ijircce.com

Scan to save the contact details