

**POST QUANTUM DECENTRALISED STORAGE SYSTEM**  
**A REPORT**

**JCS1731: PROJECT WORK PHASE – 1**

**IV YEAR / VII SEM**

**REGULATION 2021**

*Submitted by*

**DINESH KUMAR G** [130721104025]

**HARIHARAN A** [130721104031]

**KIRAN ROHITH T** [130721104044]

*in partial fulfilment for the award of the degree*

*of*

**BACHELOR OF ENGINEERING**

*in*

**DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING**



**JERUSALEM COLLEGE OF ENGINEERING**

**(An Autonomous Institution, Affiliated to Anna University, Chennai)**

**NBA & NAAC ACCREDITED INSTITUTION**

**Velachery Main Road, Narayanapuram, Pallikaranai, Chennai – 600100**

**NOVEMBER 2024**

# **JERUSALEM COLLEGE OF ENGINEERING**

**(An Autonomous Institution, Affiliated to Anna University)**

**ANNA UNIVERSITY: CHENNAI 600 025**

## **BONAFIDE CERTIFICATE**

Certified that this project report “**POST QUANTUM DECENTRALISED BLOCKCHAIN STORAGE SYSTEM - SENMON**” is the bonafide work of **DINESH KUMAR G (130721104025), HARIHARAN A (130721104031), KIRAN ROHITH T (130721104044)** who carried out the project work under my supervision.

### **SIGNATURE**

**Mrs. VANITHA SHEBA M, M.E**

**SUPERVISOR,**

**ASSISTANT PROFESSOR**

Department of Computer

Science and Engineering,

Jerusalem College of Engineering,

Pallikaranai, Chennai – 600 100.

### **SIGNATURE**

**Dr.MAYA EAPEN, M.E., Ph.D**

**PROFESSOR & HEAD**

Department of Computer

Science and Engineering,

Jerusalem College of Engineering,

Pallikaranai, Chennai – 600 100.

**Submitted for the End Semester examination held on \_\_\_\_\_.**

**INTERNAL EXAMINER**

**EXTERNAL EXAMINER**

## ABSTRACT

In an era where quantum computing threatens to compromise traditional encryption, secure data storage is essential. Senmon is a decentralized storage solution leveraging blockchain technology and post-quantum cryptography to create a quantum-safe environment for storing documents and files. It addresses growing concerns around data privacy, integrity, and resilience with advanced quantum-resistant algorithms—Crystal Kyber for key exchange, Crystal Dilithium for digital signatures, and AES-256 encryption—to protect against both current and future computational threats.

Senmon’s architecture decentralizes data storage across multiple nodes, eliminating single points of failure and enhancing data availability and redundancy. This blockchain-based structure inherently supports immutability and security, distributing encrypted data across the network to mitigate the risks of tampering, loss, or unauthorized access. Smart contract protocols further manage data access control and permissions, ensuring an immutable audit trail for every transaction and storage operation.

Senmon offers a seamless user experience that supports secure file encryption, storage, retrieval, and management. Its design layers quantum-safe cryptographic algorithms for robust data protection against classical and quantum attacks, creating a resilient and traceable storage solution.

This quantum-safe, blockchain-based storage system is scalable and suitable for industries requiring high levels of data security, such as finance, healthcare, and government. Senmon provides a forward-looking solution to current cybersecurity vulnerabilities, establishing a robust foundation that can adapt to the evolving landscape of quantum computing threats.

## ACKNOWLEDGEMENT

We extend our warmest gratitude to **Prof. Dr. M. Mala**, Chairperson, Jerusalem College of Engineering for her enduring support.

We express our sincere thanks to **Dr. Ramesh S, Ph.D.**, Principal, Jerusalem College of Engineering for his kindness, which enabled us to do this project.

We express our sincere thanks to **Dr. Maya Eapen, M.E., Ph.D.**, Professor and Head, Department of Computer Science and Engineering, Jerusalem College of Engineering for her support throughout the project.

We would like to take this opportunity to express our sincere thanks to our supervisor, **Ms. M. Vanitha Sheba, M.E.**, Assistant Professor, Department of Computer Science and Engineering, Jerusalem College of Engineering for her valuable guidance, inspirations and technical support throughout the project.

We express our gratitude to our project coordinators, **Dr.T.Dhanalakshmi, M.E., Ph.D.**, Associate Professor, **Ms.H.Mercy, M.E.**, Associate Professor, **Ms.M.Vanitha Sheba, M.E.**, Assistant Professor and **Ms.S.Devipriya, M.E.**, Assistant Professor, Department of Computer Science and Engineering, Jerusalem College of Engineering for their valuable guidance and support.

We thank all the faculty and supporting staff of the Department of Computer Science and Engineering, Jerusalem College of Engineering, for their co-operation and assistance in the successful completion of the project.

**DINESH KUMAR G (130721104025)**

**HARIHARAN A (130721104031)**

**KIRAN ROHITH T (130721104044)**

## TABLE OF CONTENTS

CHAPTER NO	TITLE	PAGE NO.
1.	<b>ABSTRACT</b>	ii
	<b>LIST OF FIGURES</b>	v
	<b>INTRODUCTION</b>	1
	1.1 GENERAL	2
	1.2 BLOCK CHAIN	2
	1.3 POST QUANTUM ALGORITHMS	2
	1.4 RUST	3
	1.5 JAVASCRIPT AND REACT	3
	1.6 SUMMARY	4
2.	<b>LITERATURE SURVEY</b>	
	2.1 GENERAL	5
	2.2 VARIOUS LITERATURE SURVEYS	5
	2.3 SUMMARY	8
3.	<b>SYSTEM ANALYSIS</b>	
	3.1 GENERAL	9
	3.2 EXISTING SYSTEM	9
	3.3 PROPOSED SYSTEM	10
	3.4 ARCHITECTURE DIAGRAM	11
	3.5 SUMMARY	12
4.	<b>SYSTEM DESIGN</b>	
	4.1 GENERAL	13
	4.2 MODULE DESCRIPTION	13
	4.2.1 USER INTERFACE MODULE	13
	4.2.2 VIRTUAL FILE SYSTEM MODULE	14
	4.2.3 ENCRYPTION/DECRYPTION MODULE	15
	4.2.4 BLOCKCHAIN DATABASE MODULE	16
	4.3 CONCLUSION	17
5.	<b>IMPLEMENTATION &amp; RESULTS</b>	
	5.1 GENERAL	19
	5.2 IMPLEMENTATION AND RESULTS	20
6.	<b>CONCLUSION AND WORK SCHEDULE FOR PHASE II</b>	
	6.1 PHASE I CONCLUSION	24
	6.2 WORK ON PHASE II	25
	6.3 CONCLUSION	27
	<b>REFERENCES</b>	28

## **LIST OF FIGURES**

<b>FIG. NO.</b>	<b>FIGURE TITLE</b>	<b>PAGE NO.</b>
3.1	ARCHITECTURE DIAGRAM	11
5.2.1	HTTPS SERVER CODE	20
5.2.2	HTTPS SERVER STARTUP	21
5.2.3	SELECTION OF FILES FOR UPLOADING	21
5.2.4	FILE UPLOADING AND PASSWORD INPUT	22
5.2.5	ENTRIES IN FILE SYSTEM	22
5.2.6	ENTRIES IN FILE SYSTEM AFTER UPLOADING	23

# CHAPTER 1

## INTRODUCTION

### 1.1 GENERAL:

Senmon, is a PostQuantum Decentralized BlockChain Storage System. It uses a custom blockchain that is connected to a Peer2Peer Network, storing all the essential details important for storing and securing data against Hackers with ill-intentions. It also features an cutting-edge Quantum Safe Algorithms like Crystal Kyber and Crystal Dilithium to encrypt and digitally sign data from the user, ensuring safety and data integrity against the coming age of Quantum Computers, being a threat to Cyber Security and Computer Networks. Senmon finds its place among other systems being ready for the future and ensuring that users find security and privacy in the ever-growing digital age.

For Senmon, we focus on security of the users at all costs. Every Module that we use in the system has to be sure that it is free from bugs and exploits that hackers could potentially use in the future to gain access into the system. To ensure this, all the modules are written in Rust and Javascript keeping OWASP Guidelines in mind. It is also kept in mind while designing the system that no central authority is capable of updating/removing/changing data in the system without the approval/consent of the data owner. The Project is entirely open source and is hosted on GitHub for the benefits of the general audience such that a simple user can easily host their own Storage System entirely of their hardware and infrastructure. The data that we collect is the bare minimum required to allow the smooth operation of the System. With the Capability of the security provided by a BlockChain and a Quantum Secure encryption cum signature algorithm; ensures that data privacy and security is available in the foreseeable future of humankind. With TLS/SSL being the absolute weaklink in the system, we look forward to witness a Session Layer Security that is secure against Quantum Computers and Cyber Attacks.

## **1.2 BLOCK CHAIN:**

Senmon is a future-proof system capable of resisting attacks against Quantum Computers combining the use of Blockchain Technology; for access control and transaction control, and Quantum Secure algorithms to encrypt/sign files. While the hashing part of a block chain is perfectly secure against Quantum Computers, it is unclear how safe the next moment will be, considering Governments and Other Central Authority could shut down/request data of a particular user without their consent. A Ledger-like system that is capable of access-control with it's strength increasing as it's users increase is extremely useful.

## **1.3 POST-QUANTUM ALGORITHMS:**

Currently classical encryption based on integer factorization, elliptical curves and discrete logarithm is extremely susceptible of breaking to Quantum Computers using Shor's algorithm. A Sufficiently powerful Quantum Computer can break classical encryption (asymmetric) in a matter of minutes. This poses a huge threat in the future, where Quantum Computers and Research towards breaking classical encryption using said computers, is coming sooner or later. Even to this day, usage of Quantum Secure algorithms against Hackers who could potentially gain classically encrypted data, could just break it in the near future where Quantum Computers are widely available to the general public. This Attack is known as Harvest-Now-Decrypt-Later and is of the main concern of Senmon and it's existence against similar systems that provide similar functionality. Usage of Quantum Secure algorithms ensure that critical documents that is of the level of National Security does not end up in the hands of malicious person, today or tomorrow.

## **1.4 RUST:**

Rust is a modern programming language focused on performance, safety, and concurrency. It was created by Mozilla Research and first released in 2010. Rust is designed to prevent memory-related errors, such as null pointer dereferencing and



buffer overflows, through its ownership system, which enforces strict rules about how memory is managed without needing a garbage collector.

Key features of Rust include Memory Safety, Rust's ownership model ensures that each piece of data has a single owner, preventing data races and memory leaks. Zero-Cost Abstractions, Rust provides high-level features without sacrificing performance, allowing developers to write efficient code. Concurrency, Rust makes it easier to write concurrent programs that are safe and efficient, helping developers take advantage of multi-core processors. Strong Type System, Rust has a robust type system that helps catch errors at compile time, reducing runtime bugs.

We use Rust in Senmon to ensure that all the data processed during runtime is not accidentally exposed to hackers having physical access to System's nodes from a server.

Rust guarantees runtime safety and compile time safety and gives the promise that if it compiles, it will never break from memory safety and concurrency that other languages are plagued with.

## **1.5 JAVASCRIPT AND REACT:**

JavaScript is a versatile, high-level programming language that is primarily known for its role in web development. Developed in 1995 by Brendan Eich, it enables interactive and dynamic content on websites. As a key technology of the web alongside HTML and CSS, JavaScript allows developers to create responsive user interfaces, handle events, and manipulate the Document Object Model (DOM) to update content dynamically. Over the years, JavaScript has evolved significantly, supporting various programming paradigms, including object-oriented, functional, and imperative programming. With the advent of frameworks and libraries like Node.js, JavaScript has also expanded to server-side development, allowing for full-stack applications.

React is a popular open-source JavaScript library developed by Facebook for building user interfaces, particularly single-page applications (SPAs). Released in 2013, React

enables developers to create reusable UI components that efficiently update and render in response to data changes.

Together, JavaScript and React empower developers to create interactive, secure and beautiful applications in the web browser without any difficulty whatsoever. Due to this overwhelming feature, Senmon chose this frontend stack for it's application.

## **1.6 SUMMARY:**

Senmon is a Post Quantum Decentralized Storage System that is used to store critically important files without ever the need of worrying about the future role of Quantum Computers and it's implications on the current network stack. Using Rust to build the Blockchain and the Encryption module, we allow security to be baked in the application without further refactoring.

## **CHAPTER 2**

### **LITERATURE SURVEY**

#### **2.1 GENERAL:**

A Literature Survey is a comprehensive summary of previous research on a topic. The Literature review surveys scholarly articles, books and other sources relevant to a particular area of research. The Review should enumerate, describe, summarize, objectively evaluate and clarify this previous research.

#### **RELATED WORKS:**

**AUTHOR:** A. Aikata, A. C. Mert, M. Imran, S. Pagliarini and S. S. Roy

**TITLE:** KaLi: A Crystal for Post-Quantum Security Using Kyber and Dilithium

**DATE:** 2024

#### **DESCRIPTION:**

The Quantum Safe Algorithms chosen by NIST QS organization are specified here. Quantum Safe Algorithms such as CRYSTAL – Kyber and CRYSTAL – Dilithium are specifically detailed here. Kyber which belongs to the family of Quantum Safe Algorithms of Key Encapsulation is used to encapsulate Symmetric Keys with Lattice Cryptography instead of normal key generation found in classical encryption. Dilithium, which belongs to the family of Quantum Safe Algorithms of Digital Signature is used sign and verify the public keys that are sent over the network to avoid replay and modification attacks by Hackers. These Algorithms provide an opportunity to use Symmetric Encryption like AES and RSA, without ever the need to worry about it being broken by cryptanalysis attacks from Quantum Computers. Usage of these algorithms also ensure that Harvest-Now-Decrypt-Later attacks do not work for the time being.

**AUTHOR:** X. Chen, K. Zhang, X. Liang, W. Qiu, Z. Zhang and D. Tu

**TITLE:** HyperBSA: A High-Performance Consortium Blockchain Storage Architecture for Massive Data

**DATE:** 2020

**DESCRIPTION:**

With the Considerable exploration of blockchain in various industrial fields, the storage architecture of mainstream consortium blockchains exhibit significant performance limitations which cannot meet the requirements of efficient data access with massive data storage in enterprise-level business scenarios. This paper provides methods to store two types of data, State Data, which is used to represent the blockchain and is essential for it's working and Continuous Data, which is the data that is stored inside the blockchain by the users. Continuous Data can represent the transactions that are performed by the users on their individual files and data uploaded into the system. HyperBSA provides a special cache and a specialized index-based storage to store the continuous data from the user into the blockchain.

**AUTHOR:** M. A. Shafique, A. Munir and I. Latif

**TITLE:** Quantum Computing: Circuits, Algorithms, and Applications

**DATE:** 2024

**DESCRIPTION:**

Quantum Computing is an emergent field of cutting-edge computer science and physics, harnessing the power of Quantum Mechanics to solve problems beyond the ability that is possible from the most powerful classical computers right now. By Taking advantage of quantum physics, Quantum Computers would be able to process massively complicated problems such TSP, Knapsack problem that would take classical computers centuries and millenia, in the span of hours. Quantum Computers use qubits which can take n number of states at the same time without ever being constricted to one particular state. This superposition of states provide the machines massive strength to compute fast

**AUTHOR:** H. Zang, H. Kim and J. Kim

**TITLE:** Blockchain-Based Decentralized Storage Design for Data Confidence Over Cloud-Native Edge Infrastructure

**DATE:** 2024

**DESCRIPTION:**

Blockchain based decentralized storage is gaining popularity among security researchers over cloud-based infrastructure due to additional security provided by blockchain and its decentralized nature. While centralized storage is cost-effective, it faces issues of scalability, performance bottlenecks and security vulnerabilities. With Decentralized storage, data are distributed across nodes, offering redundancy, data availability and enhanced security but it also introduces its own challenges such as complex data retrieval, potential inconsistencies in data versions and difficulties in ensuring data privacy and integrity.

**AUTHOR:** A. Hafid, A. S. Hafid and M. Samih

**TITLE:** Scaling Blockchains: A Comprehensive Survey

**DATE:** 2020

**DESCRIPTION:**

Blockchain has been widely deployed in recent years. However, scalability is emerging as a challenging issue. This Paper outlines the existing solutions to blockchain scalability, which can be classified into two categories: First layer and Second Layer solutions. First Layer Solutions propose modifications to the blockchain, that is changing the blockchain structure such as block size and second layer solutions propose mechanisms that are implemented outside the blockchain. The Paper proposes sharding as a solution to the first layer solution and dividing committee formation that processes a separate set of transactions which can improve performance of the blockchain

**AUTHOR:** T. M. Fernández-Caramès and P. Fraga-Lamas

**TITLE:** Towards Post-Quantum Blockchain: A Review on Blockchain  
Cryptography Resistant to Quantum Computing Attacks

**DATE:** 2020

**DESCRIPTION:**

Blockchain have evolved significantly in the last years and their use has been found in numerous applications due to their ability to provide transparency, redundancy and accountability. In the case of blockchain, such characteristics are provided through hash functions. However, the fast progress of quantum computing has opened the possibility of performing attacks based on Grover's and Shor's algorithms. Such algorithms threaten both public-key cryptography and hash functions, forcing to redesign blockchains to make use of systems that can withstand quantum-attacks. This Article provides useful guidelines on post-quantum blockchain security to future blockchain researchers and developers.

## **CHAPTER 3**

### **SYSTEM ANALYSIS**

#### **3.1 GENERAL:**

In an age where digital information has become a core asset, the need for secure, reliable, and scalable storage solutions is paramount. As quantum computing capabilities advance, traditional encryption methods become increasingly vulnerable. Senmon addresses these security concerns by introducing a decentralized, post-quantum blockchain-based storage solution for securely storing and managing digital documents and files. Utilizing robust quantum-safe algorithms—Crystal Kyber for key encapsulation, Crystal Dilithium for digital signatures, and AES-256 for symmetric encryption—Senmon provides a future-proof data storage system that ensures data confidentiality, integrity, and accessibility. By leveraging blockchain, Senmon decentralizes control and prevents single points of failure, adding a layer of security against both internal and external threats.

#### **3.2 EXISTING SYSTEMS:**

The field of digital storage has seen considerable innovation, with several established approaches to managing and securing data. However, current storage solutions often struggle to provide a balance between security, accessibility, and adaptability to emerging technologies like quantum computing. Existing systems can be broadly classified into two main categories: centralized cloud storage solutions and decentralized storage systems. Each category has its strengths and limitations, especially in terms of security, reliability, and resilience against future threats.

Currently, various storage solutions exist, but they typically fall into two primary categories:

- **Centralized Cloud Storage Solutions:** Platforms like Google Drive, Dropbox, and AWS offer scalable storage but rely on centralized servers, making them vulnerable to single points of failure and potential security breaches. Additionally, these platforms use traditional encryption methods that may be rendered obsolete by quantum computing.
- **Decentralized Storage Solutions:** Systems such as IPFS (InterPlanetary File System) and Storj utilize blockchain and decentralized networks to improve security. However, they largely rely on classical encryption, which may not offer adequate protection against quantum threats. Furthermore, most lack a dedicated focus on user-friendly interfaces and seamless integration of post-quantum cryptography.

These existing solutions fail to combine all three critical aspects: decentralization, quantum-safe encryption, and an intuitive, user-friendly interface.

### 3.3 PROPOSED SYSTEM:

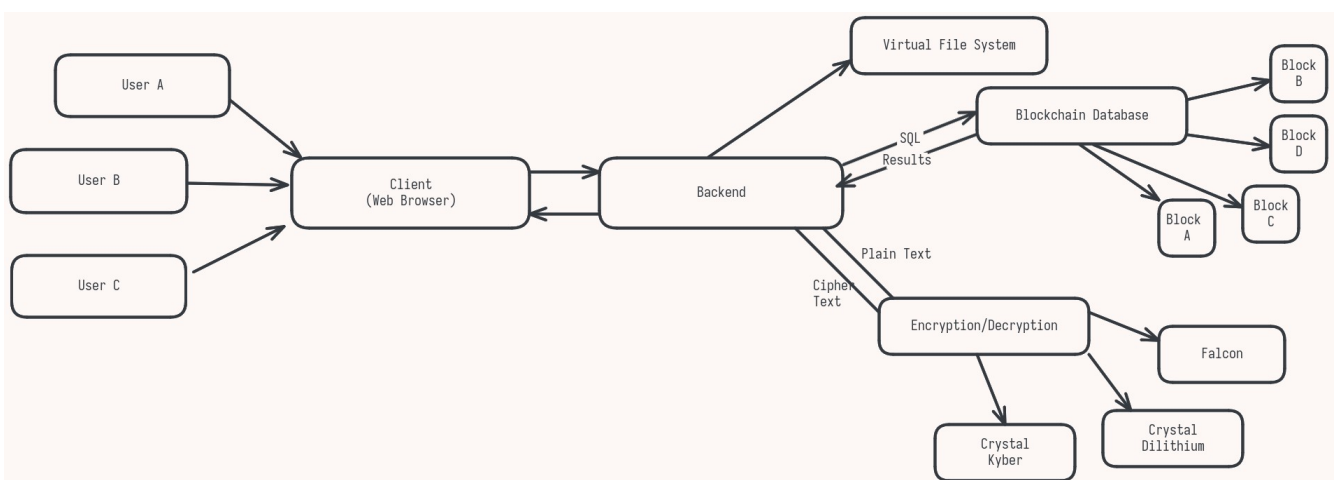
Senmon proposes a novel solution by addressing the limitations of existing storage systems through the following core features:

- **Post-Quantum Security:** Implementing NIST-approved quantum-resistant algorithms, specifically Crystal Kyber and Crystal Dilithium, alongside AES-256 ensures that data stored on Senmon remains secure against both classical and quantum-based threats.
- **Decentralized Architecture:** Using blockchain technology, Senmon achieves a decentralized structure that distributes data across multiple nodes, enhancing redundancy, availability, and security. This architecture minimizes reliance on central servers and mitigates the risk of data breaches due to single points of failure.



- **Modular Design:** Senmon’s architecture includes four key modules:
  1. **User Interface:** A streamlined and accessible front end that simplifies secure data storage, retrieval, and management.
  2. **Virtual File System:** Provides file organization, metadata management, and abstracts the underlying blockchain mechanics for an intuitive file structure.
  3. **Encryption Module:** Responsible for applying Crystal Kyber, Crystal Dilithium, and AES-256 encryption to maintain data security.
  4. **Blockchain Database Module:** Manages data replication, consensus protocols, and logs all storage transactions in an immutable ledger for auditing and transparency.
- **Smart Contract Integration:** Using smart contracts, Senmon controls data access, permissions, and sharing, which adds a layer of automation and ensures a seamless and secure user experience.

### 3.4 ARCHITECTURE DIAGRAM:



**Fig 3.1: Architecture Diagram**

### **3.5 SUMMARY:**

Senmon represents a forward-looking storage solution designed for a future where quantum computing poses a significant threat to classical encryption standards. By combining blockchain technology, decentralized storage, and quantum-safe encryption methods, Senmon mitigates vulnerabilities associated with centralized storage and traditional encryption. Its modular architecture—comprising a user-friendly interface, a virtual file system, an encryption module, and a blockchain database module—ensures a seamless, secure, and resilient environment for storing critical documents and files.

In conclusion, Senmon offers a robust, scalable, and future-proof storage system capable of addressing both current and emerging security challenges. By focusing on post-quantum encryption and decentralization, Senmon is well-positioned to support industries that demand high levels of data security, availability, and privacy.

## **CHAPTER 4**

### **SYSTEM DESIGN**

#### **4.1 GENERAL**

The system design of Senmon is built around four major modules, each serving a unique function in delivering a secure, decentralized, and user-friendly storage experience. Senmon's architecture follows a modular approach, where each component interacts seamlessly with others to achieve end-to-end security and accessibility. The system is designed with future-proofing in mind, integrating post-quantum encryption methods to resist quantum attacks and employing blockchain to decentralize data storage, eliminating reliance on a single point of control.

Senmon's design aims to provide users with a familiar file management experience while incorporating advanced security features behind the scenes. This balance between security and usability is achieved through a layered architecture that separates concerns, enhancing both scalability and maintainability. The following sections outline the structure and function of each module within the Senmon system.

#### **4.2 MODULE DESCRIPTION**

Each of Senmon's modules plays a critical role in supporting the platform's core functionalities: secure storage, file management, encryption, and decentralization. Below is a detailed description of each module and its respective components:

##### **4.2.1. User Interface Module**

The User Interface (UI) module is the front-facing component of Senmon, designed to facilitate user interactions with the platform's storage, retrieval, and file management functions. This module focuses on ensuring ease of access, simplifying the underlying complexity of a decentralized system for end-users.

- **Key Features:**
  - **File Management:** Users can upload, download, view, and organize files. The UI abstracts the underlying file system structure, giving users a familiar environment similar to traditional file storage platforms.
  - **Access Control:** Provides users with options to set permissions on files or folders, enabling secure sharing and restricted access.
  - **Security Integration:** The UI module works with the Encryption Module to manage user credentials, ensuring authentication and access control are secure.
  - **User-Friendly Design:** Prioritizes simplicity and intuitiveness to encourage adoption by non-technical users, hiding the complexities of encryption and blockchain from the end-user.
- **Interactions with Other Modules:**
  - The UI module communicates with the **Virtual File System** for displaying file structure and metadata.
  - It interacts with the **Encryption Module** to encrypt/decrypt files before upload/download.
  - The **Blockchain Database Module** is accessed to display file history and audit logs, ensuring transparency and traceability.

#### 4.2.2. Virtual File System Module

The Virtual File System (VFS) module manages the organization, storage, and retrieval of files within the decentralized network, acting as a bridge between the UI and the blockchain storage infrastructure.

##### Key Features:

- **File Structure Management:** The VFS presents files in a logical structure, organizing them into directories and providing metadata support (e.g., file type, size, creation date).

- **Indexing and Metadata:** Facilitates quick access by maintaining indexes and metadata on file location within the decentralized network. This indexing system is crucial for enabling efficient search and retrieval across distributed nodes.
- **File Abstraction Layer:** Abstracts the complexities of interacting with the decentralized storage, allowing the system to appear as a conventional file system to users while actually distributing data across nodes in the blockchain network.
- **Concurrency Management:** Ensures file consistency and prevents conflicts during simultaneous access or updates by implementing concurrency control methods.

#### **Interactions with Other Modules:**

- The VFS module interacts closely with the **Blockchain Database Module** to manage file storage and retrieval across nodes.
- It works with the **Encryption Module** to handle encrypted files and metadata, ensuring that all files in storage remain secure.
- The **UI Module** relies on the VFS to present an intuitive file structure for users.

#### **4.2.3. Encryption Module**

The Encryption Module is at the heart of Senmon's security, utilizing advanced post-quantum cryptographic algorithms to protect user data. This module is responsible for encrypting files before they are stored on the blockchain and decrypting them upon retrieval, ensuring data confidentiality and integrity at all times.

#### **Key Features:**

- **Quantum-Safe Encryption:** Implements Crystal Kyber for key encapsulation and Crystal Dilithium for digital signatures, providing robust protection against both classical and quantum attacks. AES-256 is used for symmetric encryption, offering an additional layer of security for the data at rest.

- **Key Management:** Generates, stores, and securely transmits encryption keys. Keys are handled using Crystal Kyber, ensuring they are quantum-resistant.
  - **Digital Signatures:** Uses Crystal Dilithium for signing data, which supports file integrity verification and ensures data authenticity.
  - **Data Integrity:** Ensures that encrypted data stored on the blockchain is protected from unauthorized modification, and verifies the integrity of data upon retrieval.
- **Interactions with Other Modules:**
  - Collaborates with the **VFS Module** to apply encryption to files before they are distributed across the blockchain network.
  - Works with the **UI Module** to manage user credentials and permissions, ensuring secure access and decryption.
  - The **Blockchain Database Module** relies on the encryption module to maintain data confidentiality and authenticate file modifications.

#### 4.2.4. Blockchain Database Module

The Blockchain Database Module provides the backbone of Senmon's decentralized architecture. It manages data replication, consensus protocols, and transaction records, ensuring that the storage network is both secure and distributed across multiple nodes.

##### Key Features:

- **Decentralized Storage:** Distributes encrypted file fragments across multiple nodes, preventing any single point of failure. Each node stores a portion of the encrypted data, with redundancy to ensure data availability.
- **Consensus Mechanism:** Implements a consensus protocol to validate data transactions, ensuring that all nodes in the network agree on the current state of stored data.

- **Immutability and Auditability:** Maintains a transparent and immutable ledger of file operations, providing an audit trail that allows users to track file access, modifications, and sharing history.
  - **Smart Contract Support:** Integrates smart contracts to manage permissions and automate access control, allowing users to set conditions on file sharing and access without relying on a centralized authority.
  - **Redundancy and Recovery:** Provides redundancy by storing copies of encrypted data fragments across nodes. This distributed approach ensures that data is retrievable even if some nodes become unavailable.
- **Interactions with Other Modules:**
  - Works with the **Encryption Module** to ensure that only encrypted data is stored across nodes, protecting data confidentiality even at the node level.
  - The **VFS Module** communicates with the blockchain database to manage file location, indexing, and metadata across the decentralized network.
  - The **UI Module** accesses this module to retrieve the history and audit logs, providing transparency for users.

## 4.3 CONCLUSION

Senmon's system design leverages a modular architecture to deliver a secure, user-friendly, and quantum-resistant storage solution. Each module plays a specialized role in facilitating data security, accessibility, and redundancy:

- The User Interface Module simplifies the complexity of decentralized storage, providing users with a familiar and secure experience.
- The Virtual File System Module handles file organization, metadata, and retrieval across the decentralized network.
- The Encryption Module ensures data security using advanced quantum-resistant algorithms.

- The Blockchain Database Module provides the decentralized infrastructure, enabling data redundancy, immutability, and auditability.

Through this architecture, Senmon achieves a balance of security, usability, and scalability, positioning itself as a viable storage solution for organizations and individuals requiring quantum-safe decentralized storage.



## CHAPTER 5

### IMPLEMENTATION AND RESULTS

#### 5.1 GENERAL:

The implementation of Senmon has made significant strides with the successful completion of two critical modules: the Encryption Module and the User Interface Module. These components are foundational to the system's overall functionality, ensuring both robust data security and an intuitive user experience.

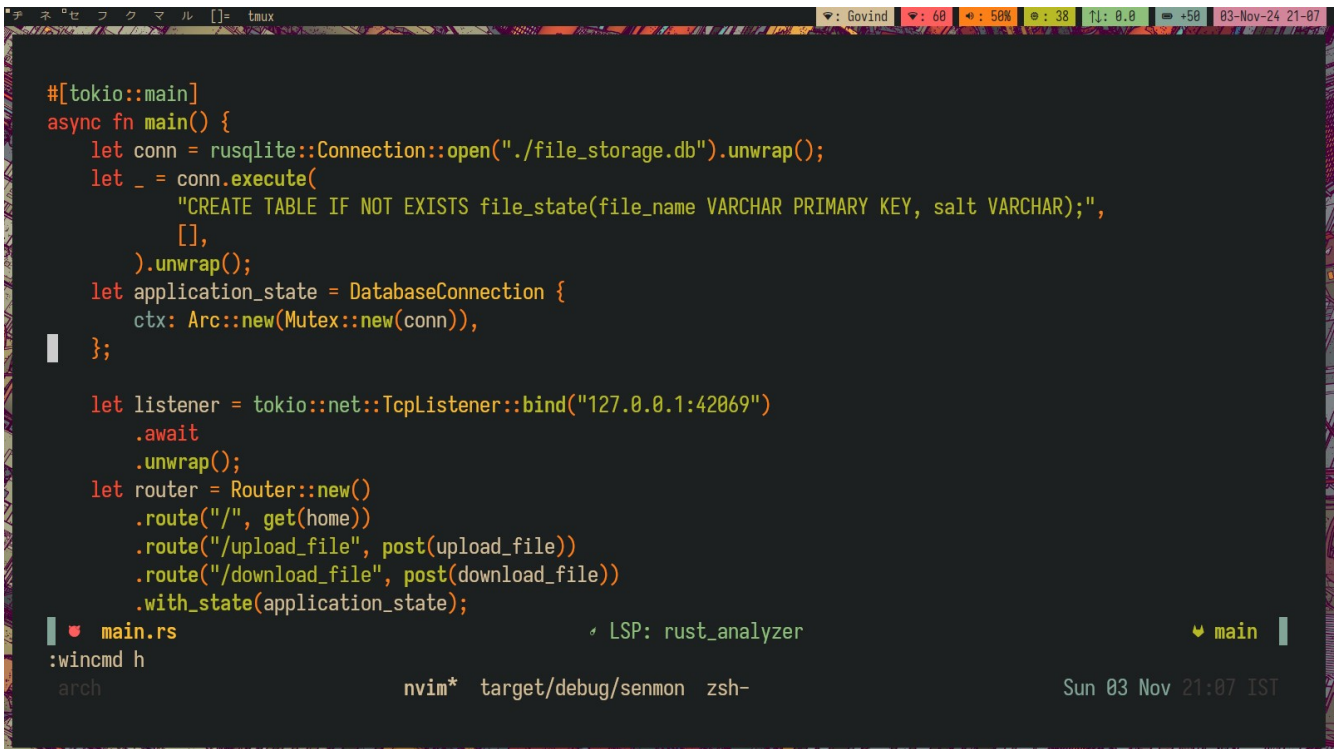
The Encryption Module has been meticulously designed to leverage advanced post-quantum cryptographic algorithms. By incorporating Crystal Kyber for key encapsulation, Crystal Dilithium for digital signatures, and AES-256 for symmetric encryption, this module ensures that all user data remains secure against current and future threats, particularly those posed by advancements in quantum computing. This robust encryption framework is designed to protect sensitive information at all stages—whether it's being uploaded, stored, or downloaded. The module efficiently handles key management, ensuring that encryption keys are generated, stored, and transmitted securely, while also providing mechanisms for data integrity verification through digital signatures. The implementation not only guarantees confidentiality but also enhances user trust by maintaining the integrity and authenticity of their data.

The User Interface Module plays a pivotal role in the overall user experience of Senmon. Designed with user-friendliness in mind, this module employs modern web technologies to create a responsive and intuitive interface that simplifies interactions with the storage system. Users can easily navigate file management tasks such as uploading, downloading, organizing, and sharing files without needing in-depth technical knowledge. The UI abstracts the complexities inherent in decentralized storage systems, presenting users with a familiar file management experience akin to traditional cloud storage solutions. It also integrates seamlessly with the Encryption Module to ensure that encryption and decryption processes are handled transparently, allowing users to focus on their files without being burdened by security concerns.

Together, the Encryption Module and User Interface Module form a cohesive foundation for Senmon, addressing the critical needs of data security and usability. The completion of these modules not only establishes the groundwork for future developments within the system but also sets the stage for further integration with additional components such as the Virtual File System and Blockchain Database Module.

The successful implementation of these modules has been validated through rigorous testing, which demonstrated their effectiveness in protecting user data while providing an accessible platform for file management. As the project progresses, the focus will shift toward integrating these completed modules with other components to create a fully operational decentralized storage solution that meets the evolving demands of users in a secure and efficient manner.

## 5.2 IMPLEMENTATION:



```
#[tokio::main]
async fn main() {
    let conn = rusqlite::Connection::open("./file_storage.db").unwrap();
    let _ = conn.execute(
        "CREATE TABLE IF NOT EXISTS file_state(file_name VARCHAR PRIMARY KEY, salt VARCHAR);",
        [],
    ).unwrap();
    let application_state = DatabaseConnection {
        ctx: Arc::new(Mutex::new(conn)),
    };

    let listener = tokio::net::TcpListener::bind("127.0.0.1:42069")
        .await
        .unwrap();
    let router = Router::new()
        .route("/", get(home))
        .route("/upload_file", post(upload_file))
        .route("/download_file", post(download_file))
        .with_state(application_state);

    main.rs
    :wincmd h
    arch
```

Fig: 5.2.1 HTTPS SERVER CODE

```
tmux
Govind 71 50% 49 248.0 +50 03-Nov-24 21-26

dinu : .software/senmon | cargo run
warning: unused import: `ring::rand::SecureRandom`
  -> src/handlers.rs:5:5
5 | use ring::rand::SecureRandom;
  |     ^^^^^^^^^^^^^^^^^^^^^^^^^
= note: `#[warn(unused_imports)]` on by default

warning: `senmon` (bin "senmon") generated 1 warning (run `cargo fix --bin "senmon"` to apply 1 suggestion)
Finished `dev` profile [unoptimized + debuginfo] target(s) in 0.08s
Running `target/debug/senmon`

arch nvim- zsh* zsh Sun 03 Nov 21:26 IST
```

FIG. 5.2.2: HTTPS SERVER STARTUP

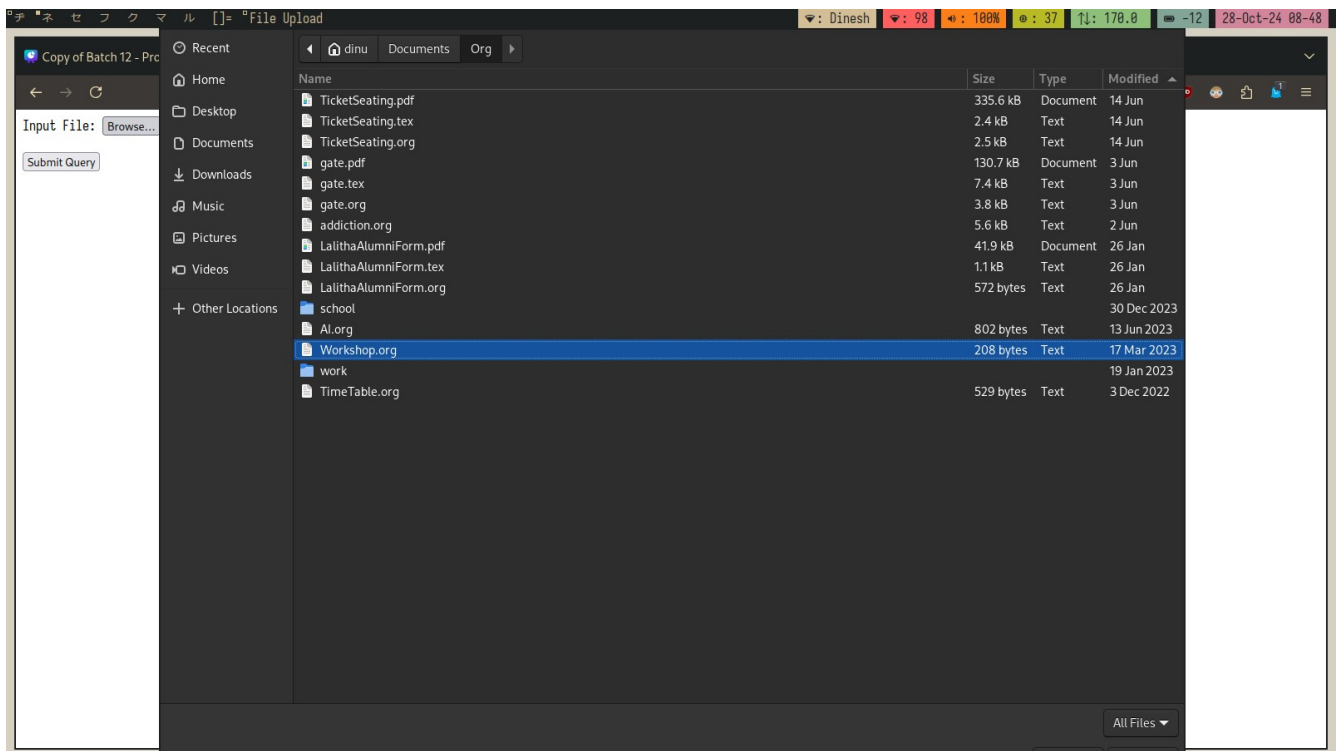


FIG. 5.2.3 SELECTION OF FILES FOR UPLOADING

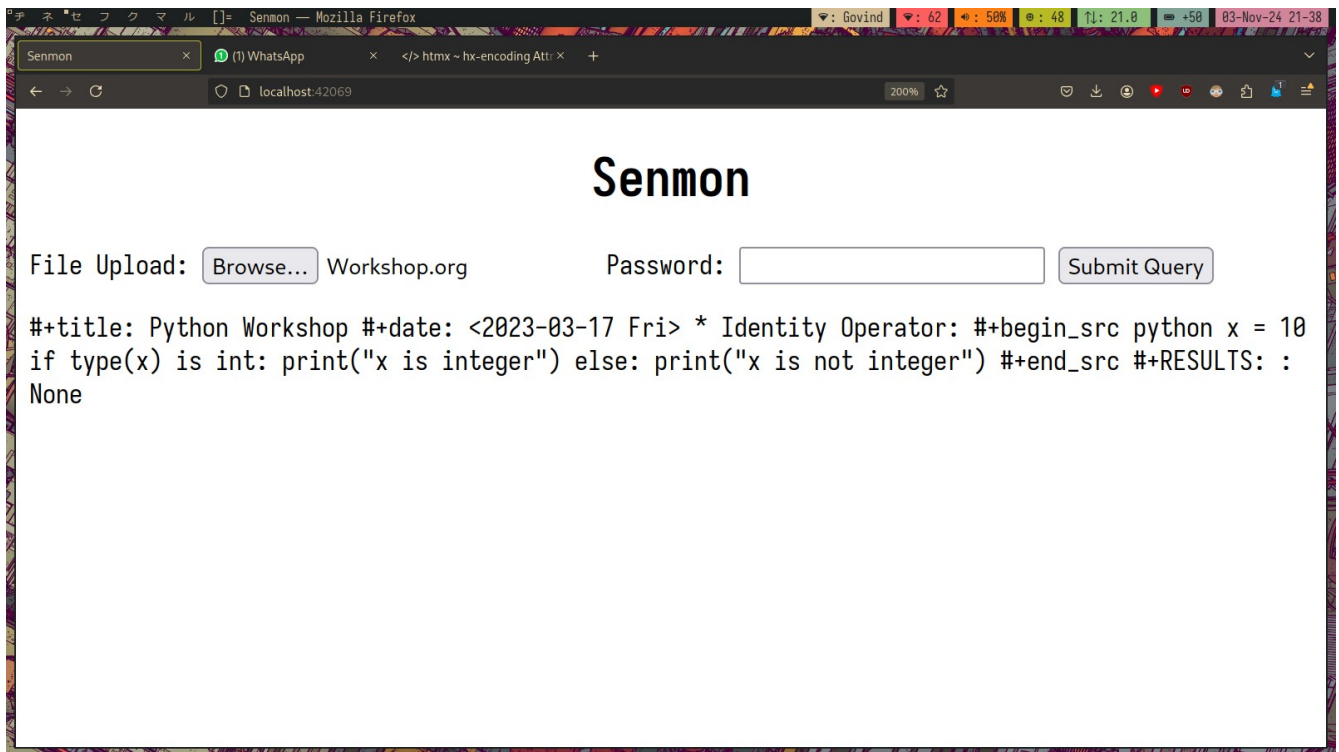


FIG. 5.2.4: FILE UPLOADING AND PASSWORD INPUT



FIG. 5.2.5: ENTRIES IN FILE SYSTEM



The image shows a terminal window with a dark background and light-colored text. The terminal is running SQLite. The commands entered are: `sqlite> SELECT * from file_state;` and `sqlite> SELECT * from file_state;`. The output of the second command is a long alphanumeric string: `Workshop.org|phhegfHHJtnhTqHE2EsmU23TW0fnE58W`. The terminal window has a title bar at the top with various system icons and a status bar at the bottom with the text `arch`, `nvim target/debug/senmon- sqlite3*`, and `Sun 03 Nov 21:39 IST`.

```
sqlite> SELECT * from file_state;
sqlite> SELECT * from file_state;
Workshop.org|phhegfHHJtnhTqHE2EsmU23TW0fnE58W
sqlite>
```

arch nvim target/debug/senmon- sqlite3\* Sun 03 Nov 21:39 IST

**FIG. 5.2.6: ENTRIES IN FILE SYSTEM AFTER UPLOAD**

## CHAPTER 6

### CONCLUSION AND WORK SCHEDULE FOR PHASE II

#### 6.1 GENERAL

As we embark on Phase 2 of the Senmon project, we are entering a pivotal stage in the development of a robust, decentralized storage solution. With the successful completion of the Encryption Module and User Interface Module, the next steps focus on implementing two integral components: the Virtual File System (VFS) Module and the Blockchain Database Module. Together, these components will significantly enhance the functionality, security, and usability of the Senmon system.

The Virtual File System (VFS) is designed to abstract the complexities associated with decentralized storage, allowing users to interact with their data in a familiar and efficient manner. The VFS will enable structured file organization, efficient indexing, and quick retrieval of documents and files. One of the key features of this module will be its ability to manage metadata effectively, allowing users to categorize and search for files based on various attributes. Additionally, the VFS will support operations such as file versioning and sharing, enabling collaborative work environments where multiple users can access and modify files without conflicts. This functionality is essential for enhancing user productivity and ensuring a smooth experience in managing their stored data.

On the other hand, the Blockchain Database Module will provide the foundational infrastructure required for decentralized storage. By utilizing blockchain technology, this module will ensure that all data is distributed across a network of nodes, eliminating the risk of single points of failure. The implementation will involve establishing a secure and transparent ledger to track all transactions related to file storage and access. Key features of the Blockchain Module will include:

- **Consensus Mechanisms:** Implementing algorithms such as Proof of Stake (PoS) or Practical Byzantine Fault Tolerance (PBFT) to validate transactions and ensure data integrity.

- **Smart Contracts:** Developing automated contracts that govern access permissions and interactions with stored data, enhancing security and streamlining operations.
- **Immutability:** Ensuring that all transactions recorded on the blockchain are immutable, thus providing an auditable history of file operations that can be reviewed by users for accountability.

Following the development of these modules, the final stage will be the integration of all—User Interface, Encryption, Virtual File System, and Blockchain Database—into a cohesive system. This integration will require careful planning and execution to ensure that all components interact smoothly, preserving the integrity and security of the user's data throughout the process.

## **6.2 FUTURE WORK/MODULES**

The successful completion of Phase 2 will involve several key tasks aimed at enhancing the system's overall capabilities and user experience:

1. **Implementation of the Virtual File System Module:**
  - **Design and Development:** Develop a comprehensive architecture for the VFS, focusing on efficient data structures for file indexing and retrieval.
  - **User Interaction:** Implement a user-friendly interface that allows users to navigate their files intuitively, facilitating actions such as upload, download, organization, and sharing.
  - **Performance Optimization:** Optimize the VFS for performance, ensuring fast access times even with a large number of files and users.
2. **Development of the Blockchain Database Module:**
  - **Infrastructure Setup:** Establish the blockchain network, determining the necessary nodes and configurations to support decentralized operations.

- **Consensus Mechanism Integration:** Select and implement the consensus algorithm that best fits the project's needs, ensuring secure and efficient transaction validation.
- **Smart Contract Development:** Create and deploy smart contracts that will govern file permissions, access controls, and transaction automation, streamlining user interactions with stored data.

### 3. Integration of All Modules:

- **Cohesive System Development:** Ensure that the User Interface, Encryption Module, VFS, and Blockchain Module work seamlessly together, maintaining data security and enhancing user experience.
- **Inter-module Communication:** Establish efficient communication protocols between modules to ensure smooth data flow and operational consistency.
- **Testing and Validation:** Conduct rigorous testing to validate the functionality and security of the integrated system, addressing any issues that arise before the final launch.

### 4. User Testing and Feedback:

- **Beta Testing:** Involve a select group of users in beta testing the integrated system, gathering feedback to identify potential areas for improvement.
- **User Training and Support:** Develop training materials and support resources to assist users in adapting to the new system, enhancing their overall experience.

### 5. Documentation and Reporting:

- **Comprehensive Documentation:** Prepare thorough documentation covering system architecture, module functionality, and user guides to facilitate understanding and usability.
- **Reporting Progress:** Regularly update stakeholders on the progress of Phase 2, providing insights into challenges encountered and solutions implemented.



## 6.3 CONCLUSION

In conclusion, Phase 2 of the Senmon project represents a critical advancement toward the realization of a fully functional and secure decentralized storage solution. The implementation of the Virtual File System and Blockchain Database modules will not only enhance the system's capabilities but also significantly improve user experience through increased functionality and streamlined interactions.

The VFS will empower users to manage their files effectively, fostering an environment conducive to collaboration and productivity. Meanwhile, the Blockchain Module will fortify the security and transparency of the storage solution, ensuring that user data is protected against unauthorized access and manipulation. Together, these components will create a robust ecosystem that prioritizes security, usability, and accountability.

As we move forward, the focus will remain on delivering a high-quality product that addresses the contemporary challenges of data storage and security in an increasingly digital world. The successful integration of all modules will position Senmon as a leader in the realm of decentralized storage solutions, making it a valuable tool for users seeking to safeguard their sensitive information. By continually refining the system and adapting to the evolving technological landscape, Senmon aims to not only meet but exceed user expectations, providing a pioneering solution in the realm of post-quantum secure storage.

## REFERENCES

- [1] "Towards Post-Quantum Blockchain: A Review on Blockchain Cryptography Resistant to Quantum Computing Attacks," T. M. Fernández-Caramès and P. Fraga-wLamas, in *IEEE Access*, vol. 8, pp. 21091-21116, 2020
- [2] "KaLi: A Crystal for Post-Quantum Security Using Kyber and Dilithium", A. Aikata, A. C. Mert, M. Imran, S. Pagliarini and S. S. Roy, in *IEEE Transactions on Circuits and Systems I: Regular Papers*, vol. 70, no. 2, pp. 747-758
- [3] "Quantum Computing: Circuits, Algorithms, and Applications," M. A. Shafique, A. Munir and I. Latif, in *IEEE Access*, vol. 12, pp. 22296-22314, 2024
- [4] "Blockchain-Based Decentralized Storage Design for Data Confidence Over Cloud-Native Edge Infrastructure", H. Zang, H. Kim and J. Kim, in *IEEE Access*, vol. 12, pp. 50083-50099, 2024
- [5] "Scaling Blockchains: A Comprehensive Survey", A. Hafid, A. S. Hafid and M. Samih, in *IEEE Access*, vol. 8, pp. 125244-125262, 2020
- [6] HyperBSA: A High-Performance Consortium Blockchain Storage Architecture for Massive Data", X. Chen, K. Zhang, X. Liang, W. Qiu, Z. Zhang and D. Tu, in *IEEE Access*, vol. 8, pp. 178402-178413, 2020
- [7] "A Proxy Re-Encryption Approach to Secure Data Sharing in the Internet of Things Based on Blockchain", K. O. -B. O. Agyekum, Q. Xia, E. B. Sifah, C. N. A. Cobblah, H. Xia and J. Gao, in *IEEE Systems Journal*, vol. 16, no. 1, pp. 1685-1696, March 2022.
- [8] C. Aristidou and E. Marcou, "Blockchain Standards and Government Applications," in *Journal of ICT Standardization*, vol. 7, no. 3, pp. 287-312, 2019
- [9] "Linear Elliptical Curve Digital Signature (LECDS) With Blockchain Approach for Enhanced Security on Cloud Server," B. Sowmiya, E. Poovammal, K. Ramana, S. Singh and B. Yoon, in *IEEE Access*, vol. 9, pp. 138245-138253, 2021.
- [10] M. Wazid, A. K. Das and Y. Park, "Generic Quantum Blockchain-Envisioned Security Framework for IoT Environment: Architecture, Security Benefits and Future Research," in *IEEE Open Journal of the Computer Society*, vol. 5, pp. 248-267, 2024.