ISSN: 0067-2904

A Development of an IoT-Based Healthcare System Using Distributed Blockchain Technology

Raad Mohammed[1], Raaid Alubady[1,2*], Ali Al-Sherbaz[3]
[1]*College of Information Technology, University of Babylon, Babylon, Iraq*
[2]*Technical Engineering College, Al-Ayen University, Thi-Qar, Iraq*
[3]*College of Science, University of Northampton, UK*

Received: 20/12/2022 Accepted: 21/9/2023 Published: 30/9/2024

Abstract

Modern healthcare systems have two main sections: electronic health records (EHRs) and remote patient monitoring (RPM). With these sections, several issues have arisen related to comprehensiveness, which these sections lack in covering all the data information about the patient's state, and other issues in privacy and security. Many types of research have been presented to solve these issues or mitigate their effects. One of these ideas is to use blockchain technology. However, some research may not utilize this technique properly, where data comprehensiveness problems remain and the preservation of individual nodes' data may be compromised. This paper introduces a new system called Patient Health Record (PHR), which will provide comprehensive patient health data and IoT device reading. PHR data is protected using a multi-peer, distributed, and decentralized system based on blockchain. A smart contract has been created for the system through which the PHR data will be cached in the blockchain. It is deployed on the Proof of Authority network to increase the speed of transaction processing, which is important in processing data for IoT devices. The proposed system is evaluated according to a range of metrics (cost, immutability, data storage, and estimated time) to validate the system.

Keywords: Internet of Things, Healthcare Systems, Blockchain Technology, Smart Contract, Ethereum.

تطوير أنظمة رعاية صحية قائمة على إنترنت الأشياء باستعمال تقنية بلوك جين الموزعة

رعد محمد[1], رائد العبيدي[1،2،*], علي شيرباز [3]
[1]كلية تكنولوجيا المعلومات, جامعة بابل, بابل, العراق
[2]الكلية التقنية الهندسية, جامعة العين, ذي قار , العراق
[3]كلية العلوم, جامعة نورثهامبتون, المملكة المتحدة

الخلاصة

تشتمل أنظمة الرعاية الصحية الحديثة على قسمين رئيسيين ، السجلات الصحية الإلكترونية (EHRs)، ومراقبة المريض عن بُعد (RPM). لذلك، ظهرت العديد من القضايا المتعلقة بالشمولية، والتي تفتقر إليها هذه الأقسام في تغطية جميع معلومات البيانات حول حالة المريض، وغيرها من القضايا المتعلقة بالخصوصية

*Email: alubadyraaid@itnet.uobabylon.edu.iq

والأمان. تم تقديم أنواع عديدة من الأبحاث لحل هذه المشكلات أو التخفيف من آثارها. تتمثل إحدى هذه الأفكار في استعمال تقنية Blockchain. ومع ذلك ، قد لا تستعمل بعض الأبحاث هذه التقنية بشكل صحيح. حيث تبقى مشاكل شمولية البيانات، والحفاظ على بيانات العقد الفردية. تقدم هذه الورقة نظامًا جديدًا يسمى سجل صحة المريض (PHR)، والذي سيكون عبارة عن بيانات صحية شاملة للمرضى مع قراءة جهاز إنترنت الأشياء. بيانات PHR محمية باستعمال نظام متعدد الأقران وموزع ولامركزي يعتمد على Blockchain. تم إنشاء عقد ذكي للنظام يتم من خلاله تخزين بيانات PHR مؤقتًا في Blockchain. يتم نشره على شبكة إثبات السلطة لزيادة سرعة معالجة المعاملات وهو أمر مهم في معالجة البيانات لأجهزة إنترنت الأشياء. يتم تقييم النظام المقترح للتحقق من صحة النظام وفقًا لمجموعة من المقاييس (التكلفة والثبات وتخزين البيانات والوقت المقدر).

1. Introduction

One of the most important parts of life is health, caring for it, and how to benefit from modern technology in preserving the safety and lives of humans [1]. In an effort to construct developed and secure healthcare systems, researchers are employing the latest technologies to develop the healthcare sector and satisfy its requirements, e.g., blockchain technology to protect healthcare data [2]. Electronic health records (EHRs) are the foundation of modern healthcare, ensuring that records are never lost. Diagnoses, prescriptions, patient medical histories, examinations, laboratory reports, and other decision-making tools for patient care are all stored in EHRs. Remote Patient Monitoring (RPM), in which patients are tracked outside of conventional clinical health settings, is another innovative technology that has helped patient care [3].

Internet of Things (IoT) devices are among the largest applications of RPM because they improve the timing of patient care through continuous monitoring of events occurring around the patient, even if the patient is away from the healthcare provider. Although incorporating innovative technology into the healthcare sector has many advantages, it may also have risks. Creating an electronic health record has several benefits in terms of providing quick access to the data and ensuring it is not lost, availability, and decreasing costs. Nevertheless, keeping data electronically makes it vulnerable to hacking and tampering [4]. Therefore, data communication based on RPM systems must be managed securely to avoid security breaches and keep patient data private. On the other hand, given the system's centralization, patients are not permitted to have access to their records, which has resulted in a slew of issues relating to patient safety and privacy. As a result, one promised solution to these problems is blockchain technology [5].

Blockchain technology is the most important technology used at present to provide a high level of protection and security for various types of digital data [6]. The decentralized environment provided by blockchain technology can also be integrated to manage data for applications outside of financial systems. By leveraging these aforementioned distinctive features, in the health sector, blockchain applications enhance the overall security of patients' electronic medical records and protect the data of IoT devices bonded to the RPM [7].
On this basis, the present study is organized in the following way: started with an illustration of a research plan in Section 1. Section 2 explains an overview of the background and related works. Section 3 explains the proposed system. Section 4 clarifies the research method, including simulation setup and performance metrics, as well as discusses the findings. Future works are highlighted in Section 6. Finally, point out the conclusions of the paper in Section 7

2. Research Plan

2.1. Summary of Notations

Table 1: A Summary of the Notations Used in the Article

Notation	Meaning
Dapps	Decentralized Applications
HER	Electronic Health Record
EVM	Ethereum Virtual Machine
IDC	International Data Corporation
IDE	Integrated Development Environment
IoT	Internet of Things
MQTT	Message Queuing Telemetry Transport
P2P	Peer to Peer
PHR	Patient Health Record
PoA	Proof of Authority
PoS	Proof of Stake
PoW	Proof of Work
RPM	Remote Patient Monitoring
SC	Smart Contract
WoT	Web of Things

2.2. Research Problem and Objectives

Currently, healthcare depends on centralized systems. These systems are weak in the face of hacking attacks, and they also do not maintain the privacy of patients as desired by the user, where the control of data entry and control is centralized by people or applications, and they may be unreliable. Despite the transfer of research from the centralized system to the decentralized system to address problems related to privacy and security and keeping user data safe using blockchain technology, several determinants have not been addressed.

The majority of the models concentrate on a single area of the healthcare industry, e.g., the e-health record or a system for RPM. The healthcare industry has no comprehensive coverage. There are also problems with maintaining the security of individual nodes, as most models refer to maintaining network security rather than individual nodes. The research also faces problems with the cost of SCs, where the user must register on two separate systems to protect his health data and safely transfer information over IoT devices, as this procedure is costly to the user.

The main contribution of our study is to develop a blockchain-based system that utilizes healthcare applications to perform better. The significance of the proposed system is demonstrated by expanding the scope of security management from the P2P system security service range to the hash-based multi-peer system security service range. Thus, the system can increase its performance in terms of immutability, cost, storage requirements, and estimated time. The other specific objectives of this research are:

1. To propose a system for improving multi-peer efficiency that can be used for a variety of blockchain-based healthcare and IoT applications.
2. To validate and evaluate the performance of the proposed system in comparison with standard solutions.
3. To test the proposed system through a special case study (testbed) to ensure its robustness and durability

2.3. Research Methodology

The research methodology of the proposed work involves several stages. It included a preliminary plan for this research, a study of related works as well as criticism of them, model design and implementation, validation and evaluation, and finally the reporting of the work. Figure 1 illustrates the design research methodology stages and the link between them, with the main process and outcomes for each stage.

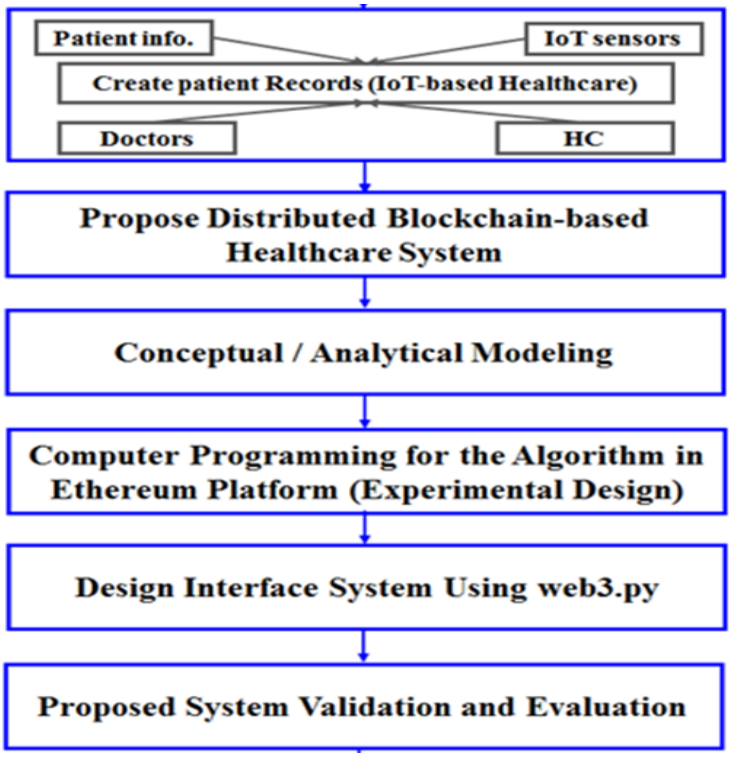


Figure 1: Design Research Methodology Stages

3. Background and Current Research of Blockchain

3.1. Healthcare System Background

Over the last few years, healthcare systems have included two fields: the EHR and RPM. By establishing an electronic record for patients, the health system has made it easier to store patient data and access it more quickly. Furthermore, new technologies have provided greater comfort to the patient by enabling him to monitor his health status remotely via an Internet-based remote patient monitoring system, for example, embedded IoT technology. On the other hand, IoT applications could be used in all aspects of our lives for better user administration [8]. Although it is a heterogeneous network and more complex as compared to other networks, it could be employed for data collection and sharing processes [9]. The direction of the researchers in IoT-based healthcare is to design intelligent systems that can observe and register the daily activities of specific people in smart environments. The IoT systems rely on multiple sensors to measure blood pressure, body temperature, oxygen, arrhythmias, etc. Hence, these systems let clinicians and family members remotely monitor their patients [10]. However, the risks of transmitting data and ensuring that it is not hacked pose a concern for subscribers to this type of system.

3.2. Blockchain Structure

A blockchain is a decentralized and distributed technology for ledgers or records that include all transactions or events that are verified by special algorithms (PoS, PoW, and PoA). Once a transaction has been validated, it cannot be altered or even erased. Digital cryptocurrencies such as Bitcoin [11] and Ethereum are the most common applications of

blockchain technology [12]. However, this technology can be used in other areas, such as healthcare and IoT. A blockchain consists of a sequence of blocks, each of which includes a comprehensive list of transaction records. If the block header has a preceding block hash, the block includes only one parent block. The genesis block, which has no parent block, is the first block on a blockchain. A block header plus a block body header make up the block. These fields of the block header are summarized as follows: Block version, Merkle tree root hash, timestamp, nBits, nonce, and hash of the parent node [13] [14].

Blockchain utilizes an asymmetric cryptography technique to verify transaction authentication, and a digital signature based on asymmetric cryptography is utilized in an unreliable environment [13]. To create blocks, miners employ a technique known as mining. Miners use specialized tools to solve the exceedingly tough arithmetic issue of discovering a previously unknown number that provides a valid hash. Because the nonce has been just 32 bits long and the hash value is 256, over four billion indivisible combinations must be retrieved before the proper combination is discovered. Every blockchain node has its own version. After validating that each node in the chain has registered, these chains are updated regularly [15].

Basically, there are two kinds of blockchain that are presented as the most significant [16]:

i. Public Blockchain: This kind of blockchain is a chain where any person can become an associate and can take part in decision-making.

ii. Private Blockchain: This kind of blockchain is not publicly available. It is available to a limited number of people, and the ledger is shared with its members only.

3.3 Blockchain-Based Open Source Project

Blockchain technology has many open-source projects, such as Hyperledger, Quorum, and Ethereum. Ethereum is just an open-source computer platform based on blockchain. The main advantage of Ethereum is that it qualifies developers to build Dapps that run on the Ethereum blockchain. The core innovation of Ethereum is the Ethereum Virtual Machine (EVM), a Turing-complete software that allows anyone to build and operate their application regardless of the programming language. The platform provides designers with the capability to develop and execute their applications without having to construct the entire blockchain from scratch [17]. In general, any contract code that types on the test network should be thoroughly tested before being published to the main network. The PoA algorithm is used by the majority of test networks. In other words, a small number of nodes are selected to validate transactions, create new blocks, and identify them [18]. The most commonly used test networks are [19]:

- Ropsten: PoW blockchain.
- Kovan: PoA blockchain, started by the Parity team.
- Rinkeby: PoA blockchain, started by the Geth team.

The PoA network is a self-contained network that is protected by a community of vetted validators. The network's validators are all notaries, and their information is available to the public. The network can provide quick and low-cost transactions thanks to this distributed community of established validators [20].

3.3. Smart Contract in Blockchain

Smart contracts (SC) are pieces of computer code that may run autonomously and perform particular functions when certain conditions are met [21]. A distributed ledger can be utilized to cache and process the code, and any modifications will be written to the ledger [22]. The SC’s main purpose is to provide better protection than traditional contract law while also cutting transaction costs. In Ethereum, the contract is given a unique address, and its code is

uploaded to the blockchain during this process. It is defined by a contract address once it has been successfully established. Any person involved in the transaction is given an Ethereum address. Each contract is associated with a predefined executable code and contains a certain amount of virtual coins. Since cryptography is used for compliance, it plays a critical role in this. A transaction's initiator pays a charge (gas) for its execution, which is also measured in units of gas [23].

The SC automatically carries out the contract terms based on the information they collect [24]. The parties come to an understanding of the contract's contents, and the contracts are carried out according to the actions written in the computer algorithms. For that, SC checks to see if the parties in a transaction follow the SC rules, which the developer writes according to the Ethereum fundamentals. The transaction is validated if they do; otherwise, it is denied. Solidity is a modern programming language for constructing EVM-compatible SC applications. The norms of networking, assembly code, and web programming are all combined in this modern language. It allows you to write contracts and compile them into bytecode for the EVM. At the moment, it is Ethereum's official language. While it is the most widely used language library for the EVM, it was not the first and is unlikely to be the last. SC written in Solidity may contain a function, function modifiers, state variables, struct types, enum types, and events [25]. The remix is employed as an IDE for SC implementation and is deployed on the PoA network.

3.5 Related Work

The most significant studies related to either the RPM system or the EHR system employing blockchain technology are presented in Table 2, according to our previous study [26].

Table 1: Summary of Related Work

Ref.	Use case	Blockchain Technology	Research Objective	Limitations
Azaria et al. [27]	HER	Ethereum platform, Proof of Work	Managing the EHR by (MedRec) and making data access faster and safer using blockchain.	The system still adopts a centralized model despite the employ of blockchain.
Usmana et al. [28]	HER	Permissioned blockchain platform, Hyperledger	The design system ensures privacy, security and accessibility, and availability of medical records.	The system still adopts a centralized model despite the use of blockchain.
Tanwar, Parekha, and Evans [2]	HER	Hyperledger platform.	Improving data accessibility between healthcare providers and helping simulate environments to execute the Hyperledger-based EHR framework which uses the chaincode principle.	Block size is still limited compared to using the Ethereum platform.
Dwivedi et al. [29]	RPM	Ethereum Platform	Make the data of IoT devices more secure with maintaining privacy for the users.	Weakness in protecting individual nodes.
Griggs et al. [30]	RPM	Ethereum Platform, consortium blockchain	Monitoring a patient in real-time by sending alerts to healthcare providers and patients. Automate notification delivery in a HIPAA-compliant manner.	Weakness in protecting individual nodes
Singh et al. [31]	RPM	Distributed, SC	Optimizing classic blockchain systems for IoT-based supply chain	Weakness in protecting individual nodes.

4. Proposed Distributed Blockchain-Based Healthcare System

In our proposed system, we are focused on criteria that enable the system to cover healthcare requirements, simplify the system, and make it more useful, reliable, and secure for the users. Figure 2 shows the proposed system block diagram.

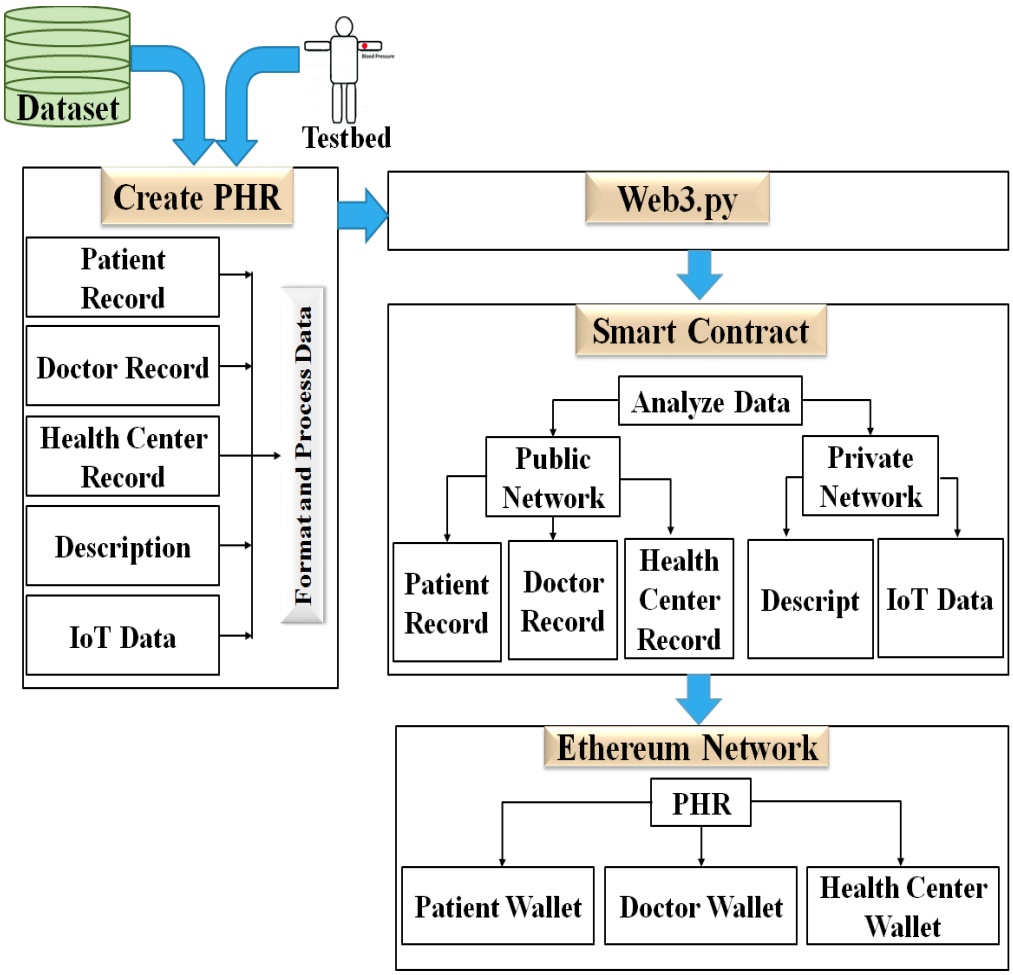


Figure 2: Block Diagram of the Proposed System

4.1 Create Patient Health Records (IoT-based Healthcare)

In this study, a new form of patient data called PHR is designed. The design seeks to create a registry that covers all personal and clinical patient data. Sub-registers (patient records, physician (doctors) records, and health center records) are included in a public health record. These records represent a peer for each one, and the whole of the records represents the concept of multiple peers that share information between them based on the proposed system. Each record (peer) includes the basic information of its user that is entered to ensure the isolation of the data of each user from the others and to indicate his/her character.

In addition, IoT device readings will be added to each log according to the proposed multi-peer system. Doctor and health center notes describe the patient's condition and the treatments used. The attending physician will add all this information based on the reading of the IoT device. Basic information from patient, doctor, and health center records will be published on the public PoA network, while IoT device information, doctor notes, and health status will be published on the private PoA network because this information is sensitive and connected to patient privacy.

4.2. Build Patient Health Record based on Blockchain

As mentioned above, PHR consists of several records. When the data for each record is entered, it is sent to the blockchain through the SC. Each record is kept in a separate block from the other records to ensure the distributed systems are applicable. The Ethereum addresses of the three peers are added (the patient wallet for the first peer, the doctor wallet for the second peer, and the health center wallet for the third, as shown in Figure 2) to the IoT device's reading to distribute this data to them. The Ethereum addresses of the three peers are also added to the description field (users' records) to ensure that the readings are distributed to them.

4.3. Smart Contract Algorithm

The SC receives the PHR data (patient, doctor, health center, doctor's notes, and IoT device reading) separately for the purpose of storing it in different blocks on the blockchain (distributed system). To check addresses that can be allowed access to the PHR, a condition has been set. If the address is not registered with the SC, that address will be rejected. Regarding the reading of the IoT device, a condition has been set for the upper and lower readings of the normal state and their classification as abnormal readings. The SC allows only abnormal readings to be stored on the blockchain. IoT reading is distributed by adding the Ethereum addresses to the sensor function. If an abnormal reading is received, the SC will send an alert to these addresses. Three addresses are also added to the description function to distribute and follow the doctor's notes. Algorithm 1 is presented in the SC that was created as a part of the proposed system.

Algorithm 1: Smart contract for PHR

```
Input:
Patient_r : patient record
Doctor_r : doctor record
HC_r : health center record
PS : pragma solidity
Begin:
1.Set endIoTPatientRate
2.Set PHR       #PHR is patient health record
3.sendMessage ← False
4.if  PS >= 0.7.0 and PS < 0.8.0
5.    if  Patient_r[patient.eth_address] == address(0x0)
6.        Add patient_r to the PHR
7.    Otherwise; Ignore
8.    End_if
9.    if  Doctor_r[doctor.eth_address] == address(0x0)
10.       Add doctor_r to the PHR
11.   Otherwise; Ignore
12.   End_if
13.   if HC_r[HC.eth_address] == address(0x0)
14.       Add HC_r to the PHR
15.   Otherwise; Ignore
16.   End_if
17.   nextIoTRate ← 1
18.   Repeat
19.           if  IoT_signal > 100
20.               Patient_r[send_message] ← B
21.               sendMessage ← True
22.           Otherwise; if  IoT_signal < 60 then
```

23. Patient_r[send_message] ← C
24. sendMessage ← True
25. Otherwise; sendMessage ← False #no action
26. End_if
27. nextIoTRate ← nextIoTRate +1
28. Until nextIoTRate == endIoTPatientRate
29. End_loop
30.End_if
End_Algorithm

4.4 Ethereum Networks

The proposed system uses the PoA network to publish the SC due to the characteristics of this network. However, in order to implement the proposed system, the real cryptocurrency (Ether) is required to publish the SC on the PoA network. On this basis, it is suggested to use the Rinkeby Test Network for SC deployment. The Rinkeby test network is one of the Ethereum networks that use the PoA algorithm, which makes the network immune to spam attacks. Rinkeby allows testing SC before deploying it on the network by giving an ether test to complete creating a SC and making a transaction inside SC.

4.5 Implementation of the Proposed System with Real Time Monitoring

In this section, the requirements for connectivity, the hardware used, the programming languages used, and the protocols used for data transfer will be explained. Starting with hardware equipment:

- Computer used: Intel (R) Core (TM) i5-3340M CPU @ 2.70GHz, Windows OS, 4GB RAM.
- The Raspberry Pi used is a Type 3 model B. The heart rate sensor is a pulse sensor and a signal converter from analog to digital.
- The Python language is used to program the IoT device.
- Raspberry pi IP (192.168.0.105).
- Pc IP (192.168.0.109).
- connection hardware:

✓ Pulse_signal = A0 of ADC.
✓ pulse Voltagecc pin of = 3.3V of Raspi.
✓ pulse sensor GND pin = GND of Pi.
✓ Ground of ADC = Ground of Raspi.
✓ VoltageCC of ADC = +5v Of Raspi.
✓ SDA and SCL of ADC module = SDA and SCL of Raspi.

Through the Raspberry Pi, Thuny was chosen as the IDE for programming the sensor in Python. Libraries related to analog-to-digital converters are defined. Then the MQTT protocol [32] for transferring data between the application and the sensor is defined. To classify the pulse reading as normal or abnormal, the proposed system adds a reading condition. If the reading is more than 100 pulses, it classifies the reading as dangerous, and the character is given as (B). If the reading is less than 60 pulses, also classify the reading as dangerous and a specific letter as (C). To display the reading on the proposed system's interface, use the WebSocket protocol and send the reading. To test the system with Ethereum, the Ethereum addresses are added to ensure the data is sent to the SC via Web3.py.

For real-time monitoring, the patient puts the pulse sensor in his finger, and when the data is read, it will be sent via Raspberry Pi using MQTT & WebSocket [33] protocols to the backend of the application for the purpose of coordinating the data according to the format

that is programmed for the interface of the IoT device data. Then the data is processed on the backend of the application. The data are analyzed according to their status (normal or abnormal). The application displays each state on the user interface of the IoT device. On the other hand, and through Web3.py, the SC receives the data and compares it with the events it has, which is shown in the following format:

event set_alarm(address indexed Patient, address indexed Doctor, address indexed Health_Center, string B);
event set_alarm2(address indexed Patient, address indexed Doctor, address indexed Health_Center, string C);

After the data is analyzed by SC, the abnormal event from the condition is logged into the blockchain, as shown in the commands below:

emit set_alarm(msg.sender,Doctor,Health_Center,"B");
emit set_alarm2(msg.sender,Doctor,Health_Center,"C");

It is important to indicate that each event is sent to the blockchain in the form of a transaction. Since the application uses a decentralized, distributed system, three addresses are included in addition to the function of the IoT device (patient, doctor, and health center). Each address will be notified of the patient's status as soon as the transaction is sent to the blockchain. It is important to note that the SC will only store abnormal readings on the blockchain and that the data will be sent to a private network.

4. Evaluation

5.1. Proposed System Validation and Evaluation

This section presents the metrics and datasets that have been adopted to validate and evaluate the proposed system.

5.1.1. Evaluation Metrics

Performance evaluation is critical in evaluating the final results of any study or research. In this study, multiple different metrics, such as cost, immutability, data storage, and estimated time, have been selected for the performance of the proposed system.

- ***Cost***: A SC's cost is the amount of cryptocurrency required to complete a transaction. To clarify the cost evaluation of the SC, it is necessary to refer to some important concepts:

✓ *Gas*: It is a unit that measures the amount of cryptocurrency required to execute each operation on Ethereum, as each operation on Ethereum, whether it is a SC instruction or a transaction, requires a certain amount of gas [34].

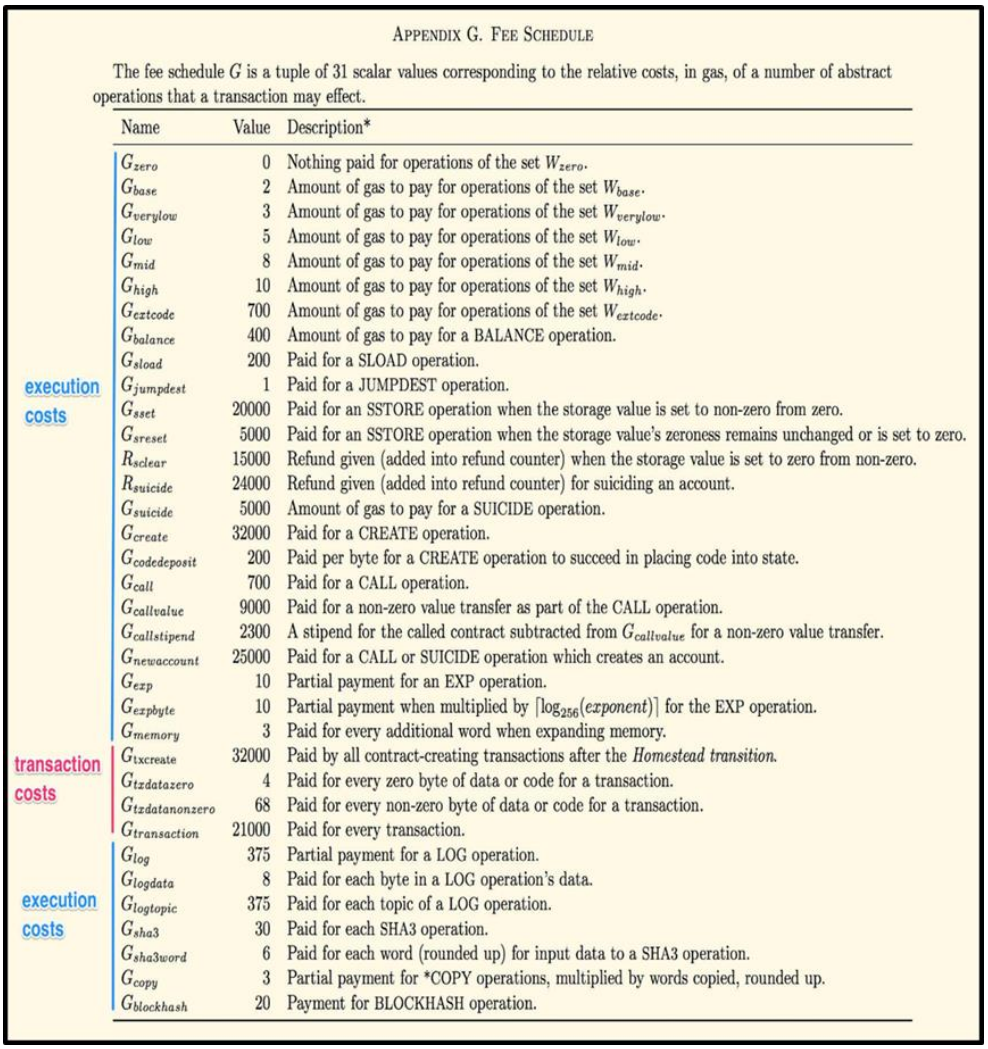
✓ *Transaction Cost*: The costs of transactions sent to the Ethereum blockchain are determined by the contract's size.

✓ Execution Cost: based on the cost of calculation operations that are executed as a result of the transaction.

Figure 3 shows the fee required for each code in the transactions, based on which the smart code will be built to reach the lowest possible cost for its deployment and the lowest cost for sending transactions.

Note that the transaction cost is measured in Wei.

- Each 1ether = 10^18 Wei
- Each 1ether = 3,410,713.98 IQD
- Each Wei = 0.000000000341071398 IQD



APPENDIX G. FEE SCHEDULE

The fee schedule G is a tuple of 31 scalar values corresponding to the relative costs, in gas, of a number of abstract operations that a transaction may effect.

	Name	Value	Description*
execution costs	G_{zero}	0	Nothing paid for operations of the set W_{zero}.
	G_{base}	2	Amount of gas to pay for operations of the set W_{base}.
	$G_{verylow}$	3	Amount of gas to pay for operations of the set $W_{verylow}$.
	G_{low}	5	Amount of gas to pay for operations of the set W_{low}.
	G_{mid}	8	Amount of gas to pay for operations of the set W_{mid}.
	G_{high}	10	Amount of gas to pay for operations of the set W_{high}.
	$G_{extcode}$	700	Amount of gas to pay for operations of the set $W_{extcode}$.
	$G_{balance}$	400	Amount of gas to pay for a BALANCE operation.
	G_{sload}	200	Paid for a SLOAD operation.
	$G_{jumpdest}$	1	Paid for a JUMPDEST operation.
	G_{sset}	20000	Paid for an SSTORE operation when the storage value is set to non-zero from zero.
	G_{sreset}	5000	Paid for an SSTORE operation when the storage value's zeroness remains unchanged or is set to zero.
	R_{sclear}	15000	Refund given (added into refund counter) when the storage value is set to zero from non-zero.
	$R_{suicide}$	24000	Refund given (added into refund counter) for suiciding an account.
	$G_{suicide}$	5000	Amount of gas to pay for a SUICIDE operation.
	G_{create}	32000	Paid for a CREATE operation.
	$G_{codedeposit}$	200	Paid per byte for a CREATE operation to succeed in placing code into state.
	G_{call}	700	Paid for a CALL operation.
	$G_{callvalue}$	9000	Paid for a non-zero value transfer as part of the CALL operation.
	$G_{callstipend}$	2300	A stipend for the called contract subtracted from $G_{callvalue}$ for a non-zero value transfer.
	$G_{newaccount}$	25000	Paid for a CALL or SUICIDE operation which creates an account.
	G_{exp}	10	Partial payment for an EXP operation.
	$G_{expbyte}$	10	Partial payment when multiplied by $\lceil \log_{256}(exponent) \rceil$ for the EXP operation.
	G_{memory}	3	Paid for every additional word when expanding memory.
transaction costs	$G_{txcreate}$	32000	Paid by all contract-creating transactions after the *Homestead transition*.
	$G_{tzdatazero}$	4	Paid for every zero byte of data or code for a transaction.
	$G_{tzdatanonzero}$	68	Paid for every non-zero byte of data or code for a transaction.
	$G_{transaction}$	21000	Paid for every transaction.
execution costs	G_{log}	375	Partial payment for a LOG operation.
	$G_{logdata}$	8	Paid for each byte in a LOG operation's data.
	$G_{logtopic}$	375	Paid for each topic of a LOG operation.
	G_{sha3}	30	Paid for each SHA3 operation.
	$G_{sha3word}$	6	Paid for each word (rounded up) for input data to a SHA3 operation.
	G_{copy}	3	Partial payment for *COPY operations, multiplied by words copied, rounded up.
	$G_{blockhash}$	20	Payment for BLOCKHASH operation.

Figure 3: Appendix G. Fee Schedule [13]

Immutability: it is defined as the number of transactions that cannot be changed or removed after they have been successfully validated and registered on the blockchain. The blocks are linked together with the hash of the genesis block, resulting in this property. Each block has been written based on the block header by providing a hash of the data transaction in the previous block in its header. Increasing the immutability of the system depends on two main factors: the block number and the miner's number. This was demonstrated in the reference [35] by measuring a system that uses dual chain design, mixed chain design, and concurrent dual chain design according to the equations below. In writing the equations, the work of Satoshi was relied upon, as Satoshi used the problem of destroying the gambler to find the probability of the attacker.

Equation of mixed-chain design:

$$Immutability = 1 - \Pr(attack) \quad (1)$$

Where (Pr(attack)) refers to probability of attack, which is the probability that an attacker can mine z blocks ahead of other miners = $(1/n)^z$

So, the equation 1 becomes:

$$Immutability = 1 - \left({}^{1}/_{n}\right)^{2z} \quad (2)$$

Where *n* refers to the number of miners and z refers to the number of blocks.

Equation of concurrent-dual-chain design:

$$Immutability = 1 - \Pr(attack) \quad (3)$$

$$Immutability = 1 - (^{2}/_{n})^{4z} \quad (4)$$

As for the number of z-blocks and how to increase them, the proposed system uses a multi-peer distribution system, where the peers are distributed in different blocks instead of everyone being in one block. For the proposed system, n refers to the number of validators in the network using the PoA network to deploy the SC.

- ***Data Storage***: The data storage metric refers to the amount of data stored in the application database [36]. Data from the IoT devices is included in the proposed system. The device sends its reading at a reading rate of 0.005. The system receives all the readings received from the IoT device and stores them. This rate of data storage is very large, as it leads to difficulty in accessing the required data. The use of SCs will limit the amount and kind of data that is cached on the blockchain, reducing the rate of data inflation.

- ***Estimated Time***: It is the measure of time that an attacker needs to hack into a block in the blockchain. If the attacker can't hack the block on time, instead of the specified honest miner, it will retry from the last created block. To calculate the estimated time, the equations mentioned in reference [25] are applied to the proposed system to prove its effectiveness, which is shown below.

T = difficulty time for Ethereum =12s
n = No. of miners.
z = No. blocks

Estimated Time for standard system:

$$E(time\ Success = T * (n)^{2z} \quad (5)$$

and

$$E(time\ Success = T * \left({}^{n}/_{2}\right)^{4z} \quad (6)$$

Estimated Time for a proposed system:

$$E(time\ Success = T * \left({}^{n}/_{3}\right)^{6z} \quad (7)$$

5.2. Dataset

The purpose of using the dataset is to test the proposed system and apply the metrics defined for system evaluation and validation. Due to the lack of a patient-specific dataset identical to the proposed system format, a virtual dataset was created to be applied to the system, containing the required information. The datasets for the doctor and the health center contain a description of the data requirements to fill in the records for each one; some fields have been deleted and others added to fit the proposed system. The IoT device dataset (heart disease sensor) contains the number of readings for 11 patients, which are included with the patient record for the purpose of sending them to the smart contract. Below are the sites from which the dataset is imported. Doctor dataset: https://data.world/city-of-ny/7btz-mnc8, Health center dataset: https://data.world/cms/hospital-general-information, and IoT dataset: https://figshare.com/articles/dataset/A_dataset_of_radar-recorded_heart_sounds_and_vital_signs_including_synchronised_reference_sensor_signals/9691544

5.3. Deploy Smart Contract to Rinkeby Ethereum Network

It is important to note that the results that will be presented are a simulation of the system since the real results require the deployment of the SC on a real network of Ethereum (PoA). Instead, the proposed system will use the Rinkeby test network as the Ethereum test network

to deploy SCs, record results, and evaluate system performance based on them. Here are the steps to deploy a SC on the Rinkeby test network to validate the SC.

1. Select a Web3 injection to connect the SC to the Rinkeby Ethereum network.
2. Select the deployment option to deploy the SC.
3. Open the wallet and submit the amount of cryptocurrency required to deploy a SC over the network.
4. The next steps for deploying the SC from Remix IDE to the Rinkeby test network.
5. To make transactions and call SC functions on the Rinkeby network, a SC address is created.

After successfully deploying the SC, we can call the functions of the SC. Then send these functions as a transaction to the Rinkeby network.

5.4. Cost of the Deploy Smart Contract

The cost of building the SC is explained in this section. Table 3 shows the various SCs (EHR, RPM, and HER-RPM) based on string data type, and PHR based on byte data type. Some notes clarify how the cost is calculated for deploying the SC.

- Each SC cost depends on the number of functions executed based on the amount, which is clearly illustrated in Figure 3.
- The unit of cost for deploying smart contracts is Wei.
- Each SC is unique based on the type of code employed to create it.

Table 2: Smart Contract Costs for Different Designs and Proposed Systems

Smart Contract	Transaction Cost (Wei)
EHR-SC based on string data type	**3054262**
RPM-SC based on string data type	**193009**
(EHR and RPM)-SC based on string data type	**3246083**
PHR-SC based on bytes data type	**1218987**

As shown in the table above, the cost of the SC for PHR has decreased when compared to the combination of all of her and RPM into a single SC. The reason for this decrease is that the proposed system's SC (PHR-SC) uses the byte32 data type rather than the string data type used by the other SCs. Since the blockchain stores the data as bytes, it converts the string data type that is included by default in SC to bytes. Hence, the data is stored as bytes. The cost of the required transaction rises as a result of this conversion. For that, the proposed system (PHR-SC) chooses the bytes32 data type directly to store data on the blockchain, which is expected to result in reducing the cost.

5.4. Evaluation the Proposed System

5.4.1. Evaluation Proposed System Using Dataset

Firstly, the proposed system results according to the dataset of the readings of the IoT device are compared with the results of the standard system (which is the system that uses string data type for coding a smart contract and uses the cloud for storing the data).

- **Cost result:**

- Each reading represents a transaction.
- Each condition represents an event.
- A transaction for executing one event takes the amount of Wei in the SC less than the transaction for executing two events.
- Transaction function cost (for events ***B*** and ***C***) = 28086 Wei.

- Reading cost= transaction function cost * number of reading.
- Total cost of readings= number of one event (condition) readings + number of two events (conditions) readings.
- 1Ether = 3,410,713.98 IQD.
- Each Wei = 0.0000000000341071398 IQD.

Example: Cost of patient1 = 3*28086Wei = 84258Wei

Cost in IQD = 84258Wei*0.0000000000341071398

= 0.00000028737993852684 IQD

According to that, the cost of the standard system and the proposed system over 10 patients' readings is presented in Table 4. The cost of the proposed system is higher than the cost of the standard system. However, it is acceptable because the proposed system sends the data and distributes it to three peers in one transaction, in addition to the SC. While the standard system sends the data to the SC only, it is assumed that the user who uses the standard system wants to share his data with the three peers, so the user should do three transactions (one transaction for each peer). As a result, the cost has risen in this case when compared to the proposed system.

Table 3: Number of Readings for Each Patient vs. Cost for the Standard System and Proposed System

Patient	No. Reading s	Reading Statues	Standard System		Proposed System	
			Cost in Wei	Cost in Iraq Currency	Cost in Wei	Cost in Iraq Currency
P 1	22	3	71670	0.0000002444458709466	84258	0.00000028737993852684
P 2	35	6	143340	0.0000004888917418932	168516	0.00000057475987705368
P 3	21	2	47780	0.0000001629639139644	56172	0.00000019158662568456
P 4	17	6	143340	0.0000004888917418932	168516	0.00000057475987705368
P 5	20	9	215010	0.0000007333376128398	252774	0.00000086213981558052
P 6	20	10	238900	0.000000814819569822	280860	0.000000957933284228
P 7	39	17	406130	0.0000013851932686974	477462	0.00000162848631831876
P 8	19	8	191120	0.0000006518556558576	224688	0.00000076634650273824
P 9	27	9	238900	0.0000007333376128398	252774	0.00000086213981558052
P 10	22	3	71670	0.0000002444458709466	84258	0.00000028737993852684

➢ **Immutability result**:

- Each reading distributed to multiple peers
- Number of readings-based dataset =73.

Immutability result for standard system:

Assume applying only one miner:

Immutability = $1 - (1/1)^{2*73}$

The result of Immutability = 0

Assume applying peer to peer miners.

Immutability = $1 - (2/73)^{4*73}$

The result of Immutability = 1

As a result, we can conclude that the result of immutability for the standard system can be 0 or 1, depending on the number of miners in the equation.

Immutability result for proposed system:

$$Immutability = 1 - (3/n)^{6*z}$$

Where *3* is the number of peers, *n* is the number of miners, 6 is the maximum block that can be attacked, and z is the number of blocks.

$$\text{Immutability} = 1 - (3/73)^{6*73} = 1$$

So, in our proposed system, as it is clear, the immutability is the highest possible, and this indicates the strength of the system's security when using a multi-peers decentralized distributed system.

- **Data storage result:**
- As shown in Figure 3 and Figure 4, transaction data can be cached on a blockchain versus transaction data cached on a cloud. In the case of transaction data cached on the cloud, it stores all the data of records (normal and abnormal). Moreover, the data type of each symbol is represented as a string based on smart contract coding. While the obtained result is reduced when records are stored on a blockchain, this is because the SC threshold rules, which are written by the developers, do not allow only abnormal readings to be stored on the blockchain.

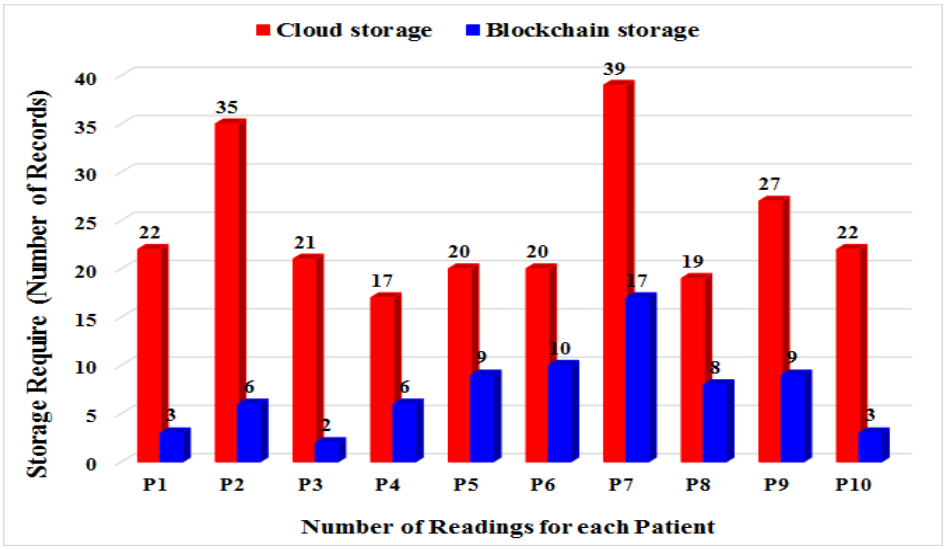


Figure 4: Number of Readings for Each Patient vs. Storage Requirement Proposed System in Blockchain and Cloud

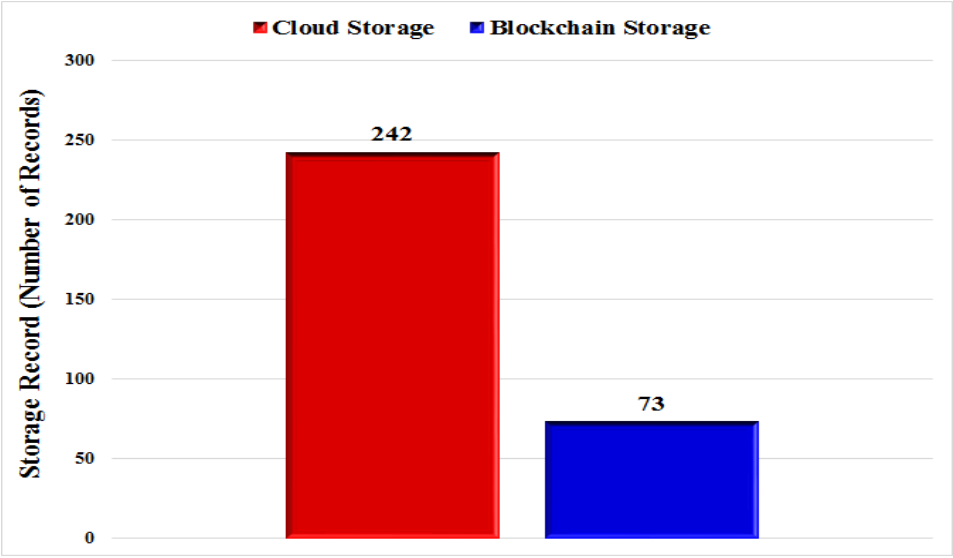


Figure 5: Depicts the Total No. of Records for All Patients Required for Storage in Both Cloud and Blockchain

➢ **Estimated time results:**

The estimated time for the attacker to perform an attack increases with the increase in the number of blocks and miners. As a result, the number of peers in the proposed system grows.

- ***Estimated time result for standard system:***

- $E(time) = T*(n)^{2*z}$
  $= 12*(1)^{2*73} = 12s$
- $E(time) = T*(n/2)^{4*z}$
  $= 12*(73/2)^{4*73}$
  $= 1.856511600748359768528569382166 5e+457s$

- ***Estimated time result for proposed system:***

$E(time) = T*(n/3)^{6*z}$
$= 12*(73/3)^{6*73}$
$= 1.719825609348671989766834164578 5e+608s$

5.4.2. Evaluation Proposed System Using Testbed

In this section, the system is evaluated based on a test performed on a patient who has an increase in heart rate. The heart rate sensor was attached to the patient's finger. The duration of the test was approximately 30 minutes. The total number of recorded readings was 1082. The number of abnormal readings was 123.

➢ **Cost results:**

The cost is calculated based on the readings that are sent to the blockchain:

- Each ether = 10^18 Wei
- Each ether = 3,410,713.98 IQD
- Each Wei = 0.000000000003410713 98IQD

Therefore, the cost in Wei = 28086*123
= 3454578

Also, cost in Iraq currency = 0.000011782577479600 44IQD

➢ **Immutability results:**

The number of reading-based testbeds is 123. The number of miners equals the number of blocks according to the PoA network, and:

$Immutability = 1-(3/123)^{6*123} = 1$

➢ **Estimated time results:**

The honest miner requires 12 seconds to mine a block of Ethereum. Assuming that the attacker needs the same amount of time to mine a block, the attacker must re-mine the entire chain to be able to hack the block data. Since each block is associated with a hash of the previous and next blocks, the time it needs can be calculated using the equation below:

$E(time) = T*(n/3)^{6*z}$

$E(time) = 12*(123/3)^{6*123}$
$= 2.059052903829010103205148510 4473e+1191s$

➢ **Data Storage results:**

Compared with storing using cloud technology, reading the heart rate sensor data for half an hour with a delay of (0.008) between one reading and another stores 1082 records, normal and abnormal. But with the use of blockchain technology and the identification of only storing abnormal readings, the number of stored readings is only 123. Therefore, the percentage compared between storage in blockchain and cloud storage is 93%. Figure 8 shows the number of readings in that storage.

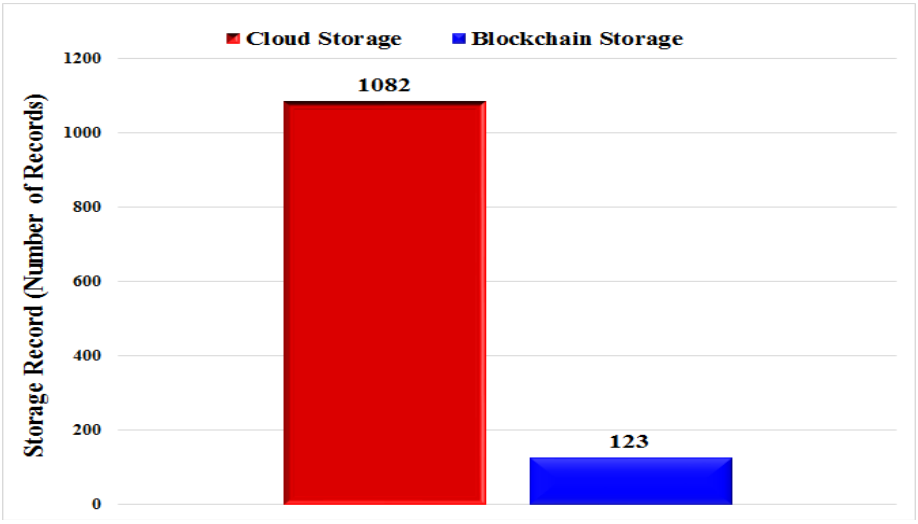


Figure 6: Number of Readings for the Patient that Storage in both Cloud and Blockchain Number of Reading

Future Works

Future work in this domain may include:

- Financial transactions: the multi-peer decentralized distribution system can be used to give cryptocurrency loans to be between the broker, the borrower, and the bank, with the possibility of adding the automatic withdrawal feature when making the borrower's transactions.
- Oil Industries: data from oil equipment sensors can be protected using blockchain technology, and the data can be monitored from multiple locations by utilizing the decentralized distributed multi-peer system.
- Blockchain Web of Things (WoT): WoT devices are heterogeneous, with varying data. These environments of data generate complexity in protecting this data. SCs can be programmed to receive different data; this feature can be used to connect the WoT devices to the blockchain through the SC.
- Scalability Healthcare: The distribution feature of blockchain technology can be used to link a group of hospitals with a SC and share data among them via their Ethereum addresses. Through this address, each hospital can save and share its data on the blockchain.

6. Conclusion

The proposed system introduced a model for a comprehensive healthcare system based on blockchain. It includes patient, doctor, and health center and IoT device readings. Patient privacy and security are important issues in the healthcare sector that have been considered in our proposed system. PHR had been evaluated according to a range of metrics: cost, immutability, data storage, and estimated time for an attacker to achieve access to the desired target. The performance of the proposed system based on blockchain was improved by increasing the immutability of the system using a decentralized distribution based on a multi-peer system. Moreover, the proposed system reduces the cost of deploying SCs on the Ethereum networks by combining the two healthcare systems (HER and RPM) into a single SC and reducing the number of tokens using the Bytes data type. Reducing the percentage of data storage by storing the specified data in the blockchain. Also, increasing the estimated attack time needed by the attacker to analyze and process data for the multi-peer system. Through testing the system (testbed), it was noted that the use of blockchain technology has reduced the number of stored records by 93% over those stored using cloud technology.

References

[1] D. Meetoo, R. Rylance, and H. A. Abuhaimid, "Health Care in a Technological World, " *British Journal of Nursins*, vol. 27, no. 20, pp. 1172–1177, 2018. https://doi.org/10.12968/bjon.2018.27.20.1172

[2] S. Tanwar, K. Parekh, and R. Evans, "Blockchain-Based Electronic Healthcare Record System for Healthcare 4.0 Applications," *Journal of Information Security and Applications*, vol. 50, pp. 1-13, 2020. https://doi.org/10.1016/j.jisa.2019.102407

[3] L. P. Malasinghe, N. Ramzan, and K. Dahal, "Remote Patient Monitoring: A Comprehensive Study," *Journal of Ambient Intelligence and Humanized Computing*, vol. 10, no. 1, pp. 57–76, 2019. https://doi.org/10.1007/s12652-017-0598-x

[4] A. Sagahyroon, "Remote Patients Monitoring: Challenges," in *2017 IEEE 7th Annual Computing and Communication Workshop and Conference (CCWC)*, 2017, pp. 57–76. https://doi.org/10.1109/ccwc.2017.7868460

[5] K. Yaeger, M. Martini, J. Rasouli, and A. Costa, "Emerging Blockchain Technology Solutions for Modern Healthcare Infrastructure," *Journal of Scientific Innovation in Medicine*, vol. 2, no. 1, pp. 1–7, 2019. https://doi.org/10.29024/jsim.7

[6] A. A. S. Al-Karkhi, N. F. Hassan, and R. A. Azeez, "A Secure Private Key Recovery Based on DNA Bio-Cryptography for Blockchain," *Iraqi J. Sci.*, vol. 64, no. 2, pp. 958–972, 2023, Doi: 10.24996/ijs.2023.64.2.38

[7] S. Angraal, H. M. Krumholz, and W. L. Schulz, "Blockchain Technology: Applications in Health Care," *Circulation: Cardiovascular quality and outcomes*, vol. 10, no. 9, pp. 1–5, 2017. https://doi.org/10.1161/CIRCOUTCOMES.117.003800

[8] S. Chowdhury, P. Mayilvahanan, and R. Govindaraj, "Optimal Feature Extraction and Classification-Oriented Medical Insurance Prediction Model: Machine Learning Integrated with the Internet of Things," *International Journal of Computers and Applications*, vol. 44, no. 3, pp. 278–290, 2022. https://doi.org/10.1080/1206212x.2020.1733307

[9] S. Mishra, R. Sagban, A. Yakoob, and N. Gandhi, "Swarm Intelligence in Anomaly Detection Systems: an Overview," *International Journal of Computers and Application*s, vol. 43, no. 2, pp. 109–118, 2021. https://doi.org/10.1080/1206212x.2018.1521895

[10] S. Nazir, Y. Ali, N. Ullah, and I. García-Magariño, "Internet of Things for Healthcare Using Effects of Mobile Computing: A Systematic Literature Review," *Wireless Communications and Mobile Computing*, vol. 2019, pp. 1-20, 2019. https://doi.org/10.1155/2019/5931315

[11] A. Rashid, A. Ali, and N. Fleih, "Wallet Key Generation for a Generic Blockchain based on Speech," *Iraqi J. Sci.*, vol. 64, no. 3, pp. 1487–1497, 2023, Doi: 10.24996/ijs.2023.64.3.37

[12] M. Mallaki, B. Majidi, A. Peyvandi, and A. Movaghar, "Off-Chain Management and State-Tracking of Smart Programs on Blockchain for Secure and Efficient Decentralized Computation," *International Journal of Computers and Applications*, vol. 44, no. 9, pp. 1–8, 2021. https://doi.org/10.1080/1206212x.2021.1948170

[13] Z. Zheng, S. Xie, H. Dai, X. Chen, and H. Wang, "An Overview of Blockchain Technology: Architecture, Consensus, and Future Trends," in *2017 IEEE International Congress on Big Data (BigData Congress)*, 2017, pp. 557–564. https://doi.org/10.1109/bigdatacongress.2017.85

[14] M. H. Jumaa and A. C. Shakir, "Review Study of E-Voting System Based on Smart Contracts Using Blockchain Technology," *Iraqi J. Sci.*, vol. 64, no. 4, pp. 2001–2022, 2023, Doi: 10.24996/ijs.2023.64.4.36

[15] D. MacKenzie, "Pick a Nonce and Try a Hash: On Bitcoin," *London Rev. Books*, vol. 41, no. 8, pp. 35–38, 2019.

[16] S. S. Sarmah, "Understanding Blockchain Technology," *Computer Science and Engineering*, vol. 8, no. 2, pp. 23–29, 2018. [Online]. Available: http://article.sapub.org/10.5923.j.computer.20180802.02.html

[17] E. M. Hreinsson and S. P. Blondal, "The Future of Blockchain Technology and Cryptocurrencies," Reykjavik University, 2018.

[18] D. P. Oyinloye, J. Sen Teh, N. Jamil, and M. Alawida, "Blockchain Consensus: An Overview of Alternative Protocols," *Symmetry*, vol. 13, no. 8, pp. 1–35, 2021. https://doi.org/10.3390/sym13081363

[19] K. Ziechmann, "Networks," *ethereum.org*, 2021. [Online]. Available: https://ethereum.org/en/developers/docs/networks/.

[20] D. Kirli *et al.*, "Smart Contracts In Energy Systems: A Systematic Review of Fundamental Approaches and Implementations," *Renewable and Sustainable Energy Reviews*, vol. 158, pp. 112013, 2022. https://doi.org/10.1016/j.rser.2021.112013

[21] C. Ene, "Smart Contracts - The New Form of the Legal Agreements," in *Proceedings of the international Conference on Business Excellence*, vol. 14, no. 1, pp. 1206–1210, 2020. https://doi.org/10.2478/picbe-2020-0113

[22] J. Earls, M. Smith, and R. Smith, "Smart Contracts: Is the Law Ready?," *Chamber of Digital Commer*ce, pp. 1–62, 2018. [Online]. Available: https://digitalchamber.s3.amazonaws.com/Smart-Contracts-Whitepaper-WEB.pdf

[23] M. Sadiku, K. Eze, and S. Musa, "Smart Contracts : A Primer," *Journal of Scientific and Engineering Research*, vol. 5, no. 5, pp. 538–541, 2018. Available: http://jsaer.com/download/vol-5-iss-5-2018/JSAER2018-05-05-538-541.pdf

[24] K. Delmolino, M. Arnett, A. Kosba, A. Miller, and E. Shi, "Step by Step Towards Creating a Safe Smart Contract: Lessons and Insights from a Cryptocurrency Lab," in *International Conference on Financial Cryptography and Data Security*, 2016, pp. 79–94. https://doi.org/10.1007/978-3-662-53357-4_6

[25] J. Jiao, S. Kan, S.-W. Lin, D. Sanan, Y. Liu, and J. Sun, "Semantic Understanding of Smart Contracts: Executable Operational Semantics of Solidity," in *2020 IEEE Symposium on Security and Privacy (SP)*, 2020, pp. 1695–1712. https://doi.org/10.1109/sp40000.2020.00066.

[26] R. Mohammed and R. Alubady and A. Sherbaz, "Blockchain-base Healthcare Applications: A Survey," in 6th *International Conference on Internet Applications, Protocols and Services (NETAPPS2020)*, 2021, pp. 130–137. [Online]. Available: https://netapps.internetworks.my/pre/netapps2020/proceedings/

[27] A. Azaria, A. Ekblaw, T. Vieira, and A. Lippman, "MedRec: Using Blockchain for Medical Data Access and Permission Management," in *2nd International Conference on Open and Big Data, (OBD 2016*), 2016, pp. 25–30. https://doi.org/10.1109/obd.2016.11

[28] M. Usmana and Usman Qamar, "Secure Electronic Medical Records Storage and Sharing Using Blockchain Technology," in *2019 International Conference on Identification, Information and Knowledge in the Internet of Things (IIKI2019)*, 2020, pp. 321–327. https://doi.org/10.1016/j.procs.2020.06.093

[29] A. D. Dwivedi, G. Srivastava, S. Dhar, and R. Singh, "A Decentralized Privacy-Preserving Healthcare Blockchain for IoT," *Sensors (Switzerland)*, vol. 19, no. 2, pp. 1–17, 2019. https://doi.org/10.3390/s19020326

[30] K. N. Griggs, O. Ossipova, C. P. Kohlios, A. N. Baccarini, E. A. Howson, and T. Hayajneh, "Healthcare Blockchain System Using Smart Contracts for Secure Automated Remote Patient Monitoring," *Journal of medical systems*, vol. 42, no. 7, pp. 1–7, 2018. https://doi.org/10.1007/s10916-018-0982-x

[31] R. Singh, A. D. Dwivedi, and G. Srivastava, "Internet of Things Based Blockchain for Temperature Monitoring and Counterfeit Pharmaceutical Prevention," *Sensors*, vol. 20, no. 14. pp. 1–23, 2020. https://doi.org/10.3390/s20143951

[32] S. Verma and M. A. Rastogi, "IoT Application Layer Protocols: A Survey," *Journal of Xi'an University of Architecture & Technology*, vol. 12, no. 8, pp. 57–63, 2020. [Online]. Available: https://xajzkjdx.cn/gallery/8-aug2020.pdf/

[33] J. Melorose, R. Perroy, and S. Careas, *Foundations of python network programming, Third Edition, Apress Berkeley, CA.* 2015.

[34] E. Albert, J. Correas, P. Gordillo, G. Román-Díez, and A. Rubio, "GASOL: Gas Analysis and Optimization for Ethereum Smart Contracts," in *International Conference on Tools and Algorithms for the Construction and Analysis of Systems*, 2020, pp. 118–125. https://doi.org/10.1007/978-3-030-45237-7_7

[35] H. S. Kim and K. Wang, "Immutability Measure for Different Blockchain Structures," in *2018 IEEE 39th Sarnoff Symposium*, pp. 1-6, 2018. https://doi.org/10.1109/SARNOF.2018.8720496

[36] J. Wu and N. K. Tran, "Application of Blockchain Technology in Sustainable Energy Systems: An Overview," *Sustainability*, vol. 10, no. 9, pp. 1–22, 2018. https://doi.org/10.3390/su10093067