JCBD
JOURNAL OF COMPUTERS AND DIGITAL BUSINESS
Volume 4, Number 2, May 2025, pp. 90-99
Homepage: https://jurnal.delitekno.co.id/index.php/jcbd

Blockchain for Secure Electronic Health Records Management

Praveen Kumar Rawat

Independent Researcher Master's in Computer Applications, PAHM, PSM, ISTQB, MCDBA, Virginia

praveen.rawat1@gmail.com

https://doi.org/10.56427/jcbd.v4i2.764

ARTICLE INFO

Article History
Received: March 12, 2025
Revised: May 22, 2025
Accepted: May 28, 2025

Keywords
Blockchain
EHR Data Security
Smart Contracts
Decentralized Ledger
Healthcare Interop

ABSTRACT

Electronic Health Records (EHRs) are essential to modern healthcare infrastructure, yet they face persistent challenges related to data security, interoperability, and unauthorized access. Blockchain technology, through its use of cryptographic protocols, smart contracts, and consensus mechanisms, offers a decentralized and tamper-resistant solution for managing EHRs. This paper explores the potential of blockchain in addressing critical limitations of conventional EHR systems, focusing on data immutability, fine-grained access control, and real-time data synchronization. By leveraging distributed ledger technology (DLT), the proposed approach reduces single points of failure and mitigates cybersecurity vulnerabilities. Furthermore, this study discusses practical implementations and case studies that demonstrate how blockchain can enhance trust, transparency, and efficiency in health data management. The analysis reveals that a well-designed blockchain-based EHR framework can minimize data breaches, protect patient data rights, and streamline operations across healthcare institutions. The paper concludes that blockchain presents a transformative path forward for secure and interoperable EHR systems. Future research should aim to refine architectural models, address scalability constraints, and ensure compliance with healthcare regulations and standards such as HIPAA and GDPR to facilitate real-world adoption and sustainable deployment.

1. Introduction

Blockchain technology has come forth as an up-to-date perfect solution to the problems faced in managing Electronic Health Records (EHRs). In today's healthcare scenario, EHR systems are very vital in storing and managing patient information, including medical histories, diagnoses, treatment, and laboratory results [1]. It allows easy communication and collaboration among health workers and a continuity of care toward better patient outcomes. But centralized data storing is the basis for most traditional EHR systems, subjecting them to cyberattacks, unauthorized access, manipulation of data, and single point failures. Interoperability issues between healthcare institutions may pose impediments to the seamless transmission of patient data, resulting in fragmented delivery of care and increasing the likelihood of medical errors [2][3][4].

The secure management of EHRs appears to have a very promising way using the decentralized, transparent, and immutable aspects of blockchain technology. Blockchain is defined as a distributed ledger where data is distributed to multiple nodes in a peer-to-peer fashion. Each block in the chain contains the cryptographic hash of the preceding block, its timestamp, and the transaction data, thus providing tamper-proofing and integrity of data. The copies of the transaction records that are part of this transaction remain synchronized and agreed-upon using consensus mechanisms like Proof of Work (PoW) or Proof of Stake (PoS). The utmost advantage of blockchain in electronic health record management is security. The cryptographic algorithms take care of encrypting patient data and allowing access to the authorized parties. The very essence of blockchain is that it provides an immutable ledger, thereby ensuring transparency and auditability in transaction tracking. In this way, stakeholders can easily trace when data was accessed or

changed, ensuring accountability and perpetually minimizing the chance of manipulation or fraud. On the other hand, blockchain also gives way to a fine-grained type of access control whereby patients remain in control of their data and may grant permission to any provider or researcher or insurer they see fit.

Conventional EHR systems are often closed in some ways, which interferes with relevance for some and usefulness for others when discussing matters related to patient information. Standardized protocols and smart contracts on the blockchain allow data sharing to flow comfortably between the many players while ensuring the integrity and security of the data shared. Moreover, with the help of smart contracts, several administrative functions such as consent management, processing of insurance claims, and medical billing can be automatically executed. The reduction of intermediaries and the automation of administrative processes promote operational efficiency and reduce costs in healthcare [5].

Moreover, the blockchain has materialized as a trustworthy technology for ensuring that medical research and clinical trials are authentic. Data from the research are directly recorded on the blockchain and timestamped. This means that anyone interested can verify the authenticity and accuracy of the findings, thus preventing manipulation of data and achieving impeccable transparency. This holds great importance for the pharmaceutical industry since accurate documentation of study is from the perspective of regulatory compliance and drug development. Although there are great potential benefits to the use of blockchains in the management of electronic health records, issues of real significance must still be resolved to achieve vast acceptance in the industry. An example that looms large in this regard is the scalability of blockchain systems, where it has been found that blockchain networks process transactions slower and require more powerful computers than their centralized counterparts. Optimizing transaction processing and reducing resource consumption through technologies, such as sharding, layer-two protocols, and permissioned blockchains, can eliminate some of these concerns. Another necessary step is to build regulatory and legal frameworks, so that one can be compliant with data protection laws: for instance, in the United States, the Health Insurance Portability and Accountability Act (HIPAA), and in the European Union, the General Data Protection Regulation (GDPR). Cooperation between policymakers, healthcare organizations,and technology providers must be created to deliver a blockchain path that is safe and compliant.

The derived findings are useful in terms of strengths and weaknesses of each gateway, to assist organizations in making suitable choices according to their security needs. Major Contributions are.

a. Decentralized and immutable nature of blockchain keeps EHR data intact from being changed or tampered with. Cryptographic hashing as well as consensus mechanisms place high-quality security on reducing the risk of data breaches, cyberattacks, and unauthorized access.
b. A patient has a greater involvement in his/her medical records with blockchain-based systems. Blockchain-enabled permission management will allow him/her to give access to/for both granting and revoking such permission for holding EHR data in a transparent and private manner.
c. Blockchain allows the adoption of standard protocols as well as smart contracts among the health care providers, insurance companies, and researchers to improve safe data sharing and thus further boost collaboration and reduce duplication of tests. Thus, patient care improves.

2. Research Methodology

The suggested Electronic Health Record (EHR) system employs a robust architecture comprising several advanced technological components, each designed to enhance the security, efficiency, and reliability of medical data management. At the core of this system lies blockchain technology, which ensures the integrity, confidentiality, and interoperability of patient health information by leveraging immutable distributed ledgers and decentralized consensus protocols. The proposed EHR model (Figure 1) integrates multiple functional layers, including secure data storage, access control mechanisms, smart contracts, and encrypted communication channels. These components work together to establish a tamper-proof environment that facilitates real-time access to medical records only by authorized stakeholders. The system also supports seamless data sharing across institutional boundaries while maintaining compliance with regulatory standards. The major functional categories and their roles within the architecture are illustrated in Figure 1.

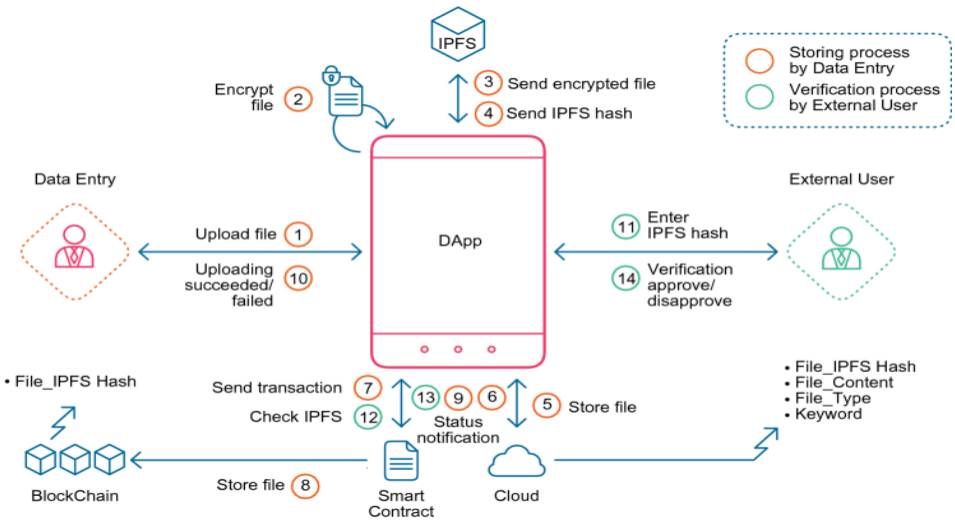


Figure 1. Suggested EHR structure.

Patient: Patients may add their medical information to a blockchain network via a website. Digital technology makes it feasible for sensors and intelligent gadgets to work together blockchain agreements. Many healthcare organizations [11] often have disparate digital health records. A healthcare facility may use the blockchain design and gadget surveillance to plan its facilities throughout their life cycle. Medical records of patients may be viewed and uploaded by hospitals. The website allows them to view and contribute material. Website: The website allows users to expressly agree to the exchange of their healthcare details with other people or health care professionals, as well as to be informed when these documents are changed. Additionally, patients may choose to set temporal limits on the duration that any third Participants may share their medical details with the internet site and view their personal data. Blockchain creates a centralized database for regularly updated medical records that authorized users may safely store and access with ease. Reducing misunderstandings between different medical experts engaged in treating the same patients may save countless mistakes, enable quicker medical evaluation and therapy, and allow for more individualized care. We outline the main elements of the proposed blockchain-based infrastructure, which employs IPFS for maintaining records and the network's decentralized structure. Utilizing blockchain, Ethereum, IPFS, and intelligent contracts, this structure enables safe and user-friendly transactions for health data administration. All components must be registered on the blockchain, with the exception of IPFS storing.

Hospitals: A special identification created by the hospital is linked to each patients document, creating an extensive dossier from the first to the last interaction. The transmission of these records is supervised by the hospital. Patients may view their health records online if a third party joins. They are responsible for carrying out the smart contracts in accordance with the guidelines set out when their most current data and histories of illness were submitted. In the decentralized architecture, patient records are encoded and distributed through the peer to peer network from a multitude of servers with the usage of IPFS. Each record is associated with a unique hash registered on distributed ledger ensuring that privacy of the information remains intact. Smart contracts built by Solidity are essential to the architects of-solutions as they smooth the core processes like authentication, verification of data and adherence to legislation. Active permission and sharing of data by patients are monitored by these intelligent contracts. This is made possible because they are empowered to actively control and monitor who can access their health records in real time. The architecture is flexible and pliable with high-end security systems for preventing the invasion and easily integrates with existing EHRs. Hardhat provides the development platform that offers quick deployment, testing, and debugging of smart contracts for all use cases above: Below, the main features of this design follow:

2.1 Hardhat (Blockchain Development & Smart Contracts)

Hardhat is a development environment for Ethereum-based blockchain applications, particularly smart contracts. Academically, it is chosen for several reasons:

a. **Robust Testing Framework** – Hardhat provides built-in testing capabilities, allowing researchers to rigorously validate smart contract logic before deployment.
b. **Advanced Debugging Tools** – It offers detailed stack traces and error messages, making it easier to analyze and optimize smart contracts.

c. **Flexible Development Workflow** – Hardhat supports custom scripts and plugins, enabling researchers to experiment with different blockchain configurations.
d. **Simulation of Blockchain Networks** – It allows local blockchain simulations, reducing the need for costly testnet transactions.
e. **Integration with Solidity** – As Solidity is the primary language for Ethereum smart contracts, Hardhat provides seamless integration, making it ideal for academic blockchain research.

2.2 Next.js (Frontend Framework for Web Applications)

Next.js is a React-based framework that enhances web application development. Academically, it is preferred due to:

a. **Server-Side Rendering (SSR) & Static Site Generation (SSG)** – These features improve performance and SEO, making it suitable for research projects requiring efficient data retrieval.
b. **Optimized Performance** – Next.js reduces load times and enhances user experience, which is crucial for real-time applications like blockchain-based EHR systems.
c. **Built-in API Routes** – It simplifies backend integration, allowing researchers to create APIs without needing a separate backend framework.
d. **Security & Scalability** – Next.js provides automatic security updates and optimizations, ensuring a secure and scalable web application.
e. **Ease of Integration with Blockchain** – It works well with Web3 libraries, enabling seamless interaction with smart contracts.

2.3 IPFS (InterPlanetary File System for Decentralized Storage)

IPFS is a decentralized storage protocol that enhances data security and accessibility. Academically, it is chosen for:

a. **Decentralized Data Storage** – Unlike traditional centralized databases, IPFS distributes data across a peer-to-peer network, reducing single points of failure.
b. **Data Integrity & Immutability** – IPFS uses cryptographic hashing to ensure that stored data remains unchanged, which is critical for secure EHR management.
c. **Efficient Content Addressing** – Instead of location-based addressing, IPFS retrieves files based on their unique hash, improving data retrieval efficiency.
d. **Interoperability with Blockchain** – IPFS can store large files off-chain while maintaining references on-chain, optimizing blockchain storage costs.
e. **Scalability & Redundancy** – IPFS ensures data redundancy across nodes, making it resilient to failures and cyberattacks.

This will be the complete development toolkit for Ethereum apps, all the while Hardhat will do the lifecycle management of the app right from programming to deployment. It brings together everyone needed for Ethereum app compilation, deployment, running, and debugging all at once to ensure that developers do not have to start from square one. Network management, error messaging, and live console logging will transform the way we do development thanks to these advanced features of Hardhat. This includes our use with the Inter-planetary Files system to answer the problems arising from a decentralized data storage: Peering–two-point in network topology lightens a lot of opportunities for nodes that are far apart because it increases the availability of such nodes. All private files in IPFS are addressed using a hash a unique one with respect to their data, thus ensuring the integrity of such data and fast access without indexation to the centralized server.

Smart Contracts: These intelligent contracts designed in Solidity are the keystones of our EHR fabric, automating crucial functions like access control, verification of information, and audit trails. These contracts are very carefully drafted to comply with very strict privacy laws by integrating procedures in patient consent management and ensuring that access to healthcare data is granted only by the very express consent of the patient.

Furthermore, Figure 2 summarizes the full system process and shows the procedure in our suggested architecture. The procedure is started by the user, or patient, who registers and uploads their medical data. A special key is then used to access these files. This special key is entered into the computer program by the individual using it and the hospital, enabling them to access the particular data or record. Establishing accounts, uploading information, and accessing or viewing the information among the main tasks for both individuals and hospitals. Blockchain innovation serves as the foundation and controller for the whole system.

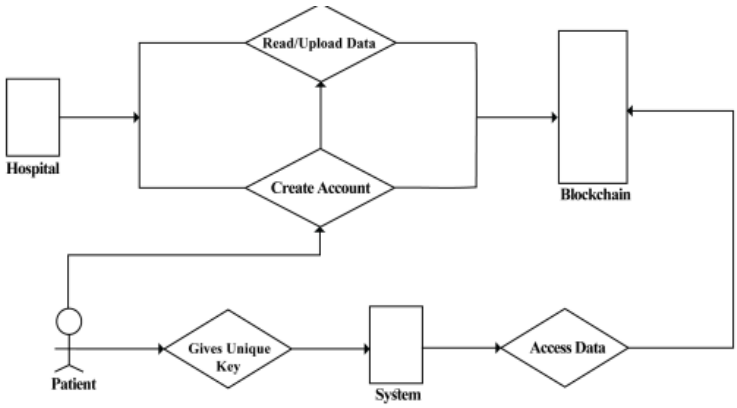


Figure 2. Patient/Hospital Registration.

2.4 Comprehensive Information Interaction and Processing

a. Relations between hospitals and patients: Hospitals start the procedure by safely uploading patient information that has been encoded to IPFS. They then store the appropriate information hashes on the blockchain. This guarantees that patient data will always be traceable and unchangeable. Through a specialized interface, individuals control who can look at their data. They can give or deny authorization in real time, and each transaction is permanently recorded on the blockchain [12].
b. Information Retrieval and Access: Through a web portal linked to the platform, individuals manage who may view their information. Real-time access granting and revocation is possible, and every exchange is transparently documented on a distributed ledger. To ensure that patients retain ownership over their confidential information, hospitals and authorized healthcare workers may seek permission to examine information about patients, but their request must be accepted by the individual being treated. The necessary information is encrypted and shown on the approved devices when other healthcare facilities seek a patient's record. The health care provider can get the patient's public key, yet only the patient's doctor can access their secret key.
c. A doctor cannot see a patient's medical records without the patient's consent.

Whenever the client receives the queue notification on the physician's site or smartphone application, they may grant admission by inputting their special key.

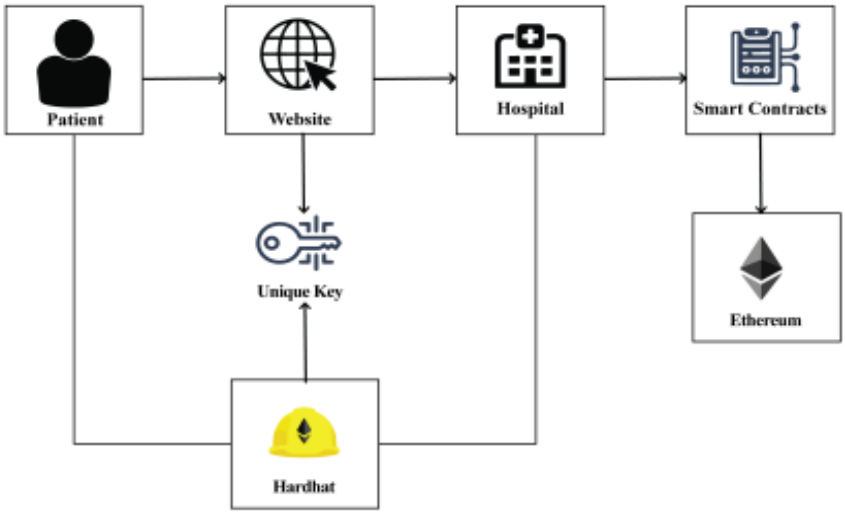


Figure 3. Patient EHR preservation and retrieving.

2.5 The Suggested Framework's Networking Architecture

Figure 4 displays the network architecture of the suggested architecture. Through the software or website, the patient may view their EHRs. In the same way, the hospital retrieves and publishes data from a system or site. Information about the system is stored on IPFS. Blockchain serves as the Hardhat framework's node, which the whole thing then accesses further. Ethereum is used to implement all smart agreements [13].

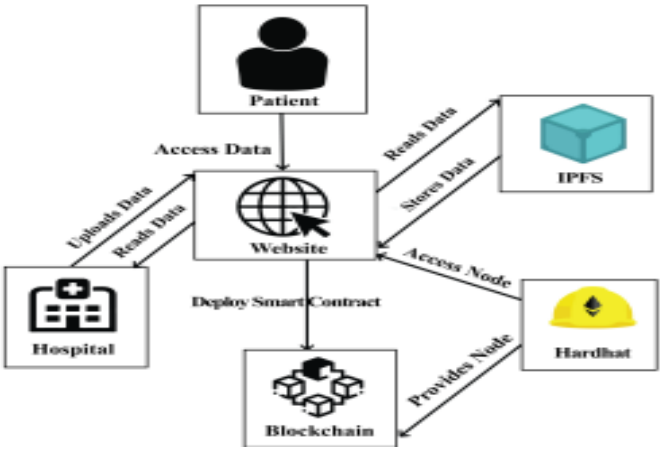


Figure 4. The suggested framework's network organization.

2.6 Implementing the Suggested Structure

This section explores the nature and many kinds of electronic contracts in order to examine the fundamental idea behind them. We chose a fictitious healthcare record software from Github because to its popularity, favorable ratings, and appropriateness. We then used this fictitious health record to put the suggested structure into practice. We used NextJS for frontend frameworks creation, JavaScript for the backend, and Solidity for intelligent contract execution. Many different kinds of variables, including diagrams, were used in the intelligent contract design. Key-value pair architectures, or mappings, are very common in the creation of smart contracts. The "Patient" map in our smart contract maintains records of all patients who have registered by designating their status as "true" & saving their home addresses as keys. Additionally, we introduced the DateTime function to guarantee that every freshly produced and mine brick has its own time. Block hashes might be needed to make this feature useful: it can be hidden under a hash.

Moreover, we used structs, which is one of the most common and widely used structure of data in computer programming languages. The smart contract is consists of three structural types: "Records," "Diagnosis," and "dateRange." The "Records" structure of data adds flexibility and usability in different environments by simply holding the health information of a person into a single variable. Furthermore, a unique class of functions known as modifier is essential for limiting usage of sure functionality. They serve as gatekeepers, making sure that only users with permission may utilise certain features.

Hospitals are added to the smart contract using the addHospitalI() method. This implies that the medical files of the patient may be revised and that they can be transferred to a clinic. When a hospital is added on the digital ledger, the patient's information along with their address remain attached to that hospital indefinitely. It is important to remember that a patient may be connected to more than one hospital. The system is going to verify the individual's accessibility and, if required, reverse the purchase with an error if the same patient is attempted to be added again. In the mean time, the "addRecord()" method is made to take in a number of arguments, such as your name, your facility address, date of admission, date of departure, cause for visit, and diagnosis. The mapping known as recordings then has all of this information. Figure 5 depicts the format of the blockchain-stored patient information. An occurrence is triggered upon the correct placement of information, and the "record quantity" variable increases by 1.


```json
{
  "providedName": true,
  "name": "NoorulAain",
  "patient": "0x9965507d1a55bc2695c58ba16f837d819804Adc",
  "hospital": "0x3C44CdDd0Ga900fa2b585dd299e03d12FA42930C",
  "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: Blockchain-based patient record architecture.

Two arguments must be supplied to the "getOwnRecord()" function: the address of the individual and the "recordID." Our intelligent contract uses these two variables to obtain and show the individual the information that exists on the digital ledger. It is crucial to remember that doctors and other patient do not have accessibility to this feature; only patients individually are authorized to use it. The patient's medical records are kept secure and confidential thanks to this limitation. Institutions that are registered may use the "getRecord()" method. Access to all pertinent information in the medical records of the individual is made possible when this feature is used. This feature adds a useful dimension to its usefulness by allowing retrieval of information within a certain period range. Furthermore, the "recordExists" and "onlyHospital" modifications are used to restrict utilisation of patient data.

Additionally, there is a method in the Solidity languages called "Object()" that is used only once before installation. Its goal is to establish the smart contract state's starting value. The standard "Object()" procedure is produced by a compiler when "Object()" is not declared manually.

3. Results and Discussion

3.1 Prototype Configuration

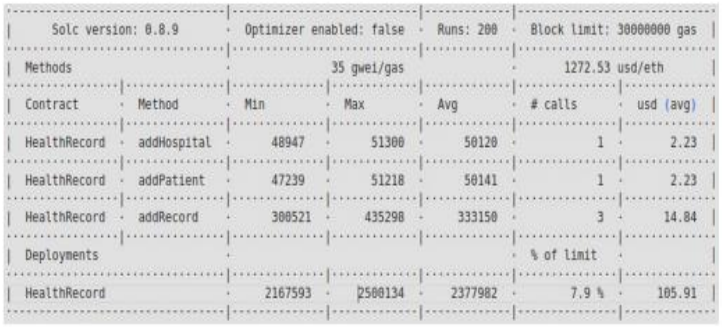
To evaluate the effectiveness of the suggested structure, we experimented with the mentioned tools—the Hardhat a framework, Mocha, and Chai Testing Structures, GasReport, Javascript, and Solidity are designed to Covering. Using the Hardhat programming structure, the suggested smart contracts were created. The suggested smart contract design was tested using two well-known JavaScript development structures, Mocha and Chai. Ether provides a programming language called Solidity, which is encased in Python and JavaScript, for creating code for intelligent contracts. To comprehend the automated processing and safe administration of information, it is essential to know the specifics of a smart contract application employed in our research. The first appendix contains the whole Solidity code for this intelligent contract, which offers an extensive examination of the data structure and functionalities used.

This study distinguishes itself from other research by focussing especially on the unique features of the proposed paradigm. These aspects include the use of IPFS for distributed data storage, the integration of intelligent contracts for automatic rule compliance, and the development of a cost-effective, scalable system that addresses the susceptibility of centralised systems to one centre of failure. By contrasting our results with those of earlier research, the paper skilfully illustrates how our study advances the subject and offers fresh perspectives that were not previously examined in other investigations.

3.2 Cost Assessment

The suggested blockchain-based system for managing patient health information underwent a thorough cost study. To assess the efficacy and accuracy of our suggested solution, we conducted a cost analysis. Since it influences the reward that miners normally get after executing the functions, Solidity function efficiency is significant. Miners may determine how to operate costs depending on data categories and movement volumes by attentively observing the tasks performed during the time a function was execution. Two smart contracts were created, deployed, and run as part of our testing. **See Figure 6** for transaction charges, miner expenditures, and the conversion procedure into USD. Additionally, it offers a comprehensive analysis of the overall costs associated with setting up and running the Ethereum blockchain's EHR smart contract. The image is crucial because it highlights the cost-effectiveness of the suggested solution when compared to conventional electronic health records and graphically depicts the monetary implications of implementing the structure. Understanding the framework's economic feasibility and scalability is made easier thorough examination of costs, particularly for large healthcare systems.

Two hospitals and two clients were included initially. Tests and assessments were conducted on additional patients and facilities for evaluation reasons. The costs of each of the functions that were used to determine their respective prices were 2.49 USD for the "AddHospital" function, 2.49 USD for the "AddPatient" function, and 16.54 USD for the "AddRecords" function. The smart contract, also known as the health information contract, required 136.96 USD to implement.



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: Full EHR smart contract analysis

The variables utilised in the smart agreement are summarised in the table listed below

3.3 Analysis of Cost per Transaction

We use the Hardhat-provided faucet Eth to conduct smart contract operations on the the Hard Hat networks. The primary functions' costs are shown in Table 1. The "addHospital()" method, for instance, makes it easier for medical information to be sent from the healthcare facility to the intelligent contract. Likewise, a patient's record is created on the blockchain using the "addRecord()" method, and it is retrieved using the "getOwnRecord()" function. Furthermore, the main method in charge of retrieving the number of patients during a certain time period is "getCurrentPatients()." Finally, "getRecord()" is used to get all patient records from the bitcoin blockchain.

Table 1. An overview of the EHR smart document's features

Variable Name	Data Type	Scope	Description
id	uint	Hospital/Patient	Unique identifier for a hospital or patient.
name	string	Hospital/Patient	Represents the name of a hospital or patient.
location	string	Hospital	Specifies the geographical location of the hospital.
phone	string	Hospital	Stores the contact phone number of the hospital.
email	string	Hospital	Stores the contact email address of the hospital.
age	uint	Patient	Holds the age value of a patient.
gender	string	Patient	Stores the gender information of a patient.
diagnosis	string	Patient	Describes the medical diagnosis of a patient.
exists	bool	Hospital	Boolean flag indicating whether a hospital exists.
hospitals	mapping(uint => Hospital)	HospitalRegistry	Data structure mapping hospital ID to hospital details.
hospitalPatients	mapping(uint => mapping(uint => Patient))	HospitalRegistry	Nested mapping associating hospital ID with patient details.
patientCount	mapping(uint =>uint)	HospitalRegistry	Counter tracking the number of patients per hospital.
hospitalCount	uint	HospitalRegistry	Counter tracking the number of hospitals added.

3.4 Medical Data Filing Cost Evaluation

The smart contract's "addRecord()" method is in charge of appending the medical information of a patient to the blockchain. Table 2 shows that the median cost evaluation for using the "addRecord()" method is 16.53 USD. Figure 7 displays the health information licensing cost assessment. These particularly look at the cost evaluation of medical records registration, emphasizing the costs related to putting patient data on the blockchain. This is critical because it demonstrates the framework's cost-effectiveness in handling large volumes of facts, which is an important consideration for any healthcare service that handles a large volume of information relating to patients. They support the assertion that the suggested architecture may efficiently manage patient data at minimal cost while maintaining its confidentiality and integrity by giving a clear picture of the specific costs involved.

Table 2: An overview of the exchange of fees associated with every hospital registration intelligent contract operation.

Function	Gas Consumption (Ether)	Gas Cost (Gwei)	Estimated Cost (USD)	Operation Type
addHospital	0.000204696 Ether	204,696Gwei	$0.812	State Modification
removeHospital	0.000103356 Ether	103,356Gwei	$0.412	State Modification
addPatient	0.000183084 Ether	183,084Gwei	$0.698	Data Insertion
addPatientRecord	0.000122658 Ether	122,658Gwei	$0.467	Data Modification
getPatientRecord	0.000061788 Ether	61,788Gwei	$0.232	Read Operation
removePatientRecord	0.000084224 Ether	84,224Gwei	$0.327	State Modification

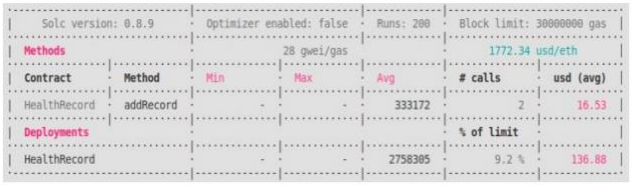


Figure 7. Evaluation of the registration costs for health information.

4. Conclusion

The proposed blockchain-based framework for Electronic Health Record (EHR) management demonstrates a secure, efficient, and scalable solution for addressing the challenges in traditional healthcare data management systems. By integrating blockchain with InterPlanetary File System (IPFS) and smart contracts, the framework ensures the confidentiality, integrity, and availability of sensitive patient data. The decentralized nature of blockchain minimizes reliance on centralized systems, reducing the risk of data breaches and unauthorized access. Role-based access control mechanisms enable patients to manage and control their medical data while allowing health care providers access to the data needed for accurate diagnosis and treatment. Smart contracts further automate a lot of essential work, including data verification, regulatory compliance, and maintenance of audit trails as a measure of enhanced transparency. Immutability and strong performance of the application are ensured through the use of Solidity for smart contract development and Hardhat for deployment and testing. The cost analysis portrays the proposed system as an affordable alternative to conventional EHR management systems. Transaction and operational costs remain affordable, making this solution viable for greater adoption across different healthcare settings. Furthermore, IPFS, while limiting the interactive performance, provides for storing and retrieving data in a bulk environment. To summarize, the blockchain EHR management system enhances security in healthcare data sharing, provides legal ownership to patients, and ensures seamless interoperability amongst healthcare institutions. This framework establishes a meaningful step toward a secure and patient-centric healthcare ecosystem, proposing data privacy, reduction in administrative overheads, and support to informed medical decision-making. Future work could consider further strengthening the resilience and adaptability of the proposed solution exploring enhancements to the consensus algorithm, additional privacy layers, and interoperability with novel healthcare technologies.

References

[1] Somepalli, S. Navigating Enterprise Software Implementation: Emphasizing the Superiority of Hybrid Methodologies Over Strict Agile or Waterfall Approaches.

[2] Deepa D, Jain G. Assessment of periodontal health status in postmenopausal women visiting dental hospital from in and around Meerut city: Cross-sectional observational study. J Midlife Health. 2016 Oct-Dec;7(4):175-179. doi: 10.4103/0976-7800.195696.

[3] Bansal, A. (2024). Enhancing Business User Experience: By Leveraging SQL Automation through Snowflake Tasks for BI Tools and Dashboards. *ESP Journal of Engineering & Technology Advancements (ESP-JETA)*, *4*(4), 1-6.

[4] Barma MD, Muthupandiyan I, Samuel SR, Amaechi BT. Inhibition of Streptococcus mutans, antioxidant property and cytotoxicity of novel nano-zinc oxide varnish. Arch Oral Biol. 2021 Jun;126:105132. doi: 10.1016/j.archoralbio.2021.105132. Epub 2021 Apr 23.

[5] Bharathy, S. S. P. D., Preethi, P., Karthick, K., & Sangeetha, S. (2017). Hand Gesture Recognition for Physical Impairment Peoples. *SSRG International Journal of Computer Science and Engineering (SSRG-IJCSE)*, 6-10.

[6] Vora, J., Nayyar, A., Tanwar, S., Tyagi, S., Kumar, N., Obaidat, M. S., & Rodrigues, J. J. (2018, December). BHEEM: A blockchain-based framework for securing electronic health records. In 2018 IEEE Globecom Workshops (GC Wkshps) (pp. 1-6). IEEE.

[7] Mahore, V., Aggarwal, P., Andola, N., & Venkatesan, S. (2019, December). Secure and privacy focused electronic health record management system using permissioned blockchain. In 2019 IEEE conference on information and communication technology (pp. 1-6). IEEE.

[8] Mondal, S., Shafi, M., Gupta, S., & Gupta, S. K. (2022). Blockchain based secure architecture for electronic healthcare record management. GMSARN Int. J, 16(4), 413-426.

[9] Zaabar, B., Cheikhrouhou, O., Jamil, F., Ammi, M., & Abid, M. (2021). HealthBlock: A secure blockchain-based healthcare data management system. Computer Networks, 200, 108500.

[10] Verma, G., Pathak, N., & Sharma, N. (2021, August). A secure framework for health record management using blockchain in cloud environment. In Journal of Physics: Conference Series (Vol. 1998, No. 1, p. 012019). IOP Publishing.

[11] Somepalli, S. Safeguarding Patient Trust: The Importance of COI, COC, and Configurable MES in CGT.

[21] Rekha, P., Saranya, T., Preethi, P., Saraswathi, L., &Shobana, G. (2017). Smart agro using arduino and gsm. International Journal of Emerging Technologies in Engineering Research (IJETER) Volume, 5.

[13] Somepalli, S. (2021). Innovations in Water and Wastewater Management: Ensuring Sustainable Water Resources. European Journal of Advances in Engineering and Technology, 8(4), 76-81.