VISVESVARAYA TECHNOLOGICAL UNIVERSITY

“JNANA SANGAMA”, BELAGAVI - 570018, KARNATAKA

A Synopsis Report
On

" QuatChat : Quantum-Secure Messaging Application, Redefining Privacy and Security in the Quantum Era"

In the partial fulfilment of the requirement for the award of degree

BACHELOR OF TECHNOLOGY
IN
COMPUTER SCIENCE AND ENGINEERING

Submitted by

PAVAN CHANDRAPPA HOTTIGOUDRA	**(4VZ22CS019)**
SHIVANI VEERESH LINGADAHALLI	**(4VZ22CS026)**
SOURABHA SHRENIKRAJA HALLI	**(4VZ22CS029)**
VIKAS	**(4VZ22CS031)**

Under the guidance of

Mr. Yogesha T
Assistant Professor,
Dept. of Computer Science and Engineering,
VTU Regional Center, Mysore.

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING
VISVESVARAYA TECHNOLOGICAL UNIVERSITY
Centre for Post Graduate Studies, Sathagalli, Mysuru – 570029.
2024 – 2025

VISVESVARAYA TECHNOLOGICAL UNIVERSITY
“JNANA SANGAMA”, BELAGAVI - 570018, KARNATAKA

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING
Centre for Post Graduate Studies, Sathagalli, Mysuru – 570029.
2024 – 2025

CERTIFICATE

This is to certify that the Project work entitled **"QuatChat : Quantum-Secure Messaging Application, Redefining Privacy and Security in the Quantum Era"** is a bonafied work carried out by **Pavan Chandrappa Hottigoudra, Shivani Veeresh Lingadahalli, Sourabha Shrenikraj Halli, Vikas** bearing USN **4VZ22CS019, 4VZ22CS026, 4VZ22CS029, 4VZ22CS031** at Department of **Computer Science and Engineering, Visvesvaraya Technological University, Centre for Post Graduate Studies, Mysuru** in partial fulfilment for the award of **Bachelor of Technology in Computer Science and Engineering, Visvesvaraya Technological University, Belagavi during the academic year 2024-2025.** It is certified that all the corrections/suggestions indicated during Internal Assessment have been incorporated in the report. The Mini Project report has been approved as it satisfies the academic requirements in respect of Project work prescribed for the Bachelor of Technology degree.

Signature of the Guide
Mr. Yogesha T
Assistant Professor,
Dept. of CS&E,
VTU, CPGS Mysuru – 570029

Signature of Program Coordinator
Dr. G. F. ALI AHAMMED
Program Coordinator,
Dept. of CS&E,
VTU, CPGS Mysuru - 570029

Name of the Examiners: **Signature with date**

1.

2.

ABSTRACT

In today's digital world, keeping our personal conversations private is more important than ever. This project introduces a **Quantum-Secure Messaging Web Application** that is designed to protect user privacy not just today, but also in a future where quantum computers could break traditional security methods. Unlike regular chat apps that rely on centralized servers, this web application uses **peer-to-peer communication**, allowing users to talk directly to each other. It ensures **end-to-end encryption** using **AES-256-GCM**, one of the strongest encryption standards available today. To protect against future quantum attacks, the system uses **Kyber**, a **post-quantum cryptographic algorithm**, to safely exchange encryption keys. This means that even powerful quantum computers in the future won't be able to decode private messages.

Each user is given a **Decentralized Identifier (DID)** that's linked to their mobile number through a secure **OTP (One-Time Password)** process. These identities are managed securely, with metadata stored in an encrypted database like **MongoDB** or **MySQL**. The application runs smoothly in the browser and is built using modern tools like **React.js** for the frontend and **Node.js with Express** for the backend. It uses **IPFS (InterPlanetary File System)** and **libp2p** for decentralized message storage and routing, removing the need for a central server and enhancing privacy and scalability.

While it mainly supports secure text messaging, **optional features** like **voice and video calls** using **WebRTC** are available for users who want them. Overall, this project delivers a **future-proof messaging platform** that gives users full control over their communication making it secure, decentralized, and ready for the next generation of the internet.

TABLE OF CONTENTS

CHAPTER 1

INTRODUCTION

In today's rapidly evolving digital landscape, securing personal communication has become more critical than ever. Conventional messaging apps are based on centralized servers and are prone to data leakage, eavesdropping, and unauthorized interception.[6] Our system a **Quantum-Secure Messaging Application** is designed to redefine privacy by combining **post-quantum cryptography**, **decentralized identity management**, and **peer-to-peer networking** into one seamless, efficient platform. Decentralized operation make use of peer-to-peer networks, this ensures that no network failure can do due to central node failure. Blockchain serves as an unalterable ledger which allows messaging to take place in a decentralized manner.[1]

At its core, the application leverages **Kyber**, a post-quantum encryption algorithm, to ensure that even the most powerful quantum computers cannot compromise private conversations. Every message is encrypted with **AES-256-GCM**, protected through Kyber-secured key exchange, and distributed securely using **IPFS (InterPlanetary File System)**, **WebSockets**, and **libp2p's decentralized peer-to-peer protocol**. IPFS establishes a permanent distributed network by using content-based reference.[5] This decentralized architecture eliminates single points of failure, enhances data redundancy, and ensures high availability of information even if some nodes go offline. This future-proof cryptographic foundation guarantees end-to-end confidentiality, forward secrecy, and resilience against quantum threats.

Rather than relying on centralized servers, the application uses a fully decentralized architecture based on Web3 principles. Peer-to-peer networks distribute resources, with each node acting as both provider and consumer, thus enhancing scalability and reducing dependency on a single messaging server.[3] Technologies like **IPFS**, **WebSockets**, and **Kademlia Distributed Hash Tables (kadDHT)** power secure storage, message routing, and global peer discovery, ensuring users can communicate directly without intermediaries.
Random connections between peers carry out resource discovery and information dissemination.[4] This design not only enhances security and privacy but also improves network scalability and fault tolerance.

User identity management is decentralized and quantum-secure. Each user is assigned a **Decentralized Identifier (DID)** linked to their mobile number through a secure, OTP-based authentication system. Each Internet-connected computer needs to install and run a computer application specific to the ecosystem they wish to participate in.[2] Critical metadata, such as DIDs and public keys, are securely managed via MongoDB (or optionally, MySQL) under strict encryption policies, ensuring real-time access without exposing sensitive information. Data and applications are kept on a single, central server in centralized servers, making them vulnerable to security flaws and hacking.[8]

The backend is powered by **Node.js with Express.js**, handling **API orchestration, WebSocket communication management, encryption services, peer session management, and message routing**. The frontend is built using **React.js** for a responsive, dynamic user experience, coupled with **Redux.js** for efficient state management across authentication, messaging, and encryption workflows.

The proposed solution uses WebRTC, a modern peer to peer web communication technology as the underlying framework that facilitates secure communication.[5] HTML5 and CSS3 standards are employed to ensure a fast, adaptive, and user-friendly interface.The system optionally supports **WebRTC** for secure peer-to-peer **voice and video calls**, maintaining the same quantum-resistant security for call metadata. WebRTC can only connect two peers if their public IPs are known.[5] WebSockets are simultaneously leveraged to maintain **real-time, low-latency messaging** between peers for text and metadata communication. Waku v2 integration for enhanced decentralized message relay is also available as an optional feature for scaling under large network loads.

By automating critical cryptographic operations, enabling decentralized direct messaging, and minimizing server reliance, the system ensures that user conversations remain private, resilient, and future-proof even against the advent of quantum computing. Decentralized chat applications, therefore, offer a safer and more private solution for those wishing to connect online.[8] This messaging application marks a significant leap forward in digital privacy, empowering users with complete control over their communications and setting a new standard for secure, decentralized interactions in the Web3 era.

1.1 Aim :

We’re building a Quantum-Secure Messaging App that takes privacy and security to the next level. It uses cutting-edge post-quantum cryptography, decentralized identity management, and peer-to-peer networking to keep your conversations safe. The app relies on Kyber for quantum-resistant key exchanges and AES-256-GCM for strong end-to-end encryption. Instead of depending on centralized servers, it uses IPFS and libp2p to store and route messages in a fully decentralized way.

Users will manage their identities with Decentralized Identifiers (DIDs), secured through simple OTP-based authentication. Beyond messaging, the app will also support quantum-secure voice and video calls, ensuring scalable, secure communication for the future. Our goal is to create a powerful, resilient platform that keeps your privacy intact, even as quantum computing becomes a reality.

1.2 Problem statement :

With the rise of quantum computing, existing encryption methods are becoming increasingly vulnerable, putting the privacy of digital communication at serious risk. At the same time, centralized messaging platforms expose users to threats like data breaches, censorship, and surveillance. This project addresses these challenges by creating a secure, decentralized messaging application that uses post-quantum cryptography and peer-to-peer networking to ensure long-term privacy, data ownership, and resilience against both current and future threats.

1.3 Motivation :

We’re developing a Quantum-Secure Messaging App because secure, private communication is more important than ever—especially with the rise of quantum computing threats. Our goal is to give users full control over their conversations by using post-quantum cryptography, decentralized identity management, and peer-to-peer networking.

By cutting out centralized servers, we’re working to eliminate risks like data breaches and ensure that sensitive information stays private and protected, even against future quantum attacks. With technologies like Kyber encryption, AES-256-GCM, IPFS, and libp2p, we're building a system that not only boosts user trust and privacy but also sets a new standard for secure, decentralized communication.

At the heart of this project is a simple mission: to protect personal conversations, keep digital interactions authenticpro, and drive innovation toward a future where communication stays safe—no matter what new technologiesemerge.

1.4 Objective :

- **Protect Communication Against Quantum Threats:** Build a messaging platform that uses post-quantum cryptography to stay secure even against future quantum computer attacks.
- **Ensure Complete Privacy and User Control:** Give users full ownership of their conversations and identities by removing reliance on centralized servers and using decentralized technologies.
- **Deliver End-to-End Encryption:** Use strong, proven encryption methods like Kyber for key exchange and AES-256-GCM for securing messages, ensuring that only intended recipients can read the conversations.
- **Implement Decentralized Identity Management:** Let users create and manage their digital identities safely through Decentralized Identifiers (DIDs) secured with simple OTP-based authentication.
- **Enable Real-Time, Peer-to-Peer Communication:** Support fast, direct messaging without intermediaries by using IPFS and libp2p for decentralized data storage and routing.
- **Support Quantum-Secure Voice and Video Calls:** Extend security beyond text by offering optional voice and video communication that's protected against quantum threats.
- **Set a New Standard for Privacy and Trust:** Create a platform that not only meets today’s privacy needs but also inspires confidence and trust for the future of digital communication.

CHAPTER 2

LITERATURE SURVEY

This paper presents a decentralized messaging platform built on blockchain technology, replacing centralized servers with a peer-to-peer network. By leveraging blockchain's immutable ledger and encryption capabilities, it strengthens communication security and ensures continuity even if some nodes go offline. However, the system still relies on classical cryptographic methods, which may become vulnerable as quantum computing advances.[1] To address these concerns, our proposed Quantum-Secure Messaging Application integrates post-quantum cryptography, using Kyber for key exchanges. It also builds on decentralized principles by utilizing IPFS for storage and libp2p for peer-to-peer communication, while implementing Decentralized Identifiers (DIDs) for secure identity management. This approach ensures greater resilience against quantum threats while maintaining decentralization and enhancing forward secrecy.

The system described focuses on building a decentralized chat platform through blockchain, eliminating reliance on central servers. It uses Ethereum and the Whisper protocol to enable encrypted peer-to-peer messaging. Although it successfully tackles server dependency and improves user privacy, it continues to depend on conventional cryptographic techniques that are expected to be broken by future quantum computers.[2] Our solution strengthens this model by incorporating quantum-safe encryption, specifically Kyber for secure key exchanges and AES-256-GCM for data encryption. Rather than relying solely on blockchain frameworks like Ethereum, we integrate IPFS and libp2p to create a more scalable, efficient, and quantum-resistant peer-to-peer messaging system designed to withstand both current and emerging cyber threats.

SendingNetwork introduces a robust decentralized messaging protocol using libp2p, dynamic relay nodes, and Proof of Relay/Availability mechanisms. It enhances scalability and privacy through encryption schemes like the Double Ratchet Algorithm for group communications. Although the system greatly advances decentralized infrastructure, it still depends on classical cryptographic standards, leaving it vulnerable to future quantum decryption methods.[3] Our Quantum-Secure Messaging Application builds upon these foundations by embedding post-quantum encryption techniques at the core, particularly Kyber for key establishment. It further enhances decentralized communication using IPFS for distributed storage and DIDs for secure, user-controlled identity. This results in a highly scalable and quantum-resilient messaging platform that ensures privacy even against next-generation computing attacks.

This paper focuses on optimizing message transmission in libp2p's GossipSub protocol by introducing fragmentation and staggering techniques. These improvements significantly reduce network latency and improve bandwidth usage for decentralized applications. However, while it addresses efficiency challenges, it does not address encryption robustness, especially concerning quantum-era threats.[4] Our system retains the scalability benefits offered by GossipSub optimizations while embedding quantum-resistant security measures. By integrating Kyber-based encryption for key exchanges and AES-256-GCM for message protection, along with decentralized content storage via IPFS and secure identity management through DIDs, the application ensures both high performance and strong, future-proof communication security.

The paper proposes a decentralized messaging solution that employs WebRTC for direct peer-to-peer communication and uses a blockchain with Proof of Authority (PoA) for authentication. This setup ensures communication persists even when users go offline by involving temporary relay nodes. However, its reliance on classical encryption makes it potentially vulnerable to future quantum computing capabilities.[5] In our design, WebRTC remains a core component for real-time peer-to-peer connections but is strengthened through quantum-safe encryption methods. Kyber is used for secure key exchanges and AES-256-GCM for data confidentiality. Additionally, IPFS decentralized storage and OTP-verified DIDs provide stronger security layers, ensuring communication remains confidential and tamper-proof even in a quantum-empowered environment.

This system presents a secure messaging application leveraging blockchain for decentralized storage and smart contracts for authentication and key management. It offers end-to-end encryption and ensures data integrity. Nonetheless, its reliance on traditional asymmetric cryptography leaves it susceptible to decryption by quantum computers in the near future.[6] Our Quantum-Secure Messaging Application upgrades this approach by utilizing Kyber, a post-quantum encryption algorithm selected by NIST, for key exchanges. IPFS enables decentralized storage, and user identities are managed securely with DIDs, making the platform highly resistant to both classical and quantum cyberattacks.

The paper outlines a decentralized messaging web application using blockchain to ensure data integrity, availability, and resistance to censorship. It successfully addresses challenges associated with centralization, such as single points of failure. However, its dependence on classical encryption methods could make it vulnerable to attacks from future quantum computers.[7] Our proposed system extends the decentralization benefits while future-proofing security by incorporating Kyber for quantum-safe key exchanges and AES-256-GCM for data encryption. Combined with IPFS storage and libp2p networking for peer discovery, and secure DIDs for identity management, the system ensures robust, tamper-resistant communication even against emerging quantum threats.

This paper discusses a secure peer-to-peer communication model built on a private blockchain network using Go-Ethereum (GETH) and Proof of Authority (PoA) consensus. While the system effectively tackles issues of censorship, data security, and centralization, its encryption and identity verification methods still rely on classical cryptographic techniques, posing risks in the quantum computing era.[8] Our application strengthens this approach by embedding Kyber-based quantum-resistant encryption for secure key exchanges and employing IPFS for decentralized storage. It also introduces OTP-secured Decentralized Identifiers (DIDs) for tamper-proof identity management.

CHAPTER 3

SYSTEM ANALYSIS

Today’s digital communication is increasingly threatened by cyberattacks, data breaches, and emerging technologies like quantum computing. Most messaging platforms rely heavily on centralized servers or traditional cryptographic techniques, leaving them vulnerable to surveillance, censorship, and future quantum decryption attacks. Since these systems are not designed to withstand the capabilities of quantum computers, there is a pressing need for a messaging platform that ensures privacy, protects user identities, and maintains communication resilience — all without depending on a central authority.

3.1 Existing Systems

- **Decentralized Blockchain-Based Messaging:** Researches [1][2] introduced peer-to-peer messaging using blockchain technology, Ethereum, and the Whisper protocol to eliminate centralized control and strengthen security.
- **Libp2p-Based Messaging Networks:** Project [3] took a step further by integrating libp2p for peer-to-peer networking, incorporating group encryption and relay systems to improve scalability and reliability.
- **GossipSub Optimization:** The paper [4] focused on optimizing decentralized communication by proposing message fragmentation and staggering techniques to improve GossipSub protocol performance for large messages.
- **WebRTC and Blockchain Hybrid Systems: S**olutions such as [5][8] combined WebRTC technology and private blockchain networks (like GETH) to enable direct, secure messaging between users.
- **Secure Messaging with Blockchain and Smart Contracts:** Systems discussed in paper[6] used blockchain-based smart contracts and AES encryption to offer secure messaging features but remained reliant on traditional cryptography methods.

3.2 Limitations of Existing Systems

- **Lack of Quantum Resistance:** Most current systems use classical cryptographic algorithms like RSA and ECC, which are vulnerable to quantum attacks through Shor’s and Grover’s algorithms.
- **Partial Decentralization:** Many platforms still have elements of centralization, such as reliance on smart contract hosting services or bootstrap nodes, limiting their resilience and autonomy.
- **Weak Identity Management:** Identity handling often relies on semi-centralized databases and lacks full decentralized identifiers (DIDs), which makes users susceptible to identity theft and tracking.
- **Absence of Secure Real-Time Media Communication:** Only a few systems address encryption for voice and video calls, and none currently offer quantum-secure protection for real-time media streams.

- **Scalability and Efficiency Issues:** Solutions heavily dependent on blockchain often face bottlenecks such as high transaction fees, limited throughput, and network congestion, restricting their scalability.

3.3 Proposed System:

To overcome the gaps identified in the existing systems, the proposed project introduces a **Quantum-Secure Messaging Application** that provides secure, decentralized, and quantum-resistant communication. Key features include:

- **Post-Quantum Cryptography**: The system implements **Kyber**, a NIST-recommended algorithm, to ensure that key exchanges are safe against quantum decryption attempts, securing the future of private communications.
- **AES-256-GCM Encryption**: All message content is encrypted with AES-256-GCM, offering strong, efficient, and authenticated encryption to preserve message integrity and confidentiality.
- **Decentralized Storage & Networking**: **IPFS** is used for decentralized message distribution, and **libp2p** with **Kademlia DHT** is employed for peer discovery, eliminating the need for central servers and improving fault tolerance.
- **Decentralized Identity Management (DIDs):** Each user is assigned a Decentralized Identifier linked to their mobile number, authenticated through secure OTP verification. Metadata and cryptographic keys are protected in encrypted MongoDB/MySQL databases.
- **Quantum-Secure Real-Time Communication:** Optional support for **WebRTC** enables encrypted voice and video calls, securing both data and metadata with quantum-resistant protections.
- **Scalable Communication Relay: Waku v2** is optionally integrated to handle message relay efficiently over large decentralized networks, improving scalability without compromising privacy.
- **Modern Architecture:** The system uses **React.js** and **Redux** for the frontend and **Node.js** with **Express.js** for backend services, ensuring a highly responsive, scalable, and secure communication experience.

3.4 Advantages of the Proposed System:

- **Built for the Future with Quantum-Safe Security**: The app uses **Kyber**, a next-generation encryption algorithm that’s resistant to attacks from quantum computers. This means even as technology advances, your messages will stay private and protected.

- **Strong End-to-End Encryption**: All messages are protected with **AES-256-GCM**, one of the strongest encryption methods available today. Only the people involved in the conversation can read the messages—no one else.

- **No Central Server, No Single Point of Failure**: Instead of storing messages on one central server, the app uses **peer-to-peer networking** through **IPFS** and **libp2p**. This makes the system more secure, more private, and less likely to go offline.

- **Private and Secure Identity System**: Every user gets a **Decentralized Identifier (DID)**, verified through a secure OTP process. This keeps your identity safe and under your control, without needing a central database.

- **Fast, Real-Time Messaging and Calls**: With support for **WebSockets** and **WebRTC**, the app enables smooth, instant messaging and even voice/video calls—all while keeping everything secure and encrypted.

- **Designed to Scale Up Easily**: As more users join the network, the system remains fast and efficient thanks to **Waku v2**, which helps relay messages across large networks without slowing things down.

- **Maximum Privacy by Design**: Your messages are never stored on a central server. Even metadata like your public key or DID is stored in encrypted databases, keeping everything private and tamper-proof.

- **Works Everywhere, Feels Great to Use**: The app is built using modern tools like **React.js** and **Redux**, making it responsive, fast, and easy to use across desktops and mobile devices.

- **Safe Data Storage**: Any sensitive information—like keys or IDs—is stored securely using encryption in databases like **MongoDB** or **MySQL**, so even behind-the-scenes data is protected.

- **Ready for What's Next**: The app is modular and easy to upgrade, whether that's adding new encryption methods, language options, or features like biometric login in the future.

CHAPTER 4

SYSTEM REQUIREMENTS

This chapter outlines the system requirements necessary to develop and deploy the **Quantum-Secure Messaging Application**. These specifications ensure optimal performance, high security, decentralization, and real-time communication capabilities even under quantum-era threats.

4.1 Hardware Specification

To ensure smooth operation of the decentralized, encrypted messaging platform, the following hardware specifications are recommended:

- **Processor**: Intel Core i3 / AMD Ryzen 3 or higher
- **RAM**: Minimum 6GB or more (for handling real-time encryption, WebRTC media streams, and decentralized networking)
- **Storage**: SSD preferred, 256GB minimum (for caching IPFS data and running local nodes)
- **Network**: Stable broadband internet connection for peer-to-peer communications
- **Display**: 13-inch or larger screen, 1366x768 resolution minimum (recommended Full HD for better UI experience)
- **Keyboard & Mouse**: Standard USB or wireless peripherals

4.2 Software Specification

The development and deployment of the application require the following software environment:

- **Frontend Development**: React.js, Redux.js (for user interface and state management)
- **Backend Development**: Node.js with Express.js (for server-side API orchestration and encryption services)
- **Cryptography**: Kyber (for quantum-safe key exchange), AES-256-GCM (for message encryption)
- **Networking**: libp2p and Kademlia DHT (for decentralized peer-to-peer networking)
- **Storage**: IPFS (InterPlanetary File System) for decentralized message storage
- **Identity Management**: DID (Decentralized Identifiers) linked with OTP-based verification
- **Voice/Video Communication**: WebRTC (for secure peer-to-peer real-time communication)
- **Scalable Relay Option**: Waku v2 (for efficient decentralized message relaying)
- **Database**: MongoDB / MySQL (for secure metadata storage, encrypted at rest)

- **Code Editor**: Visual Studio Code (preferred)
- **Browser**: Latest versions of Google Chrome or Firefox (for testing WebRTC and libp2p integrations)
- **Version Control**: Git with GitHub/GitLab for collaborative development

4.3 Functional Requirements

1. **Quantum-Secure Messaging**

    - Enable real-time, end-to-end encrypted text messaging with Kyber-secured key exchanges and AES-256-GCM encryption.
    - Ensure complete forward secrecy and tamper-resistance across all communications.

2. **Peer-to-Peer Networking**

    - Implement decentralized messaging via libp2p protocols without centralized servers.
    - Achieve peer discovery using Kademlia DHT for efficient routing.

3. **Decentralized Storage**

    - Utilize IPFS to store message content in a distributed manner.
    - Ensure message retrieval and publishing are resistant to censorship and node failures.

4. **User Identity Management**

    - Assign users unique DIDs linked to their mobile numbers through OTP authentication.
    - Securely store and manage public keys and metadata in encrypted databases.

5. **Secure Voice and Video Calling**

    - Integrate WebRTC to provide real-time, quantum-safe encrypted voice and video communication.

6. **Scalability and Message Relay**

    - Support optional Waku v2 integration to relay encrypted messages efficiently over large networks.

7. **Cross-Platform Compatibility**

    - Ensure the application works across major operating systems (Windows, Linux, macOS) and is mobile-optimized for Android/iOS.

4.4 Non-Functional Requirements

1. **Performance**
   - The application should support sending/receiving encrypted messages with minimal latency.
   - Handle up to 100+ simultaneous peer connections with smooth encryption and decryption operations.
2. **Reliability**
   - Guarantee consistent delivery of messages even with fluctuating network conditions.
   - Resilient against single-point failures due to decentralized storage and networking.
3. **Usability**
   - Provide an intuitive and responsive user interface using React.js.
   - Offer seamless onboarding with simple DID creation and OTP-based authentication.
4. **Scalability**
   - Allow for easy expansion to accommodate more users, additional decentralized storage nodes, and enhanced WebRTC group calling features.
   - Modular codebase to easily integrate new cryptographic standards or scaling solutions.
5. **Security**
   - All encryption and decryption should happen locally on the user’s device to maintain absolute privacy.
   - Ensure protection against browser vulnerabilities (e.g., XSS, CSRF) during WebRTC signaling and UI interactions.
6. **Maintainability**
   - Codebase must be modular, well-documented, and ready for future updates such as multi-language support, new DID standards, or next-generation post-quantum algorithms.
   - Ensure easy debugging, monitoring, and logging with minimal overhead on decentralized communications.

CHAPTER 5

METHODOLOGY

1. **User Registration and Decentralized Identity Setup**

   - Implement mobile OTP verification for user registration.
   - Generate and assign a **Decentralized Identifier (DID)** upon successful registration.
   - Use **Twilio/Nodemail** for OTP verification and **Verifiable Credentials** for DID management.
   - Ensure DID generation follows W3C DID standards, utilizing public/private key pairs for authentication.
   - Securely store the DID and user data.

2. **Secure Login and Authentication**

   - Implement login using **DID** for secure authentication.
   - Use **Elliptic Curve Cryptography (ECC)** to verify and sign the DID.
   - Encrypt user data using **AES-256** and securely store it in the database.

3. **Quantum-Secure Key Exchange for Communication**

   - Implement **Kyber** algorithm for quantum-secure key exchange.
   - Use **AES-256-GCM** for encrypting messages after key exchange.
   - Ensure communication is protected against quantum threats.

4. **Real-Time Messaging with End-to-End Encryption**

   - Use **WebSockets** for real-time messaging.
   - Encrypt messages using **AES-256-GCM** before transmission.
   - Ensure message integrity with **HMAC-SHA256**.
   - Decrypt messages upon receipt using the shared key from the Kyber exchange.

5. **Peer-to-Peer (P2P) Network Setup for Serverless Communication**

   - Integrate **libp2p** for direct, secure peer-to-peer communication.
   - Use **IPFS** for decentralized file storage, ensuring secure file sharing across peers.

6. **Voice and Video Calls**

   - Implement voice and video calls using **WebRTC**.

   - Secure calls with **Kyber** for key exchange and **AES-256** for encrypting audio and video.

- Ensure real-time communication remains private and protected.

7. **Scalable Messaging System with Decentralized Relay**

- Integrate **Waku v2** for decentralized message relay.
- Ensure messages are encrypted before being relayed across the distributed network.

8. **Cross-Platform Compatibility and UI/UX**

- Build a responsive **UI/UX** using **React.js** (web) and **React Native** (mobile).
- Use **CSS Grid/Flexbox** for responsive layouts.
- Implement **service workers** to enable offline functionality.

CHAPTER 6

EXPECTED OUTCOME

Once the development of *QuatChat – A Quantum-Secure Messaging Application* is complete, the system will offer a secure and fully decentralized platform for real-time communication. The key expected outcomes are:

- **Secure User Onboarding with Decentralized Identity:**
  Users will be able to register by verifying their mobile number using a simple OTP process. After verification, the system will assign them a unique Decentralized Identifier (DID), which is securely managed without relying on any centralized identity server.

- **Private and Protected Login System:**
  When a user logs in, their DID and associated data will be securely retrieved from an encrypted database. This allows them to access their account without compromising privacy or control over their identity.

- **Safe Key Exchange Protected Against Quantum Attacks:**
  Before any conversation starts, a quantum-secure key exchange will take place using the Kyber algorithm. This ensures that all future communication between users remains protected, even from powerful quantum computers.

- **Real-Time Messaging with End-to-End Encryption:**
  The chat interface will allow users to send and receive messages in real time. Each message will be encrypted on the sender’s side and only decrypted on the receiver’s side, using AES-256-GCM encryption. This keeps messages fully confidential and tamper-proof.

- **Serverless Peer-to-Peer Architecture:**
  Instead of sending messages through a central server, all communication will be routed directly between users using peer-to-peer technology (libp2p). Files and attachments will be stored using IPFS, a decentralized storage system that ensures reliability and resistance to censorship.

- **Fast and Reliable Message Delivery:**
  Messages will be delivered quickly and efficiently through WebSockets, providing a seamless chat experience without delays.

- **Secure Voice and Video Calls (Optional):**
  Users will also have the option to make voice and video calls using WebRTC. These calls will be protected using the same post-quantum encryption methods to keep conversations private and secure.

- **Scalable Messaging for Larger Networks:**
  To support a growing number of users, the system will optionally integrate Waku v2, a decentralized message relay protocol that helps scale message delivery while maintaining user privacy.

- **Compatible Across Devices and Browsers:**
  QuatChat will run smoothly in modern web browsers like Chrome and Firefox and be accessible on various platforms, including Windows, Linux, and Android devices.

- **Designed for the Future:**
  The platform will be built in a modular way, making it easy to add new features such as group messaging, biometric login, support for additional languages, or integration of advanced cryptographic standards as they become available.

CONCLUSION

The development of **QuatChat** represents a significant leap toward safeguarding digital communication in the age of advancing quantum technologies. By integrating **post-quantum cryptography**, **decentralized identity management**, and **peer-to-peer networking**, this system provides robust protection against both present-day and future cybersecurity threats. Through the use of **Kyber** for quantum-resistant key exchanges and **AES-256-GCM** for secure message encryption, alongside technologies like **IPFS**, **libp2p**, and **WebRTC**, QuatChat eliminates centralized vulnerabilities and empowers users with full control over their data and identities. Optional features like quantum-secure voice and video calls further expand the platform's capabilities. Ultimately, this project delivers a resilient, future-proof messaging solution that addresses the urgent need for privacy, authenticity, and scalability in the era of quantum computing and Web3.

References

[1] Bagade, S. D., & Wankhade, N. R. (2022). Decentralized Secure Messaging Application Using Blockchain Technology. International Research Journal of Modernization in Engineering Technology and Science, 4(9).

[2] Khalkar, K., Dhake, N., Kelzarkar, S., & Shinde, T. (2023). Decentralized Chat Application using Blockchain Technology. International Journal for Research in Applied Science & Engineering Technology (IJRASET), 11(1).

[3] Yeung, M. (2024). SendingNetwork: Advancing the Future of Decentralized Messaging Networks. Sending Labs Whitepaper

[4] Farooq, M. U., Cizain, T., & Kaiser, D. (2025). Staggering and Fragmentation for Improved Large Message Handling in libp2p GossipSub. arXiv preprint arXiv:2504.10365.

[5] Sanjay, B. N., Sivaram, N. A., Shankaranarayanan, N. S., Jaiprakash, V. A., & Govilkar, S. (2022). A Decentralized Application for Secure Private and Group Messaging in a Peer-to-Peer Environment. International Research Journal of Engineering and Technology (IRJET), 9(2).

[6] Raghuwanshi, A., Gour, G., Choudhary, H. K., Rai, A., Ahirwar, A., & Ahirwar, D. S. (2025). Secure Messaging WebApp using Blockchain Technology. International Research Journal of Modernization in Engineering Technology and Science, 7(3).

[7] Vikram, V. J., Krishna, M. A., Reddy, J. K., Kumar, G. S., Balaji, A., & Mariappan, R. (2024). Decentralized Messaging Web Application: A Blockchain-Based Approach for Secure Communication. International Journal of Engineering Research in Computer Science and Engineering (IJERCSE), 11(6).

[8] Syed, M. R., Nair, S., Shinde, M., Patade, A., & Unny, S. (2023). Secure Peer-to-Peer Communication using Private Network Blockchain Technology. 2023 International Conference on Advanced Computing Technologies and Applications (ICACTA).