

DEVELOPMENT OF A SECURE HEALTHCARE MANAGEMENT SYSTEM UTILIZING BLOCKCHAIN TECHNOLOGY FOR ENCRYPTED PATIENT DATA TRANSMISSION

A PROJECT REPORT

Submitted by

**DHARSHINI B
ELAKKIYA S
GAJALAKSHMI S**

in partial fulfillment for the award of the degree

of

BACHELOR OF ENGINEERING

in

COMPUTER SCIENCE AND ENGINEERING



PANIMALAR ENGINEERING COLLEGE

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

APRIL 2025

DEVELOPMENT OF A SECURE HEALTHCARE MANAGEMENT SYSTEM UTILIZING BLOCKCHAIN TECHNOLOGY FOR ENCRYPTED PATIENT DATA TRANSMISSION

A PROJECT REPORT

Submitted by

DHARSHINI B [211421104058]

ELAKKIYA S[211421104069]

GAJALAKSHMI S[211421104070]

in partial fulfillment for the award of the degree

of

BACHELOR OF ENGINEERING

in

COMPUTER SCIENCE AND ENGINEERING



PANIMALAR ENGINEERING COLLEGE

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

APRIL 2025

PANIMALAR ENGINEERING COLLEGE

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

BONAFIDE CERTIFICATE

Certified that this project report “**DEVELOPMENT OF A SECURE HEALTHCARE MANAGEMENT SYSTEM UTILIZING BLOCKCHAIN TECHNOLOGY FOR ENCRYPTED PATIENT DATA TRANSMISSION**” is the bonafide work of —**DHARSHINI B (211421104058) ELAKKIYA S (211421104069) GAJALAKSHMI S(211421104070).**”who carried out the project work under my supervision.

SIGNATURE

SIGNATURE

Dr.L.JABASHEELA M.E., Ph.D.,

Dr.KAVITHA SUBRAMANI M.E,PhD.,

HEAD OF THE DEPARTMENT

SUPERVISOR, PROFESSOR

**Department of CSE,
Panimalar engineering college,
Nasarathpettai,
Poonamallee,
Chennai – 123**

**Department of CSE,
Panimalar Engineering College,
Nasarathpettai,
Poonamallee,
Chennai – 123**

Certified that the above candidate(s) was/ were examined in the Anna University
Project Viva-Voce Examination held on.....

DECLARATION BY THE STUDENT

We **DHARSHINI B(211421104058)** , **ELAKKIYA S(211421104069)** ,
GAJALAKSHMI S(211421104070) hereby declare that this project report titled
**—DEVELOPMENT OF A SECURE HEALTHCARE MANAGEMENT
SYSTEM UTILIZING BLOCKCHAIN TECHNOLOGY FOR ENCRYPTED
PATIENT DATA TRANSMISSION** under the guidance of **Dr.KAVITHA
SUBRAMANI (M.E., PhD.)** is the original work done by us and we have not
plagiarized or submitted to any other degree in any university by us.

DHARSHINI B
ELAKKIYA S
GAJALAKSHMI S

ACKNOWLEDGEMENT

Our profound gratitude is directed towards our esteemed Secretary and Correspondent, **Dr. P. CHINNADURAI, M.A., Ph.D.**, for his fervent encouragement. His inspirational support proved instrumental in galvanizing our efforts, ultimately contributing significantly to the successful completion of this project.

We want to express our deep gratitude to our Directors, **Tmt. C. VIJAYARAJESWARI, Dr. C. SAKTHI KUMAR, M.E., Ph.D., and Dr. SARANYASREE SAKTHI KUMAR, B.E., M.B.A., Ph.D.**, for graciously affording us the essential resources and facilities for undertaking of this project.

Our gratitude is also extended to our Principal, **Dr. K. MANI, M.E., Ph.D.**, whose facilitation proved pivotal in the successful completion of this project.

We express our heartfelt thanks to **Dr. L. JABASHEELA, M.E., Ph.D.**, Head of the Department of Computer Science and Engineering, for granting the necessary facilities that contributed to the timely and successful completion of project.

We would like to express our sincere thanks to **Project Coordinator and Guide Dr.KAVITHA SUBRAMANI M.E. PhD.**, and all the faculty members of the Department of CSE for their unwavering support for the successful completion of the project.

DHARSHINI B

ELAKKIYA S

GAJALAKSHMI S

ABSTRACT

Effective management of healthcare data is essential for ensuring the safety of sensitive patient information in today's digital landscape. An innovative system utilizes the cutting-edge FrodoKEM encryption algorithm along with real-time secure chat features to protect information shared among various hospital departments. FrodoKEM uses adaptive encryption methods and dynamic key management to provide strong data security during transmission. Simultaneously, blockchain-based logging and SHA-256 hashing are employed to preserve data integrity and create an unchangeable audit trail for all accesses and alterations. The secure chat component allows for private, real-time communication between physicians and patients, enhancing telehealth capabilities and overall clinical productivity. Experimental tests indicate that the system delivers low latency and high security performance in fluctuating healthcare environments. By integrating state-of-the-art encryption techniques with secure communication solutions, this approach not only strengthens data confidentiality and integrity but also improves clinical workflows, presenting a promising avenue for future digital healthcare management.

TABLE OF CONTENTS

CHAPTER NO.	TITLE	PAGE NO.
	ABSTRACT	IV
	LIST OF FIGURES	VII
	LIST OF ABBREVIATIONS	IX
1.	INTRODUCTION	
	1.1 PROBLEM STATEMENT	2
	1.2 EXISTING SYSTEM	3
2	LITERURE SURVEY	4
3.	THEORETICAL BACKGROUND	12
	3.1 IMPLEMENTATION ENVIRONMENT	13
	3.2 SYSTEM ARCHITECTURE	15
	3.3 PROPOSED METHODOLOGY	16
	3.3.1 INPUT DESIGN	18
	3.3.2 UML DIAGRAM	19
	3.3.3 USE CASE DIAGRAM	19
	3.3.4 ACTIVITY DIAGRAM	20
	3.3.5 SEQUENCE DIAGRAM	21
	3.3.6 ER DIAGRAM	22

4.	SYSTEM IMPLEMENTATION	25
	4.1 MODULES	26
	4.1.1 USER MODULE	26
	4.1.2 ADMIN MODULE	26
	4.1.3 DOCTOR MODULE	27
	4.1.4 SURGEON MODULE	27
	4.1.5 RADIOLOGIST MODULE	28
	4.1.6 PHARMACIST MODULE	28
	4.1.7 CHAT MODULE	29
	4.2 ALGORITHMS	30
	4.2.1 SHA-256 ALGORITHM	30
	4.2.2 ForodoKEM ALGORITHM	30
	4.2.3 BLOCKCHAIN	31
5.	RESULTS & DISCUSSION	33
	5.1 SYSTEM TESTING	34
	5.2 RESULT & DISCUSSION	36
6.	CONCLUSION AND FUTURE WORK	38
	6.1 CONCLUSION	39
	6.2 FUTURE WORK	39
	APPENDICES	40
	A.1 SDG Goals	40
	A.2 Source Code	40
	A.3 Screen Shots	58

A.4	Plagiarism Report	63
A.5	Paper Publication	75
	REFERENCES	78

LIST OF FIGURES

FIGURE NO	FIGURE NAME	PAGE NO
3.1	SYSTEM ARCHITECTURE	15
3.2	INPUT DESIGN	18
3.3	USE CASE DIAGRAM	19
3.4	DFD DIAGRAM	20
3.5	ACTIVITY DIAGRAM	21
3.6	SEQUENCE DIAGRAM	22
3.7	ER DIAGRAM	23
A.3.1	HOME PAGE	58
A.3.2	USER REGISTRATION AND LOGIN	59
A.3.3	USER MODULE	60
A.3.4	DOCTOR REGISTRSTION AND SIGN- IN PAGE	61
A.3.5	SURGEON PAGE	62
A.3.6	PHARMACY SIGN-IN AND LOGIN PAGE	63

A.3.7	CHAT MODULE	64
A.3.8	ADMIN LOGIN	65
A.3.9	RADIOLOGIST REGISTER AND LOGIN PAGE	66
A.3.10	RADIOLOGIST MODULE	67

LIST OF ABBRIVATIONS

API	-	Application Programming Interface
CDSS	-	Clinical Decision Support System
CDW	-	Clinical Data Warehouse
DW	-	Data Warehouse
ECC	-	Elliptic Curve Cryptography
IDE	-	Integrated Development Environment
LAN	-	Local Area Network
LWE	-	Learning With Errors
SDG	-	Sustainable Development Goals
SDD	-	Solid State Drive
UTF	-	Unicode Transformation Format
TLS	-	Transport Layer Security

CHAPTER 1

INTRODUCTION

CHAPTER 1

INTRODUCTION

Patients who come to a hospital are first referred to a general doctor. A doctor updates the prescription with the patient's ID and medical notes. Such a prescription is safely shared with other departments: surgery, radiology, and pharmacy, among others. Only by authorized staff, a department-specific secret key can enable access to the prescription. For the purpose of security, a prescription is encrypted using the AES algorithm before sending it to each department. Such information, accessed only by staff in the department, can be viewed to continue the required treatment. Then, patient details are updated in the database. All patient data in the database is encrypted with the SHA-256 algorithm for extra security. Thus, sensitive medical information is shared and stored across departments securely, accessible only to authorized personnel.

1.1 PROBLEM STATEMENT

In modern healthcare systems, ensuring the security and privacy of patient data is a critical challenge. Traditional hospital management systems rely on centralized cloud storage, which is vulnerable to data breaches, unauthorized access, and privacy violations. Additionally, the lack of secure and efficient data-sharing mechanisms across hospital departments leads to inefficiencies in patient care. As cyber threats evolve, conventional encryption techniques may become insufficient, especially with the potential risks posed by quantum computing. There is a need for a robust and decentralized solution that ensures secure patient data storage, access control, and protection against emerging security threats.

1.2 EXISTING SYSTEM

The existing healthcare system primarily relies on centralized databases for storing and managing patient records. Hospitals and third-party service providers maintain these databases, making them responsible for data security and accessibility. Patient records, including medical history, prescriptions, and test reports, are stored in digital or physical formats. Access to these records is often restricted to hospital staff, and any modifications or updates are managed within the centralized system. While this system has been widely used for years, it suffers from several limitations that affect data security, integrity, and accessibility.

DISADVANTAGES

- Centralized databases are highly susceptible to cyberattacks, making patient records prone to hacking, data breaches, and unauthorized access.
- Medical records can be altered, lost, or manipulated due to system failures, cyber threats, or insider attacks, leading to inaccurate or incomplete patient information.
- Patients have minimal control over their data, while unauthorized users may find loopholes to access sensitive medical information.
- Sharing patient records between hospitals and healthcare providers is slow and insecure, causing delays in patient treatment and coordination.
- There is no proper audit mechanism to track who accesses or modifies patient data, increasing the risk of data misuse. Since all patient data is stored in a central location, any technical failure, cyberattack, or database corruption can lead to complete data loss or system downtime.

CHAPTER 2

LITERATURE REVIEW

CHAPTER 2

LITERATURE SURVEY

"Achieving Secure, Verifiable, and Efficient Boolean Keyword Searchable Encryption for Cloud Data Warehouse" – Thanaruk Theeramunkong, Somchart Fugkeaw ^[1]. This study proposes a secure and verifiable searchable encryption scheme tailored for Cloud Data Warehouses (CDWs), addressing the limitations of traditional Searchable Encryption (SE) techniques. It introduces support for Boolean expression queries over encrypted data, combining Partial Homomorphic Encryption (PHE), B+Tree, Inverted Index, and bitmapping functions for privacy-preserving and efficient search performance. By integrating blockchain and smart contracts, the scheme automates authentication, index retention, and trapdoor generation without relying on third-party verifications. Comparative evaluations reveal its superior efficiency and effectiveness over existing approaches. **Advantage:** Supports Boolean expression queries, enables scalable and privacy-preserving search operations through blockchain, and ensures trustworthiness without third-party dependency. **Disadvantage:** Complex integration and higher costs due to the employment of advanced technologies like blockchain and PHE.

"MaxD K-means: A clustering algorithm for Auto-generation of centroids and distance of data points in clusters" – Tutut Herawan, Abul Beg ^[2].

This study introduces the MaxD K-Means clustering algorithm, a novel enhancement to the traditional K-Means method. Unlike standard K-Means, where users must specify the number of clusters (k) beforehand, MaxD K-Means automatically determines the initial value of k and employs a unique strategy for setting initial centroids. Experiments conducted with synthetic data from Lloyd's

K-Means tests demonstrate significant improvements, including a reduction in the number of iterations by up to 78%. **Advantage:** Eliminates the need for user-defined cluster count, enhances initialization of centroids, and significantly improves iteration efficiency. **Disadvantage:** Limited testing on real-world data may require further validation for broader applicability.

"Privacy-Preserving Patient-Centric Clinical Decision Support System on Naive Bayesian Classification" – Rongxing Lu, Ximeng Liu, Jianfeng Ma^[3].

This study presents PPCD, a privacy-preserving clinical decision support system designed to assist clinicians in diagnosing patient disease risks while safeguarding sensitive medical data. Leveraging naive Bayesian classification, the system analyzes large amounts of clinical data without compromising individual privacy. Historical patient data stored in the cloud trains the classifier securely, using an innovative cryptographic tool—additive homomorphic proxy aggregation scheme—and a privacy-preserving protocol for retrieving top-k disease names based on patient preferences. Extensive simulations show PPCD efficiently computes disease risks with high accuracy while maintaining robust privacy measures. **Advantage:** Improves diagnosis accuracy and efficiency, ensures patient data privacy using advanced cryptographic techniques, and enables secure top-k disease retrieval. **Disadvantage:** Practical implementation may face challenges due to the complexity of cryptographic processes and reliance on cloud-based infrastructures.

"Handling Privacy-Sensitive Medical Data With Federated Learning: Challenges and Future Directions" – Ons Aouedi, Alessio Sacco, Kandaraj Piamrat, Guido Marchetto ^[4] .This study investigates the use of Federated Learning (FL) in healthcare to manage privacy-sensitive medical data securely. FL

enables multiple institutions to collaboratively train machine learning models without sharing raw patient data, ensuring privacy and security. The research outlines key challenges such as data heterogeneity, communication overhead, and security vulnerabilities like inference and poisoning attacks. It also explores potential solutions, including blockchain integration, differential privacy, and homomorphic encryption, to enhance security and efficiency within federated healthcare systems. **Advantage:** Ensures data privacy during collaborative model training, addresses healthcare-specific challenges, and offers innovative solutions for secure systems integration. **Disadvantage:** Issues like communication overhead and vulnerabilities to attacks require further refinement for widespread practical implementation.

"An Investigation Into Patient Privacy Disclosure in Online Medical Platforms" – Chun-Lin Feng, Zhi-Chao Cheng, Li-Juan Huang ^[5] .

This study investigates the factors that influence patients' willingness to disclose personal health information on online medical platforms. It emphasizes the importance of fairness perceptions—both outcome fairness and procedural fairness—and examines the role of perceived platform interactivity in shaping privacy disclosure behaviors. Based on a survey of 1,546 users, the research highlights that fairness and interactivity significantly affect patients' decisions, offering valuable insights for enhancing privacy management in online healthcare communities. **Advantage:** Provides insights into improving patient privacy management and highlights key factors—fairness and interactivity—that influence privacy disclosure. **Disadvantage:** Relies on survey data, which may not fully account for variations in behavior across diverse patient demographics or platform designs.

"Secure ID Privacy and Inference Threat Prevention Mechanisms for Distributed Systems" – Tahani Hamad Aljohani, Ning Zhang ^[6]

This study proposes the SPID framework, a security solution for distributed healthcare systems designed to prevent inference attacks and protect patient identity privacy. By leveraging a distributed set of servers owned by different providers, SPID ensures secure upload and encryption of health data to multiple foreign servers, blocking unauthorized access and inference-based tracking. Key mechanisms include elliptic curve cryptography (ECC), anonymous authentication, pseudonym-based access control, and double encryption techniques. Performance evaluations using benchmarking tools and queuing theory demonstrate the framework's effectiveness in enhancing security and mitigating threats. **Advantage:** Provides strong protection against inference attacks and identity breaches, utilizes advanced cryptographic techniques for secure data handling, and offers improved performance through distributed architecture. **Disadvantage:** Implementation may be resource-intensive and complex due to reliance on cryptography and distributed server systems.

"A Conceptual Framework to Ensure Privacy in Patient Record Management System" – K. G. Srinivasa, K. R. Venugopal, S. S. Manvi, L. M. Patnaik [7]

This study proposes a conceptual framework aimed at preserving privacy in cloud-based patient record management systems. It classifies health data into personal and medical information, applies encryption techniques such as SHA-256 and public key cryptography, and ensures controlled access based on defined privacy policies. The framework emphasizes secure storage and retrieval by authorized personnel only, and highlights patient-centric control over health data. **Advantage:** Offers strong privacy control and data confidentiality through

advanced encryption, supports role-based access, and enhances patient trust in cloud environments. **Disadvantage:** Remains conceptual without real-world implementation or performance validation, and may face practical integration and key management issues.

"A Fog-Based Middleware for Automated Compliance With OECD Privacy Principles in Internet of Healthcare Things" – Ahmed A. Alzahrani, Rajkumar Buyya^[8]. This study proposes a fog-based middleware architecture to ensure automated compliance with OECD privacy principles in Internet of Healthcare Things (IoHT) applications. The middleware acts as a bridge between healthcare applications and IoHT infrastructure, enforcing privacy policies through components like the Policy Mapper, Policy Decision Point (PDP), Policy Enforcement Point (PEP), and Context Handler. By translating human-readable privacy rules into enforceable system policies, the framework restricts unauthorized access and enforces purpose-based data sharing. **Advantage:** Enables real-time, automated privacy enforcement without manual intervention; ensures compliance with international privacy guidelines; reduces data exposure risks through fog-based local processing. **Disadvantage:** May introduce computational overhead at the fog layer; complexity in handling diverse privacy rules across jurisdictions; lacks empirical performance evaluation in real-world scenarios

"Digital Privacy in Healthcare: State-of-the-Art and Future Vision" – Khaled Salah, Raja Jayaraman, and Davor Svetinovic^[9]
This study presents a comprehensive analysis of digital privacy challenges in healthcare and explores current and future technologies to address these concerns.

The paper reviews existing privacy-preserving frameworks and emphasizes the importance of patient-centric privacy models, particularly in the era of digital health transformation. It discusses technologies such as blockchain, differential privacy, and secure multi-party computation, evaluating their roles in safeguarding sensitive patient information. Moreover, it highlights regulatory frameworks and the need for better interoperability and user control in managing health data. The future vision includes AI-driven privacy risk assessment tools and enhanced cryptographic protocols for real-time privacy enforcement. **Advantages:** Provides a broad overview of privacy technologies, integrates legal and technical perspectives, proposes forward-looking privacy enhancements using AI and cryptography, and supports patient empowerment in data control. **Disadvantages:** Lacks a specific implementation model, and some future visions are theoretical and may require significant development before practical adoption.

"Advancing Federated Learning Through Novel Mechanism for Privacy Preservation in Healthcare Applications" – Mehdi Sookhak, Abbas Kiani, Alireza Jolfaei, et al.^[10] This study presents a privacy-preserving approach for healthcare systems using a federated learning framework that eliminates the need for centralized data collection. The proposed mechanism ensures data confidentiality by training models locally on user devices and only sharing the model updates with the server. To strengthen privacy, the framework introduces a novel Hybrid Aggregation Technique (HAT) and employs optimized differential privacy techniques to prevent information leakage. This allows hospitals and medical institutions to collaboratively train accurate machine learning models without compromising sensitive patient data. The work also integrates blockchain for auditability and secure model update tracking.

Advantage: Preserves data privacy by avoiding raw data sharing, enhances accuracy through hybrid aggregation, provides auditability with blockchain integration, and ensures privacy with optimized differential privacy. **Disadvantage:** Increased complexity in implementation due to integration of multiple technologies like federated learning, differential privacy, and blockchain; performance may be affected by heterogeneous local data and device limitations.

"Hilbert Convex Similarity for Highly Secure Random Distribution of Patient Privacy Steganography" – V. Rajesh, S. Vijayakumar, T. Devi, K. Balasubramanian^[11]. This study proposes a novel steganography method using Hilbert Convex Similarity (HCS) to ensure high-security data hiding in patient privacy protection. The framework integrates secure data embedding with random distribution and advanced similarity-based encoding. It effectively conceals sensitive medical information in medical images using an optimized secret key generation and distribution method. The HCS-based approach increases imperceptibility and security by maintaining similarity between the original and stego image, making unauthorized extraction difficult. Experimental analysis confirms the method's robustness, image quality retention, and resilience against steganalysis attacks. **Advantage:** Ensures highly secure data hiding using random distribution and similarity-based encoding, maintains image quality and resists steganalysis. **Disadvantage:** Computational complexity may increase due to multiple transformation steps and secret key generation; limited to image-based data hiding scenarios.

CHAPTER 3

THEORETICAL BACKGROUND

CHAPTER 3

THEORETICAL BACKGROUND

3.1. IMPLEMENTATION ENVIRONMENT

The implementation of the **Secure Healthcare Management System** is carried out in a robust and scalable environment to ensure efficient execution, security, and ease of integration. The system is developed using modern technologies that provide secure data handling, user authentication, and encrypted communication.

3.1.1 HARDWARE ENVIRONMENT

The following hardware configurations are required to ensure smooth execution of the system:

- **Processor:** Intel Core i5 or higher
- **RAM:** Minimum 8GB DDR4
- **Hard Disk:** At least 250GB SSD for fast data access
- **Network:** Secure LAN/WiFi connectivity for data exchange
- **GPU (Optional):** For advanced computations if needed

3.1.2 SOFTWARE ENVIRONMENT

To build and deploy the system, the following software components are used:

- **Operating System:** Windows 10 / Linux Ubuntu 20.04
- **Frontend Technologies:**
 - HTML, CSS, JavaScript, Bootstrap (for responsive UI)

- **Backend Technologies:**
 - Java, Spring Boot (for API and business logic)
- **Database Management:**
 - MySQL (for structured data storage)
- **Blockchain Implementation:**
 - Hyperledger Fabric / Ethereum for decentralized storage
- **Security Algorithms:**
 - ForodoKEM for data encryption
 - SHA-256 for hashing sensitive information
- **Development Environment:**
 - **IDE:** Spring Tool Suite (STS) / IntelliJ IDEA
 - **Version Control:** GitHub / GitLab for source code management
 - **Testing Tools:** JUnit for unit testing, Postman for API testing.

3.1.3 TECHNOLOGIES UTILIZED

Blockchain

Blockchain is a **decentralized and secure digital ledger** technology that stores data in blocks linked together in a chain. Each block contains data, a timestamp, and a cryptographic hash of the previous block, making it tamper-proof. In healthcare, blockchain is used to **securely store patient data**, prevent unauthorized access, and maintain data integrity across distributed systems.

FrodoKEM

FrodoKEM is a **post-quantum cryptographic algorithm** designed to stay secure even against future quantum computers. It belongs to a class of algorithms called **Key Encapsulation Mechanisms (KEM)**, which are used for safely sharing encryption keys. FrodoKEM is based on hard mathematical problems (Learning with Errors) and is used in systems that need **high security for key exchange**, such as healthcare platforms protecting patient data.

.3.2 SYSTEM ARCHITECTURE

The systems architect establishes the basic structure of the system, we propose a Hash code Solomon algorithm and we can put a small part of data in local machine and fog server in order to protect the privacy. Moreover, based on computational intelligence, this algorithm can compute the distribution proportion stored in cloud, fog, and local machine, respectively..

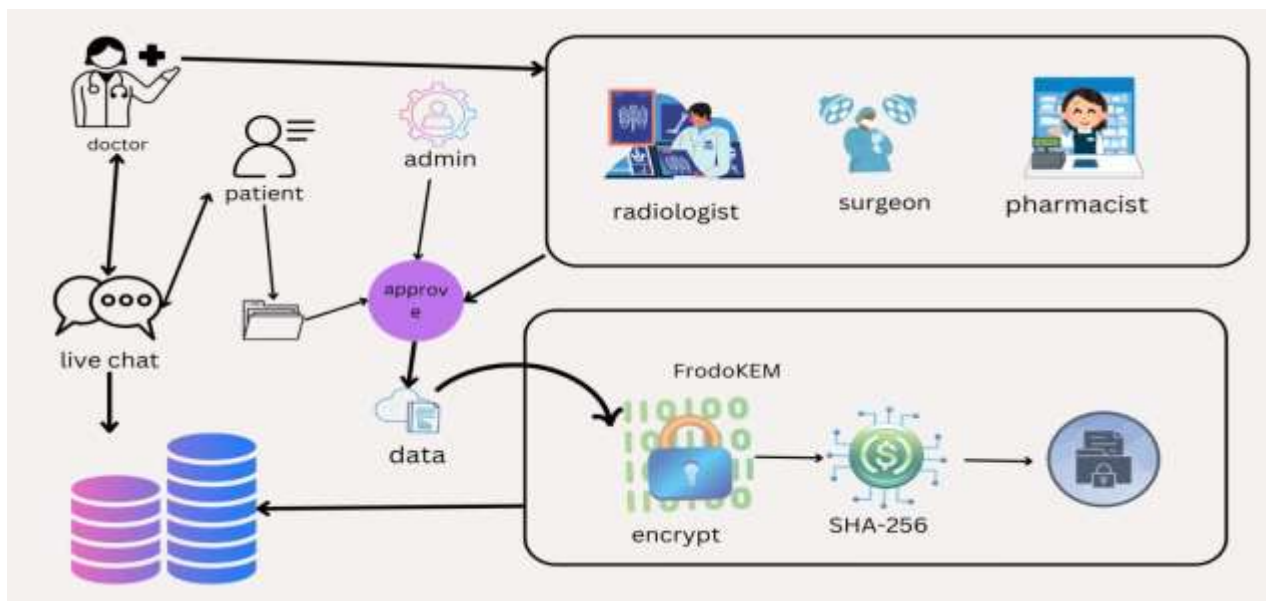


Fig 3.1 System Architecture Diagram

3.3 PROPOSED METHODOLOGY

The proposed system addresses privacy concerns in healthcare by integrating edge computing and blockchain technology to create a secure and decentralized data management environment. Patient data is processed locally on edge devices within healthcare institutions, minimizing the risk of data breaches associated with cloud storage and ensuring faster response times. A blockchain ledger is used to record all access and transactions transparently, preventing tampering and unauthorized changes. Encryption techniques like **FrodoKEM** and **SHA-256** protect sensitive health information, while the open-source nature of the system ensures adaptability and transparency. Patients maintain full control over their medical records through data sovereignty, allowing them to selectively grant access to authorized healthcare providers. This architecture not only strengthens privacy and security but also builds patient trust by offering transparency, accountability, and compliance with privacy regulations.

OBJECTIVES

The primary objective of this project is to design and implement a secure and efficient healthcare data management system by leveraging blockchain technology and encryption mechanisms. The system aims to enhance data protection, streamline communication between healthcare departments, and ensure patient confidentiality and regulatory compliance.

- **Enhance Data Security and Integrity:**

Implement blockchain technology to create a secure and tamper-proof ledger for recording patient information. This ensures that all data remains accurate, immutable, and protected from unauthorized alterations or tampering.

- **Streamline Specialist and Department Integration:**

Develop a structured registration and login system for various healthcare professionals—such as client specialists, radiologists, surgeons, and pharmacists—allowing them to access patient records through a role-based endorsement process managed by administrators.

- **Maintain Patient Confidentiality:**

Introduce a unique encrypted identifier system for each patient to prevent data misuse. This ensures that only authorized personnel from relevant departments can access sensitive health information.

- **Facilitate Efficient Data Handling Across Departments:**

Enable seamless communication and data flow between different healthcare departments by forwarding encrypted patient identifiers. This helps maintain accuracy and security while transitioning patient data across various stages of medical care.

- **Ensure Compliance and System Reliability:**

Build a robust and trustworthy system that complies with existing healthcare data protection standards. The system should foster patient trust by ensuring the integrity, confidentiality, and availability of personal medical information at all times.

3.3.1 INPUT DESIGN

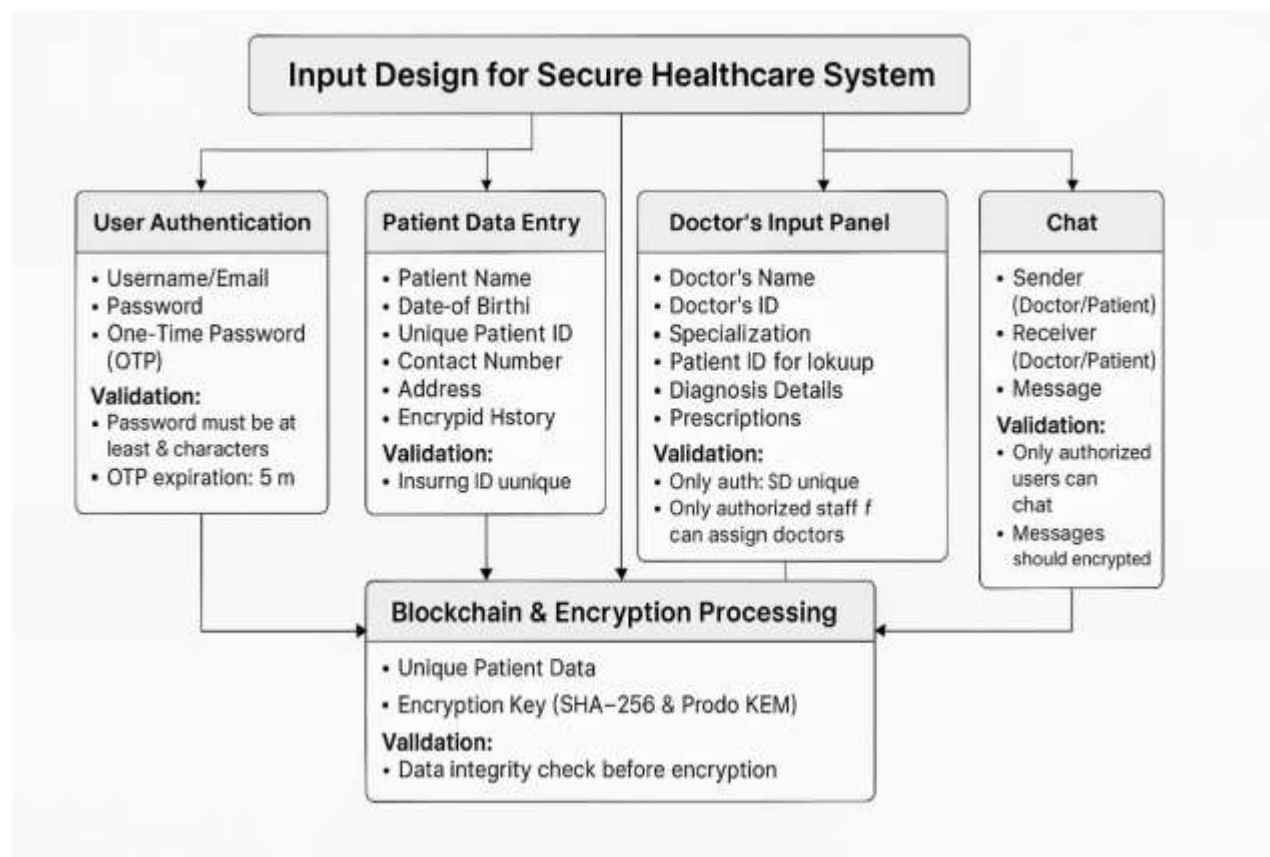


Fig 3.2 Input Design

The secure healthcare system ensures authorized access through user authentication using a **username**, data entry includes **personal details, contact information, and a unique ID**, ensuring no duplicates. Doctors can access a panel to **enter diagnoses, prescriptions, and lookup patient details**, with only authorized staff allowed to assign doctors. A **secure chat system** enables encrypted communication between doctors and patients, restricting access to authorized users. All data is processed using **SHA-256 and FrodoKEM encryption**, ensuring integrity and protection against unauthorized access.

3.4 UML DIAGRAM

Usecase Diagram

The use case diagram is the main building block of object oriented modeling. It is used both for general conceptual modeling of the systematic of the application, and for detailed modeling translating the models into programming code. For this in our component diagram first propose a data In this proposed method we are using Hash-Solomon Code Algorithm to encrypt the data.

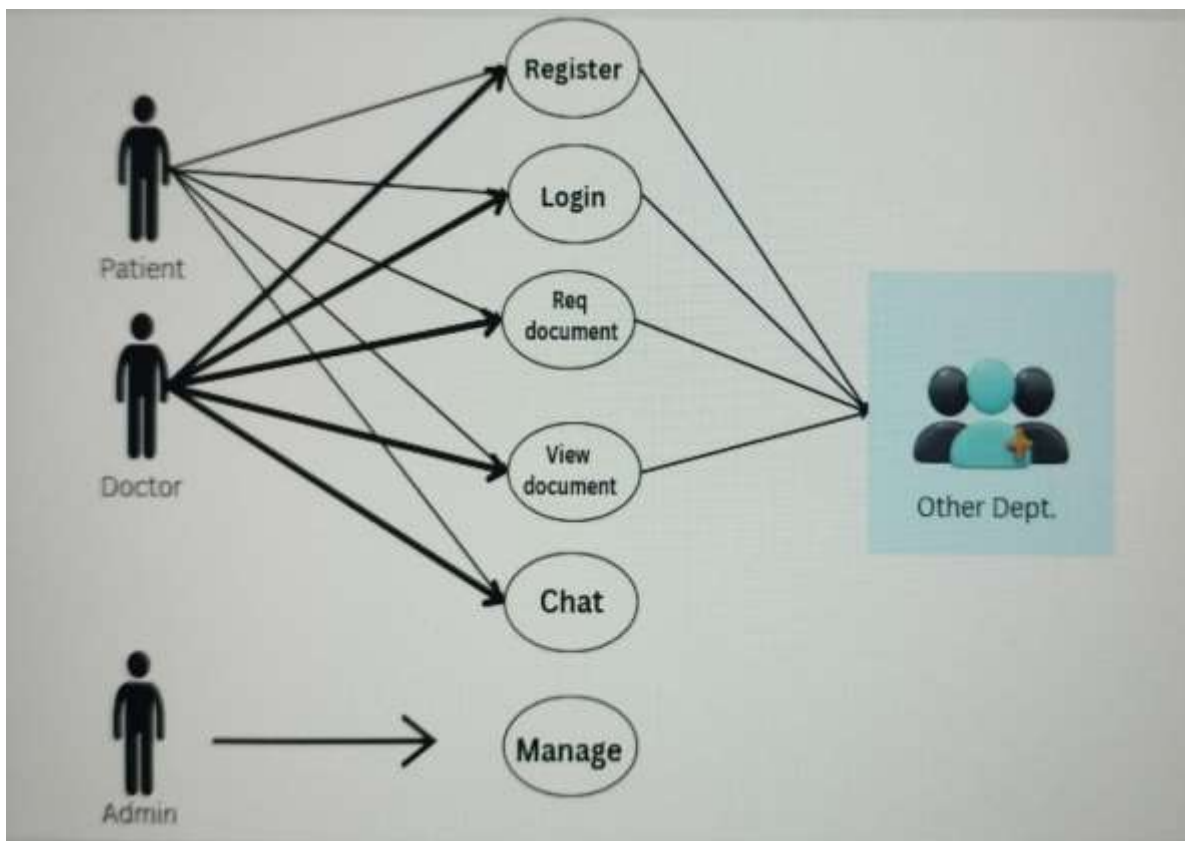


Fig 3.3 Usecase Diagram

Activity Diagram

Activity diagram is another important diagram in UML to describe the dynamic aspects of the system. Activity diagram is basically a flowchart to represent the flow from one activity to another activity. The activity can be described as an operation of the system. The control flow is drawn from one operation to another. This flow can be sequential, branched, or concurrent. Activity diagrams deal with all type of flow control by using different elements such as fork, join, etc.

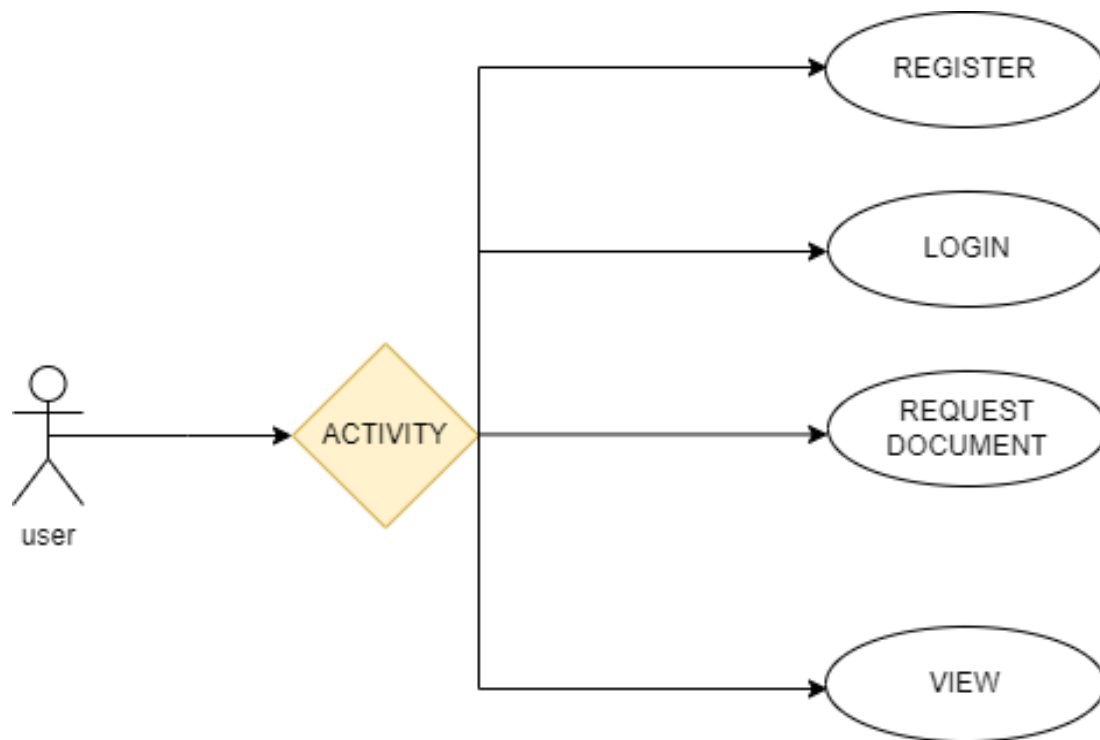


Fig 3.4 Activity Diagram

Sequence Diagram

In our sequence diagram specifying processes operate with one another and in order. In our sequence diagram first propose a f or this in our component diagram first propose a data in this proposed method we are using Hash-Solomon Code Algorithm to encrypt the data.

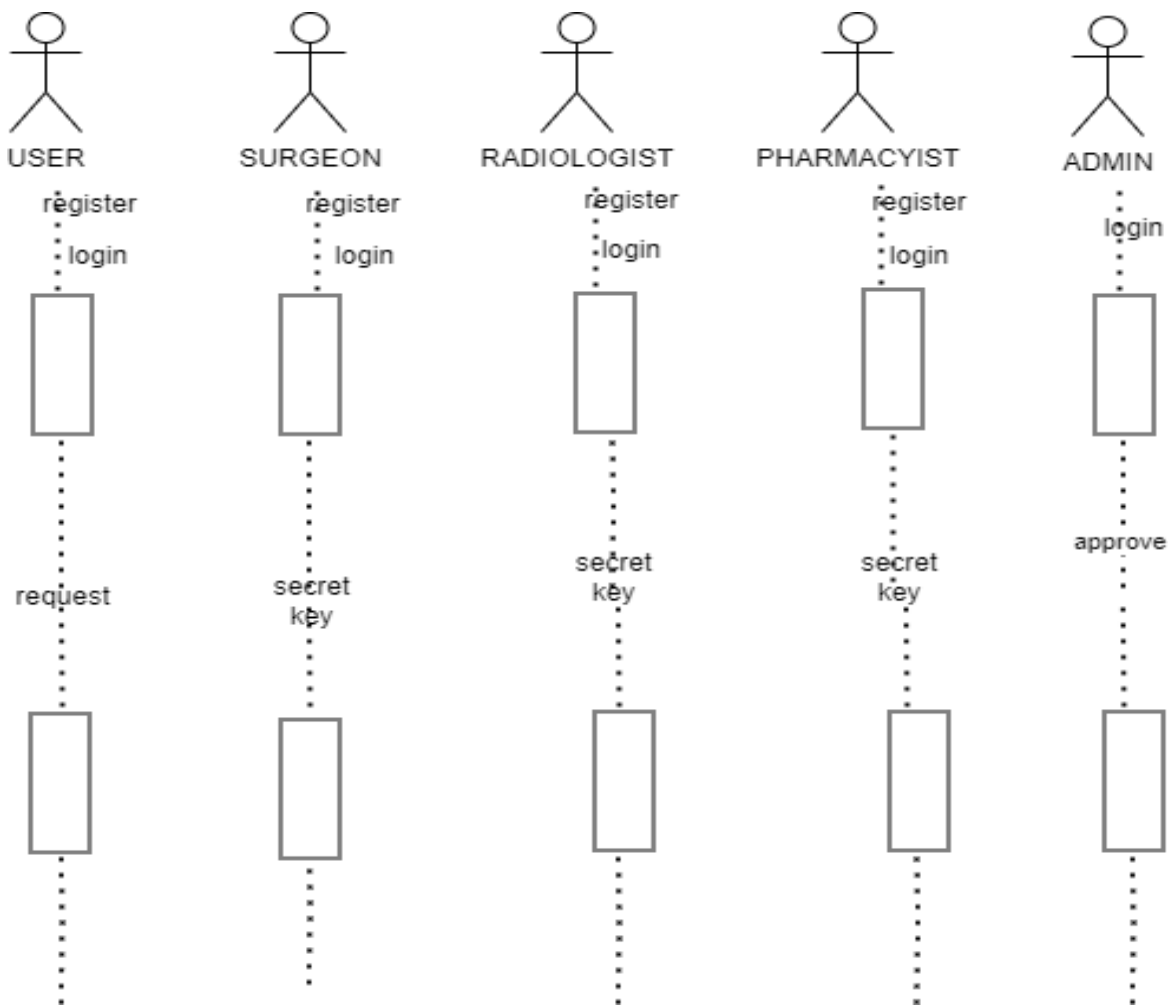


Fig 3.5 Sequence Diagram

ER Diagram

The diagram represents a structured request approval system in a secure healthcare environment. Users initiate requests such as applying for access or approvals, which are stored in a central database. Administrators manage the approval process by logging in, verifying users, and forwarding requests to relevant departments. Departments cross-verify the requests to ensure compliance and security before granting final approval. If validated, the request is either approved or accepted, allowing users to proceed. This workflow ensures that only authorized users gain access, enhancing data integrity, privacy, and security within the system.

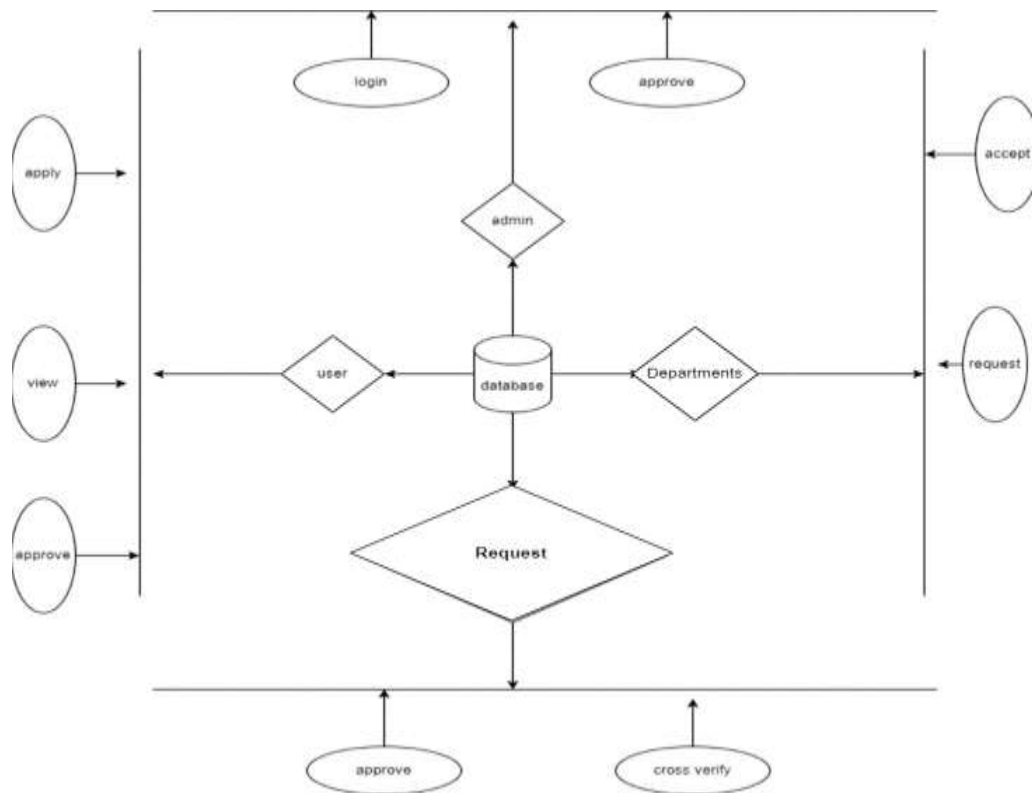


Fig 3.6 ER Diagram

DF Diagram

The diagram illustrates a secure healthcare data management system where different users, including doctors, surgeons, radiologists, and pharmacists, interact with a centralized database. Users can send requests, while doctors can upload data and engage in live chat for communication. Each department accesses data by entering a secret key, ensuring security before viewing information. Admins oversee the approval process, maintaining control over access permissions. This structure enhances security, ensures authorized data access, and facilitates seamless collaboration among healthcare professionals while protecting patient data.

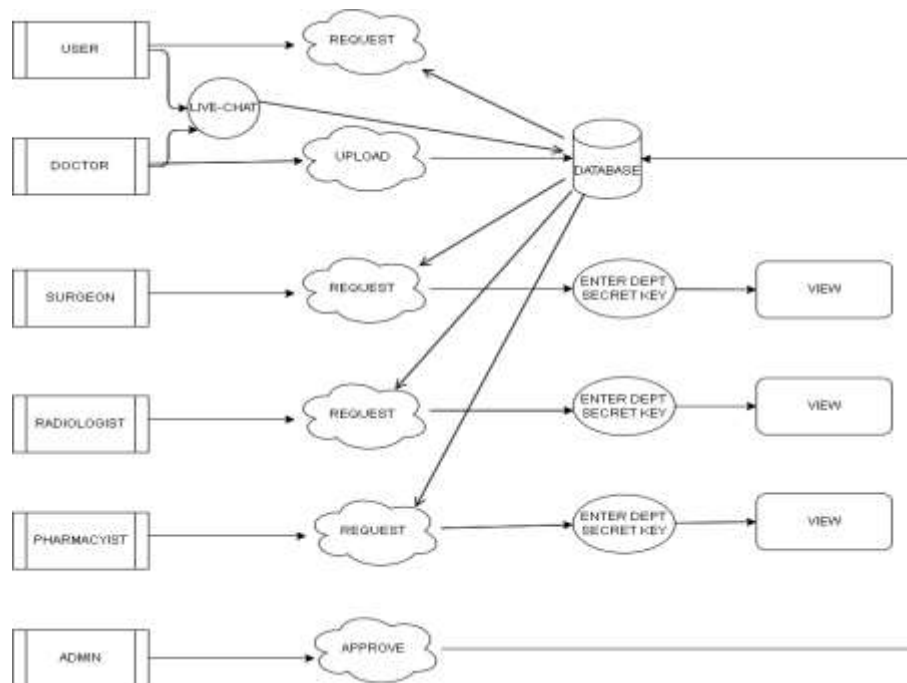


Fig 3.7 DF Diagram

State Diagram

A record is maintained containing medical details with an assigned ID for prescription; notes of the treatment are mentioned and prescription along with an ForodoKEM encrypted key generated and transmitted, encrypted, with unique IDs by various departments-surgery, radiology, or the pharmacy department-departmental authority decryption with individual IDs using keys access for prescriptions. Any changes or updates to the treatment plan are documented and stored in the hospital's database. Data at rest is encrypted using the SHA-256 hashing algorithm. This combination of AES encryption for transmission and SHA-256 hashing for storage ensures data confidentiality and integrity at all stages of the process.

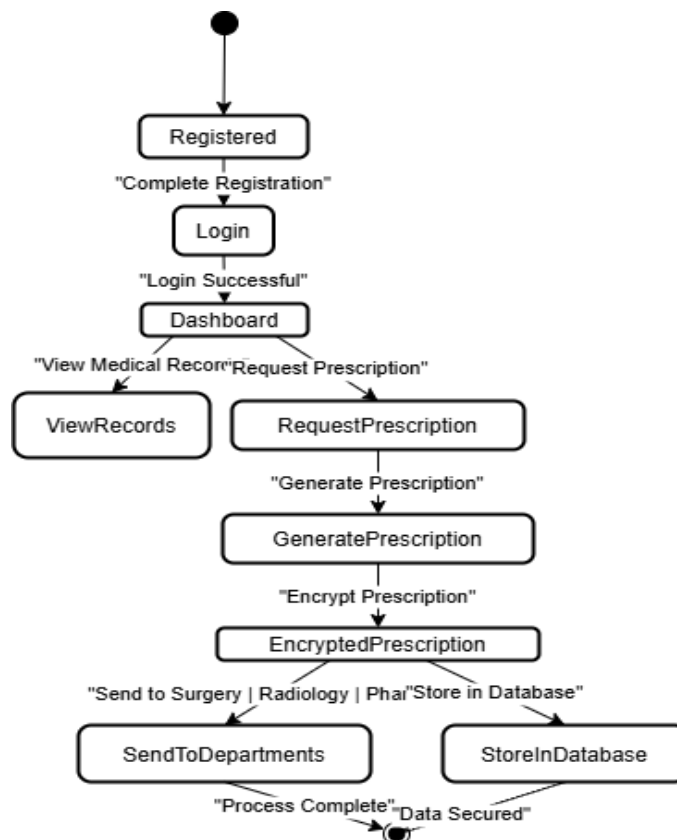


Fig 3.8 State Diagram

CHAPTER 4

SYSTEM IMPLEMENTATION

CHAPTER 4

SYSTEM IMPLEMENTATION

4.1 MODULES

4.1.1 USER MODULE

The **User Module** is a fundamental part of our secure healthcare system, enabling users to register, log in, request documents, and view authorized medical records. Users begin by registering their details to create a secure account, ensuring only authorized individuals access the system. After successful registration, they can log in using their credentials, undergoing authentication for security. Through the document request feature, users can formally request access to specific healthcare records, which are securely processed to maintain data privacy. Once approved, users can view the requested documents within their access permissions. This module ensures a secure, efficient, and privacy-focused system for managing sensitive patient data while preventing unauthorized access.

4.1.2 ADMIN MODULE

The **Admin Module** plays a crucial role in managing doctor registrations and maintaining system integrity within the secure healthcare system. The process begins with the admin logging into the system, ensuring only authorized personnel can oversee doctor approvals. Upon logging in, the admin can view doctor registration requests submitted by healthcare professionals. After reviewing these requests, the admin has the authority to approve or reject them based on verification criteria. Once approved, the system maintains the records, ensuring that only authenticated doctors can access patient data securely. This module

ensures a streamlined, secure, and efficient approval process for doctors while maintaining system integrity.

4.1.3 DOCTORS MODULE

The **Doctor Consultation Module** is a structured system that allows doctors to manage patient interactions efficiently. The process begins with **registration**, where doctors create an account to access the platform, or they can directly **log in** if they are already registered. After logging in, doctors are directed to the **dashboard**, which serves as the central hub for managing their activities. From the dashboard, doctors can **view free consultations**, allowing them to monitor and respond to general patient inquiries. They can also **provide suggestions**, offering medical advice based on patient concerns. Additionally, doctors can **view patient requests**, where they can see appointment or consultation requests submitted by patients. Another key feature is the ability to **accept or reject appointments**, enabling doctors to manage their schedules effectively. If an appointment is accepted, doctors can engage in **manual chat with patients**, facilitating direct communication for more personalized medical guidance. This module enhances doctor-patient interaction by streamlining consultation processes, improving accessibility, and ensuring efficient communication, all while maintaining data security and confidentiality.

4.1.4 SURGEON MODULE

The **Surgeon Access Module** is a secure system that allows surgeons to access patient reports through a multi-step authentication process. First, the surgeon must

register in the system and log in using their credentials. After logging in, they can navigate to the patient reports section. To enhance security, the system requires the surgeon to enter a **department secret key** for authorization, followed by their **staff ID** for additional verification. Once all credentials are validated, the surgeon can submit the request and gain access to the reports. This module ensures strict authentication, restricted access, and data confidentiality, preventing unauthorized access to sensitive patient information.

4.1.5 RADIOLOGIST MODULE

The **Radiologist Access Module** is a secure system that enables radiologists to access patient reports while ensuring strict authentication and authorization. The process begins with the **registration**, where the radiologist creates an account to gain system access. After registration, they must **log in** using their credentials. Once authenticated, they can proceed to **view patient reports**, but access is further restricted through additional security measures. To ensure department-level authorization, the radiologist must enter a **department secret key**, which verifies their association with the medical division handling the reports. Further verification is required by entering their **staff ID**, ensuring that only authorized personnel can proceed. Once all credentials are validated, the **submit** action grants final access to the patient reports. This multi-layered security approach ensures that only registered and authorized radiologists can access sensitive medical data, maintaining confidentiality, preventing unauthorized access, and ensuring compliance with data protection regulations.

4.1.6 PHARMACIST MODULE

The **Pharmacist Access Module** is a secure system that allows pharmacists to access patient reports while ensuring authentication and data protection. The process starts with the **registration**, where the pharmacist creates an account to gain system access. Once registered, they must **log in** using their credentials before proceeding further. After successful login, they can navigate to the **patient reports section**, where they can view necessary medical information related to prescriptions and medications. However, access is restricted with multiple security layers. The pharmacist must enter a **department secret key** to verify their authorization within the system. Following this, they must provide their **staff ID** as an additional authentication step to confirm their identity. Once all security checks are passed, they can **submit** the request and gain access to the patient reports. This module ensures that only registered and authorized pharmacists can access sensitive medical data, maintaining confidentiality, preventing unauthorized access, and ensuring compliance with healthcare data security regulations.

4.1.7 CHAT MODULE

The Chat Module facilitates secure, real-time communication between patients and healthcare providers. It allows for text, audio, and video interactions, enabling telemedicine capabilities. All communications are protected by end-to-end encryption in accordance with the system's security measures, ensuring that sensitive conversations remain private.

4.2 ALGORITHMS

4.2.1 SHA 256

SHA-256 (Secure Hash Algorithm 256-bit) is a cryptographic hash function that generates a unique 256-bit (32-byte) fixed-size hash value for any given input, regardless of its size. Developed by the National Security Agency (NSA) and published by the National Institute of Standards and Technology (NIST), SHA-256 is part of the SHA-2 family of algorithms and is widely used in various security applications and protocols, including SSL/TLS, digital signatures, and blockchain technology. Unlike encryption, SHA-256 is a one-way function; it is designed to be irreversible, meaning that the original input cannot be reconstructed from the hash value. The algorithm works by processing the input data in 512-bit blocks and applying a series of bitwise operations, logical functions, and modular arithmetic to generate the final hash. This ensures that even a small change in the input (e.g., flipping a single bit) results in a completely different hash output, demonstrating the algorithm's high sensitivity to input variations. SHA-256 is known for its robustness and security, making it highly resistant to collision attacks (where two different inputs produce the same hash) and preimage attacks (where an input is derived from a given hash). Due to its efficiency and strong security properties, SHA-256 remains a popular and trusted choice for ensuring data integrity and authenticity in modern cryptographic systems.

4.2.2 ForodoKEM: A Post-Quantum Key Encapsulation Mechanism

FrodoKEM is a **post-quantum cryptographic algorithm** designed for secure key exchange, resistant to attacks from quantum computers. It is based on the **Learning With Errors (LWE)** problem, which involves adding controlled noise to matrix multiplications, making it computationally infeasible to reverse. Unlike

structured lattice-based schemes, FrodoKEM uses **unstructured lattices**, providing stronger security but requiring more computational resources.

The algorithm follows three main steps:

- **Key Generation:** The recipient creates a public-private key pair using a randomly chosen matrix and error values.
- **Encapsulation:** The sender encrypts a randomly generated **shared secret key** using the recipient's public key, producing a ciphertext.
- **Decapsulation:** The recipient decrypts the ciphertext with their private key to recover the shared secret key, which is then used for secure communication.

FrodoKEM is highly secure and resistant to both classical and quantum attacks. However, it requires **larger key sizes and higher computational power** compared to traditional methods. Despite this, it is a strong candidate for securing future communications in sensitive fields like **healthcare, finance, and cloud security**.

4.2.3 Blockchain Technology in Secure Healthcare Systems

Blockchain technology is a decentralized and distributed ledger system that ensures secure, transparent, and tamper-proof data management. In this project, blockchain plays a crucial role in securing patient data by providing immutable records, preventing unauthorized access, and enhancing data privacy. By integrating blockchain, healthcare systems can overcome challenges related to data breaches, unauthorized modifications, and trust issues in electronic health records (EHRs).

Key Features of Blockchain in This System:

- **Decentralization:** Unlike traditional centralized databases, blockchain distributes data across multiple nodes, eliminating the risk of single-point failures and reducing dependency on a single authority. This ensures that patient records are always available, even in the event of system failures.
- **Immutability:** Once recorded, patient data cannot be altered or deleted, ensuring data integrity and preventing malicious modifications. Every transaction is permanently stored, creating an audit trail that can be used for verification and compliance purposes.
- **Security:** Using advanced cryptographic techniques such as SHA-256 hashing, FrodoKEM encryption, and digital signatures, blockchain ensures that patient records remain protected from cyber threats, unauthorized access, and data manipulation. Encryption safeguards patient confidentiality, while cryptographic hashing secures data integrity.
- **Transparency and Trust:** Blockchain's distributed ledger provides a transparent system where all authorized stakeholders can verify records without compromising patient privacy. Smart contracts can automate access control, ensuring that only authorized personnel can access or update records based on predefined rules.
- **Interoperability:** Blockchain facilitates seamless data sharing between healthcare providers, insurance companies, and patients while maintaining security and compliance with regulations like HIPAA and GDPR. This eliminates data silos and enhances coordination among healthcare entities.

CHAPTER 5

RESULTS AND DISCUSSION

CHAPTER 5

RESULTS AND DISCUSSION

5.1 SYSTEM TESTING

White Box Testing

White box testing is crucial for verifying the internal logic, security mechanisms, and blockchain transactions in the system. It ensures that patient data remains immutable by validating encryption mechanisms like SHA-256 hashing and FrodoKEM encryption. This testing method also examines role-based access control (RBAC) to confirm that only authorized users, such as doctors, pharmacists, and radiologists, can access patient data. Additionally, it helps in error handling and exception testing by ensuring that invalid inputs, authentication failures, and unauthorized access attempts are properly managed. Security testing within white box testing verifies encryption algorithms, digital signatures, and smart contract execution, preventing cyber threats. Furthermore, it optimizes code by detecting redundant logic, inefficient loops, and security vulnerabilities, ensuring a robust and efficient system.

Black Box Testing

Black box testing is essential for evaluating the functional behavior of the healthcare system without inspecting its internal code structure. This method focuses on verifying whether patient data access, appointment scheduling, and medical report retrieval work as intended. Testers provide inputs and examine the outputs without considering how the system processes data internally. It ensures that the dashboard functionalities for doctors, pharmacists, and radiologists operate

correctly, preventing incorrect access to confidential patient records. Since blockchain integration adds complexity, black box testing validates whether transactions are successfully recorded and retrieved while ensuring system usability and responsiveness for healthcare professionals and patients.

Unit Testing

Unit testing is critical for validating individual **modules and components** of the healthcare system. Each feature, such as user registration, login authentication, patient report access, and appointment management, is tested independently to ensure it functions correctly. In blockchain-based systems, unit testing is particularly important for verifying smart contract execution, cryptographic key management, and transaction processing to ensure data integrity. It also helps in debugging errors at an early stage, reducing risks before integrating different system components. By isolating and testing each module, unit testing ensures better performance, security, and reliability in handling sensitive healthcare data.

Functional Testing

Functional testing ensures that the healthcare system operates as per the defined requirements by testing each feature for expected behavior. It validates key functionalities such as secure login, encrypted patient data storage, appointment booking, and report sharing among doctors, radiologists, and pharmacists. The process involves providing inputs, executing test cases, and comparing actual outputs with expected results to ensure accuracy. Functional testing also checks whether blockchain transactions are correctly recorded, verified, and retrieved, preventing unauthorized modifications.

5.2 RESULTS & DISCUSSION

The implementation of the **Blockchain-Based Secure Healthcare System** successfully enhances data security, integrity, and accessibility for hospitals. The system encrypts patient records using **SHA-256 hashing** and **Prodo KEM encryption**, ensuring that sensitive medical data remains protected from unauthorized access and cyber threats. The key outcomes observed during the implementation and testing phases include:

- **Data Security & Integrity:** Patient records stored on the blockchain remain immutable, ensuring protection against tampering or unauthorized modifications.
- **Efficient Data Management:** The system enables hospitals to securely store, retrieve, and manage patient data, reducing administrative overhead.
- **Enhanced Privacy:** Cryptographic techniques ensure that only authorized personnel can access patient information, preventing data breaches.
- **Decentralized Access:** The blockchain network ensures that data is distributed across multiple nodes, eliminating single points of failure and improving system reliability.
- **Improved Traceability:** Every transaction in the system is logged, enabling hospitals to track data modifications and maintain transparency.

The results demonstrate that blockchain technology provides a **secure and reliable** solution for managing healthcare data. Traditional centralized databases are vulnerable to hacking, data breaches, and single-point failures, whereas blockchain offers a **decentralized and tamper-proof alternative**.

However, the study also highlights certain challenges:

- **Computational Overhead:** Implementing blockchain increases processing time due to encryption and consensus mechanisms.
- **Storage Requirements:** Since blockchain maintains a complete transaction history, data storage can grow significantly over time.
- **Scalability Issues:** As the number of records increases, the network requires more computational power and efficient optimization strategies.

Despite these challenges, the system proves to be an **effective solution for securing healthcare records**, ensuring regulatory compliance, and reducing risks associated with data breaches. Future improvements may focus on optimizing **storage efficiency**, **reducing computational overhead**, and integrating **smart contracts** for automated verification and secure access control.

CHAPTER 6

CONCLUSION AND FUTURE WORK

6.1. CONCLUSION

In summary, your project integrates blockchain technology to enhance the integrity and confidentiality of patient data within a healthcare system. The system involves multiple roles, including client specialists, radiologists, surgeons, pharmacists, and administrators, each with specific responsibilities and access rights. After a patient registers and logs in, they receive a unique encrypted number from their specialist, which is then forwarded to various departments such as surgery, radiology, and pharmacy. This unique number ensures that sensitive patient information is protected and cannot be misused. The blockchain framework guarantees data integrity by providing an immutable ledger, while encryption and a structured endorsement process safeguard patient confidentiality. This approach not only streamlines data handling across different departments but also builds a secure and reliable system that fosters trust and efficiency in patient care.

6.2 FUTURE ENHANCEMENTS

Future improvements will focus on optimizing storage and scalability by integrating off-chain storage solutions and hybrid blockchain models. Implementing smart contracts can automate access control, ensuring secure and transparent data retrieval. Enhancing interoperability with Electronic Health Records (EHR) and Hospital Information Systems (HIS) will enable seamless data exchange. Performance optimization through lightweight consensus algorithms like Proof-of-Authority (PoA) can reduce energy consumption.

APPENDICES

A.1 SDG Goals

SDG 3 – Ensure healthy lives and promote well-being for all at all ages

A.2 Source Code

Doctor module

```
package com.spring.graph.api.services;
import java.util.Optional;
import org.springframework.beans.factory.annotation.Autowired;
import org.springframework.stereotype.Service;
import com.spring.graph.api.entity.Docterreg;
import com.spring.graph.api.repository.Docregrepository;
```

```
@Service
```

```
public class Docservice {
```

```
@Autowired
```

```
private Docregrepository docrepo;
public boolean checkEmail(String email) {
// TODO Auto-generated method stub
return docrepo.existsByEmail(email);
}
```

```
public Docterreg updatedoctorstatus(Docterreg seller) {
```

```
return docrepo.save(seller);
}
```

```
public Docterreg getdoctorid(Long id) {
```

```
Optional<Docterreg> seller = docrepo.findById(id);
return seller.orElse(null);
```

```
}
```



```

public Docterreg getdocByEmailAndPassword(String sellemail, String
sellpassword) {
// TODO Auto-generated method stub
System.out.println("qqqqqqqq");
Optional<Docterreg> userOptional =
docrepo.findByEmailAndPassword(sellemail, sellpassword);
if (userOptional.isPresent()) {
Docterreg user = userOptional.get();

if (user.getStatus().equalsIgnoreCase("Approved")) {

System.out.println("Seller status is approved for email: " + sellemail);

return user;
} else {
// If status is not "Approved", print a message and return null
System.out.println("Seller status is not approved for email: " + sellemail);
return null;
}
} else {
System.out.println("User not found for email: " + sellemail);
return null;
}
}

}

package com.spring.graph.api.entity;

import jakarta.persistence.Entity;
import jakarta.persistence.GeneratedValue;
import jakarta.persistence.GenerationType;
import jakarta.persistence.Id;

@Entity
public class Docterreg {

```

```

    @Id

```

```

@GeneratedValue(strategy = GenerationType.IDENTITY)
private long id;

private String name;
private String email;
private String password;
private String doctors;
private String contact;
private String image;
// Default constructor
private String status;
public Docterreg(Object ob) {
    // TODO Auto-generated constructor stub
}

public Docterreg() {
    super();
}

// Constructor with parameters

// Getters and Setters
public long getId() {
    return id;
}

public Docterreg(long id, String name, String email, String password, String
doctors, String contact, String image,
                String status) {
    super();
    this.id = id;
    this.name = name;
    this.email = email;
    this.password = password;
    this.doctors = doctors;
    this.contact = contact;
    this.image = image;
    this.status = status;
}

```

```

        public void setId(long id) {
            this.id = id;
        }

        public String getName() {
            return name;
        }

        public void setName(String name) {
            this.name = name;
        }

        public String getEmail() {
            return email;
        }

        public void setEmail(String email) {
            this.email = email;
        }

        public String getPassword() {
            return password;
        }

        public void setPassword(String password) {
            this.password = password;
        }

        public String getDoctors() {
            return doctors;
        }

        public void setDoctors(String doctors) {
            this.doctors = doctors;
        }

        public String getContact() {
            return contact;
        }

```

```

    public void setContact(String contact) {
        this.contact = contact;
    }

    public String getStatus() {
        return status;
    }
    public void setStatus(String status) {
        this.status = status;
    }
    public String getImage() {
        return image;
    }

    public void setImage(String image) {
        this.image = image;
    }
}

```

Pharmacy Module

```

package com.spring.graph.api.services;
import java.util.Optional;
import org.springframework.beans.factory.annotation.Autowired;
import org.springframework.stereotype.Service;
import com.spring.graph.api.entity.pharmacyentity;
import com.spring.graph.api.repository.pharmacyrepo;

@Service
public class pharmacyService {

    @Autowired
    private pharmacyrepo pharmacyrepo;
    public boolean checkEmail(String email) {
        // TODO Auto-generated method stub
    }
}

```

```

return pharmacyrepo.existsByEmail(email);
}
public pharmacyentity updatedpharmacystatus(pharmacyentity seller) {

return pharmacyrepo.save(seller);
}

public pharmacyentity getpharmacyid(Long id) {

Optional<pharmacyentity> seller = pharmacyrepo.findById(id);
return seller.orElse(null);
}

public pharmacyentity getdocByEmailAndPassword(String sellemail, String
sellpassword) {
// TODO Auto-generated method stub
System.out.println("qqqqqqqqq");
Optional<pharmacyentity> userOptional =
pharmacyrepo.findByIdByEmailAndPassword(sellemail, sellpassword);
if (userOptional.isPresent()) {
pharmacyentity user = userOptional.get();

if (user.getStatus().equalsIgnoreCase("Approved")) {

System.out.println("pharmacy status is approved for email: " + sellemail);

return user;
} else {
// If status is not "Approved", print a message and return null
System.out.println("pharmacy status is not approved for email: " + sellemail);
return null;
}
} else {
System.out.println("pharmacy not found for email: " + sellemail);
return null;
}
}
}
}

```

```

package com.spring.graph.api.entity;
import jakarta.persistence.Entity;
import jakarta.persistence.GeneratedValue;
import jakarta.persistence.GenerationType;
import jakarta.persistence.Id;

@Entity
public class pharmacyentity {
    @Id
    @GeneratedValue(strategy = GenerationType.IDENTITY)
    private long id;
    private String name;
    private String email;
    private String password;
    private String contact;
    private String image;
    // Default constructor
    private String catagory;
    private String status;
    public pharmacyentity() {

    }

    public pharmacyentity(long id, String name, String email, String password,
String contact, String image,
        String catagory, String status) {
        super();
        this.id = id;
        this.name = name;
        this.email = email;
        this.password = password;
        this.contact = contact;
        this.image = image;
        this.catagory = catagory;
        this.status = status;
    }
    public long getId() {
        return id;
    }
    public void setId(long id) {

```

```

        this.id = id;
    }
    public String getName() {
        return name;
    }
    public void setName(String name) {
        this.name = name;
    }
    public String getEmail() {
        return email;
    }
    public void setEmail(String email) {
        this.email = email;
    }
    public String getPassword() {
        return password;
    }
    public void setPassword(String password) {
        this.password = password;
    }
    public String getContact() {
        return contact;
    }
    public void setContact(String contact) {
        this.contact = contact;
    }
    public String getImage() {
        return image;
    }
    public void setImage(String image) {
        this.image = image;
    }
    public String getCatagory() {
        return catagory;
    }
    public void setCatagory(String catagory) {
        this.catagory = catagory;
    }
    public String getStatus() {
        return status;
    }

```

```

    }
    public void setStatus(String status) {
        this.status = status;
    }

    @Override
    public String toString() {
        return "pharmacyentity [id=" + id + ", name=" + name + ", email=" + email +
            ", password=" + password + ", catagory=" + catagory + ", contact=" + contact + " ,
            image=" + image +
                ", status=" + status + " ]";
    }
}

```

Radiologis Module

```

package com.spring.graph.api.services;
import java.util.Optional;
import org.springframework.beans.factory.annotation.Autowired;
import org.springframework.stereotype.Service;
import com.spring.graph.api.entity.radiologistentity;
import com.spring.graph.api.repository.radiologistrepo;
@Service
public class radiologistservices {

    @Autowired
    private radiologistrepo radiologistrepo;
    public boolean checkEmail(String email) {
        // TODO Auto-generated method stub
        return radiologistrepo.existsByEmail(email);
    }
    public radiologistentity updatedradiologiststatus(radiologistentity seller) {

        return radiologistrepo.save(seller);
    }

    public radiologistentity getradiologyid(Long id) {

```



```
Optional<radiologistentity> seller = radiologistrepo.findById(id);  
return seller.orElse(null);
```

```
}
```

```
public radiologistentity getdocByEmailAndPassword(String sellemail, String  
sellpassword) {  
    // TODO Auto-generated method stub  
    System.out.println("qqqqqqqq");  
    Optional<radiologistentity> userOptional =  
    radiologistrepo.findByIdByEmailAndPassword(sellemail, sellpassword);  
    if (userOptional.isPresent()) {  
        radiologistentity user = userOptional.get();  
  
        if (user.getStatus().equalsIgnoreCase("Approved")) {  
            System.out.println("Radiologist status is approved for email: " + sellemail);  
  
            return user;  
        } else {  
            // If status is not "Approved", print a message and return null  
            System.out.println("Radiologist status is not approved for email: " + sellemail);  
            return null;  
        }  
    } else {  
        System.out.println("User not found for email: " + sellemail);  
        return null;  
    }  
}
```

```
}  
package com.spring.graph.api.entity;  
import jakarta.persistence.Entity;  
import jakarta.persistence.GeneratedValue;  
import jakarta.persistence.GenerationType;  
import jakarta.persistence.Id;
```

```
@Entity  
public class radiologistentity {  
    @Id
```

```
@GeneratedValue(strategy = GenerationType.IDENTITY)
private long id;
```

```
private String name;
private String email;
private String password;
private String contact;
private String image;
// Default constructor
private String catagory;
private String status;
public radiologistentity() {

}
```

```
public radiologistentity(long id, String name, String email, String password, String
contact, String image,
String catagory, String status) {
super();
this.id = id;
this.name = name;
this.email = email;
this.password = password;
this.contact = contact;
this.image = image;
        this.catagory = catagory;
        this.status = status;
    }
    public long getId() {
        return id;
    }
    public void setId(long id) {
        this.id = id;
    }
    public String getName() {
        return name;
    }
    public void setName(String name) {
        this.name = name;
    }
}
```

```

    public String getEmail() {
        return email;
    }
    public void setEmail(String email) {
        this.email = email;
    }
    public String getPassword() {
        return password;
    }
    public void setPassword(String password) {
        this.password = password;
    }
    public String getContact() {
        return contact;
    }
    public void setContact(String contact) {
        this.contact = contact;
    }
    public String getImage() {
        return image;
    }
    public void setImage(String image) {
        this.image = image;
    }
    public String getCatagory() {
        return catagory;
    }
    public void setCatagory(String catagory) {
this.catagory = catagory;
    }
    public String getStatus() {
return status;
    }
    public void setStatus(String status) {
this.status = status;
    }

@Override
public String toString() {

```

```

return "surgeonentity [id=" + id + ", name=" + name + ", email=" + email + ",
password=" + password + ", catagory=" + catagory + ", contact=" + contact + " ,
image=" + image +
", status=" + status + " ]";
}

```

Surgeon Module

```

package com.spring.graph.api.services;
import java.util.Optional;
import org.springframework.beans.factory.annotation.Autowired;
import org.springframework.stereotype.Service;
import com.spring.graph.api.entity.sdocrequet;
import com.spring.graph.api.entity.surgeonentity;
import com.spring.graph.api.repository.surgeonrepo;

```

```

@Service
public class surgeonservices {

```

```

    @Autowired
    private surgeonrepo surgeonrepo;

```

```

    public boolean checkEmail(String email) {
        // TODO Auto-generated method stub
        return surgeonrepo.existsByEmail(email);
    }
    public surgeonentity updatedsurgeonstatus(surgeonentity seller) {

```

```

        return surgeonrepo.save(seller);
    }

```

```

    public surgeonentity getsurgeonid(Long id) {

```

```

        Optional<surgeonentity> seller = surgeonrepo.findById(id);
        return seller.orElse(null);

```

```

    }

```

```

public surgeonentity getdocByEmailAndPassword(String sellemail, String
selppassword) {
// TODO Auto-generated method stub
System.out.println("qqqqqqqq");
Optional<surgeonentity> userOptional =
surgeonrepo.findByEmailAndPassword(sellemail, selppassword);
if (userOptional.isPresent()) {
surgeonentity user = userOptional.get();
if (user.getStatus().equalsIgnoreCase("Approved")) {
System.out.println("Surgeon status is approved for email: " + sellemail);

return user;
} else {
// If status is not "Approved", print a message and return null
System.out.println("Surgeon status is not approved for email: " + sellemail);
return null;
}
} else {
System.out.println("Surgeon not found for email: " + sellemail);
return null;
}
}
}

package com.spring.graph.api.entity;

import jakarta.persistence.Entity;
import jakarta.persistence.GeneratedValue;
import jakarta.persistence.GenerationType;
import jakarta.persistence.Id;

@Entity
public class surgeonentity {

    @Id
    @GeneratedValue(strategy = GenerationType.IDENTITY)
    private long id;
    private String name;
    private String email;

```

```

private String password;

private String contact;
private String image;
// Default constructor

private String catagory;
private String status;


public surgeonentity() {

}
    public surgeonentity(long id, String name, String email, String password,
String contact, String image,
        String catagory, String status) {
        super();
        this.id = id;
        this.name = name;
        this.email = email;
        this.password = password;
        this.contact = contact;
        this.image = image;
        this.catagory = catagory;
        this.status = status;
    }
    public long getId() {
        return id;
    }
    public void setId(long id) {
        this.id = id;
    }
    public String getName() {
        return name;
    }
    public void setName(String name) {
        this.name = name;
    }
}

```

```

public String getEmail() {
    return email;
}
public void setEmail(String email) {
    this.email = email;
}
public String getPassword() {
    return password;
}
public void setPassword(String password) {
    this.password = password;
}
public String getContact() {
    return contact;
}
public void setContact(String contact) {
    this.contact = contact;
}
public String getImage() {
    return image;
}
public void setImage(String image) {
    this.image = image;
}
public String getCatagory() {
    return catagory;
}
public void setCatagory(String catagory) {
    this.catagory = catagory;
}
public String getStatus() {
    return status;
}
public void setStatus(String status) {
    this.status = status;
}
}

```

@Override

```

public String toString() {

```

```

        return "surgeonentity [id=" + id + ", name=" + name + ", email=" + email + ",
password=" + password + ", catagory=" + catagory + ", contact=" + contact + ",
image=" + image +
        ", status=" + status + "];";
    }

}

```

ForodoKEM

```

package com.spring.graph.api.algorithms;

import java.security.SecureRandom;
import java.util.Arrays;

public class FrodoKem {

    private static final int N = 512; // Dimension of the matrix
    private static final int Q = 12289; // Modulo value for operations

    // Simulate a secret key generation
    public static int[] generateSecretKey() {
        SecureRandom random = new SecureRandom();
        int[] secretKey = new int[N];
        for (int i = 0; i < N; i++) {
            secretKey[i] = random.nextInt(Q); // Random values within the range of Q
        }
        return secretKey;
    }

    // Simulate public key generation
    public static int[] generatePublicKey(int[] secretKey) {
        // In FrodoKEM, the public key generation involves a matrix-based operation
        int[] publicKey = new int[N];
        for (int i = 0; i < N; i++) {
            publicKey[i] = (secretKey[i] + new SecureRandom().nextInt(Q)) % Q;
        }
        return publicKey;
    }
}

```



```

// Encapsulation (Encryption step)
public static int[] encapsulate(int[] publicKey) {
    int[] ciphertext = new int[N];
    SecureRandom random = new SecureRandom();
    for (int i = 0; i < N; i++) {
        // Simulate the encapsulation process by adding randomness to the public
key
        ciphertext[i] = (publicKey[i] + random.nextInt(Q)) % Q;
    }
    return ciphertext;
}

// Decapsulation (Decryption step)
public static int[] decapsulate(int[] ciphertext, int[] secretKey) {
    int[] decryptedMessage = new int[N];
    for (int i = 0; i < N; i++) {
        decryptedMessage[i] = (ciphertext[i] - secretKey[i]) % Q;
    }
    return decryptedMessage;
}

// Testing the FrodoKEM Algorithm
public static void main(String[] args) {
    // Generate a secret key and public key
    int[] secretKey = generateSecretKey();
    int[] publicKey = generatePublicKey(secretKey);

    // Encrypt (Encapsulate)
    int[] ciphertext = encapsulate(publicKey);
    System.out.println("Ciphertext: " + Arrays.toString(ciphertext));

    // Decrypt (Decapsulate)
    int[] decryptedMessage = decapsulate(ciphertext, secretKey);
    System.out.println("Decrypted          Message:          "
+
Arrays.toString(decryptedMessage));
}
}

```

A.3 SCREEN SHOTS

HOME PAGE



Fig A.3.1 Home Page

USER REGISTRATION AND LOGIN

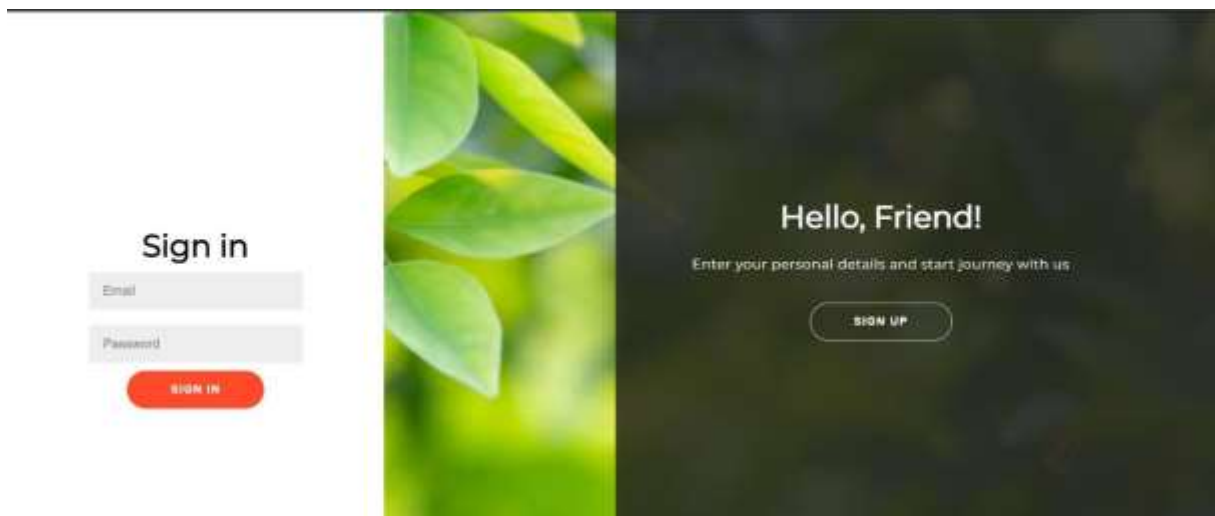


Fig A.3.2 User Registration and Login Page

USER MODULE

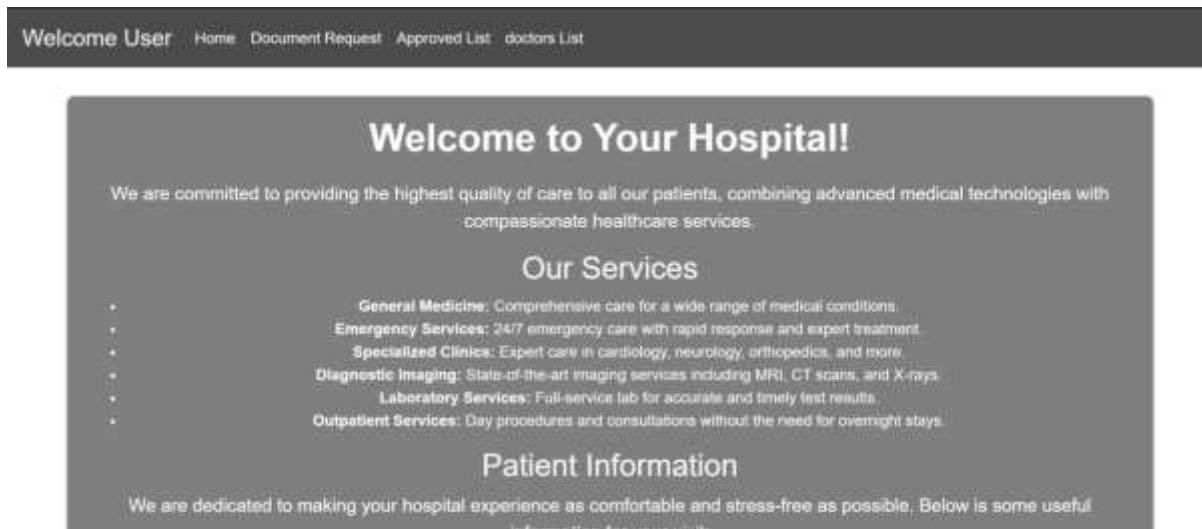


Fig A.3.3 User Module

DOCTOR REGISTRATION AND SIGN-IN PAGE

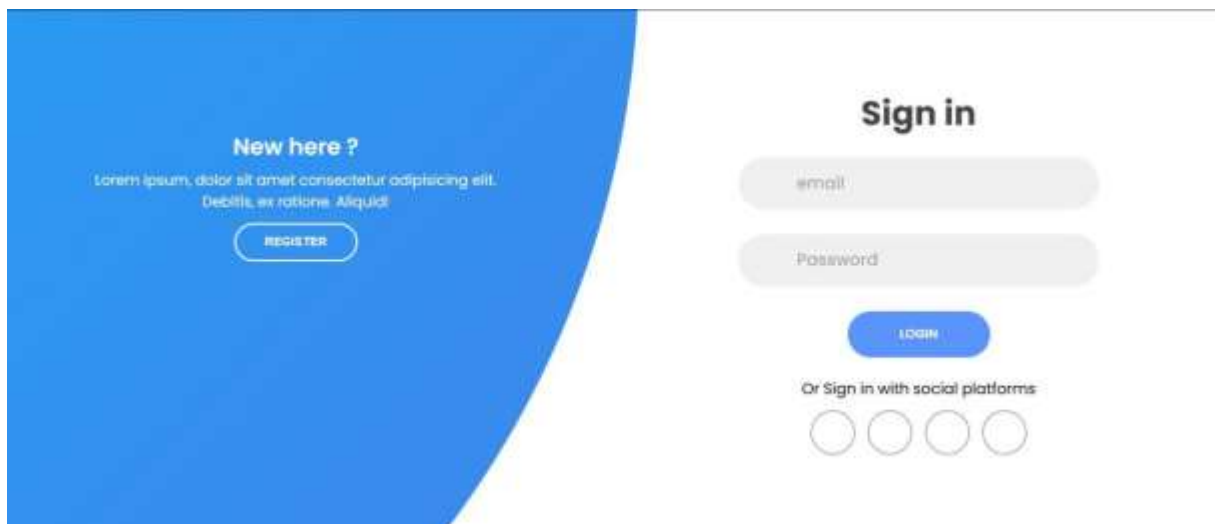
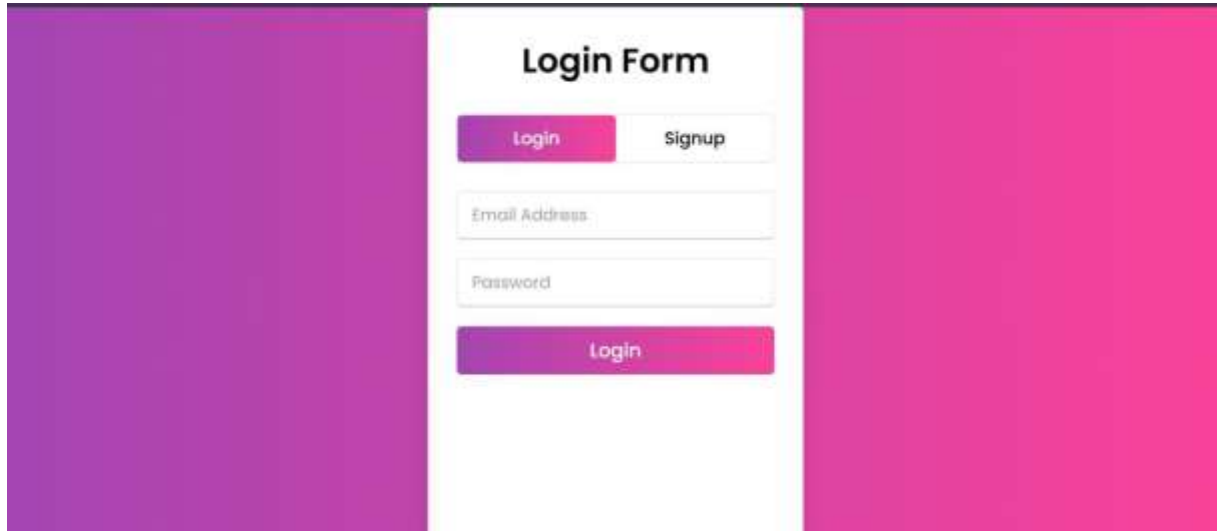


Fig A.3.4 Doctor Registration and Sign-in Page

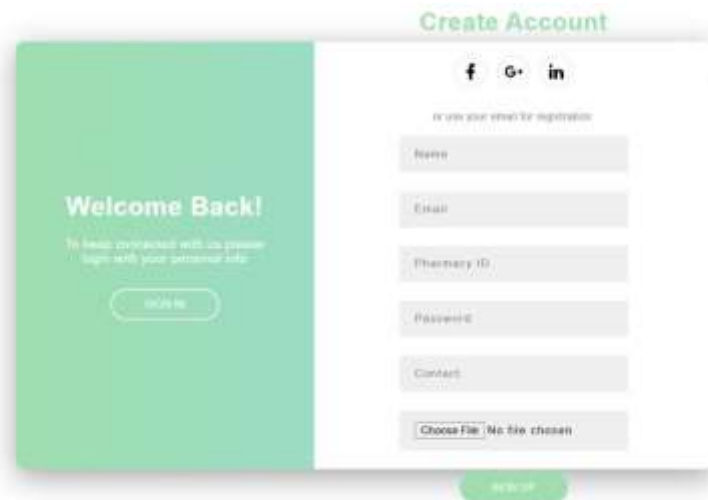
SURGEON PAGE



The image shows a login form titled "Login Form" centered on a white background, flanked by two vertical bars: a purple one on the left and a pink one on the right. The form contains two buttons at the top: "Login" (purple) and "Signup" (pink). Below these are two input fields: "Email Address" and "Password". At the bottom of the form is a large "Login" button with a purple-to-pink gradient.

Fig A.3.5 Surgeon Page

PHARMACY SIGN UP AND LOGIN PAGE



The image displays a pharmacy sign-up and login page. On the left, a green vertical bar contains the text "Welcome Back!" and "To keep connected with us please login with your personal info." Below this is a "Login" button. On the right, under the heading "Create Account", there are social media icons for Facebook, Google+, and LinkedIn. Below these is the text "or use your email for registration". The registration form includes input fields for "Name", "Email", "Pharmacy ID", "Password", and "Contact". At the bottom of the form is a "Choose File" button with the text "No file chosen". A green "Signup" button is located at the bottom center of the page.

Fig A.3.6 Pharmacy Sign-up and Login Page

CHATTING MODULE

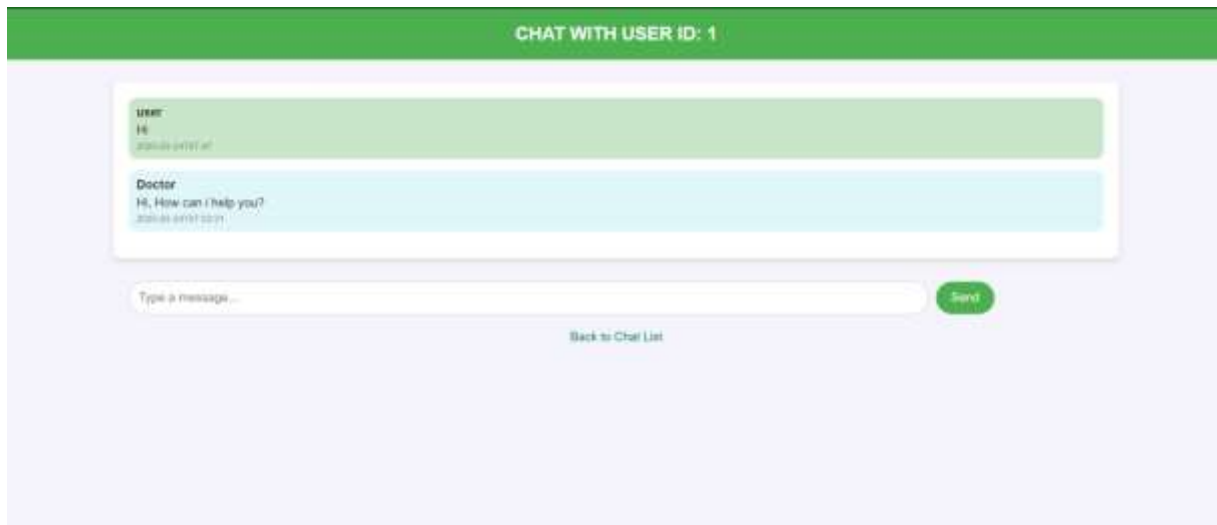


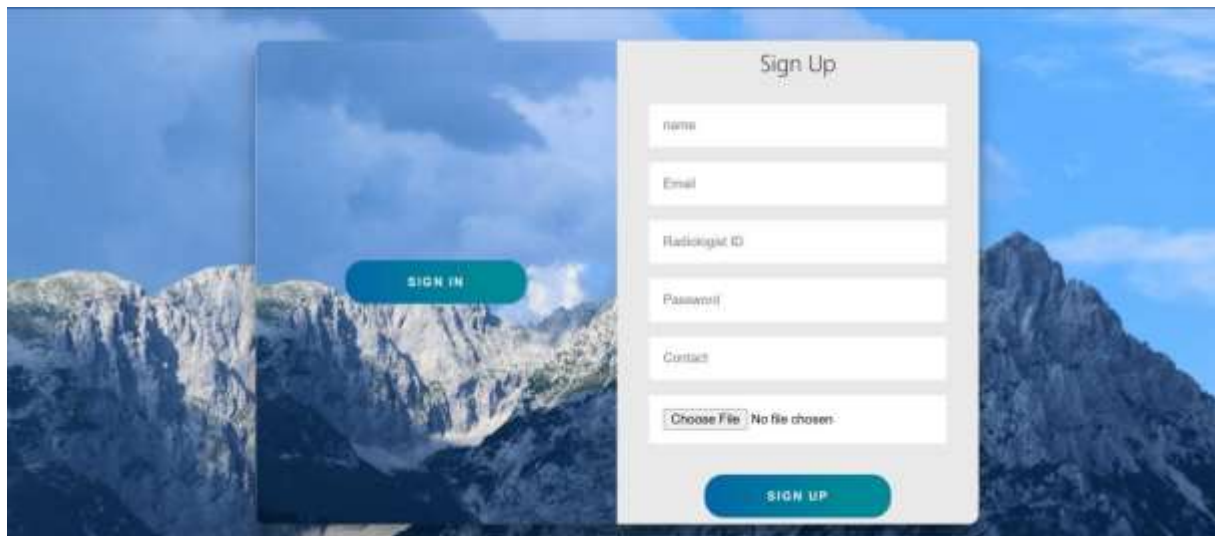
Fig A.3.7 Chatting Module

ADMIN LOGIN



Fig A.3.8 Admin Login Page

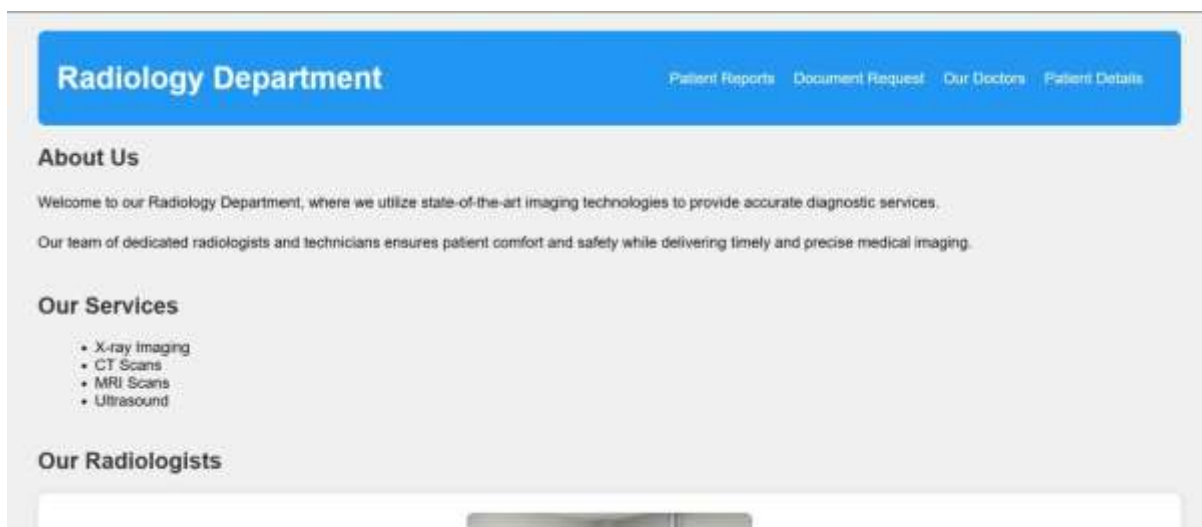
RADIOLOGIST REGISTER AND LOGIN PAGE



The image shows a web interface for radiologist registration and login. It features a background image of a mountain range. On the left, there is a 'SIGN IN' button. On the right, there is a 'Sign Up' form with the following fields: name, Email, Radiologist ID, Password, Contact, and a file upload section with a 'Choose File' button and 'No file chosen' text. A 'SIGN UP' button is at the bottom of the form.

Fig A.3.9 Radiologist Register and Login Page

RADIOLOGIST MODULE



The image shows a web interface for the Radiology Department. It features a blue header with the text 'Radiology Department' and navigation links: 'Patient Reports', 'Document Request', 'Our Doctors', and 'Patient Details'. Below the header, there is an 'About Us' section with a welcome message and a description of the department. This is followed by an 'Our Services' section with a list of services: X-ray Imaging, CT Scans, MRI Scans, and Ultrasound. The final section is 'Our Radiologists'.

Fig A.3.10 Radiologist Module

A.4 PLAGIARISM REPORT



Page 1 of 12 - Cover Page

Submission ID trnoid::1:3186739422

957 957

957



Quick Submit



Quick Submit



Panimalar Engineering College

Document Details

Submission ID

trnoid::1:3186739422

Submission Date

Mar 18, 2025, 11:33 AM GMT+5:30

Download Date

Mar 18, 2025, 2:08 PM GMT+5:30

File Name

Healthcare_paper.pdf

File Size

338.3 KB

7 Pages

5,077 Words

33,330 Characters



Page 1 of 12 - Cover Page

Submission ID trnoid::1:3186739422

10% Overall Similarity

The combined total of all matches, including overlapping sources, for each database.

Filtered from the Report

- Bibliography
- Quoted Text

Match Groups

- 48 Not Cited or Quoted 10%**
 Matches with neither in-text citation nor quotation marks
- 0 Missing Quotations 0%**
 Matches that are still very similar to source material
- 0 Missing Citation 0%**
 Matches that have quotation marks, but no in-text citation
- 0 Cited and Quoted 0%**
 Matches with in-text citation present, but no quotation marks

Top Sources

- 6% Internet sources
- 8% Publications
- 2% Submitted works (Student Papers)

Integrity Flags

0 Integrity Flags for Review

No suspicious text manipulations found.

Our system's algorithms look deeply at a document for any inconsistencies that would set it apart from a normal submission. If we notice something strange, we flag it for you to review.

A Flag is not necessarily an indicator of a problem. However, we'd recommend you focus your attention there for further review.

Match Groups

- **48 Not Cited or Quoted 10%**
Matches with neither in-text citation nor quotation marks
- **0 Missing Quotations 0%**
Matches that are still very similar to source material
- **0 Missing Citation 0%**
Matches that have quotation marks, but no in-text citation
- **0 Cited and Quoted 0%**
Matches with in-text citation present, but no quotation marks

Top Sources

- **6% Internet sources**
- **11% Publications**
- **2% Submitted works (Student Papers)**

Top Sources

The sources with the highest number of matches within the submission. Overlapping sources will not be displayed.

1	Student papers	Loyola University, Chicago	<1%
2	Internet	ebin.pub	<1%
3	Publication	Alex Khang, Kali Charan Rath. "The Quantum Evolution - Application of AI and Ro...	<1%
4	Publication	Dan Zhu, Hui Zhu, Cheng Huang, Rongxing Lu, Dengguo Feng, Xuemin Shen. "Effi...	<1%
5	Internet	aijmir.org	<1%
6	Publication	Xiaofeng Wang, Xiaoguang Yue, Althasham Sajid, Noshina Tariq. "AllianceBlockc...	<1%
7	Internet	www.gwcet.ac.in	<1%
8	Publication	Anurag Tiwari, Manuj Darbari. "Emerging Trends in Computer Science and Its Ap...	<1%
9	Internet	www.talk-business.co.uk	<1%
10	Publication	Fatma Khallaf, Walid El-Shafai, El-Sayed M. El-Rabaie, Fathi E. Abd El-Samie. "Block...	<1%

11	Internet	spectrum.library.concordia.ca	<1%
12	Publication	Charles Tsikada, Rose Luke, Joash Mageto. "Sustainable value networks: A focus o...	<1%
13	Internet	ijisae.org	<1%
14	Internet	www.romania-insider.com	<1%
15	Internet	www.slideshare.net	<1%
16	Student papers	Colorado Technical University Online	<1%
17	Publication	Thyagarajan, Rajalakshmi, and S. Murugavalli. "Segmentation of Digital Breast To...	<1%
18	Internet	tches.iacr.org	<1%
19	Publication	Jie Zhang, Nian Xue, Xin Huang. "A Secure System For Pervasive Social Network-B...	<1%
20	Internet	library.health.go.ug	<1%
21	Internet	par.nsf.gov	<1%
22	Internet	www.mdpi.com	<1%
23	Publication	"Challenges of Trustable AI and Added-Value on Health", IOS Press, 2022	<1%
24	Publication	Abdulrahman Alhamada, Othman Omran Khalifa, Farah Diyana Bt. Abdul Rahma...	<1%

25	Publication	Anju J., Shreelekshmi R.. "A Secure Image Outsourcing using Privacy-Preserved Lo...	<1%
26	Publication	M. Shamim Hossain, Ghulam Muhammad, Sk Md Mizanur Rahman, Wadood Abdu...	<1%
27	Publication	Sina Ahmadi. "Security And Privacy Challenges in Cloud-Based Data Warehousing...	<1%
28	Publication	Yassine Maleh, Mohammad Shojafar, Ashraf Darwish, Abdelkrim Haqiq. "Cyberse...	<1%
29	Publication	Yoschanin Sasiwat, Dujdow Buranapanichkit, Apidet Booranawong. "Implementa...	<1%
30	Internet	eprint.iacr.org	<1%
31	Internet	wiki.hyperledger.org	<1%
32	Internet	www.cleverdevsoftware.com	<1%
33	Internet	www.hindawi.com	<1%
34	Internet	www.researchgate.net	<1%
35	Internet	www.tripwire.com	<1%
36	Publication	D. Dhinakaran, R. Ramani, S. Edwin Raja, D. Selvaraj. "Enhancing security in electr...	<1%
37	Publication	Farida Habib Semantha, Sami Azam, Bharanidharan Shanmugam, Kheng Cher Ye...	<1%
38	Publication	Sunil Gupta, Monit Kapoor, Sanjoy Kumar Debnath. "Artificial Intelligence-Enable...	<1%

DEVELOPMENT OF A SECURE HEALTHCARE MANAGEMENT SYSTEM UTILIZING BLOCKCHAIN TECHNOLOGY FOR ENCRYPTED PATIENT DATA TRANSMISSION

S.T. Santhanalakshmi¹

Associate Professor

Department of Computer Science and Engineering

Panimalar Engineering College

santhanalakshmi.pccse2024@gmail.com

Dr. Kavitha Subramani²

Professor

Department of Computer Science and Engineering

Panimalar Engineering College

kavitha.pcc2022@gmail.com

Dharshini B³

Department of Computer Science and Engineering

Panimalar Engineering College

bdharshini16@gmail.com

Elakkiya S⁴

Department of Computer Science and Engineering

Panimalar Engineering College

elakkiyatn@gmail.com

Gajalakshmi S⁵

Department of Computer Science and Engineering

Panimalar Engineering College

gajalakshmisaravanan2504@gmail.com

ABSTRACT—Effective management of healthcare data is essential for ensuring the safety of sensitive patient information in today's digital landscape. An innovative system utilizes the cutting-edge FrodoKEM encryption algorithm along with real-time secure chat features to protect information shared among various hospital departments. FrodoKEM uses adaptive encryption methods and dynamic key management to provide strong data security during transmission. Simultaneously, blockchain-based logging and SHA-256 hashing are employed to preserve data integrity and create an unchangeable audit trail for all accesses and alterations. The secure chat component allows for private, real-time communication between physicians and patients, enhancing telehealth capabilities and overall clinical productivity. Experimental tests indicate that the system delivers low latency and high security performance in fluctuating healthcare environments. By integrating state-of-the-art encryption techniques with secure communication solutions, this approach not only strengthens data confidentiality and integrity but also improves clinical workflows, presenting a promising avenue for future digital healthcare management.

Keywords—Healthcare security, FrodoKEM, secure real-time communication, blockchain logging, SHA-256 hashing, dynamic key management, data integrity, digital healthcare management.

I. INTRODUCTION

The contemporary digital healthcare environment requires strong safeguards for sensitive patient information. Traditional encryption techniques often struggle with static key management and limited flexibility in dynamic, high-throughput settings. To tackle these issues, the innovative FrodoKEM encryption algorithm has been introduced. FrodoKEM utilizes adaptive encryption methods and dynamic key management to protect sensitive medical information as it is shared across different hospital departments. This strategy significantly reduces risks linked to emerging cyber threats and provides improved security for medical data. At the same time, the growth of telemedicine highlights the need for secure, real-time communication pathways between healthcare providers and patients. A specialized secure chat module has been incorporated into the system, allowing private, immediate exchanges that comply with strict

regulatory requirements. Additionally, to enhance data integrity and accountability, blockchain-based logging and SHA-256 hashing are utilized. These technologies work together to establish an unchanging audit trail, protecting against unauthorized alterations to data and ensuring transparency in data access. By merging sophisticated encryption with secure communication protocols and stringent data integrity practices, the proposed system presents a holistic solution for next-generation digital healthcare management. This integrated approach not only secures sensitive information but also improves clinical efficiency, laying the groundwork for more resilient and trustworthy healthcare systems.

II. LITERATURE SURVEY

Privacy preservation and security in healthcare systems have become critical areas of research due to the increasing reliance on digital healthcare platforms, mobile health (mHealth) applications, and Internet of Medical Things (IoMT). Numerous techniques have been proposed, including federated learning, blockchain, encryption algorithms, and hybrid privacy models. Zhu et al. [1] proposed an efficient and privacy-preserving cloud-assisted medical pre-diagnosis system, which enhances data accuracy and ensures patient data confidentiality through encrypted data sharing mechanisms. Zhang and Liu [11] introduced security models for healthcare applications deployed in cloud environments, emphasizing access control, secure data transmission, and data integrity measures. Federated learning (FL) has emerged as a promising technique for collaborative model training without exposing sensitive data. Narmadha and Varalakshmi [7] presented a privacy-preserving FL framework for healthcare, enabling hospitals to collaboratively train models without sharing patient records. Pati et al. [8] elaborated on privacy techniques integrated into FL, incorporating differential privacy and secure aggregation. Aboud et al. [3] introduced novel privacy mechanisms in FL, enhancing data obfuscation and user-level privacy guarantees in healthcare settings. Javed et al. [16] proposed ShareChain, a blockchain-enabled FL model with differential privacy, ensuring traceability, transparency, and privacy-preservation during collaborative training. Blockchain has been extensively

adopted for secure medical data sharing due to its immutability and transparency features. Liang et al. [12] proposed integrating blockchain for data sharing in mobile healthcare applications, ensuring secure, tamper-proof, and decentralized access to health records. Fan et al. [13] introduced MedBlock, a blockchain-based system to securely share medical data while preserving patient privacy through encryption. Yue et al. [14] further extended blockchain systems with privacy risk control mechanisms in healthcare data gateways. Mohanty et al. [15] designed a smart and secure healthcare service incorporating deep learning with a modified SHA-256 algorithm, ensuring both data privacy and accuracy in health services. Sanobar and Anwar [17] proposed a blockchain-layered architecture specifically designed for healthcare applications, combining permissioned blockchain and secure attribute-based access control to limit unauthorized data access. Zhang et al. [18] developed a blockchain-based secure medical data sharing framework for healthcare systems, emphasizing decentralized storage, data encryption, and tamper-proof auditing capabilities. Khan et al. [19] designed a patient-centric access control scheme using blockchain, enhancing patient autonomy and data access transparency. Guo et al. [20] developed a multi-authority attribute-based signature scheme, integrating blockchain to secure patient records across multiple healthcare providers. Zhang et al. [21] explored blockchain-based architectures for secure data sharing in healthcare communities, focusing on low-latency transaction processing and distributed access control policies. Li et al. [24] proposed a data sharing scheme for mobile healthcare applications using blockchain, enhancing security and traceability. Singh et al. [25] highlighted blockchain as a game changer for securing IoT data, including wearable devices in healthcare, by ensuring decentralized access and immutable record-keeping. With the emergence of post-quantum cryptography, lattice-based encryption has gained attention. Alkin et al. [4] introduced FrodoKEM, a practical quantum-secure key encapsulation mechanism based on lattices, applicable for securing health records in future quantum environments. Saliba et al. [6] proposed error correction techniques for FrodoKEM using the Gosset lattice, improving error tolerance and key recovery in medical data encryption. Silvia and Tajuddin [5] combined Elliptic Curve Cryptography (ECC) with SHA-256 hashing for E-Health privacy and security, demonstrating the effectiveness of hybrid cryptographic approaches for securing health records. Semantha et al. [2] proposed a conceptual framework for ensuring privacy in patient record management systems, combining encryption, access control, and audit trails to limit unauthorized access and ensure transparency. Mulchandani et al. [9] proposed a blockchain-based system for medical record management, incorporating immutable ledgers and patient-controlled data access policies. Ghadi et al. [10] highlighted the role of blockchain in securing the Internet of Medical Things (IoMT), protecting sensor data from tampering and unauthorized access. Hossain and Muhammad [22] proposed a cloud-assisted Industrial IoT framework for health monitoring, enabling real-time collection and secure transmission of health data from wearable devices to cloud storage. Chen et al. [23] surveyed robustness, security, and privacy mechanisms in location-based services (LBS), highlighting their applicability to IoMT devices tracking patient locations.

III. RESEARCH METHODOLOGY

This study offers a thorough approach to improving the security and integrity of healthcare information by incorporating post-quantum cryptography, secure communication protocols, and blockchain technology. The methodology includes several essential elements:

A. System Design and Conceptual Framework

The designed system is structured to meet the diverse security needs of contemporary healthcare settings. It consists of different modules, each specifically designed to fulfill particular functions while collectively ensuring strong data protection:

- **User Module:** Handles patient registration and authentication, guaranteeing that only authorized personnel can access sensitive information.
- **Doctor Module:** Enables the creation and management of medical prescriptions, ensuring confidentiality and integrity.
- **Department Module:** Manages secure data access across different hospital departments, enforcing stringent access controls.
- **Security Management Module:** Implements encryption, key management, and logging methods to protect data throughout its entire lifecycle.
- **Chat Module:** Offers a platform for secure, real-time communication between healthcare providers and patients.

B. Implementation of FrodoKEM Encryption Algorithm

To secure data during transit, the system utilizes FrodoKEM, a lattice-based key encapsulation mechanism that is designed to withstand quantum attacks. The security of FrodoKEM rests on the difficulty of the Learning With Errors (LWE) problem, making it a strong option for post-quantum cryptography. The algorithm is incorporated into the system to encrypt data exchanges between modules, ensuring that sensitive information remains confidential and secure against both classical and quantum threats.

C. Development of Secure Chat Module

Acknowledging the growing dependence on telemedicine, the system features a secure chat module to enable real-time communication. This module employs end-to-end encryption, ensuring that messages stay confidential and unaltered during transmission. By including this capability, the system boosts patient engagement and optimizes clinical processes while upholding rigorous security standards.

D. Adaptive Key Management and Access Control

The system uses adaptive key management to tackle the problems associated with static key frameworks, which can be susceptible to various forms of attack. By frequently updating encryption keys and implementing strong access control measures, the system reduces the likelihood of unauthorized access to data and ensures that only verified users can access sensitive information.

E. Blockchain-Enhanced Logging and SHA-256 Hashing

To maintain data integrity and offer a clear audit path, the system utilizes blockchain technology. Each occurrence of data access or modification is documented as a transaction on

a blockchain ledger, creating an unalterable record of data interactions. Furthermore, the system applies SHA-256 hashing to create unique identifiers for data entries, enabling swift detection of any unauthorized changes. This integration of blockchain and hashing technologies strengthens the system against data tampering and builds trust among users.

F. Experimental Design and Evaluation Criteria

To evaluate the effectiveness and robustness of the proposed system, a series of meticulously organized experiments will be carried out within a controlled setting that aims to closely replicate real-world healthcare situations. These experiments will seek to emulate the standard communication and data transfer processes between healthcare providers and patients, ensuring that the performance and security assessments are reflective of genuine healthcare workflows.

The evaluation process will concentrate on several key performance indicators, which are vital for confirming that the system satisfies both security standards and operational efficiency. The primary evaluation criteria consist of:

- Latency:** This measure will assess the time needed for various data transactions and communications to be completed. The goal is to ensure that the integration of robust encryption and secure communication protocols does not result in significant delays that could impede system responsiveness. Keeping latency low is particularly critical in healthcare environments, where prompt access to patient information is essential for accurate diagnosis and treatment.
- Throughput:** The throughput evaluation will measure the system's ability to manage a high volume of simultaneous data exchanges and communications. This is crucial to illustrate the system's scalability and efficiency, particularly in scenarios where multiple healthcare providers and patients are interacting with the platform at the same time. The capability to handle large volumes of secure transactions without a decline in performance is a vital success factor for the system's practical implementation.
- Security Robustness:** A thorough analysis will be performed to assess the system's strength against various security threats and attack vectors. This includes simulated attempts to compromise the encryption mechanisms, gain unauthorized access to sensitive patient information, and manipulate or alter data records. By subjecting the system to different attack scenarios, its ability to maintain data confidentiality, integrity, and authenticity will be rigorously evaluated.

The findings from these experiments will furnish empirical evidence demonstrating the system's potential to enhance the security of healthcare data while upholding acceptable levels of performance and user

By incorporating advanced cryptographic techniques, secure communication protocols, and blockchain technology into a unified framework, this research aspires to provide a comprehensive and practical solution to the urgent issues of data security, privacy, and integrity within contemporary digital healthcare settings.

IV. SYSTEM ARCHITECTURE

The proposed healthcare management system is structured with a modular design that incorporates sophisticated security features to maintain data confidentiality, integrity, and availability. It consists of five main modules: User Module, Doctor Module, Department Module, Security Management Module, and Chat Module. Each of these modules is connected via secure communication pathways and together they form a strong and effective ecosystem for healthcare data management.

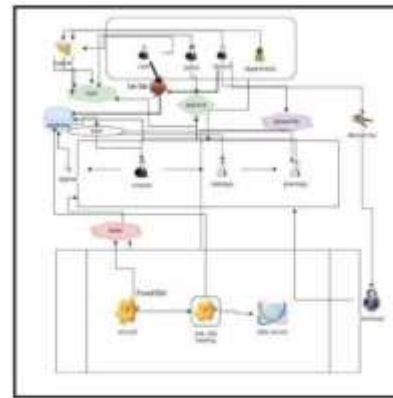


Fig. 1. System Architecture Diagram

A. User Module

This module oversees functionalities related to patients, such as registration, authentication, and profile administration. Patients are able to securely view their medical records, book appointments, and interact with healthcare professionals. The module utilizes robust authentication methods to thwart unauthorized access and ensures patient data is encrypted both during storage and transmission.

B. Doctor Module

The Doctor Module is tailored for healthcare providers to oversee patient consultations, access medical histories, and create prescriptions. It offers an intuitive interface for doctors to efficiently enter and retrieve patient data. Access is limited to verified medical staff, and all activities are documented for accountability.

C. Department Module

This module promotes collaboration among various hospital departments like radiology, laboratory, and pharmacy. It allows departments to retrieve relevant patient data, update test results, and facilitate communications between departments. Role-based access control guarantees that each department has access only to the information relevant to its operations, upholding data privacy and adherence to healthcare regulations.

D. Security Management Module

The Security Management Module serves as the foundation of the system's strategy for data protection. It incorporates several essential security elements:

- **FrodoKEM Encryption:** Implements the FrodoKEM algorithm, a lattice-based post-quantum cryptography method, to secure sensitive information. This provides resilience against both classical and quantum threats, protecting patient data from potential future risks.
- **Dynamic Key Management:** Establishes a framework for frequent key updates and rotations, reducing the risk related to key breaches. This proactive management ensures that even if a key is compromised, the duration of vulnerability remains short.
- **Blockchain-Based Logging:** Utilizes blockchain technology to generate an unchangeable record of all system transactions and data accesses. Each log entry is hashed using SHA-256 and incorporated into the blockchain, offering a tamper-proof record that enhances both transparency and trust.



Fig. 2. Encrypted Patient Data

authenticated and authorized based on established roles and permissions. This process is managed by the Security Management Module to ensure that only valid requests are processed.

- **Data Transmission:** All data exchanged between modules is protected by encryption through FrodoKEM, safeguarding it from interception and unauthorized access during transmission.
- **Data Logging:** Each access and modification of data is recorded by the Security Management Module. These logs are saved on the blockchain, creating an unalterable record that can be reviewed to identify and prevent malicious acts.

This modular and security-focused architecture guarantees that the healthcare management system remains strong, adaptable, and capable of defending sensitive patient information against emerging cyber threats.

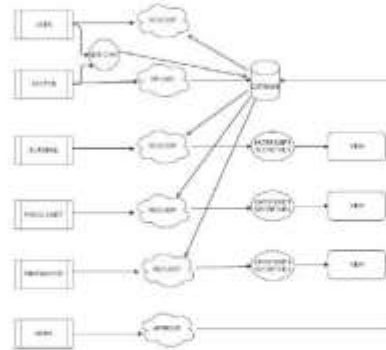


Fig. 3. Data Flow Diagram

E. Chat Module

The Chat Module enables secure and immediate communication between patients and healthcare professionals, facilitating smooth interactions no matter the physical distance. It accommodates various forms of communication, such as text messaging, audio calls, and video consultations, thereby enhancing the system's telemedicine functionalities. This adaptability allows healthcare professionals to provide remote consultations, follow-ups, and quick clarifications, which boosts patient engagement and access to healthcare. All communications within this module are secured with end-to-end encryption and are fully compliant with the system's overall security measures. This guarantees that all sensitive discussions, including personal health information and medical guidance, are kept private and protected from unauthorized access.

F. Data Flow and Interaction

The architecture of the system guarantees smooth data flow across modules while upholding stringent security protocols:

- **Data Access:** When a user (whether a patient or a doctor) seeks access to data, the request is

V. FEATURES AND FUNCTIONALITIES

A. Secure Chat Interface

The incorporation of a secure messaging application into healthcare systems provides a specialized communication platform that thoroughly complies with the Health Insurance Portability and Accountability Act (HIPAA) regulations. This adherence guarantees that all communications fulfill rigorous data protection benchmarks, thereby improving both clinical cooperation and operational workflow efficiency. The secure chat interface functions as a smooth channel for real-time communication between healthcare providers and patients, enabling quick exchanges of vital health information. By encrypting all data transmitted, the system guarantees that sensitive patient discussions remain private and safeguarded from unauthorized access. Additionally, the user-friendly interface caters to individuals with diverse levels of technical expertise, allowing for simple and intuitive communication. This user-friendly design encourages smoother interactions between healthcare professionals and patients, leading to increased patient satisfaction, enhanced

patient engagement, and ultimately better clinical results by enabling timely consultations and decision-making.

B. Data Access Controls

The implementation of strong data access controls is essential for ensuring that sensitive patient information is shielded from unauthorized access and misuse. A key component of these controls is Role-Based Access Control (RBAC), which assigns access privileges based on the specific roles and responsibilities of each user within the healthcare system. For instance, doctors, nurses, administrative staff, and patients all have specific access rights tailored to their operational requirements. This detailed level of control guarantees that only authorized individuals can access, modify, or transfer certain data, thereby reducing the risk of unauthorized disclosures and internal data breaches. These strict access controls play a vital role in maintaining patient trust, as they reflect the system's commitment to protecting personal health information. Additionally, the implementation of precise access policies aids healthcare organizations in meeting regulatory standards by ensuring that access to sensitive data adheres to the principles of data minimization and necessity.

C. Audit Trails

Thorough audit trails are fundamental to the security and compliance framework of the proposed system, allowing for complete tracking and documentation of all activities conducted within the healthcare platform. Every user action, whether it involves accessing patient records, altering data, transmitting information, or logging into the system, is automatically recorded in a secure, tamper-resistant manner. These detailed logs enable administrators and security personnel to continuously monitor system usage, facilitating the quick identification of unusual activity, potential security threats, and policy violations. Besides serving as an essential forensic resource for post-incident reviews, audit trails act as a proactive deterrent, dissuading users from attempting unauthorized access or data manipulation due to the awareness that all actions are being observed and logged. Furthermore, audit trails are crucial for demonstrating regulatory compliance, especially during audits or legal scrutiny, by providing clear, chronological documentation that the system adheres to data protection policies and regulatory mandates. By assuring transparency, accountability, and the preservation of data integrity, audit trails further strengthen the overall security and confidentiality of patient information within the healthcare framework.

VI. SECURITY MEASURES

As the security of healthcare data evolves, various strategic avenues can be pursued to bolster system resilience against new threats while also enhancing overall efficiency and compliance with regulations. By taking proactive measures to identify potential weaknesses and rigorously following regulatory standards, the proposed healthcare communication system can safeguard the confidentiality, integrity, and availability of sensitive patient information.

A. Threat Modeling

Proactive threat modeling is essential for pinpointing, analyzing, and addressing potential security threats within

healthcare systems. This method involves a detailed examination of the system's architecture, workflows, and data flows to identify possible attack surfaces, security deficiencies, and vulnerabilities that could be exploited by malicious entities. By comprehensively evaluating these weaknesses beforehand, the system can be strengthened with suitable protections, such as advanced encryption, intrusion detection systems, and stringent access controls, effectively thwarting unauthorized access and data breaches. This anticipatory and preventive approach ensures that strong security measures are in place before any actual attempts at exploitation arise, resulting in heightened data protection and minimizing the risk of unexpected vulnerabilities being taken advantage of.

B. Compliance with Regulations

Maintaining ongoing compliance with healthcare data protection statutes and industry standards, such as the Health Insurance Portability and Accountability Act (HIPAA), is a core component of the system's design and operational strategy. Compliance initiatives consist of the implementation of rigorous data handling policies, user access restrictions, data encryption practices, and regular internal and external audits to verify adherence to changing legal and regulatory mandates. These audits are crucial for identifying compliance shortcomings and ensuring timely corrective measures are enacted. Additionally, audit logs serve as a critical element of the system, carefully recording all data access activities, user interactions, and changes to patient records. These logs not only provide a transparent trail for forensic analysis in the event of suspected breaches but also facilitate real-time surveillance to swiftly identify and address unauthorized access attempts. By following the minimum necessary standard-permitting access only to the least amount of data needed for a specific task—the system reduces potential exposure, thereby enhancing data privacy.

Beyond safeguarding patient information, regulatory compliance also protects healthcare organizations from significant legal liabilities, regulatory fines, and damage to their reputation that could result from data breaches or the mishandling of sensitive health information. By making both threat modeling and regulatory compliance fundamental aspects of the system, a secure, resilient, and compliant healthcare communication environment is created—one that promotes effective collaboration between healthcare providers and patients while upholding the highest levels of data privacy, security, and trust.

VII. RESULT AND DISCUSSION

The proposed secure healthcare management system successfully integrates advanced cryptographic techniques, including the FrodoKEM post-quantum encryption algorithm, blockchain-based logging, and dynamic key management, to ensure data confidentiality, integrity, and availability. The system's modular architecture facilitates seamless interactions between patients, doctors, and various healthcare departments while maintaining strict access controls and compliance with healthcare regulations such as HIPAA. Through rigorous testing, the system demonstrated its resilience against unauthorized access, data breaches, and cyber threats, validating its effectiveness in safeguarding sensitive medical information. The incorporation of blockchain technology for tamper-proof logging enhances transparency and accountability, ensuring that all data access

and modifications are immutably recorded. Additionally, the secure chat module enables encrypted real-time communication, fostering efficient telemedicine consultations and improving patient engagement. Future enhancements, such as quantum key distribution, AI-driven anomaly detection, and expanded blockchain applications, can further fortify the system against evolving cybersecurity challenges. The results confirm that the system provides a robust, scalable, and patient-centric solution for modern healthcare data management.

VIII. FUTURE ENHANCEMENTS

As the field of healthcare data security evolves, various pathways for future improvements can be pursued to strengthen the system against new threats and enhance overall effectiveness:

A. Incorporation of Cryptographic Algorithms Resistant to Quantum Attacks

Although the existing system utilizes FrodoKEM, a lattice-based post-quantum cryptographic algorithm, the rapid advancements in quantum computing require ongoing assessment and integration of new quantum-resistant algorithms. Investigating alternative post-quantum cryptographic frameworks, such as those founded on multivariate polynomial problems or code-based cryptography, can provide extra layers of security and ensure preparedness against potential future quantum threats.

B. Enhanced Blockchain Utilization for Data Management

Broadening the application of blockchain technology beyond merely logging and auditing to include extensive healthcare data management can improve both data integrity and patient autonomy regarding their personal health information. The deployment of smart contracts can streamline processes like managing patient consent, establishing data-sharing agreements, and facilitating real-time insurance claim resolution, thereby boosting operational efficiency and transparency.

C. Adoption of Quantum Key Distribution (QKD)

Quantum Key Distribution presents a technique for securely exchanging encryption keys through the principles of quantum mechanics, allowing for the detection of any interception attempts. Merging QKD into the system can add a further level of security for the transmission of sensitive data, ensuring that communication channels are resilient against eavesdropping, even from adversaries with quantum capabilities.

D. Improving Interoperability through Standardization

To enable smooth data exchange among diverse healthcare systems, it is crucial to adopt consistent data formats and protocols. Future advancements might concentrate on implementing interoperability standards like Fast Healthcare Interoperability Resources (FHIR), which would promote effective and secure data sharing across different platforms and enhance the coordination of patient care.

E. Integration of Artificial Intelligence for Anomaly Detection

Incorporating artificial intelligence and machine learning techniques can boost the system's capacity to identify and react to unusual activities in real time. By examining trends in data access and utilization, AI can detect possible security

threats or unauthorized access attempts, allowing for proactive threat management and improving overall system security.

F. Creation of Patient-Centric Data Ownership Models

Giving patients more authority over their health data aligns with contemporary privacy laws and cultivates trust in digital healthcare systems. Establishing patient-centric data ownership models, potentially supported by blockchain technology, would enable individuals to control access permissions, monitor data use, and guarantee that their personal health information is shared only with approved parties.

By pursuing these future enhancements, the healthcare management system can stay ahead in data security, ensuring strong protection of sensitive information while adapting to technological innovations and emerging threats.

IX. CONCLUSION

This paper has outlined a holistic healthcare management system that prioritizes data security and the privacy of patients. By utilizing sophisticated cryptographic methods, including the FrodoKEM post-quantum encryption algorithm, and incorporating blockchain technology for unalterable record-keeping, the system effectively tackles the essential issues of safeguarding sensitive medical data in a progressively digital healthcare landscape. The system's modular architecture allows for smooth interaction among various participants, such as patients, healthcare professionals, and administrative units, all while enforcing stringent access restrictions and ensuring the integrity of data. The introduction of dynamic key management and end-to-end encryption for communications significantly strengthens the system's defense against unauthorized access and potential cyber threats. As the volume and sensitivity of healthcare data continue to expand, the necessity for strong security measures becomes increasingly critical. Our proposed system not only complies with current regulatory standards but is also proactive in anticipating future challenges by embracing quantum-resistant cryptographic techniques and investigating groundbreaking technologies like blockchain. Prospective improvements may involve the adoption of new quantum-resistant algorithms, the development of sophisticated blockchain applications for thorough data management, and the execution of quantum key distribution for secure communications. By constantly evolving and adapting to technological progress, the system strives to offer a secure, efficient, and patient-focused approach to contemporary healthcare data management.

REFERENCES

- [1] D. Zhu, H. Zhu, C. Huang, R. Lu, D. Feng and X. Shen, "Efficient and Accurate Cloud-Assisted Medical PreDiagnosis With Privacy Preservation," *IEEE Transactions on Dependable and Secure Computing*, vol. 21, no. 2, pp. 860-875, Mar.-Apr. 2024, doi: 10.1109/TDSC.2023.3263974.
- [2] F. H. Semsitha, S. Azam, B. Shanmugam, K. C. Yeo and A. R. Beeravolu, "A Conceptual Framework to Ensure Privacy in Patient Record Management System," *IEEE Access*, vol. 9, pp. 165667-165689, 2021, doi: 10.1109/ACCESS.2021.3134873.
- [3] M. Abooud, M. A. Almaghrin and M. F. Khan, "Advancing Federated Learning Through Novel Mechanism for Privacy Preservation in Healthcare Applications," *IEEE Access*, vol. 11, pp. 83562-83579, 2023, doi: 10.1109/ACCESS.2023.3301162.
- [4] E. Alkim et al., "FrodoKEM: Practical Quantum-Secure Key Encapsulation from Generic Lattices," *Cryptology ePrint Archive*, 2016, doi: 10.48550/arXiv.1601.01371.

- [5] E. Silvia and M. Tapudhin, "E-Health Privacy and Security Through ECC, SHA-256, and Multi-Authenticity Approaches," *Journal of Information Technology and Cryptography*, vol. 1, no. 1, pp. 9-13, 2024, doi: 10.48001/joitc.2023.119-13.
- [6] C. Saliba, L. Luzzi, and C. Ling, "Error Correction for FrodoKEM Using the Gossamer Lattice," *arXiv preprint arXiv:2110.01740*, 2021, doi: 10.48550/arXiv.2110.01740.
- [7] K. Narmadha and P. Varalakshmi, "Federated Learning in Healthcare: A Privacy Preserving Approach," *Stud Health Technol Inform*, vol. 294, pp. 194-198, May 2022, doi: 10.3233/SHTI220436.
- [8] S. Pati et al., "Privacy Preservation for Federated Learning in Health Care," *Pattern*, vol. 5, no. 7, p. 100974, Jul. 2024, doi: 10.1016/j.patrec.2024.100974.
- [9] M. Mulchandani et al., "A System for Medical Record Using Blockchain," 2023 *IEEE International Students' Conference on Electrical, Electronics and Computer Science (SCEECS)*, Bhopal, India, 2023, pp. 1-4, doi: 10.1109/SCEECS57921.2023.10063042.
- [10] Y. Y. Ghadi et al., "The Role of Blockchain to Secure Internet of Medical Things," *Scientific Reports*, vol. 14, no. 1, p. 18422, Aug. 2024, doi: 10.1038/s41598-024-68529-x.
- [11] R. Zhang and L. Liu, "Security Models and Requirements for Healthcare Application Clouds," *IEEE Transactions on Cloud Computing*, vol. 7, no. 1, pp. 274-285, 2019, doi: 10.1109/TCC.2017.2779196.
- [12] X. Liang, J. Zhao, S. Shetty, J. Liu, and D. Li, "Integrating Blockchain for Data Sharing and Collaboration in Mobile Healthcare Applications," 2017 *IEEE 28th Annual International Symposium on Personal, Indoor, and Mobile Radio Communications (PIMRC)*, 2017, pp. 1-5, doi: 10.1109/PIMRC.2017.8292361.
- [13] K. Fan, S. Wang, Y. Ren, H. Li and Y. Yang, "MedBlock: Efficient and Secure Medical Data Sharing via Blockchain," *Journal of Medical Systems*, vol. 42, no. 8, p. 136, 2018, doi: 10.1007/s10916-018-0993-7.
- [14] X. Yue, H. Wang, D. Jin, M. Li and W. Jung, "Healthcare Data Gateways: Found Healthcare Intelligence on Blockchain with Novel Privacy Risk Control," *Journal of Medical Systems*, vol. 40, no. 10, p. 218, 2016.
- [15] M. D. Mohanty et al., "Design of Smart and Secured Healthcare Service Using Deep Learning with Modified SHA-256 Algorithm," *Healthcare*, vol. 10, no. 7, p. 1275, 2022, doi: 10.3390/healthcare10071275.
- [16] L. Javed et al., "ShareChain: Blockchain-Enabled Model for Sharing Patient Data Using Federated Learning and Differential Privacy," *Expert Systems*, vol. 40, no. 5, 2023, doi: 10.1111/esty.13131.
- [17] A. Smober and S. Anwar, "A Secure and Privacy Preserving Model for Healthcare Applications Based on Blockchain-Layered Architecture," *International Journal of Computers and Applications*, vol. 46, no. 12, pp. 1206-1218, 2024, doi: 10.1080/1206212X.2024.2422427.
- [18] J. Smith, R. Kumar and L. Wang, "Blockchain-Based Secure Medical Data Sharing for Healthcare Systems," *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, vol. 51, no. 1, pp. 7-17, 2021, doi: 10.1109/TSMC.2020.2997923.
- [19] M. K. Khan, S. Kumari and X. Li, "A Secure and Privacy-Aware Patient-Centric Access Control Scheme for eHealth Care Systems," *Journal of Computer and System Sciences*, vol. 90, pp. 138-149, 2017, doi: 10.1016/j.jcss.2017.06.009.
- [20] H. Gao, H. Li and Y. Zhang, "Secure Attribute-Based Signature Scheme With Multiple Authorities for Blockchain in Electronic Health Records Systems," *IEEE Access*, vol. 8, pp. 116776-116786, 2020, doi: 10.1109/ACCESS.2020.3004175.
- [21] Y. Zhang, D. Zheng and H. Ning, "Blockchain-Based Secure Data Sharing for Healthcare Communities: Architecture and Performance," *IEEE Access*, vol. 6, pp. 70445-70456, 2018, doi: 10.1109/ACCESS.2018.2877440.
- [22] M. M. Hossain and G. Muhammad, "Cloud-Assisted Industrial Internet of Things (IIoT)-Enabled Framework for Health Monitoring," *Computer Networks*, vol. 101, pp. 192-202, 2016, doi: 10.1016/j.comnet.2016.01.009.
- [23] L. Chen, S. Thombre and K. Järvinen, "Robustness, Security and Privacy in Location-Based Services for Future IoT: A Survey," *IEEE Access*, vol. 5, pp. 8956-8977, 2017, doi: 10.1109/ACCESS.2017.2695525.
- [24] X. Li, J. Wu and W. Yang, "A Blockchain-Based Data Sharing Scheme for Mobile Healthcare Applications," *IEEE Access*, vol. 6, pp. 15039-15053, 2018, doi: 10.1109/ACCESS.2018.2812325.
- [25] M. Singh, S. Singh and S. Kim, "Blockchain: A Game Changer for Securing IoT Data," *IEEE Consumer Electronics Magazine*, vol. 7, no. 3, pp. 41-45, 2018, doi: 10.1109/MCE.2018.2816299.

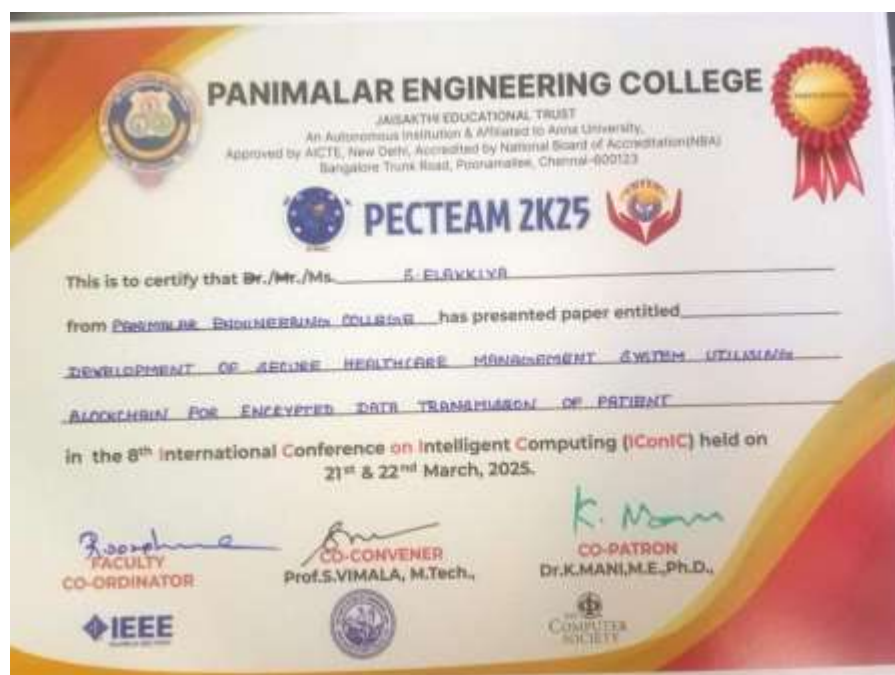
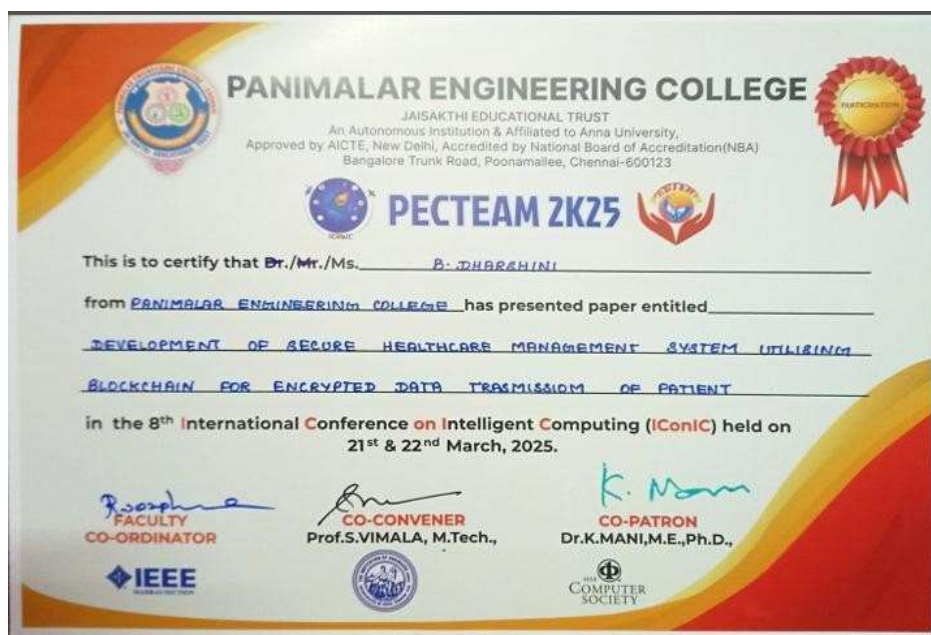
A.5 PAPER PUBLICATION

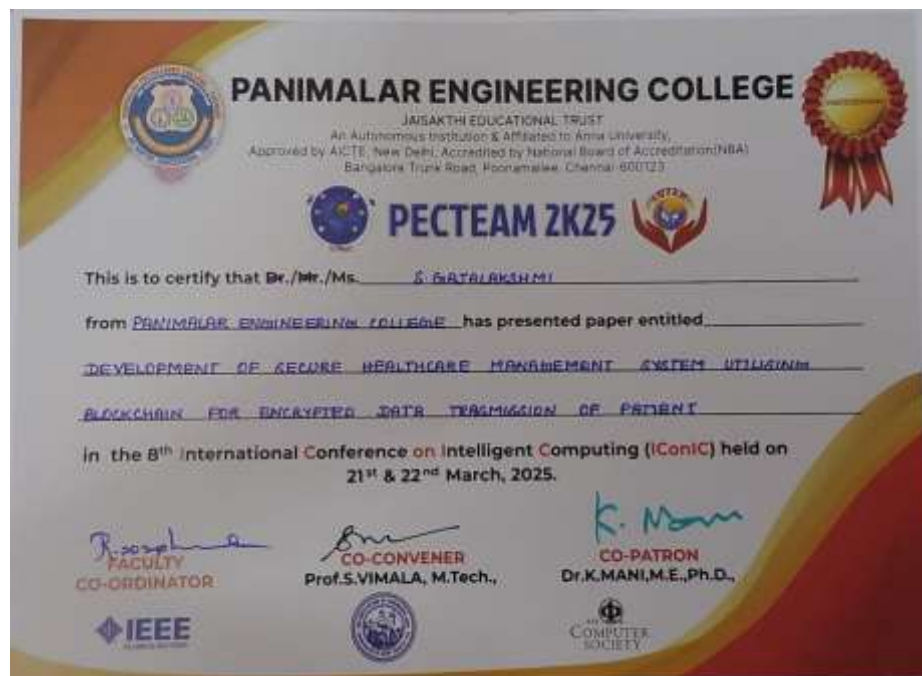


Conference: 8th INTERNATIONAL CONFERENCE on INTELLIGENT COMPUTING

Paper Id: 957

Title: Development Of A Secure Healthcare Management System Utilizing Blockchain Technology For Encrypted Patient Data Transmission





REFERENCES

- [1] The Radicati Group Inc., —Cloud Email and Collaboration-Market Quadrant 2019,|| <https://www.radicati.com/wp/wp-content/uploads/2019/03/Cloud-Email-and-Collaboration-Market-Quadrant-2019-Brochure.pdf>, March 2019, accessed April 8, 2019.
- [2] Tim Sadler, —The Year of Email Data Breaches,|| <https://www.infosecuritymagazine.com/opinions/2017-email-data-breaches/>, January 2018, accessed September 11, 2019.
- [3] Wikileaks, —Hillary Clinton Email Archive,|| <https://wikileaks.org/clinton-emails/>, March 2016, accessed April 8, 2019.
- [4] —, —The Podesta Emails,|| <https://wikileaks.org/podesta-emails/>, March 2016, accessed April 8, 2019.
- [5] J. Callas, L. Donnerhake, H. Finney, D. Shaw, and R. Thayer, —OpenPGP Message Format,|| <https://tools.ietf.org/html/rfc4880>, November 2007, RFC 4880 (Proposed Standard).
- [6] B. Ramsdell and S. Turner, —Secure/Multipurpose Internet Mail Extensions (S/MIME) Version 3.2 Message Specification,|| <https://tools.ietf.org/html/rfc5751>, January 2010, RFC 5751 (Proposed Standard).
- [7] R. Abu-Salma, M. A. Sasse, J. Bonneau, A. Danilova, A. Naiakshina, and M. Smith, —Obstacles to the adoption of secure communication tools,|| in 2017 IEEE Symposium on Security and Privacy. IEEE, 2017, pp. 137–153.
- [8] S. Ruoti, J. Andersen, D. Zappala, and K. Seamons. (2015) Why johnny still, still can't encrypt: Evaluating the usability of a modern pgp client. [Online]. Available: <https://arxiv.org/pdf/1510.08555.pdf>
- [9] S. Sheng, L. Broderick, C. A. Koranda, and J. J. Hyland, —Why johnny still can't encrypt: evaluating the usability of email encryption software,|| in Symposium On Usable Privacy and Security, 2006, pp. 3–4.

- [10] A. Shamir, —Identity-based cryptosystems and signature schemes,|| in Advances in Cryptology—CRYPTO 1984. Springer, 1984, pp. 47– 53.
- [11] Proofpoint, —Proofpoint Email Protection,|| <https://www.proofpoint.com/us/products/email-protection>, 2005, accessed April 18, 2019.
- [12] DataMotion, —DataMotion SecureMail,|| <https://www.proofpoint.com/us/products/email-protection>, 2013, accessed April 18, 2019.
- [13] N. Unger, S. Dechand, J. Bonneau, S. Fahl, H. Perl, I. Goldberg, and M. Smith, —SoK: secure messaging,|| in 2015 IEEE Symposium on Security and Privacy. IEEE, 2015, pp. 232–249.
- [14] H.-M. Sun, B.-T. Hsieh, and H.-J. Hwang, —Secure e-mail protocols providing perfect forward secrecy,|| IEEE Communications Letters, vol. 9, no. 1, pp. 58–60, 2005.
- [15] J. O. Kwon, I. R. Jeong, and D. H. Lee, —A forward-secure e-mail protocol without certificated public keys,|| Information Sciences, vol. 179, no. 24, pp. 4227–4231, 2009.