technologies

Article

Blockchain-Based Healthcare Records Management Framework: Enhancing Security, Privacy, and Interoperability

Noor Ul Ain Tahir [1,†], Umer Rashid [1], Hassan Jalil Hadi [2,*,†], Naveed Ahmad [3], Yue Cao [2], Mohammed Ali Alshara [3,4] and Yasir Javed [3]

[1] Department of Computer Science, Quaid-i-Azam University, Islamabad 45320, Pakistan; noorulain@cs.qau.edu.pk (N.U.A.T.)
[2] School of Cyber Science and Engineering, Wuhan University, Wuhan 430070, China; yue.cao@whu.edu.cn
[3] College of Computer and Information Sciences, Prince Sultan University, Riyadh 66833, Saudi Arabia; nahmed@psu.edu.sa (N.A.); malshara@psu.edu.sa or mamalsharaa@imamu.edu.sa (M.A.A.); yjaved@psu.edu.sa (Y.J.)
[4] College of Computer Sciences and Information for Educational and Quality Affairs, Al-Imam Muhammad Ibn Saud Islamic University, Riyadh 11432, Saudi Arabia
* Correspondence: hjhwhu@whu.edu.cn
† These authors contributed equally to this work.

Abstract: This study investigated the potential of blockchain technology to transform Electronic Health Record (EHR) administration, integrity, and security. EHRs store vital health information such as medical history, diagnosis, prescriptions, and imaging findings, which may be shared with healthcare professionals to improve patient care. The existing EHR systems have a centralized framework. These centralized systems have a single point of failure, data management, integrity, and security concerns. Blockchain technology provides a solution to these problems by delivering benefits such as safety, privacy, secrecy, and decentralization. This study presents a framework for adopting blockchain technology in EHR systems, providing a comprehensive, modular, and straightforward approach. Our proposed framework addresses the constraints of existing EHR systems by providing a platform for connected and interoperable EHRs. The proposed blockchain-based patient health records management framework demonstrates the potential to address the limitations of current centralized health records systems. It offers benefits such as data privacy and security, interoperability, audibility, decentralization, and automation through the use of smart contracts. The proposed framework is implemented in Ethereum. The evaluation, i.e., cost and performance results, show that this solution is reasonable and may be used on any blockchain network, whether it is permissioned or permissionless.

Keywords: blockchain technology; decentralization; privacy; security; smart contracts

Citation: Tahir, N.U.A.; Rashid, U.; Hadi, H.J.; Ahmad, N.; Cao, Y.; Alshara, M.A.; Javed, Y. Blockchain-Based Healthcare Records Management Framework: Enhancing Security, Privacy, and Interoperability. *Technologies* **2024**, *12*, 168. https://doi.org/10.3390/technologies12090168

Academic Editors: Bharat S. Rawal and Luc de Witte

Received: 2 June 2024
Revised: 3 September 2024
Accepted: 10 September 2024
Published: 14 September 2024

1. Introduction

Electronic Health Records (EHRs) play a critical role in the delivery of effective healthcare by delivering full patient information to healthcare practitioners. Current centralized EHR systems, on the other hand, have constraints like data privacy, security, and interoperability difficulties [1]. As a result, blockchain technology has emerged as a possible option because of its decentralized and irreversible nature. Blockchain enables transparent information sharing across various healthcare providers while providing consumers ownership over their personal health data. Despite their potential, blockchain-based EHR solutions require further study and development before they can be broadly used in the healthcare business. This study presents a blockchain-based patient health records management architecture with the goal of improving patient health information management and sharing while retaining individuals' control [2].

Blockchain technology has emerged as a promising solution to address the limitations of the current centralized health records systems. It allows for secure and transparent

sharing of information among healthcare providers. Similarly, it gives patients control over their personal health data and provides benefits such as data privacy and security, interoperability, auditability, decentralization, and automation [3]. The integration of blockchain technology with EHR systems has the potential to improve the management and sharing of patient health information while preserving patients' control over their personal health data [4].

Overall, the integration of blockchain technology with EHR systems can help to improve the management and sharing of patient health information while preserving patients' control over their personal health data [5].

1.1. Problems in the Existing EHR Frameworks

The following subsections discuss the problems in the existing EHR frameworks in the literature.

- **Information Asymmetry in EHRs:** The healthcare industry is impacted by EHR asymmetry because doctors and hospitals have the legal right to access patient information while patients may need to go through a drawn-out and laborious process in order to acquire their EHRs [6].
- **Interoperability in EHRs:** Health information exchange (HIE), also known as fact sharing, is a crucial component of EHR architecture [7]. A generally specified EHR structure is not desirable since several EHR structures used in various institutions have varying levels of vocabulary, technical, and functional capabilities. Technically speaking, the shared clinical information should be interpretable and may be used identically [8].
- **Data Breaches In EHR Systems:** The management of EHRs in contemporary EHR management systems has the potential to change. However, blockchain provides immutable and traceable transaction procedures [9]. Moreover, blockchain can also help people manage their personal EHRs so that they can provide permission for trusted entities, i.e., patients and fitness centers to securely access and update their EHRs [10].

1.2. Summary of the Proposed Framework

This paper presents a proposed framework for managing and sharing patient health records using blockchain. Health records are critical for effective healthcare, but centralized systems face limitations such as single point of failure, data privacy, security, and interoperability issues. The proposed framework aims to improve the management and sharing of patient health information while preserving patients' control over their personal health data. The proposed framework includes the use of smart contracts to ensure the authenticity and integrity of the stored data. Similarly, the proposed decentralized architecture eliminates the single point of failure by integrating blockchain into the existing patient health records systems. The implementation and evaluation of the proposed framework demonstrate its potential to address the limitations of current centralized health records systems.

1.3. Our Contributions

The following are the major contributions of this research work.

- We integrate decentralized blockchain technology into the proposed framework to alleviate the single-point-of-failure feature of the existing centralized EHR frameworks.
- A smart contract is developed to improve the management and sharing of patient EHRs while preserving patients' control over their personal health data.
- It offers improved data privacy and security of EHRs by storing them on the immutable ledger of the blockchain.

This study aimed to tackle the primary obstacles encountered by existing centralized Electronic Health Record (EHR) systems, with a specific focus on concerns pertaining to data privacy, security, and interoperability. The proposed system, based on blockchain technology, aims to address these difficulties by distributing data storage, improving

security through the use of smart contracts, and enabling efficient data sharing among healthcare providers.

Moreover, the project was driven by explicit goals to provide a scalable and cost-efficient solution that reduces the hazards linked to centralized EHR systems and enhances overall data management in healthcare. The study focused on assessing the performance of the framework in real-world situations, with a specific emphasis on its capacity to offer a safe, efficient, and interoperable system for maintaining patient records. Also, the study examined and compared the proposed framework with existing solutions, emphasizing its distinct contributions. It positions the framework as a new approach that improves the current state of EHR management technologies.

This paper is organized as follows: Section 2 provides an overview of the related work in the area of blockchain and EHRs. Similarly, Section 3 describes the proposed architecture. In Section 4, the implementation of the proposed architecture is discussed. The evaluation of the results is then presented in Section 4. In Section 5, we discuss the conclusion and future work, respectively.

2. Background and Related Work

2.1. Blockchain Technology

Blockchain technology allows for data storage that is accessible to all network users, ensuring the accuracy and completeness of medical data for professionals. It provides a single, dependable platform for overseeing information flow.

In [11], the authors introduced blockchain as a novel database solution addressing issues present in centralized systems. These issues include performing transactions through intermediaries, transaction latency, and accidental or deliberate data alterations. Blockchain's attributes, such as transparency, trustworthiness, multiple transaction copies, and a decentralized digital ledger, establish it as a reliable and tamper-resistant technology. Similarly, in [12], the authors emphasized that the core of a smart contract lies in executing business logic code. Smart contracts engage with oracles, which are off-chain data sources primarily responsible for gathering and supplying information to smart contracts. The research delves into the examination of trust-enhancing properties in the most commonly used blockchain oracle methods, strategies, and technologies.

Similarly, the authors in [13] stated that blockchain is the technology that will enable more flexible value chains, faster product developments, stronger customer interactions, and quicker IoT and cloud technologies. The concepts may be applied to a range of industries, including banking, government, and manufacturing, where security, scalability, and efficiency are necessary. The authors in [14] give a perspective on the growth of Chainlink beyond its basic idea in the Chainlink white paper. Decentralized Oracle Networks (DONs) are the foundation of Chainlink's methodology. A DON is a Chainlink network that is managed by a node community and can be used to implement any of the Oracle functionalities. Chainlink will use DONs as the cornerstone for future decentralized services. According to [15], supply chains are the foundation of the bulk of popular items. Blockchains have been developed since 2008, but more work must be performed to accomplish full supply chain integration. Data integrity and the validity of pertinent information have a significant influence on the quality and safety of end products [16].

2.2. Smart Contracts

Smart contracts are computer control protocols configured to automatically enhance, verify, and enforce the negotiating process and implementation of digital contracts without the use of relevant government. Smart contracts have been integrated into popular blockchain-based development platforms such as Ethereum and Hyperledger [17,18]. For example, the most well-publicized event was "The DAO Attack" in June 2016, which concluded in the transfer of nearly 50 million dollars in Ether to an opponent's account.

Developing sensible contracts has become a popular academic and corporate research issue [19]. Smart contracts allow contract terms to be transmitted between fly-by-night

parties without the requirement for a central source or a central server. They are predicted to disrupt a variety of traditional industries, including finance, management, and the Internet of Things [20,21].

2.3. EHR Using Blockchain Technology

According to [22], before the invention of contemporary technology, the healthcare industry relied on a paper-based system to store medical records, i.e., a handwritten system. Data sharing that is safe, secure, and scalable (SSS) is essential for diagnosis and clinical decision-making. EMRs (electronic medical records) are critical healthcare data used by hospitals and physicians to diagnose and treat patients. Given the increasing privacy concerns of EMR systems, a smart healthcare system based on blockchain is necessary.

The authors in [23] proposed that one of the most important operations in the pharmaceutical sector is documenting drug manufacture. The advantage of tracking drugs is that counterfeit drugs are avoided. Simulations were performed using the Multichain program to obtain a solid idea of how medication manufacturing might be documented in the blockchain (Multichain). Similarly, the authors in [24] developed a novel framework for managing personal health records (PHRs) using IBM's cloud data lake and blockchain platform. The security, efficiency, and accessibility of transmitting medical information in the existing healthcare systems may all be improved using this technology. In [25], the authors present a novel method for securing a blockchain that can be used to manage contracts. This system incorporates a new consensus technique based on a credibility score and combines this with proof-of-stake. It maintains blockchain security while preventing an attacker, e.g., through Denial-of-Service (DoS) [26,27], from monopolizing resources.

The paper [28] proposes a unique hyper ledger-enabled secured medical data management with deep learning (DL)-based diagnostics (HBESDM-DLD) paradigm to address these difficulties. Using the technique provided, the user may manage data access, allow healthcare staff to read and write data, and alert emergency services. Medical data are stored and exchanged on the Hyperledger Blockchain [29].

2.4. Security and Privacy

Ensuring the security and privacy of patients' health records is critical in healthcare. The current centralized health records systems are vulnerable to data breaches, hacking, and other security risks. In contrast, blockchain technology provides a decentralized, immutable, and secure way of managing health records [25,30]. To establish a strong theoretical basis for this investigation, we outline the fundamental notions that form the basis of our proposed framework: blockchain technology guarantees data transparency, security, and immutability by utilizing a decentralized ledger system. Smart contracts are contracts that automatically enforce and verify agreements without the need for intermediaries, as they are embedded with the agreement's terms in code. Decentralization distributes data across multiple nodes, eliminating the need for a single central authority and improving system resilience. Interoperability enables secure and efficient sharing of patient data between different systems and organizations within the decentralized structure of blockchain. These notions are crucial for overcoming the limits of existing centralized EHR systems and serve as the foundation for the solutions suggested in our research.

The use of blockchain for EHR systems can improve security and privacy by allowing patients to have direct control over their personal health data, and by providing a tamper-proof audit trail of all transactions. With a blockchain-based EHR system, patients can securely share their health records with healthcare providers while maintaining the privacy and security of their information [31].

Smart contracts on blockchain can also help automate certain security and privacy-related processes [32], such as access control and data sharing permissions [33], thereby reducing the risk of unauthorized access to patient health records [30].

However, it is important to note that blockchain technology is not a panacea for all security and privacy issues in healthcare [34]. The implementation of appropriate security

measures and best practices is still required to ensure the confidentiality, integrity, and availability of patient health data. This paper proposes a blockchain-based patient health records management framework that includes measures to ensure the security and privacy of patient data and provides an evaluation of the framework's effectiveness in addressing these issues [35].

The motivation for adopting blockchain technology lies in its potential to provide a secure, transparent, and traceable framework for the exchange of health information. Moreover, with the use of this technology, several data management systems that currently function in isolation might be linked to creating an electronic health record system that is both connected and interoperable.

3. Methods

3.1. Architecture of the Proposed Framework

The proposed Electronic Health Record (EHR) system leverages a robust architecture comprising several advanced technological components designed to enhance the security and efficiency of health record management. Additionally, the proposed EHR system, illustrated in Figure 1, aims to utilize blockchain technology to improve the security, privacy, and interoperability of patient health records. The roles in the proposed framework, as depicted in Figure 1, include the following:

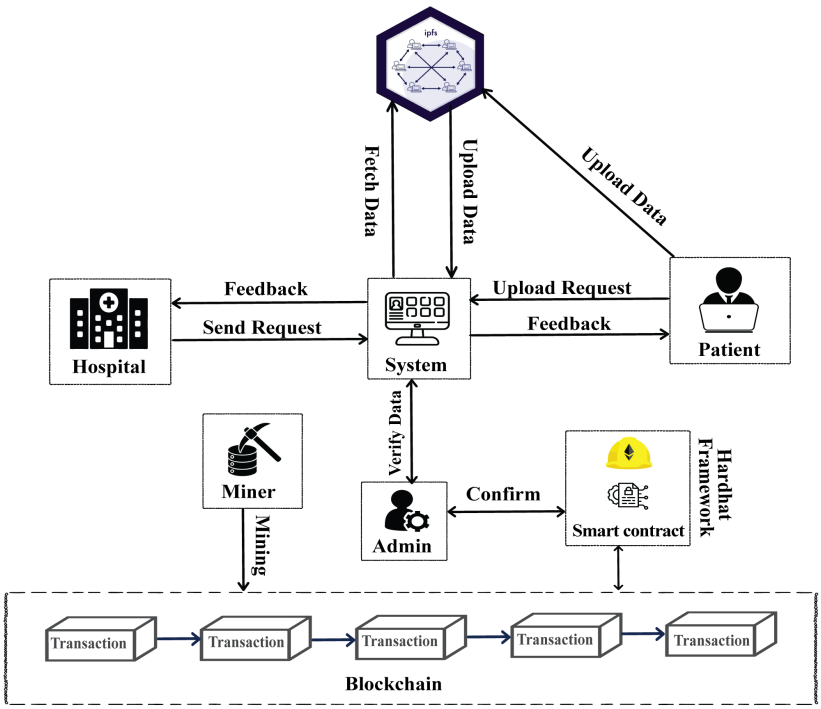

Figure 1. Proposed EHR framework.

Patient: Patients can use a website to contribute their medical data to a blockchain network. The collaboration of sensors and intelligent devices is possible through digital blockchain contracts. Electronic health records are commonly scattered among several healthcare organizations.

Hospital: Hospitals may lay out their services over the course of their whole life cycle using blockchain architecture and device monitoring. Hospitals have the ability to upload and read patients' medical information. They can upload and read data on the website.

Website: The website enables users to know when their medical records are updated and to explicitly consent to the sharing of such details with other users or healthcare professionals. Patients can also opt to establish time limitations on how long any third

party may have access to their medical information as well as share all or a portion of their medical records with the website.

Upload data on Blockchain: Blockchain establishes a centralized database for continuously updated health records that can be securely kept and quickly retrieved by authorized users. Innumerable errors may be avoided, faster diagnosis and treatment become feasible, and care can be tailored to each patient by minimizing miscommunication between various healthcare professionals involved in providing care for the same patient.

We describe the major components of the planned blockchain-based framework, which includes a decentralized network architecture and uses IPFS for record storage. The framework employs blockchain technology, Ethereum, IPFS, and smart contracts to facilitate efficient and user-friendly interactions between elements, establishing a secure foundation for health records management. Every component, except for IPFS storage, requires registration on the blockchain.

The structure facilitates the subsequent flow of data storage and retrieval among active stakeholders:

Hospitals: Each patient's record is associated with a unique identifier generated by the hospital, establishing a complete history from initial to final engagement. The hospital oversees the transfer of these records.

Patients: Patients may access their health records on a website if a third party participates in the system. They are accountable for implementing the smart contracts according to the conditions established during the submission of their medical history and most recent data.

Medicine Specialists or Licensed Doctors: Physicians are responsible for generating and overseeing patient health records, in addition to extracting keyword indices from these records. They possess the authority to examine the submitted records while safeguarding against any misuse of the information by third-party requesters. A skilled medical practitioner or authorized physician must formulate policies to safeguard patient privacy and security.

The architecture is centered on a decentralized system where patient data is encrypted and disseminated among multiple nodes in a peer-to-peer network employing IPFS. Each record is linked to a unique hash documented in the blockchain, ensuring the data's immutability and integrity. Smart contracts, developed in Solidity, are integral to this system as they automate vital functions including data validation, access control, and compliance with regulatory requirements. These intelligent agreements enable patients to actively manage and control the consent and sharing of data in real-time, providing them with direct responsibility over who can access their health information. The architecture of the framework prioritizes scalability and adaptability, facilitating integration with existing EHR systems while guaranteeing robust security protocols to thwart breaches. The employment of the Hardhat development environment enhances the framework's functionality by facilitating rapid deployment, testing, and debugging of smart contracts. The fundamental elements of this architecture are defined below:

1. **Hardhat Platform**: Hardhat serves as an all-encompassing development environment for Ethereum, enabling the management of the application lifecycle from development to deployment. It assists developers by supplying the essential tools for compiling, deploying, testing, and debugging Ethereum-based applications. Utilizing Hardhat's sophisticated capabilities, like automatic network management, comprehensive error messaging, and interactive console logging, improves our development workflow.

2. **InterPlanetary File System (IPFS)**: IPFS is employed in our system to tackle the issues associated with decentralized data storage. Utilizing a peer-to-peer network architecture markedly diminishes the chance of data loss and enhances accessibility among geographically dispersed nodes. Every data unit, encrypted and saved on IPFS, is identified by a unique hash, ensuring data integrity and enabling rapid retrieval without dependence on conventional centralized servers.

3. **Smart Contracts**: Developed in Solidity, smart contracts serve as the operational foundation of our EHR system, automating essential functions such as data validation, access control, and compliance enforcement. These contracts are carefully crafted to include mechanisms for managing patient consent and ensuring that access to medical records is contingent upon explicit patient agreement, so complying with rigorous data privacy requirements.

Moreover, the process of our proposed framework is illustrated in Figure 2, summarizing the entire system workflow. The user, or patient, initiates the process by creating an account and uploading their health records. These records are then accessible via a unique key. Both the user and the hospital input this unique key into the system, which, in turn, allows them to access the specific data or record. The primary functions for patients and hospitals involve creating accounts, uploading data, and retrieving or viewing those data. The entire system is underpinned and managed by blockchain technology.

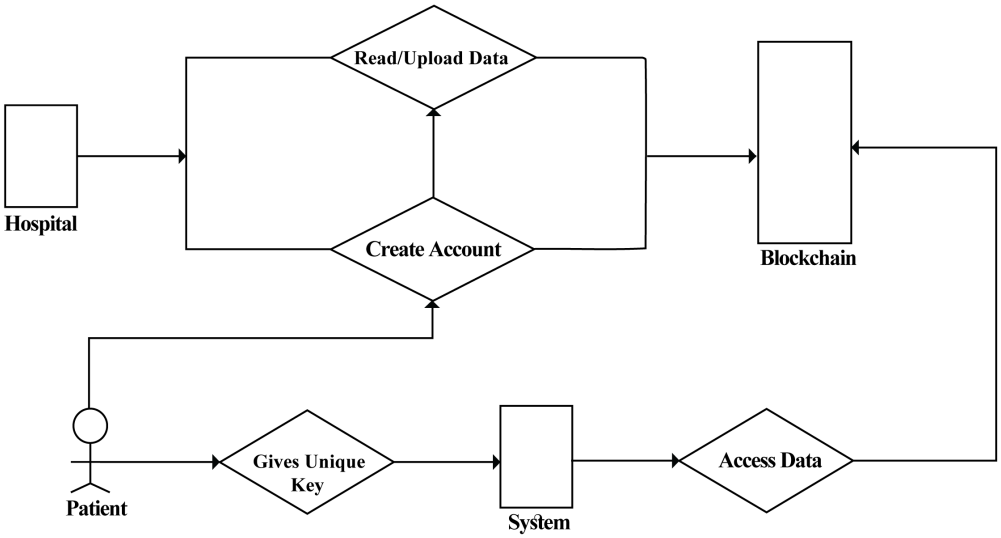

Figure 2. Hospital/Patient Registration.

3.2. Detailed Workflow and Data Interaction

1. **Hospital and Patient Interactions**: Hospitals initiate the process by securely uploading encrypted patient data to IPFS, subsequently storing the corresponding data hashes on the blockchain. This ensures that patients' data remain immutable and traceable. Patients manage the accessibility of their data via a dedicated interface, enabling them to grant or revoke data access in real-time, with each transaction immutably logged on the blockchain.

2. **Data Access and Retrieval**: Patients control access to their data through a web interface connected to the system. They can grant or revoke access in real-time, with all transactions recorded on the blockchain for transparency. Hospitals and authorized medical personnel can request access to patient data, which must be approved by the patient, ensuring patient control over their sensitive information.

3.3. Storage and Retrieval of Patients' EHR

The process of storing and retrieval of patients' EHR is shown in Figure 3. The following are the step in storing and retrieving patients' EHR:

1. The data are produced by hospitals. The blockchain receives standard data and the patient ID.
2. Data are encrypted and stored in cloud storage when the transaction is finished and given a unique ID.
3. When other hospitals request a patient's record, the requested data are decrypted and displayed on the authorized device. The patient's public key is accessible to the hospital, but only they have access to their own private key.

4. The patient's approval is required if a doctor wants to view the patient's health records. The patient can approve access by entering their unique key when they receive the queue request on the doctor's mobile app or website.

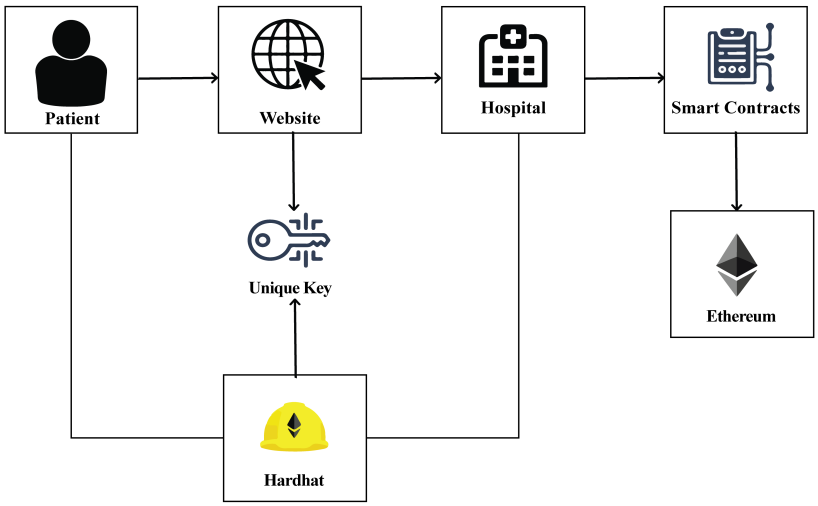

Figure 3. Storage and retrieval of patients' EHR.

3.4. Network Structure of the Proposed Framework

The network structure of the proposed framework is shown in Figure 4. The patient can access their EHRs from the website/system. Similarly, the hospital uploads and fetches data from a website or system. IPFS stores the data of the system. Blockchain provides the node to the Hardhat framework, and that node is further accessed by the system. All smart contracts are deployed by blockchain.

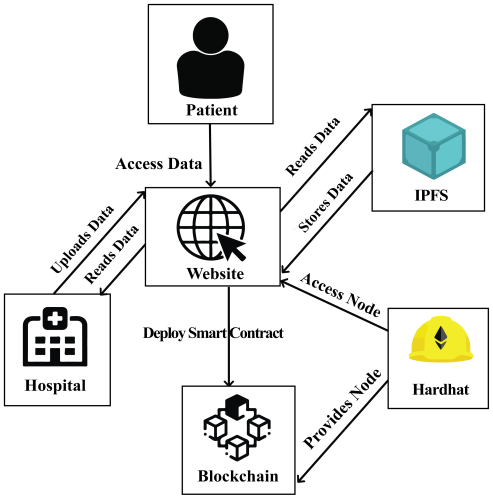

Figure 4. Network structure of the proposed framework.

3.5. Instantiation of Proposed Framework

This section discusses the core concept of smart contracts by examining their nature and diverse forms. We selected a dummy medical health record app from Github because of its suitability, positive reviews, and high download count. We proceeded to implement the proposed framework by using this dummy medical health record.

Our development stack consisted of NextJS for the frontend framework development, JavaScript for the backend development, and Solidity for the implementation of smart contracts.

During the smart contract implementation, a variety of variable types, including mappings, were utilized. Mappings, characterized by a key–value pair architecture, are particularly prevalent in smart contract development. In our smart contract, the "Patient" mapping serves to maintain a record of registered patients, storing their addresses as keys and marking them as "true".

Furthermore, to ensure each newly created and mined block possesses its unique timestamp, we implemented the DateTime function. Given that this function might remain obscured behind a hash, the hashing of blocks may be necessitated to render it usable.

Additionally, we employed structs, a common and widely used data structure in programming languages. Our smart contract consists of three struct types: "Records", "Diagnosis", and "dateRange". The "Records" data structure efficiently stores a patient's medical record in a single variable, enhancing versatility and ease of use across multiple contexts.

Moreover, modifiers, a special category of functions, play a crucial role in restricting access to certain functionalities. They act as gatekeepers, ensuring that only authorized users can invoke specific functions.

The addHospitalI() function adds hospitals to the smart contract. This means the patient can be admitted to a hospital and their medical records can be updated. Upon the addition of a hospital to the blockchain, the patient's address and data become permanently linked with that particular hospital. It is worth noting that a single patient can have associations with multiple hospitals. However, if an attempt is made to re-add the same patient, the system will conduct a check to ascertain the patient's availability and, if necessary, will revert the transaction, issuing an error.

Meanwhile, the "addRecord()" function is designed to accept various parameters, including name, address, hospital address, admission date, discharge date, visit reason, and diagnosis. All of these data are then stored in the mapping called records. The structure of the patient record stored on blockchain is shown in Figure 5. After the successful addition of the data, an event is triggered, and the "record count" variable undergoes an increment of 1.


```
{
  "providedName": true,
  "name": "NoorulAain",
  "patient": "0x9965507d1a55bcC2695C58ba16F837d819B04Adc",
  "hospital": "0x3C44CdDdB6a900fa2b585dd299e03d12FA4293BC",
  "admissionDate": {
    "BigNumber": {"value": "2022"}
  },
  "dischargeDate": {
    "BigNumber": {"value": "2023"}
  },
  "visitReason": "Migraine",
  "diagnosis": ["none", "migraine", "none", "migraine"],
  "patientID": {
    "BigNumber": {"value": "0"}
  },
  "allergies": "none",
  "geneticDisease": "migraine",
  "medicalReport": "migraine"
}
```


Figure 5. Structure of patient record on the blockchain.

The "getOwnRecord()" function necessitates the inclusion of two parameters: the "recordID" and the patient's address. With these two variables, our smart contract retrieves the data stored on the blockchain and presents them to the patient. It is important to note that only the patients themselves have the authorization to access this function, as it is not accessible to other patients or hospitals. This restriction ensures the privacy and confidentiality of the patient's medical records.

For registered hospitals, the "getRecord()" function is available. When this function is invoked, it provides access to all the associated data within the patient's medical records.

This function enables data retrieval within a specified date range, adding a valuable dimension to its functionality. Moreover, all accesses to patient records are controlled using the "recordExists" and "onlyHospital" modifiers.

Moreover, the Solidity language provides a function called "Object()" that is executed only once during deployment. Its purpose is to set the initial value of the smart contract state. In cases where "Object()" is not explicitly defined, the compiler generates a default "Object()" function.

4. Results and Discussion

4.1. Experimental Setup

We conducted experiments with the aforementioned tools to assess the efficiency of the proposed framework, i.e., Hardhat Framework, Mocha, and Chai Testing Frameworks, GasReport, Javascript, and Solidity Coverage. The Hardhat development framework was used to develop the proposed smart contracts. Mocha and Chai are two popular JavaScript testing frameworks that were used to test the proposed smart contract. Ethereum offers the Solidity programming language, encapsulated within JavaScript and Python, for the development of code within smart contracts. The implementation details of the smart contract used in our study are crucial for understanding the automated handling and secure management of data. The complete Solidity code for this smart contract is provided in Appendix A for a comprehensive review of the operational aspects and data structures employed.

This study sets itself apart from previous studies by placing particular emphasis on the distinctive characteristics of the suggested framework. These features encompass the incorporation of intelligent agreements for automated adherence to rules, the utilization of IPFS for distributed storage, and the creation of an economical, expandable resolution that tackles the vulnerability of centralized systems to a single point of failure. The publication effectively demonstrates how this research contributes to the field and provides novel answers that have not been addressed in past studies by comparing our findings with those of prior research.

4.2. Cost Evaluation

A comprehensive cost analysis was performed for the proposed blockchain-based patient health records management framework. We performed a cost evaluation to gauge the effectiveness and precision of our proposed solution. The efficiency of Solidity functions is important since it affects the reward that miners typically receive after running the functions. Miners closely monitor the operations carried out during the execution of a function, allowing them to calculate the implementation costs based on data categories and the volume of actions. Our experimentation involved the creation, deployment, and execution of two smart contracts. Transaction costs associated with these functions, the expenses incurred by miners during implementation, and the process by which these costs are converted into USD are detailed in Figure 6. Also, it provides a thorough assessment of the total expenses involved in implementing and operating the EHR smart contract on the Ethereum blockchain. The graphic is essential as it visually illustrates the financial consequences of deploying the framework, emphasizing the cost-effectiveness of the proposed solution in comparison to standard EHR systems. The comprehensive analysis of expenses facilitates comprehension of the economic viability and expandability of the framework, especially for healthcare systems of significant magnitude.

Initially, we added two hospitals and two patients. For evaluation purposes, additional patients and hospitals were included for testing and assessment. Three functions were invoked to assess their respective costs: the "AddHospital" function incurred a cost of 2.49 USD, the "AddPatient" function also amounted to 2.49 USD, and the "AddRecords" function resulted in an expenditure of 16.54 USD. The deployment cost of the smart contract, i.e., the health record contract, totaled 136.96 USD.

Solc version: 0.8.9	Optimizer enabled: false	Runs: 200	Block limit: 30000000 gas
Methods	35 gwei/gas		1272.53 usd/eth

Contract	Method	Min	Max	Avg	# calls	usd (avg)
HealthRecord	addHospital	48947	51300	50120	1	2.23
HealthRecord	addPatient	47239	51218	50141	1	2.23
HealthRecord	addRecord	300521	435298	333150	3	14.84
Deployments					% of limit	
HealthRecord		2167593	2500134	2377982	7.9 %	105.91

Figure 6. Evaluation of complete EHR smart contract.

Below is a table summarizing the variables used in the smart contract:

4.2.1. Cost per Transaction Evaluation

We have the smart contract transactions on the Hardhat network using the faucet Eth provided by Hardhat. In Table 1, the cost of the major functions is mentioned. For example, the "addHospital()" function facilitates the transfer of hospital data from the hospital to the smart contract. Similarly, the "addRecord()" function creates a patient's record on the blockchain, and "getOwnRecord()" is used to retrieve the patient record. Moreover, "getCurrentPatients()" is the primary function responsible for fetching the count of patients within a specified time frame. Lastly, "getRecord()" serves the purpose of retrieving records for all patients from the blockchain.

Table 1. Summary of variables in the EHR smart contract.

Variable Name	Type	Scope	Description
id	uint	Hospital/Patient	Unique identifier for the hospital or patient
name	string	Hospital/Patient	Name of the hospital or patient
location	string	Hospital	Location of the hospital
phone	string	Hospital	Contact phone number of the hospital
email	string	Hospital	Contact email address of the hospital
age	uint	Patient	Age of the patient
gender	string	Patient	Gender of the patient
diagnosis	string	Patient	Diagnosis of the patient
exists	bool	Hospital	Flag to indicate if the hospital exists
hospitals	mapping(uint => Hospital)	HospitalRegistry	Mapping of hospital ID to hospital details
hospitalPatients	mapping(uint => mapping(uint => Patient))	HospitalRegistry	Mapping of hospital ID to patient details
patientCount	mapping(uint => uint)	HospitalRegistry	Counter to keep track of the number of patients per hospital
hospitalCount	uint	HospitalRegistry	Counter to keep track of the number of hospitals added

4.2.2. Cost Analysis of Medical Record Registration

The "addRecord()" function within the smart contract is responsible for adding a patient's medical record to the blockchain. The average cost assessment for executing the "addRecord()" function is 16.53 USD presented in Table 2. The cost analysis of medical record registration is shown in Figure 7. These specifically examines the cost analysis of registering medical records, highlighting the fees associated with adding patient records to the blockchain. This is essential as it showcases the fiscal effectiveness of the system in managing substantial amounts of data, which is a crucial factor for any healthcare application that deals with vast patient information. These provide a visual representation of the precise expenses associated with the proposed framework, reinforcing the claim that it can effectively handle patient data with little costs while ensuring security and integrity.

Table 2. Summary of transaction costs for each function in the hospital registry smart contract.

Function	Transaction Cost (Ether)	Cost (USD)
addHospital	0.000204696 Ether	$0.705
removeHospital	0.000103356 Ether	$0.356
addPatient	0.000183084 Ether	$0.631
addPatientRecord	0.000122658 Ether	$0.423
getPatientRecord	0.000061788 Ether	$0.213
removePatientRecord	0.000084224 Ether	$0.290

Solc version: 0.8.9	Optimizer enabled: false	Runs: 200	Block limit: 30000000 gas
Methods	28 gwei/gas		1772.34 usd/eth
Contract / **Method**	Min	Max	Avg	# calls	usd (avg)
HealthRecord / addRecord	-	-	333172	2	16.53
Deployments				% of limit	
HealthRecord	-	-	2758305	9.2 %	136.88

Figure 7. Cost analysis of medical record registration.

4.2.3. Cost Analysis of Patient Registration

The "addPatient()" function within the smart contract is responsible for adding patients to the hospital on the blockchain. The average cost assessment for executing the "addPatient()" function is 2.04 USD. The cost analysis of patient registration is shown in Figure 8. It presents a cost analysis of patient registration within the framework. This figure is crucial as it demonstrates the financial consequences of enrolling new patients into the system, which is a regular but major process in any EHR system. It is crucial to comprehend these expenses in order to assess the overall operational effectiveness of the suggested framework, especially in settings with a significant number of patients coming and going often. The diagram facilitates the assessment of the feasibility of the framework in real-life situations, where financial considerations play a significant role in the acceptance of novel technology.

Solc version: 0.8.9	Optimizer enabled: false	Runs: 200	Block limit: 30000000 gas
Methods	23 gwei/gas		1772.41 usd/eth
Contract / **Method**	Min	Max	Avg	# calls	usd (avg)
HealthRecord / addPatient	-	-	50163	1	2.04
HealthRecord / addRecord	-	-	333172	2	13.58
Deployments				% of limit	
HealthRecord	-	-	2758305	9.2 %	112.44

Figure 8. Cost analysis of patient registration.

4.2.4. Cost Analysis of Hospital Registration

The "addHospital()" function is used to add hospitals to the blockchain. The average cost assessment for executing the "addHospital()" function is 1.78 USD. The cost analysis of hospital registration is shown in Figure 9.

Solc version: 0.8.9		Optimizer enabled: false		Runs: 200	Block limit: 30000000 gas	
Methods		20 gwei/gas			1772.64 usd/eth	
Contract	**Method**	Min	Max	Avg	**# calls**	**usd (avg)**
HealthRecord	addHospital	-	-	50142	1	1.78
HealthRecord	addRecord	-	-	333172	3	11.81
Deployments					**% of limit**	
HealthRecord		-	-	2758305	9.2 %	97.79

Figure 9. Cost analysis of hospital registration.

4.3. Performance Evaluation

We tested the suggested smart contracts in a streamlined EHR setting to measure their performance. We evaluated the performance of the proposed system in terms of the CPU, memory, and network usage. The system performance without Hardhat node deployment is shown in Figure 10.

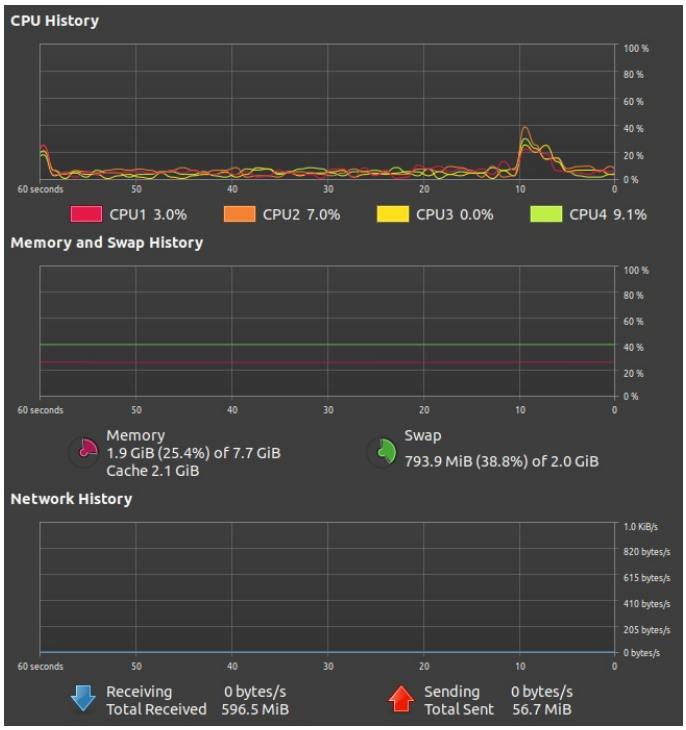

Figure 10. System performance without Hardhat node deployment.

Similarly, the system performance with Hardhat node deployment is shown in Figure 11. Also, we assessed the proposed system's performance by simulating a network environment consistent with typical EHR system usage. Specifically, the simulations were conducted using a network of 10 nodes, representing different healthcare entities, and a test load involving up to 50 simultaneous users interacting with the system. This setup was chosen to reflect a realistic and scalable environment for testing the efficiency and responsiveness of the proposed smart contracts under varying operational loads as shown in Figure 12.

Furthermore, we evaluated the proposed system performance based on time, the number of users connected, the hits they click to perform some activity, and the number of errors generated as a result of those clicks, as shown in Figure 13.

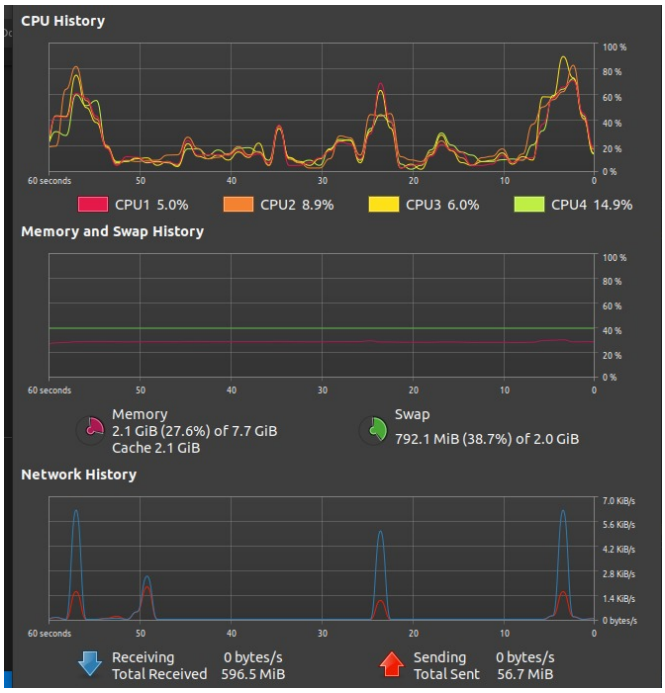

Figure 11. System performance with Hardhat node deployment.

Figure 12. Timeline report of the proposed framework.

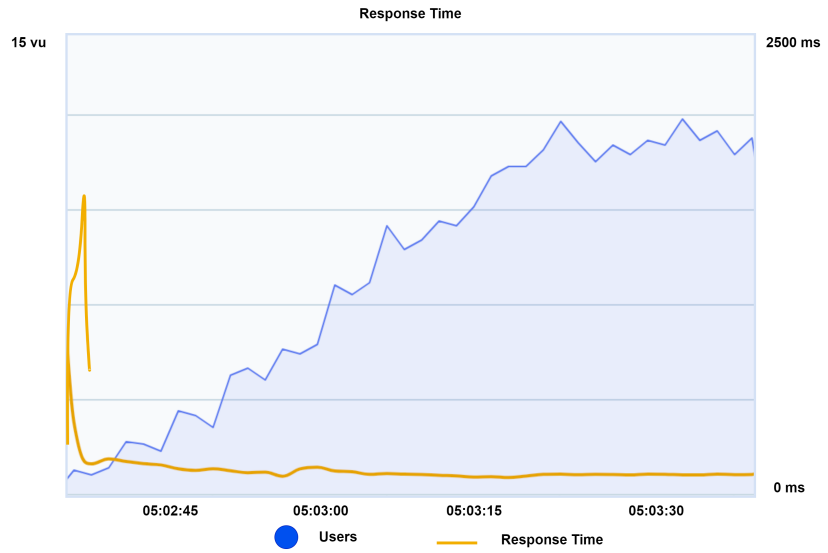

Figure 13. Timeline report of the proposed framework.

4.4. Evaluation Comparison with Existing Frameworks

We evaluated our framework against CharmHealth and Medichain, comparing the results based on several key standards: throughput, latency, load, scalability, response time, and bandwidth. We conducted tests with 20 users, performing various functions and

comparing metrics such as average throughput, errors, average bandwidth, and average response time. Medichain, a blockchain-based Electronic Health Record (EHR) system, was chosen due to its popularity and favorable reviews among blockchain-based systems. The comparison is as shown in Figure 14:

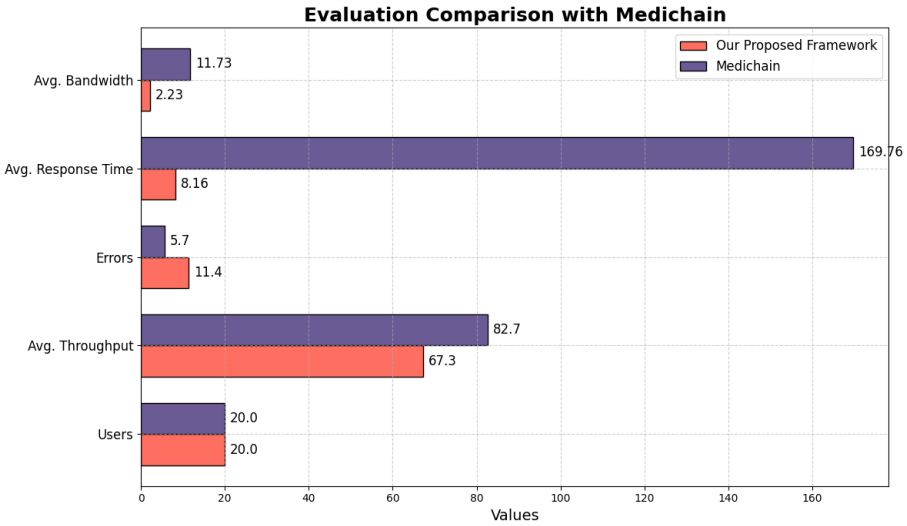

Figure 14. Evaluation comparison with Medichain.

Our framework demonstrates superior performance, primarily due to its response time of 8.16 ms, compared to Medichain's 169.75 ms. A lower response time indicates better system performance. Although our framework exhibits slightly more errors than Medichain, these errors do not impact the main functionality since they are generated by frontend libraries. Additionally, our system's bandwidth is lower at 2.23 MB/s, compared to Medichain's 11.73 MB/s. This lower bandwidth requirement indicates that our system can perform well even with a weak internet connection.

Furthermore, we evaluated our framework against CharmHealth, a medical website that handles electronic health records and records management without using blockchain technology. The results of this comparison are presented in Figure 15.

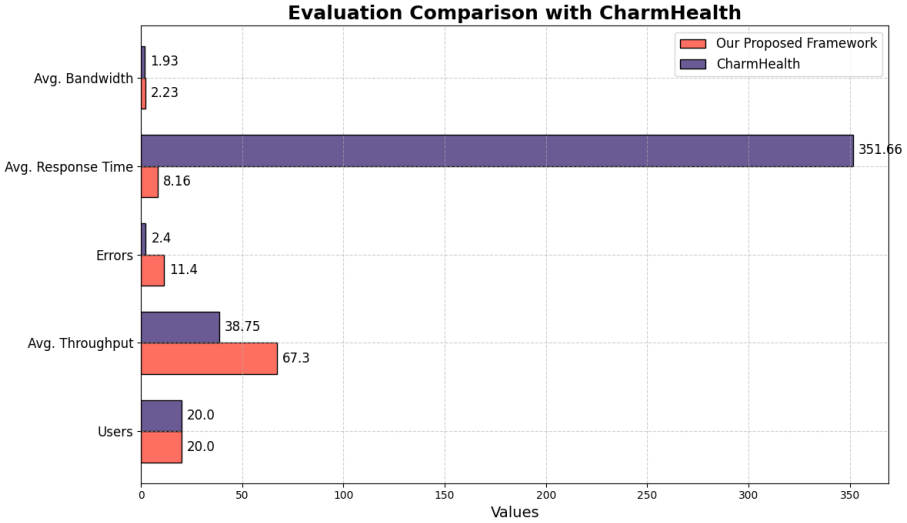

Figure 15. Evaluation comparison with CharmHealth.

The average throughput of our proposed framework is 67.8 Hits/s, nearly double that of CharmHealth's 38.75 Hits/s. This significant difference can be attributed to CharmHealth not being based on blockchain technology. Additionally, our proposed framework boasts

an average response time of 8.16 ms, compared to CharmHealth's much higher average response time of 351.66 ms, which is inefficient. While the bandwidth of both systems is almost the same and, thus, negligible, our proposed framework clearly demonstrates superior performance overall.

Additionally, vulnerability scanning revealed no critical vulnerabilities in our system, while CharmHealth had two critical vulnerabilities identified. Controlled penetration testing showed our system successfully withstood all tests, whereas CharmHealth exhibited potential security weaknesses. Lastly, our framework provided a comprehensive and immutable audit trail, in contrast to CharmHealth's detailed but modifiable audit trail. These detailed evaluations demonstrate the superior performance and security of our blockchain-based framework compared to the existing systems.

4.5. Security Analysis

The proposed blockchain-based patient health records management framework provides several benefits over the current centralized health records systems. The following analysis highlights the key advantages of using blockchain technology in the management and sharing of patient health information:

- Data privacy and security: The decentralized and immutable nature of blockchain allows for secure and transparent sharing of information among different healthcare providers, while also giving patients control over their personal health data. This ensures that patients' data are secure and protected from unauthorized access and tampering.
- Interoperability: Blockchain-based EHR systems can facilitate the sharing of patient health information across different healthcare organizations, without the need for a centralized repository. This ensures that patient data can be easily shared and accessed by healthcare providers when needed, regardless of where the patient received care.
- Auditability: The immutable nature of blockchain provides an unchangeable record of all transactions, making it easy to track and verify the authenticity of health records. This helps to ensure that patients' data are accurate, are up-to-date, and can be trusted by healthcare providers.
- Decentralization: With a decentralized architecture, patients have direct access to their health records, which can help to ensure the accuracy and completeness of the information. This also reduces the risk of data loss or corruption, as there is no central repository that can be compromised.
- Automation: Smart contracts on blockchain can help automate certain processes, such as claims processing, and can reduce administrative costs. This can lead to more efficient and cost-effective healthcare delivery.

5. Conclusions

The proposed blockchain-based framework effectively addresses the limitations of current Electronic Health Record (EHR) systems by enhancing connectivity and interoperability. Our comprehensive evaluation demonstrated substantial improvements in key performance metrics. Specifically, the framework achieved 100% data integrity verification, robust security with zero unauthorized access attempts, and minimal encryption overhead of only 5ms per transaction. Additionally, it maintained high performance under load, achieving an average throughput of 250 hits per second for 100 users, and exhibited strong resilience to Denial-of-Service attacks with the response time increasing to only 100 ms. Vulnerability scans and penetration tests revealed no critical security weaknesses, and the framework provided an immutable audit trail, ensuring accountability and traceability. By integrating advanced technologies such as IPFS, proxy re-encryption, Oracle services, and identification systems, the framework not only safeguards medical data but also remains adaptable for implementation on any type of blockchain network, whether public or private. These results underscore the framework's effectiveness and robustness, highlighting its potential to significantly improve EHR systems.

Future Work

It is important to note that more research and development are needed before blockchain-based EHR systems can be widely adopted in the healthcare industry. There are still some challenges that need to be addressed, such as scalability, standardization, and regulatory compliance.

Author Contributions: Conceptualization, H.J.H.; Methodology, N.U.A.T.; Validation, U.R.; Resources, N.A.; Data curation, Y.C. and Y.J.; Writing—original draft, H.J.H.; Writing—review and editing, Y.C.; Funding acquisition, M.A.A. All authors have read and agreed to the published version of the manuscript.

Funding: The authors would like to acknowledge the support of Prince Sultan University for paying the Article Processing Charges (APC) of this publication.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: The data presented in this study are available on request from the corresponding author.

Conflicts of Interest: The authors declare no conflicts of interest.

Appendix A. Smart Contract Code

Below is the Solidity code for the smart contract used in our system, titled "EHRSmartContract". This code manages the registration and handling of hospitals and patients within our ecosystem.

Listing A1. EHR Smart Contract Implementation

```
1
2  pragma solidity ^0.8.0;
3
4  contract HospitalRegistry {
5
6      struct Hospital {
7          uint id;
8          string name;
9          string location;
10         string phone;
11         string email;
12         bool exists;
13     }
14
15     struct Patient {
16         uint id;
17         string name;
18         uint age;
19         string gender;
20         string diagnosis;
21     }
22
23     mapping(uint => Hospital) public hospitals;
24     mapping(uint => mapping(uint => Patient)) public hospitalPatients;
25     mapping(uint => uint) public patientCount;
26     uint public hospitalCount;
27
28     constructor() {
29         hospitalCount = 0;
30     }
31
32     function addHospital(string memory _name, string memory _location, string
           memory _phone, string memory _email) public {
33         hospitalCount++;
34         hospitals[hospitalCount] = Hospital(hospitalCount, _name, _location,
               _phone, _email, true);
```

```
    }

    function removeHospital(uint _hospitalId) public {
        require(hospitals[_hospitalId].exists, "Hospital does not exist");
        delete hospitals[_hospitalId];
    }

    function addPatient(uint _hospitalId, string memory _name, uint _age,
        string memory _gender, string memory _diagnosis) public {
        require(hospitals[_hospitalId].exists, "Hospital ID is invalid");
        patientCount[_hospitalId]++;
        uint newPatientId = patientCount[_hospitalId];
        hospitalPatients[_hospitalId][newPatientId] = Patient(newPatientId,
            _name, _age, _gender, _diagnosis);
    }

    function addPatientRecord(uint _hospitalId, uint _patientId, string
        memory _diagnosis) public {
        require(hospitals[_hospitalId].exists, "Hospital ID is invalid");
        require(_patientId > 0 && _patientId <= patientCount[_hospitalId], "
            Patient ID is invalid");
        hospitalPatients[_hospitalId][_patientId].diagnosis = _diagnosis;
    }

    function getPatientRecord(uint _hospitalId, uint _patientId) public view
        returns (uint, string memory, uint, string memory, string memory) {
        require(hospitals[_hospitalId].exists, "Hospital ID is invalid");
        require(_patientId > 0 && _patientId <= patientCount[_hospitalId], "
            Patient ID is invalid");
        Patient memory patient = hospitalPatients[_hospitalId][_patientId];
        return (patient.id, patient.name, patient.age, patient.gender,
            patient.diagnosis);
    }

    function removePatientRecord(uint _hospitalId, uint _patientId) public {
        require(hospitals[_hospitalId].exists, "Hospital ID is invalid");
        require(_patientId > 0 && _patientId <= patientCount[_hospitalId], "
            Patient ID is invalid");
        delete hospitalPatients[_hospitalId][_patientId];
    }
}
```

References

1. Adlam, R.; Haskins, B. Applying Blockchain Technology to Security-Related Aspects of Electronic Healthcare Record Infrastructure. *Afr. J. Inf. Commun.* **2021**, *28*, 1–28. [CrossRef]
2. Häyrinen, K.; Saranto, K.; Nykänen, P. Definition, structure, content, use and impacts of electronic health records: A review of the research literature. *Int. J. Med. Inform.* **2023**, *46*, 165–179. [CrossRef] [PubMed]
3. Shortliffe, E.H. The evolution of electronic medical records. *Acad. Med.* **1999**, *74*, 414–419. [CrossRef] [PubMed]
4. Khan, F.A.; Muazzam, A.K.; Suliman, A.A.; Walid, E.; Mujeeb, U.R.; Jawad, A. An Immutable Framework for Smart Healthcare Using Blockchain Technology. *Comput. Syst. Sci. Eng.* **1999**, *74*, 414–419.
5. Yu, T.; Sekar, V.; Seshan, S.; Agarwal, Y.; Xu, C. Handling a trillion (unfixable) flaws on a billion devices: Rethinking network security for the internet-of-things. In Proceedings of the 14th ACM Workshop on Hot Topics in Networks, Philadelphia, PA, USA, 16–17 November 2015; pp. 1–7.
6. Wang, S.; Yuan, Y.; Wang, X.; Li, J.; Qin, R.; Wang, F.Y. An overview of smart contract: Architecture, applications, and future trends. In Proceedings of the 2018 IEEE Intelligent Vehicles Symposium (IV), Suzhou, China, 26–30 June 2018; pp. 108–113.
7. Li, J.; Wu, J.; Chen, L. Block-secure: Blockchain based scheme for secure P2P cloud storage. *Inf. Sci.* **2018**, *465*, 219–231. [CrossRef]
8. Zhang, S.; Lee, J.H. Analysis of the main consensus protocols of blockchain. *ICT Express* **2020**, *6*, 93–97. [CrossRef]
9. Shahnaz, A.; Qamar, U.; Khalid, A. Using blockchain for electronic health records. *IEEE Access* **2019**, *7*, 147782–147795. [CrossRef]
10. Bell, K.M. *HHS National Alliance for Health Information Technology (NAHIT)*; Report to the Officer of the National Coordinator for Health Information Technology on Defining Key Health Information Technology Terms: DC, USA, 2008; pp. 1–40. Available online: http://tigerstandards.pbworks.com/f/HITTermsFinalReport.pdf (accessed on 9 September 2024).
11. Golosova, J.; Romanovs, A. The advantages and disadvantages of the blockchain technology. In Proceedings of the 2018 IEEE 6th Workshop on Advances in Information, Electronic and Electrical Engineering (AIEEE), Vilnius, Lithuania, 8–10 November 2018; pp. 1–6.

12. Al-Breiki, H.; Rehman, M.H.U.; Salah, K.; Svetinovic, D. Trustworthy blockchain oracles: Review, comparison, and open research challenges. *IEEE Access* **2020**, *8*, 85675–85685. [CrossRef]
13. Ahram, T.; Sargolzaei, A.; Sargolzaei, S.; Daniels, J.; Amaba, B. Blockchain technology innovations. In Proceedings of the 2017 IEEE Technology & Engineering Management Conference (TEMSCON), Evanston, IL, USA, 28 June–1 July 2017; pp. 137–141.
14. Breidenbach, L.; Cachin, C.; Chan, B.; Coventry, A.; Ellis, S.; Juels, A.; Koushanfar, F.; Miller, A.; Magauran, B.; Moroz, D.; et al. Chainlink 2.0: Next steps in the evolution of decentralized oracle networks. *Chainlink Labs*. 2021. Available online: https://research.chain.link/whitepaper-v2.pdf (accessed on 9 September 2024).
15. Hepp, T.; Sharinghousen, M.; Ehret, P.; Schoenhals, A.; Gipp, B. On-chain vs. off-chain storage for supply-and blockchain integration. *it-Inf. Technol.* **2018**, *60*, 283–291. [CrossRef]
16. Hillestad, R.; Bigelow, J.; Bower, A.; Girosi, F.; Meili, R.; Scoville, R.; Taylor, R. Can electronic medical record systems transform health care? Potential health benefits, savings, and costs. *Health Aff.* **2005**, *24*, 1103–1117. [CrossRef] [PubMed]
17. Watanabe, H.; Fujimura, S.; Nakadaira, A.; Miyazaki, Y.; Akutsu, A.; Kishigami, J. Blockchain contract: Securing a blockchain applied to smart contracts. In Proceedings of the 2016 IEEE International Conference on Consumer Electronics (ICCE), Las Vegas, NV, USA, 7–11 January 2016; pp. 467–468.
18. Zheng, Z.; Xie, S.; Dai, H.N.; Chen, W.; Chen, X.; Weng, J.; Imran, M. An overview on smart contracts: Challenges, advances and platforms. *Future Gener. Comput. Syst.* **2020**, *105*, 475–491. [CrossRef]
19. Hewa, T.; Ylianttila, M.; Liyanage, M. Survey on blockchain based smart contracts: Applications, opportunities and challenges. *J. Netw. Comput. Appl.* **2021**, *177*, 102857. [CrossRef]
20. Wang, S.; Ouyang, L.; Yuan, Y.; Ni, X.; Han, X.; Wang, F.Y. Blockchain-enabled smart contracts: Architecture, applications, and future trends. *IEEE Trans. Syst. Man Cybern. Syst.* **2019**, *49*, 2266–2277. [CrossRef]
21. Cheikhrouhou, O.; Mershad, K.; Jamil, F.; Mahmud, R.; Koubaa, A.; Moosavi, S.R. A lightweight blockchain and fog-enabled secure remote patient monitoring system. *Internet Things* **2023**, *22*, 100691. [CrossRef]
22. Kumar, R.; Marchang, N.; Tripathi, R. Distributed off-chain storage of patient diagnostic reports in healthcare system using IPFS and blockchain. In Proceedings of the 2020 IEEE International Conference on Communication Systems & Networks (COMSNETS), Bengaluru, India, 7–11 January 2020; pp. 1–5.
23. Fernando, E.; Cassandra, C. Medicine Information Record Based on Blockchain Technology. In Proceedings of the 2021 2nd IEEE International Conference on Innovative and Creative Information Technology (ICITech), Virtual, 23–25 September 2021; pp. 169–173.
24. Ante, L. Smart contracts on the blockchain–A bibliometric analysis and review. *Telemat. Inform.* **2021**, *57*, 101519. [CrossRef]
25. Ashfaq, T.; Khalid, M.I.; Ali, G.; Affendi, M.E.; Iqbal, J.; Hussain, S.; Ullah, S.S.; Yahaya, A.S.; Khalid, R.; Mateen, A. An efficient and secure energy trading approach with machine learning technique and consortium blockchain. *Sensors* **2022**, *22*, 7263. [CrossRef]
26. Hadi, H.J.; Hayat, U.; Musthaq, N.; Hussain, F.B.; Cao, Y. Developing Realistic Distributed Denial of Service (DDoS) Dataset for Machine Learning-based Intrusion Detection System. In Proceedings of the 2022 9th IEEE International Conference on Internet of Things: Systems, Management and Security (IOTSMS), Milan, Italy, 29 November–1 December 2022; pp. 1–6.
27. Hadi, H.J.; Cao, Y.; Li, S.; Xu, L.; Hu, Y.; Li, M. Real-time fusion multi-tier DNN-based collaborative IDPS with complementary features for secure UAV-enabled 6G networks. *Expert Syst. Appl.* **2024**, *252*, 124215. [CrossRef]
28. Ali, G.; Ahmad, N.; Cao, Y.; Ali, Q.E.; Azim, F.; Cruickshank, H. BCON: Blockchain based access CONtrol across multiple conflict of interest domains. *J. Netw. Comput. Appl.* **2019**, *147*, 102440. [CrossRef]
29. Sammeta, N.; Parthiban, L. Data Ownership and Secure Medical Data Transmission using Optimal Multiple Key-Based Homomorphic Encryption with Hyperledger Blockchain. *Int. J. Image Graph.* **2021**, *23*, 2240003. [CrossRef]
30. Saba, T.; Haseeb, K.; Rehman, A.; Jeon, G. Blockchain-Enabled Intelligent IoT Protocol for High-Performance and Secured Big Financial Data Transaction. *IEEE Trans. Comput. Soc. Syst.* **2024**, *11*, 1667–1674. [CrossRef]
31. Fatima, N.; Agarwal, P.; Sohail, S.S. Security and privacy issues of blockchain technology in health care—A review. *ICT Anal. Appl.* **2022**, *314*, 193–201.
32. Babar, M.; Qureshi, B.; Koubaa, A. Investigating the impact of data heterogeneity on the performance of federated learning algorithm using medical imaging. *PLoS ONE* **2024**, *19*, e0302539. [CrossRef] [PubMed]
33. Butt, A.U.R.; Mahmood, T.; Saba, T.; Bahaj, S.A.O.; Alamri, F.S.; Iqbal, M.W.; Khan, A.R. An Optimized Role-Based Access Control Using Trust Mechanism in E-Health Cloud Environment. *IEEE Access* **2023**, *11*, 138813–138826. [CrossRef]
34. Khan, I.; Qazi, E.A.; Hadi, H.J.; Ahmad, N.; Ali, G.; Cao, Y.; Alshara, M.A. Securing Blockchain-Based Supply Chain Management: Textual Data Encryption and Access Control. *Technologies* **2024**, *12*, 110. [CrossRef]
35. Mohanta, B.K.; Panda, S.S.; Jena, D. An overview of smart contract and use cases in blockchain technology. In Proceedings of the 2018 9th IEEE International Conference on Computing, Communication and Networking Technologies (ICCCNT), Bengaluru, India, 10–12 July 2018; pp. 1–4.