Research Article GU J Sci, Part A, 12(1): 72-95 (2025) 10.54287/gujsa.1624623

Gazi University
Journal of Science
PART A: ENGINEERING AND INNOVATION
http://dergipark.org.tr/gujsa

ChainHealth: Blockchain-Based IoT-Edge Model for Secure Management of Health Data

Rukiye Nur ÇAYAN[1*] Feyza YILDIRIM OKAY[1]

[1] *Gazi University, Ankara, Türkiye*

Keywords

IoT
Blockchain
Edge
Health Data
Privacy

Abstract

The Internet of Things (IoT) is rapidly expanding and seamlessly integrating into our daily lives, with an increasing number of objects connecting to the Internet. It operates as a networked architecture that enables communication between connected devices. IoT applications span various domains, including smart homes, cities, transportation, and healthcare. Among these, smart healthcare is particularly important, allowing specialists to monitor patients remotely, anytime, and anywhere. In this system, patient data is transmitted through networked systems, enabling remote health monitoring. However, significant challenges remain regarding the privacy and integrity of patient health data. This study addresses these challenges by proposing a model named ChainHealth that leverages IoT devices for data collection, edge infrastructure for processing, smart contracts on blockchain to ensure data integrity, and blockchain to store data securely. Experimental results demonstrate that ChainHealth significantly outperforms traditional models in terms of data transmission efficiency, scalability, and overall system performance. The model enhances throughput, reduces latency even as the number of users increases, and strengthens data encryption and transmission processes. Additionally, the smart contract mechanism is evaluated and shown to be reliable for managing data integrity. As a result, the proposed model ensures secure data transfer across the network and secure critical health information. By maintaining data integrity, confidentiality, and security, ChainHealth improves both the quality and reliability of healthcare services compared to traditional approaches.

Cite

Çayan, R. N., & Okay Yıldırım, F. (2025). ChainHealth: Blockchain-Based IoT-Edge Model for Secure Management of Health Data. *GU J Sci, Part A, 12*(1), 72-95. doi:10.54287/gujsa.1624623

Author ID (ORCID Number)

0000-0002-6034-0678 Rukiye Nur ÇAYAN
0000-0002-6239-3722 Feyza YILDIRIM OKAY

Article Process

Submission Date 21.01.2025
Revision Date 18.02.2025
Accepted Date 12.03.2025
Published Date 26.03.2025

1. INTRODUCTION

The innovations of Internet of Things (IoT)-based health technologies have increased the neccesity to monitor individuals' health status. With IoT, access to healthcare services becomes more effective, and it brings greater emphasis on the collection and analysis of personal health data. Today, wearable devices such as smartwatches and smartphones enable individuals to monitor their health data continuously. In addition, there are specialized sensors that can also measure the heart rate using electrocardiograms (ECG) and monitor the blood sugar levels (Al-Kahtani et al., 2022). These technologies allow people to monitor their health status themselves in their daily lives and offer immediate access to healthcare services as it is required. Furthermore, continuous monitoring of these data provides opportunities to detect health conditions early and, therefore, intervention as a proactive way of managing an individual's health (Ranjan & Sahana, 2024). The recent studies classify

*Corresponding Author, e-mail: rukiye06nur@gmail.com

health data into four categories: the first category includes the data produced by the medical system, such as electronic medical records, prescriptions, lab data, pathology images, radiographic images, and payer claims data. The second category includes consumer-generated health and wellbeing data from wearable fitness trackers, medical devices such as insulin pumps and pacemakers, health-monitoring applications, and patient-reported outcome surveys, along with types of direct-to-consumer testing (DNA analysis) and treatment (such as insurance) that are not tied to specific health services. The third category encompasses what is referred to as digital exhaust, which arises from a consumer's activities, such as social media posts, search histories, and location data. Although these data are not health-related patterns, they can provide helpful relevant understanding of behaviors and patterns related to health. Lastly, the non-health demographic and socioeconomic data, including age, income, employment status, and education level, fall under the fourth category. Though not health-specific, these data help define broader social determinants pertaining to health and well-being (McGraw & Mandl, 2021). The first two categories contain quite sensitive data because their scope includes personally identifiable information that can easily identify an individual and provide comprehensive insights into their personal health state. With the IoT concept, it becomes easy to collect and share such sensitive data over networks. However, there is a necessity to secure such personal and sensitive information during both transmission and storage due to potential cyberattacks. Furthermore, there are issues rising from the secondary use of these data for privacy and data integrity. The integrity and correctness of these datasets are important in remote treatment, diagnosis, and monitoring. Thus, data security becomes a vital objective for effective healthcare service management (Rayan et al., 2021; Boopathi, 2023). Many different encryption algorithms and security protocols are used to protect such sensitive data. Current research has found the use of multiple technological frameworks for the securing of sensitive data. Especially, the combination of IoT with blockchain and cloud computing constitutes of innovative approaches for data security (Xiang & Cai, 2021; Nowrozy et al., 2024).

Increasing complexity and volume of health data are prompting the need for secure and efficient management, storage, and sharing mechanisms. To address this, the authors propose an innovative model named ChainHealth, which integrates IoT, edge computing, and blockchain technologies to ensure the confidentiality, security, and integrity of sensitive health data. With wearable devices and specialized sensors, the data can be collected in near real-time in order to continuously monitor health metrics while assuring the privacy of the data collection. Also, edge computing allows for processing the data at the local source that minimizing the latencies or risk of the data breach with respect to centralized storage. The decentralized and immutable characteristics of blockchain technology are perfectly aligned with the advantages of edge computing in guaranteeing that health data is kept secure, remains untampered, and is always traceable back to the source. The combination of all these technologies, therefore, provides a robust, secure, and efficient model addressing both privacy and integrity concerns for personal health data, thus further contributing towards a secure ecosystem for healthcare.

The ChainHealth contributes to the literature by offering a comprehensive and combined approach for securing health data, which addresses critical concerns related to privacy and data integrity. The first phase of this model involves encrypting data at the time of collection and securely transferring it to the edge computing infrastructure, thereby protecting against network-level attacks and ensuring data integrity throughout the entire transmission lifecycle. The next phase is the smart contract, which automates decision-making processes based on real-time data analysis to make healthcare much more responsive. The third phase involves the storage of abnormal health data in an immutable ledger, employing blockchain technology for full accountability and transparency. The model, furthermore, maintains health data integrity across networks through its transmission. In the final phase, the system's interface allows authenticated people to provide feedback, subsequently promoting communication streams between patients and healthcare providers. Thus, the model not only strengthens data security but also provides a scalable solution for real-time health interventions. Overall, this model improves conditions for responding to health situations in a more proactive and timely manner, thereby advancing the health ecosystem as a whole.

The rest of this paper is organized as follows: Section II reviews related studies. Section III gives the background information. Then section IV gives important details of the proposed model, and section V presents the performance evaluation and discussion of experimental findings. Finally, section VI summarize the paper with discussions on the results.

2. RELATED WORK

In the past few years, there has been quite an attention about the use of innovative technologies, such as blockchain, for managing health data, especially for protecting confidentiality, security, and integrity, given the sensitive nature of health information. Many research studies have been attempting to see the possibility of combining blockchain with other technologies, especially cloud computing and IoT, for better monitoring, storage, and sharing of health data. Such pioneering studies have created frameworks for security and efficiency in healthcare systems. The section reviews important studies in chronological order that have paved the way for blockchain solutions in healthcare, with an emphasis on health data management, data security, and remote patient monitoring.

Among the initial studies was 'HealthSense' (Dey et al., 2017), which investigated the use of blockchain-based technological applications for remote patient monitoring and securing sensitive health information—patient vital phenomena that originated from the sensors installed in a patient's bed. Such data was stored through a smart contract on a blockchain system. This implementation allowed for input, reading, updating, and writing to the blockchain through a REST-based API. Moreover, it introduced a method for computing final patients' hospital billing using smart contracts. In this relatively simple structure, the blockchain architecture was not specified, nor was a cloud infrastructure utilized.

The study, 'BlockCloud' (Kaur et al., 2018), focused on the storage and processing of heterogeneous health data. According to the study, apart from homogeneous health data, such as heart and blood glucose levels, there were heterogeneous data, which included ECGs and X-Rays. The storage of such kind of diverse data was very challenging on its own, and given the sensitive nature of health data, it required extra attention. The proposed model integrated both blockchain and cloud architecture. The cloud infrastructure processed and analyzed the heterogeneous data, after which the processed data was stored on the blockchain. However, with the cloud infrastructure, storing sensitive data securely was still not viable. Thus, it incorporated blockchain support for securing data.

Research concerning health data has dramatically surged since 2019 through the implementation of blockchain technology into health data. Chakraborty et al. (2019) suggested that health data storage should have been digitized through wearable devices based on a model integrating cloud architecture and blockchain technology. Machine learning algorithms detected abnormalities in the data, ensuring the integrity, privacy, and seamless security of health data. In another study, Dilawar et al. (2019) proposed a model in which doctors remotely monitored patients and added reports through blockchain. These reports were incorporated into the patient's historical data. The study also highlighted the potential high costs associated with storing data collected through remote health systems on the blockchain. As a solution, it was recommended that only the hash values of the data be stored in the blockchain rather than the data itself, minimizing costs. Health data collected via an IoT sensor was processed through an API gateway in conjunction with a smart contract that triggered the proposal of a transaction, as discussed in the study by Bhawiyuga et al. (2019). The peers in the network simulated the transaction proposal, ensuring validation of the transaction prior to recording it on the blockchain. Once the peers approved the transaction proposal, they sent their approvals back to the API gateway, which aggregated the approvals and forwarded them to the orderer. The orderer, upon receiving the approvals, broadcasted the block containing the transaction to all peers, who performed a final validation check of the transaction. This guaranteed confidentiality, security, and integrity of health data. In fact, no cloud structure was utilized; instead, the Hyperledger Fabric blockchain framework was employed as the basis. Identity verification was carried out through a smart contract written in JavaScript. API analysis of that study was done using JMeter at the end of the study. The model proposed by Dwivedi et al. (2019) made IoT, blockchain, and cloud architecture all work together for the privacy and security of health data. Due to the intense computing power required from traditional blockchains, an overlay network was created to mimic the behavior of blockchain instead of working with mining mechanisms. Data was not saved directly but was instead stored on cloud storage servers, and each block of bits was hashed using a Merkle Tree; these data were then sent to nodes in an overlay network that verified and chained the hash values of the blocks. Patients, healthcare providers, and other authorized entities shared data in the network. An expert was automatically notified if anomalies were detected by the smart contract, which monitored the data. Access to a patient's health data remained under the patient's control, and thus the patient could revoke the sharing of such data after treatment. A three-layered model was proposed by Shahnaz et al. (2019), in which electronic health data were

first processed through an initial layer where a smart contract was invoked. A smart contract was written using the Solidity programming language and the Open Zeppelin tool; identity authentication within the system was enabled through the smart contract. Following the authentication process, updates, additions, viewing, and deletion operations occurring in the second layer, the user layer, were recorded on the Ethereum blockchain. The electronic health data contained the essential parameters of an individual, such as identification number, name, blood type, and professional reports constructed as required by the medical personnel for the patient's treatment. These expert reports were stored using a distributed file system protocol known as IPFS. Toward the end of the study, performance testing was conducted using the Apache JMeter tool. The system was tested with 100-500 users, and it was observed that as the number of users increased, the system's efficiency also improved. In another work proposed by Attia et al. (2019), two different blockchain infrastructures were used for this model. The function of the first chain in this model was to store health data collected from wearable sensors, while the second chain was tasked with keeping the entire health history of the patient. Hence, according to the model, both the patients and the healthcare professionals could view information found on the first blockchain, while only the healthcare professional could view information on the second chain. Both of the two chains were based on the Hyperledger Fabric blockchain framework. To enable emergency notifications, a smart contract structure was used, which was written in the Go programming language. The model did not include cloud infrastructure.

In the later study, Satamraju (2020) gave the fundamental aim of keeping health data integrity and preventing unauthorized access to it. This study proposed a three-layer model. The first layer, the application layer, acted as an interface between the IoT devices and the blockchain. The second layer, the business layer, contained all the functionalities. These included functions such as viewing data, pulling prescriptions for a patient, or adding new data. The last layer was the storage layer, where all the data was stored. As a great cost was incurred to keep data and transactions on the blockchain, only transaction data were transferred to the blockchain and all relevant encrypted data were stored in an encrypted database. Ethereum was used as the blockchain structure for the model. The data could be visualized through a user interface; authorized personnel such as doctors, pharmacists, and insurance companies had access to this interface. In a similar study, the model proposed by Khatoon (2020) enabled access to health data by multiple stakeholders. The database contained encrypted and hashed actual values of health data, and hash value records were also kept on the blockchain. Hence, the integrity of the data could be verified by comparing the hash values against each other. The adopted blockchain framework was Ethereum, with data access authorization managed through a smart contract scripted in Solidity. Additionally, all operations performed on the data were recorded on the blockchain. Ismail et al. (2020) proposed a model named BlockHR with faster and safer patient monitoring over existing client-server architectures. The model demonstrated 20 times faster performance than existing client-server systems. Similar to most existing studies, patients could simply add or query data over the network. The advantage of the BlockHR model lay in its predictive tool, which learned from the patient data and made diagnoses based on this information. The data could be controlled and modified only by authorized individuals, with each

transaction being recorded to ensure transparency and accountability. Jamil et al. (2020) proposed a model for visualizing and processing health data collected from seven types of sensors. All of them were accessible only to authenticated individuals. The Hyperledger Fabric blockchain framework served as a backbone to the architecture of this model. The architecture had four layers. The first layer, the sensor layer, was responsible for data collection; the second layer, the network layer, was responsible for transmitting the data collected to the blockchain; the third layer, the blockchain layer, ensured, by means of the smart contract, that it checked whether the data was below the threshold level, with identity authentication handled through this same contract; and, finally, the application layer, which enabled functionality for users to interact with and input data into the system. The performance of this blockchain model was tested using the Caliper tool with 300, 500, and 1000 users, showing improved performance as the number of users increased.

Banotra et al. (2021) stated that the remote healthcare service models based on classical client-server architecture were insecure due to their central authority. Thus, they proposed a four-layered model to provide a secure platform for remote healthcare services: a physical layer where the patients from whom data was collected were situated; a second layer for sensor devices gathering such data; a third layer called the communication layer, which had an Application Programming Interface (API) implemented for sending data to the cloud; and an uppermost layer for normalizing and analyzing the transmitted data. The last layer was called the distributed ledger system, which concerned the data storage, while access to the ledger was possible either through web application or mobile application. Another model proposed by Ngabo et al. (2021) integrated both cloud infrastructure and blockchain technology. Here, health data collected through sensors was encrypted using elliptic curve algorithms and transmitted to a fog computing environment. The use of fog computing helped reduce latency during data transmission. While the data was stored in a cloud computing environment, all transactions performed on the data were recorded on the blockchain, as an authentication mechanism was required to execute these operations. The model also implemented a cloud computing structure that stored copies of transaction records on the blockchain. The subsequent proposed model in the literature was the FogChain model (Mayer et al., 2021). The main aim of this model was for the security of health data. The proposed model shared similarities with the structure of our study and utilized Hyperledger Fabric as the blockchain framework. It also incorporated fog computing architecture. In this particular model, health data collected with patients was sent to the fog computing layer. When data values were outside given threshold values, the fog computing layer processed the data and triggered a smart contract on the blockchain to record it. The adoption of fog computing minimized transmission delay and thereby improved the model's efficiency.

The ERTCA model was suggested in another study by Bataineh et al. (2022) that addressed the security of health data through the integration of rich and thin clients along with blockchain technology. In this model, Ethereum was the blockchain platform, and Proof of Work was the consensus mechanism. This model represented the thin client, which functioned as an IoT device that collected health data and sent it to the

blockchain through an API. A rich client with a larger memory had access to the blockchain and participated in mining activities. No cloud infrastructure was utilized in this model.

As shown by the research by Elvas et al. (2023), the model involved not only patients and specialists but also pharmacies in the process. The major goal of the study was to give the data owner, the patient, the ability to determine those who could access his or her data, thus making the authentication process more secure. It was also supposed to assure security for the data itself. According to the model's operation, after the patient registered and underwent an examination at the hospital, the specialist created a smart contract on the blockchain, encrypting the patient's data and the required medication list, and then storing it on the blockchain. As a result of the examination, the patient did not require second visits to pharmacies because he or she was prescribed the necessary medications. After performing identity authentication, that pharmacy was able to get the medications ready for the patient. This approach made the treatment of patients faster as well as ensuring a safer way of protecting the data. The model did not leverage any cloud infrastructure. A new study proposed by Cheikhrouhou et al. (2023) was conducted in line with the previous research by Jamil et al. (2020), which highlighted the vulnerability of data to attacks while in the network layer. Therefore, Cheikhrouhou et al. (2023) proposed a lightweight blockchain and fog-enabled secure remote patient monitoring system to enhance data security and improve system efficiency. The new approach suggested that data collected by sensors was first stored on a local blockchain and then transferred to a cloud environment for tests against threshold values. If an acceptance condition was met, the data transfer to the global blockchain was initiated. Only authorized individuals could access the two blockchains and modify the database entries, while the cloud kept the information concerning the data and its operations safely. The model used two cloud structures and showed a 40% improvement in response times. Hyperledger Fabric was selected as a blockchain platform. The study further subjected the proposed model to various attack scenarios that included key attacks, replay attacks, and impersonation attacks; and found that the model successfully prevented these threats.

BEHeDaS, introduced by Oladele et al. (2024), was a private blockchain system designed to ensure the secure and transparent storage of healthcare data. Healthcare personnel and patients authenticated themselves using unique usernames and passwords, with authorization managed through the blockchain. Healthcare professionals could access patients' medical histories, diagnose conditions, conduct tests, and upload medical results, which were encrypted and subjected to consensus within the blockchain network. Patients, in turn, could review their medical results and request modifications to their personal data, with such requests being processed only upon approval by network participants. The system operated on a Proof-of-Trust mechanism, which enhanced security through a penalty-based enforcement model, while authentication and authorization were governed by smart contracts. Since it was a P2P-based private blockchain, the system focused on security and data integrity, even though its TPS was lower than that of public blockchains. In the proposed model (Said et al., 2024), which integrated NFTs with Hyperledger Fabric, the process began with the registration of healthcare facilities (clinics, hospitals, laboratories, and pharmacies) on the blockchain network. Patient-

related transactions, including laboratory results, prescriptions, and consultation reports, were securely stored on the blockchain and directly linked to the corresponding patient, ensuring accurate identity verification. Once the patient's identity was authenticated by cross-referencing their identification number with blockchain records, a digitally verified NFT was issued. This NFT served as a cryptographic representation of the patient's validated medical records and enabled healthcare facilities within the network to authenticate identity and access medical data. However, access was time-restricted and automatically expired after a predefined period. Access control policies were enforced through a smart contract developed in Go, ensuring that patients retained control over their data.

All the studies mentioned above are summarized in Table 1. The symbol `✓' denotes that the study is integrated with the respective technology in the corresponding column; conversively, the symbol `-' in the column indicates that there is no such integration. The term NS stands for 'Not Specified', meaning the study does not provide information regarding the integration of technology.

3. BACKGROUND INFORMATION

3.1. IoT systems and Their Vulnerabilities

An IoT device is an Internet-connected device that uses sensors and actuators to collect, store, and share data from the surrounding environment with other devices in the network. Also, through connectivity to other devices, it can access data without human intervention. Another ability is intelligent communication with other devices and making decisions by itself (Torğul et al., 2016). Life cycle of an IoT system usually contains four sequential stages: collecting data using sensors, storing the data in the analyzed way, transmission of the analyzed data back to the device, and the device’s response based on the analysis results. The applications are in various fields, from smart homes to healthcare and intelligent transportation systems, and even everyday devices such as coffee machines. However, IoT systems are inherently vulnerable to a range of security threats, including physical, software-based, and Denial of Service (DoS) attacks. Moreover, the networks enabling data sharing among IoT devices are also exposed to security risks stemming from factors such as diverse protocols, multiple data structures, and the limited resources of IoT devices (Dai et al., 2019; Ratta et al., 2021).

3.2. Technological Solutions for IoT Security: Cloud, Edge Computing and Blockchain

A robust and sufficient security framework is required to address the security vulnerabilities in IoT systems and protect the collected sensitive data. Since its emergence in 2007, cloud computing offers different solutions to challenges posed by IoT heterogeneity, decentralization, and resource limitations (Darwish et al., 2019). However, due to its centralized nature, cloud computing has raised some concerns regarding real-time data transfer, which requires high bandwidth and minimal latency (Hamdan et al., 2020). Since 2017, several edge computing frameworks, especially cloudlets, fog computing, and mobile edge computing, have emerged to handle these concerns (Dang et al., 2019). Although edge computing effectively addresses challenges related

to remote resources and services, security and privacy concerns remain paramount. Therefore, some additional technologies like blockchain technology have gained attention for IoT systems.

Table 1. *Summary of Recent Studies on IoT-Based Blockchain Models for Health Data*

	Blockchain Type	Cloud Integration	Smart Contract Implementation	Smart Contract Language	Objective
(Dey et al., 2017)	NS	-	✓	NS	Security
(Kaur et al., 2018)	NS	✓	-	-	Security and storage of heterogeneous data
(Chakraborty et al., 2019)	NS	✓	-	-	Integrity, privacy, security
(Dilawar et al., 2019)	NS	-	-	-	Security
(Bhawiyuga et al., 2019)	Hyperledger Fabric	-	✓	Node.js	Confidentiality, security, and integrity
(Dwivedi et al., 2019)	NS	✓	✓	NS	Privacy and security
(Shahnaz et al., 2019)	Ethereum	-	✓	Solidity	Security, integrity
(Attia et al., 2019)	Hyperledger Fabric	-	✓	Go	Secured access
(Satamraju, 2020)	Ethereum	-	✓	Solidity	Security, integrity
(Khatoon, 2020)	Ethereum	-	✓	Solidity	Privacy, security, availability
(Ismail et al., 2020)	Permissioned blockchain	-	✓	NS	Confidentiality, integrity, availability
(Jamil et al., 2020)	Hyperledger Fabric	-	✓	Node.js	Privacy, security, scalability
(Banotra et al., 2021)	NS	✓	✓	NS	Privacy, security
(Ngabo et al., 2021)	NS	✓	✓	NS	Confidentiality, security, privacy
(Mayer et al., 2021)	Hyperledger Fabric	✓	✓	Node.js	Integrity, security, privacy
(Bataineh et al., 2022)	Ethereum	-	✓	Solidity	Security
(Elvas et al., 2023)	Ethereum	-	✓	NS	Security and storage of heterogeneous data
(Cheikhrouhou et al., 2023)	Hyperledger Fabric	✓	✓	NS	Privacy, security, confidentiality
(Oladele et al., 2024)	Hyperledger Fabric	-	✓	Go	Security, interoperability, privacy
(Said et al., 2024)	Hyperledger Fabric	-	✓	Go	Privacy, security, scalability, interoperability, traceability
This Study	Hyperledger Fabric	✓	✓	Go	Integrity, privacy, security

Blockchain enables all participants in a distributed network to maintain copies of the generated ledger. Thus, the decentralization in recording information guarantees the integrity and security of data without going through any intermediaries. Initially proposed in 1991 to prevent the forgery of electronic documents, the phrase gradually gained popularity with Bitcoin, which was first developed in 2009 (Nasir et al., 2018; Foschini et al., 2020). Integrity, time-stamping, and immutability of records are ensured through cryptographic techniques, preventing unauthorized modification. The main features of blockchain technology consist of blocks (transactions, timestamps, and cryptographic mechanisms), consensus mechanisms to approve transactions, and smart contracts to automate actions when conditions are satisfied. A variety of consensus protocols such as Proof of Work (PoW), Proof of Stake (PoS), and Practical Byzantine Fault Tolerance (PBFT) have been used for different scenarios (Nasir et al., 2018; Dai et al., 2019; Almagrabi et al., 2021; Tripathi et al., 2023). Together with the fundamental intrinsic features of blockchain like immutability, decentralization, security, anonymity, non-repudiation, and traceability (Atlam et al., 2018; Dai et al., 2019), these mechanisms make blockchain highly useful for securing IoT systems. The specified characteristics are applicable in various categories, including banking, finance, governance, healthcare, logistics and supply chain, food and agriculture, transport, real estate, and education (Tripathi et al., 2023).

These mechanisms prove their effectiveness in securing information storage and privacy within IoT networks, making them one of the best solutions to such security concerns. In particular, Hyperledger Fabric, which has developed into a very renowned open blockchain framework provided by the Linux Foundation, has gained much more importance in the field of IoT system security. It implements a property called a 'channel' to keep the transaction information confidential because it ensures that only the nodes within a specified channel can access the transaction data. Hyperledger Fabric consists of two main data structures: the world state, which keeps current data and gives direct access to the up-to-date information, and the blockchain, which captures the historical changes to the world state, thus making the transaction history transparent and auditable. To summarize, there are three main types of peers within a Fabric network: endorsers, which are responsible for validating transactions; orderers, which organize those transactions; and committer peers, which maintain the ledger (Nasir et al., 2018; Foschini et al., 2020). In addition, Hyperledger Fabric has chaincode (smart contract) development facilities in Go, Java, and Node.js, with recent studies saying that Go is the best language for chaincode development (Foschini et al., 2020).

4. PROPOSED MODEL: CHAINHEALTH

The development of the remote healthcare technology has transformed how medical conditions are monitored and treated, thus ensuring greater flexibility and accessibility for the patients (Boopathi, 2023). Particularly, continuous monitoring of health parameters such as heart rate, blood pressure, and blood glucose levels is increasingly recognized as essential for the effective management of chronic diseases, such as diabetes (Attia et al., 2019). These health conditions require constant and precise monitoring to optimize patient treatment and prevent complications. Among these parameters, the secure tracking and management of blood glucose levels

are particularly important, as they directly impact the patient's ability to control their condition and reduce the risk of chronic health complications (Ranjan & Sahana, 2024).

There are numerous challenges in dealing with remote monitoring of sensitive health data, particularly concerning data security and privacy issues. For patients to continue trusting the healthcare systems, the security of their information from unauthorized access and its integrity during the monitoring process is paramount. In response to the above-mentioned problems, the proposed model, ChainHealth, integrates IoT, edge computing, blockchain, and smart contract. This model focuses on providing a secure, trustworthy, and efficient means of monitoring blood glucose level in such a way that the information is encrypted and tamper-proof during transmission and storage. The phases of the proposed model are described in subsequent sections, depicting how every phase will help secure and efficient monitoring of blood glucose levels and preservation of data integrity with respect to the patient. In Figure 1, the architecture of the proposed model is outlined, giving an overview of the main phases involved in secure monitoring and management of glucose levels using IoT, edge computing, blockchain, and smart contract.

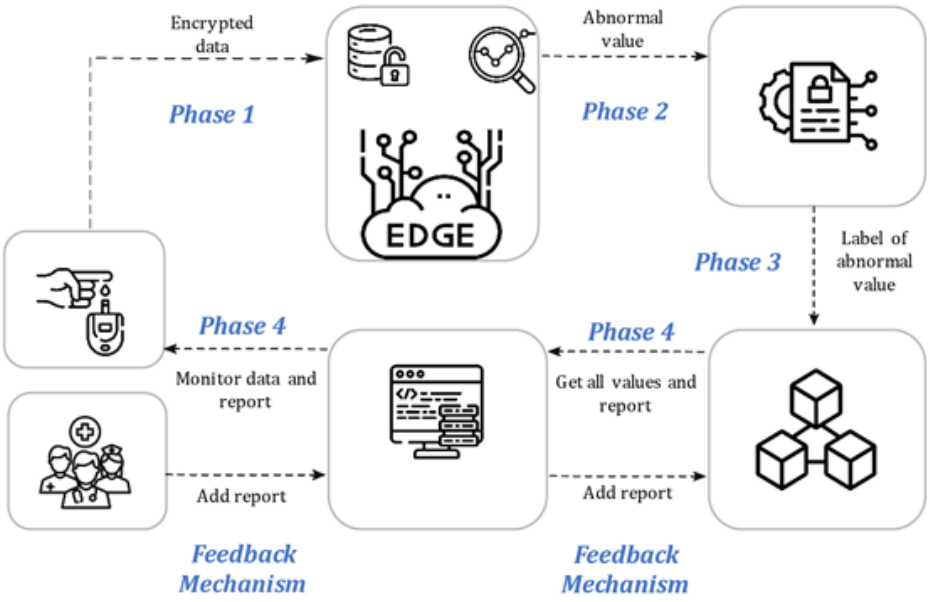

Figure 1. *The ChainHealth for secure monitoring and management of blood glucose levels using IoT, edge computing, blockchain, and smart contract*

4.1. Phase 1: IoT to Edge Computing

In the first phase, the collection of patient data is done by sensors with the help of IoT devices. Health data is vulnerable to network-based threats because it is highly sensitive. Therefore, encryption must be a key point where data, after being collected by IoT devices, is verified and kept secure during its transmission. The AES algorithm is employed to securely encrypt the data before being transmitted to the edge computing infrastructure. The encryption is such that it guarantees protection of information throughout its entire lifespan:

data collection, and transmission. The mathematical formulation of the encryption mechanism employed in the proposed model is as follows:

Step 1: Generation of AES Key through Diffie-Hellman Key Exchange

In the first step, the IoT device (A), which collects health data, specifically blood glucose levels, from the patient, and the Edge server (B), responsible for processing this data, share a common modulus p and base g. Each party independently selects its private key and exchanges public keys to establish a shared secret.

The IoT device (A) and the Edge server (B) each generate their private keys and exchange their corresponding public keys, as shown in Eq. (1):

$$A_{pub} = \mathrm{g}^{a} \mod \mathrm{p}, \qquad B_{pub} = \mathrm{g}^{b} \mod \mathrm{p} \tag{1}$$

Here, A_{pub} and B_{pub} represent the public keys of the IoT device and the Edge server, respectively, while a and b are the private keys of each party.

Both parties then calculate a shared secret key $\mathrm{K_{shared}}$ using the Diffie-Hellman key exchange algorithm, as expressed in Eq. (2):

$$\mathrm{K_{shared}} = \mathrm{B_{pub}^{a}} \bmod \mathrm{p} = \mathrm{A_{pub}^{b}} \bmod \mathrm{p} \tag{2}$$

This shared key $\mathrm{K_{shared}}$ becomes the common secret key known to both the IoT device and the Edge server. This key is then used to generate an AES key for secure data encryption.

The shared key $\mathrm{K_{shared}}$ is subsequently transformed into the AES key, as expressed in Eq. (3):

$$\mathrm{K_{AES}} = H\,(\mathrm{K_{shared}}) \tag{3}$$

Here, H denotes a hash function. This function takes the shared key and converts it into a 256-bit AES key.

Step 2: Salt Generation

In the second step, a random salt S_{salt} is generated for key derivation. This salt is an additional security measure to ensure that even if the same AES key is reused across multiple data points, the encryption process remains unique and resistant to attacks, as described in Eq. (4):

$$\mathrm{S}_{salt} = \mathrm{GenerateSalt()} \tag{4}$$

The salt S_{salt} helps to prevent the risk of two identical data points (e.g., two identical blood glucose measurements) resulting in the same encrypted output, thus maintaining the confidentiality of the data.

Step 3: Key Derivation

For each new data point (such as a new blood glucose measurement), the AES key is re-derived using the generated S_{salt} and the previously generated AES key K_{AES}. This is achieved using a Key Derivation Function (KDF), which typically involves multiple iterations to enhance security by increasing computational complexity and resistance to brute-force attacks. The derived key $K_{derived}$ is computed as shown in Eq. (5):

$$K_{derived} = KDF(K_{AES}, S_{salt}, iterations) \quad (5)$$

Here, the KDF ensures that even if an attacker gains access to one AES key, they cannot easily derive subsequent keys. The number of iterations is chosen to balance security and efficiency, ensuring that the derived key remains resistant to brute-force decryption attempts.

Step 4: Data Encryption

Once the AES key $K_{derived}$ is re-derived, it is used to encrypt the data, such as the patient's blood glucose levels. The encryption process ensures that the data remains confidential and secure during transmission to the Edge server for further analysis, as demonstrated in Eq. (6):

$$E_{AES}(m) = AES(m, K_{derived}) \quad (6)$$

In this equation, m represents the patient's blood glucose measurement. The newly derived AES key $K_{derived}$ guarantees that the encryption process is unique for each data point, even when using the same base AES key. The encrypted data is then transmitted securely to the Edge server for further processing and analysis.

Through this multi-stage process, the proposed model ensures not only the security of the AES key but also the confidentiality of the patient's health data with respect to the blood glucose levels while guaranteeing enhanced performance and scalability in terms of key generation and encryption operations. The pseudocode outlining the described process is presented in Algorithm 1 below:

Algorithm 1: IoT System (Key Generation, Key Exchange, Encryption, and Data Transmission)

```
for Every predefined time interval do
    Collect environmental data from the IoT sensor (blood sugar level)
    Get the current time → currentTime = LocalTime.now()
    if 12:00:00 ≤ currentTime ≤ 12:00:10 then
        Reject the transmission request
    else
        Retrieve the secret key from memory cache
        Retrieve the salt from memory cache
        Derive the AES key using the secret key and salt with a key
```

```
            derivation function (e.g., PBKDF2)
10:         Encrypt the data (e.g., blood sugar level) using the derived AES
            encryption key
11:         Send the encrypted data to the Edge Server for further processing
12:     end if
13: end for
14: for Every midnight (12:00 AM) do
15:     Generate Diffie-Hellman keys for key exchange with the Edge Server
16:     Encrypt the public key using base64 encoding
17:     Wait for the Edge Server's public key
18:     Use Diffie-Hellman to generate a shared secret after receiving the
        Edge Server's public key
19:     Derive the AES secret key from the shared secret
20:     Generate a salt using a key derivation function (e.g., PBKDF2)
21:     Derive the encryption key using the AES secret key and salt
22:     Encrypt the salt using AES encryption
23:     Store the derived secret key and salt in memory cache for future use
24: end for
```

4.2. Phase 2: Edge Computing to Smart Contract

The encrypted data is transmitted to the edge and securely decrypted. As soon as the decryption is completed, a detailed analysis of the data is carried out to determine its value. Based on the outcome of this analysis, the data are categorized according to the anticipated outcome on health. Prior to defining the ranges, real-world health data were analyzed to develop values that accurately reflect prevailing physiological conditions. (Memorial, 2023, accessed January 21, 2025). The classification process involves allocating the data to one of several predefined categories: 'normal', 'caution', 'high', 'very high' or 'emergency.' Data between 70 and 139 is considered to be 'normal'—an indication there is no significant health risk and that the values are within acceptable range. Data in the range of 140-179 is considered to be at 'caution', meaning that there may be a potential risk to health and that this might need close monitoring and/or further investigation. The 'high' category, involving the range 180-199, indicates that the data point toward a significant health issue, for example, the onset of diabetes, and require intervention. Data points classified as 'very high' belong to a range from 200 to 249, which means that intervention is urgently needed, since there are very high levels of a health indicator that pose a great risk. Values exceeding 250 are those that fall under the 'emergency' classification, suggesting an immediate threat to life such as a threatening coma or death-and requiring immediate medical attention. Any data that falls outside the normal range can be classified as abnormal, triggering the smart contract to execute predefined actions, such as sending alerts to medical professionals or implementing other measures to mitigate potential health risks. This system ensures that health data is accurately managed,

allowing for timely interventions and improving overall healthcare outcomes. The process of decrypting the encrypted data on the Edge server is presented in Algorithm 2:

```
Algorithm 2: Edge Server (Key Generation, Key Exchange, Decryption)
```

```
for Every midnight (12:00 AM) do
    Take the IoT system's public key
    Use Diffie-Hellman to generate a shared secret after receiving the IoT system's public key
    Decrypt the encrypted salt using the AES key
    Store the derived secret key and salt in memory cache for future use
end for
for After receiving valid data from the IoT system do
    Retrieve the secret key from memory cache
    (key = keyCache.get("dailySecretKey"))
    Retrieve the salt from memory cache
    (salt = keyCacheSalt.get("dailySalt"))
    Derive the AES key using the secret key and salt with a key derivation function (e.g., PBKDF2)
    Decrypt the data (e.g., blood sugar level) using the derived AES encryption key
end for
```

4.3. Phase 3: Smart Contract to Blockchain

During Phase 3, the smart contract interacts with the blockchain. The label of abnormal data is securely stored on the blockchain, leveraging its decentralized and immutable nature to ensure data integrity. The blockchain is an open ledger that provides complete security and transparency, since it records all transactions, labels of data and timestamps permanently. Therefore, any modifications to the data can be traced back to the source, which prevents tampering and unauthorized alterations. Functions contained in the smart contract are detailed in Table 2.

Table 2. *List of Functions Implemented in the Smart Contract*

Function Name	Description
InitLedger()	Initializes the ledger by creating the initial set of records or entries.
AddValue()	Adds a new label to the ledger.
AddReport()	Allows users to add a new report to the ledger.
GetAllValues()	Retrieves all the stored labels in the ledger.
GetAllReports()	Retrieves all the reports stored in the ledger.

4.4. Phase 4: Blockchain to İnterface

In Phase 4, abnormal data can be monitored through an interface that can be accessed by patients and authorized individuals. This interface has a time-stamped line graph for data labels, along with a time stamp of when the data was added to the blockchain. In addition, authorized individuals can provide feedback to the patient through this interface. Therefore, it ensures real-time feedback and intervention. More details will be given in the ‘Feedback Mechanism’ section.

4.5. Feedback Mechanism

The feedback mechanism allows authorized individuals to interact with the blockchain and provide real-time feedback to the patient. The data and labels are projected onto a time-tagged line graph. When an authorized individual clicks on a specific time point on the graph, a status-adding interface is opened. The authorized individual may choose to add statuses based on their own evaluation, such as: "good," "should be monitored," "stable," "consult a doctor," "emergency," or "critical." These statuses are saved in the system and displayed on the patient's interface, organized by their timestamp.

In an emergency situation, another feedback mechanism becomes activated. If an authorized individual assesses a situation to be critical, he or she may assign an "emergency" or "critical" status. Such a notice is then prompted to the patient via email. This mechanism ensures that healthcare providers also give guidance immediately, proactively managing the patient's condition.

5. EXPERIMENTAL EVALUATION

5.1. Experimental Setup

The model for this study was developed using RESTful principles. Both the IoT and edge architectures were experimentally developed using Spring Boot integrated into the IntelliJ IDE. The IoT system described in the model collected and encrypted the data, while the edge system received the encrypted data, decrypted and analyzed it, and triggered the smart contract according to analysis results. In this scenario, AES was used for encryption, while the Diffie-Hellman algorithm was employed for key exchange The blockchain structure was built on Hyperledger Fabric, following the implementation patterns provided in the fabric samples. The blockchain network was created using Docker containers while its management was done through Docker PS. The smart contract, according to the model, was written using the Go programming language and deployed into the network. Concerning the web application, both React and Spring Boot were utilized. Instead of providing actual blood glucose levels, sensor data between 70 and 250 were randomly generated for simulating glucose measurements and sent using the Postman.

5.2. Experimental Results

5.2.1. Performance Testing of Data Transmission with Apache JMeter

The study tested the transfer process of encrypted data from the IoT domain to the edge infrastructure, using Apache JMeter as a performance testing tool. The application allows simulation of multiple requests and measures the performance of the data transfer process in terms of data encryption, transmission, and processing in the edge environment. Then, the process of data transmission was analyzed on the basis of various performance metrics, throughput, response time, and resource utilization, to enable a broad examination of all stages comprising the full pipeline.

Two different models were evaluated for the test stages of data encryption, transmission, and analysis. The traditional model involves collecting sensor data, which is then secured using AES encryption before being forwarded to an edge computing environment. While AES encryption is set up for ensuring data security, it is computationally expensive, which means long processing times and higher resource consumption. To address these issues, the novel proposed model also uses the same AES encryption but introduces many optimizations to speed up both encryption and transmission processes. These optimizations include parallelization techniques and the use of more efficient data encoding methods, which together reduce encryption time, improve transmission efficiency, and lower latency, all while maintaining the data's security.

To assess the performance of the ChainHealth, it is compared against a traditional model. In the traditional model, the AES key generated at the end of Stage 1 in 'Phase 1: IoT to edge computing' (where Diffie-Hellman is used for key exchange and the AES key is derived) is directly used for encrypting the data. Although this standard AES encryption approach is secure, it lacks the optimizations of the proposed model, such as parallelization and efficient data encoding, resulting in higher computational overhead and slower transmission times. This standard AES encryption approach in the traditional model is represented by Eq. (7), where the data m is encrypted using the AES key $\mathrm{K_{AES}}$ generated at the end of Stage 1 in Phase 1.

$$\mathrm{E_{AES}\,(m)} = AES\ (m, \mathrm{K_{AES}}) \tag{7}$$

Table 3 provides an overview of throughput and response times (minimum, maximum, and average) for models at different user counts. As the table indicates, ChainHealth has consistently outperformed the traditional model, with throughput much higher and better scalability as the number of users increases**.**

The test results highlighted significant differences between the two models in terms of system stability, throughput, and resource utilization. In the traditional model, longer ramp-up times were required to stabilize system performance under load. More specifically, the ramp-up time for 100 users was 600 seconds; for 200 users, it was 1200 seconds; and for 500 users, it was 3000 seconds. The main reason behind this prolonged ramp-up time was the very high computational requirement of AES encryption, which affected the

responsiveness of the system and utilization of resources. The throughput of the traditional model was almost 0.1 requests per second for 100 and 200 users and 0.076 requests per second for 500 users.

***Table 3**. Response Times and Throughput for Different HTTP Request Implementations*

Models	Number of Users	Ramp-up (sec)	Min. Resp. Time (ms)	Max. Resp. Time (ms)	Average Resp. Time (ms)	Throughput (sec)
Traditional Model	100	600	7	266.33	17.33	≈ 0.1
	200	1200	4.33	275.33	14	≈ 0.1
	500	3000	5.33	220	12	≈ 0.076
Proposed Model: ChainHealth	100	100	8	42.66	14.33	≈ 1.0
	200	200	5	77.33	10	≈ 1.0
	500	500	2.66	71.33	5	≈ 1.0

The test results highlighted significant differences between the two models in terms of system stability, throughput, and resource utilization. In the traditional model, longer ramp-up times were required to stabilize system performance under load. More specifically, the ramp-up time for 100 users was 600 seconds; for 200 users, it was 1200 seconds; and for 500 users, it was 3000 seconds. The main reason behind this prolonged ramp-up time was the very high computational requirement of AES encryption, which affected the responsiveness of the system and utilization of resources. The throughput of the traditional model was almost 0.1 requests per second for 100 and 200 users and 0.076 requests per second for 500 users.

However, in comparison to system optimizations, ChainHealth has attained far better ramp-up times. Measured ramp-up times were 100 seconds, 200 seconds, and 500 seconds for 100, 200, and 500 users, respectively. The throughput of the ChainHealth model was 1 request per second across all user loads. This represents a 900% increase for 100 and 200 users, and a 1216% increase in throughput for 500 users compared to the conventional models. The optimized system has clearly improved encryption due to enhanced throughput and reduced ramp-up times.

The average values were obtained after 10 test runs to build reliability and consistency in evaluating performance. On the one hand, the traditional model performed well under low-load conditions, with increased load it started suffering performance-wise due to resource consumption and latencies. On the other hand, the new model proved to be more scalable and therefore was able to keep performance and throughput stable as user load increased. These results confirm that the proposed model is a more efficient and scalable solution for secure data transmission within an IoT framework. Throughput efficiency of the proposed model highlights the effectiveness of the optimizations in reducing computational overhead and improving overall system performance.

5.2.2. Smart Contract Functionality and Reliability Testing with Postman

The functionality and reliability of performing the implemented smart contract operations were tested using Postman to send HTTP requests to RESTful APIs developed using the Spring Boot framework. There were four major functions of the smart contract: AddValue(), AddReport(), GetAllValues() and GetAllReports(). The AddValue() function is used to add new labels into the ledger, while AddReport() is used to add new reports to the ledger. The GetAllValues() function returns all stored labels while GetAllReports() gives all stored reports. When valid input is given through the AddValue() and AddReport() paths, the system responds with HTTP 200 OK to show proper execution. Otherwise, it returns an HTTP 400 Bad Request on submission of invalid data to prove fault generation and input validation. Performance testing with JMeter, which measures the professional tests and efficiency of smart contract operations, can also be used across multiple conditions. Moreover, they were tested for functionality and reliability of smart contract methods deployed into the Hyperledger Fabric network via HTTP requests sent using Postman. The primary methods of the smart contract are AddValue(), AddReport(), GetAllValues(), and GetAllReports(). Under Spring Boot endpoints, valid data was confirmed with these methods, which then returned an HTTP 200 OK status, indicating successful entry of the data set. Invalid or incomplete submissions return an appropriate HTTP 400 Bad Request status, thus showing that proper input validation and error handling mechanisms are triggered. GetAllValues() and GetAllReports() were also tested. Health data stored on the blockchain for retrieval using these methods were successfully retrieved without error. It can thus be concluded that smart contract functions are strong and reliable in their operation within the blockchain network.

5.3. Discussion

The proposed model has several advantages in secure and efficient data transmission. It integrates AES encryption, edge computing, and blockchain technologies to enhance the security of data, reduce latency, and improve scalability for remote healthcare systems. Glucose levels are continuously monitored, and only abnormal values may be recorded into the blockchain to reduce storage and also network load. The emphasis is on postprandial glucose measurement, which is within the critical period of 1-2 hours after meals when an increase in insulin secretion is expected in response to rising glucose levels in the body (Dimitriadis et al. 2021). Only salient deviations in glucose levels—spikes or dips—are recorded into the blockchain. This selective strategy improves the efficiency of the system by filtering out normal variations and, at the same time allows for the identification of certain abnormal health threats, such as insulin resistance or blood sugar imbalances, which may require some type of intervention. Blockchain technology also guarantees the integrity and immutability of the data against unauthorized access or tampering. A smart contract inbuilt within the system allows real-time analysis and feedback for rapid, effective healthcare interventions for chronic patients. The model also secures data communication among hospitals and other health institutions, improving interoperability of health institutions, and ensuring compliance with data protection regulations globally with GDPR and HIPAA and locally under Turkey's KVKK. Real-time health data sharing can also accelerate clinical decision-making, ultimately leading to better patient outcomes through proactive intervention.

Although it offers several advantages, the proposed model also has limitations that should be considered. The proposed model processes the data through the database rather than real physical IoT sensors and gateways which incurs time-consuming operations on the database. Furthermore, the testing was conducted using the free test network of Hyperledger Fabric, which may not fully reflect the performance characteristics expected from a production-ready system. These limitations are anticipated to be addressed in future iterations of the model, leading to improved performance once actual IoT devices and optimized network configurations are incorporated, removing the need for database-related operations and allowing the system to operate more efficiently in real-world applications. In addition, with the increase in the network and with more health providers and IoT devices onboard, transaction throughput, latency, and computational overhead will also have to be managed very carefully for performance and reliability.

6. CONCLUSION

With a rapid increase in the number of devices, particularly in healthcare, the amount of sensitive data, such as health information, has also increased tremendously. Ensuring the security, privacy, and integrity of this data during transmission and storage is paramount. Existing IoT models are mostly dependent on a centralized server that may lead to high costs and a single point of failure. Such communication lacks secure transmission and is mutable with regard to transactions, leaving it vulnerable to attacks. In this study, a blockchain and edge-based IoT model called ChainHealth is proposed to facilitate healthcare device transactions and communication, as a substitute for traditional IoT models. The proposed model is a hybrid IoT and edge computing blockchain framework, designed to ensure security, privacy, and integrity of data. By incorporating blockchain, the transactions are designed to be immutable and secure, while the communication is encrypted between devices. The proposed model optimizes processing capabilities of the IoT devices due to its decentralized structure. The experimental tests regarding blood glucose level data shows the effectiveness of the model in terms of security, privacy and integrity of health data. This model can be a possible solution for addressing the security and integrity of health data. Consequently, the proposed ChainHealth offers promising solutions to the challenges of secure data management in healthcare and fills a significant gap in healthcare technologies.

Future work could address the model's limitations, expand its application to real-world IoT sensors and gateways for more accurate and dynamic data capture, and integrate advanced networking configurations to improve performance. Additionally, scalability and reliability can be further enhanced, while emerging technologies such as artificial intelligence and machine learning can be incorporated for predictive analytics and improved decision-making. Beyond the healthcare sector, this model may be applied in other fields that require privacy, security, and data integrity, including finance, personal identification systems, and smart cities. These sectors increasingly demand robust and secure data management solutions, positioning the model as a viable option for broader applications.

AUTHOR CONTRIBUTIONS

Methodology and writing-reviewing, R.N.Ç and F.Y.O; editing, F.Y.O; conceptualization and software, R.N.Ç. All authors have read and legally accepted the final version of the article published in the journal.

CONFLICT OF INTEREST

The authors declare no conflict of interest.

REFERENCES

Al-Kahtani, M. S., Khan, F., & Taekeun, W. (2022). Application of internet of things and sensors in healthcare. *Sensors*, *22*(15), 5738. https://doi.org/10.3390/s22155738

Almagrabi, A. O., Ali, R., Alghazzawi, D., AlBarakati, A., & Khurshaid, T. (2021). Blockchain-as-a-Utility for Next-Generation Healthcare Internet of Things. *Computers, Materials & Continua*, *68*(1). https://doi.org/10.32604/cmc.2021.014753

Atlam, H. F., Alenezi, A., Alassafi, M. O., & Wills, G. (2018). Blockchain with Internet of Things: Benefits, challenges, and future directions. *International Journal of Intelligent Systems and Applications*, *10*(6), 40-48. https://doi.org/10.5815/ijisa.2018.06.05

Attia, O., Khoufi, I., Laouiti, A., & Adjih, C. (2019, June). An IoT-blockchain architecture based on Hyperledger framework for healthcare monitoring application. In *NTMS 2019-10th IFIP International Conference on New Technologies, Mobility and Security* (pp. 1-5). IEEE Computer Society. https://doi.org/10.1109/NTMS.2019.8763849

Banotra, A., Sharma, J. S., Gupta, S., Gupta, S. K., & Rashid, M. (2021). Use of Blockchain and Internet of Things for Securing Data in Healthcare Systems. In K. J. Giri, S. A. Parah, R. Bashir, & K. Muhammad (Eds.), *Multimedia Security: Algorithms for Intelligent Systems* (pp. 255–267). Springer. https://doi.org/10.1007/978-981-15-8711-5_13

Bataineh, M. R., Mardini, W., Khamayseh, Y. M., & Yassein, M. M. B. (2022). Novel and secure blockchain framework for health applications in IoT. *IEEE Access*, *10*, 14914-14926. https://doi.org/10.1109/ACCESS.2022.3147795

Bhawiyuga, A., Wardhana, A., Amron, K., & Kirana, A. P. (2019, December). Platform for integrating Internet of Things based smart healthcare system and blockchain network. In *2019 6th NAFOSTED Conference on Information and Computer Science (NICS)* (pp. 55-60). IEEE. https://doi.org/10.1109/NICS48868.2019.9023797

Boopathi, S. (2023). Internet of things-integrated remote patient monitoring system: Healthcare application. In *Dynamics of swarm intelligence health analysis for the next generation* (pp. 137-161). IGI Global. https://doi.org/10.4018/978-1-6684-6894-4.ch008

Chakraborty, S., Aich, S., & Kim, H. C. (2019, February). A secure healthcare system design framework using blockchain technology. In *2019 21st International Conference on Advanced Communication Technology (ICACT)* (pp. 260-264). IEEE. https://doi.org/10.23919/ICACT.2019.8701983

Cheikhrouhou, O., Mershad, K., Jamil, F., Mahmud, R., Koubaa, A., & Moosavi, S. R. (2023). A lightweight blockchain and fog-enabled secure remote patient monitoring system. *Internet of Things*, *22*, 100691. https://doi.org/10.1016/j.iot.2023.100691

Dai, H. N., Zheng, Z., & Zhang, Y. (2019). Blockchain for Internet of Things: A survey. *IEEE internet of things journal*, *6*(5), 8076-8094. https://doi.org/10.1109/JIOT.2019.2920987

Dang, L. M., Piran, M. J., Han, D., Min, K., & Moon, H. (2019). A survey on Internet of Things and cloud computing for healthcare. *Electronics*, *8*(7), 768. https://doi.org/10.3390/electronics8070768

Darwish, A., Hassanien, A. E., Elhoseny, M., Sangaiah, A. K., & Muhammad, K. (2019). The impact of the hybrid platform of Internet of Things and cloud computing on healthcare systems: Opportunities, challenges, and open problems. *Journal of Ambient Intelligence and Humanized Computing*, *10*, 4151-4166. https://doi.org/10.1007/s12652-017-0659-1

Dey, T., Jaiswal, S., Sunderkrishnan, S., & Katre, N. (2017, December). HealthSense: A medical use case of Internet of Things and blockchain. In *2017 International Conference on Intelligent Sustainable Systems (ICISS)* (pp. 486-491). IEEE. https://doi.org/10.1109/ISS1.2017.8389459

Dilawar, N., Rizwan, M., Ahmad, F., & Akram, S. (2019). Blockchain: Securing Internet of Medical Things (IoMT). *International Journal of Advanced Computer Science and Applications*, *10*(1), 82-89. https://doi.org/10.14569/IJACSA.2019.0100110

Dimitriadis, G. D., Maratou, E., Kountouri, A., Board, M., & Lambadiari, V. (2021). Regulation of postabsorptive and postprandial glucose metabolism by insulin-dependent and insulin-independent mechanisms: an integrative approach. *Nutrients*, *13*(1), 159. https://doi.org/10.3390/nu13010159

Dwivedi, A. D., Srivastava, G., Dhar, S., & Singh, R. (2019). A decentralized privacy-preserving healthcare blockchain for IoT. *Sensors*, *19*(2), 326. https://doi.org/10.3390/s19020326

Elvas, L. B., Serrão, C., & Ferreira, J. C. (2023). Sharing health information using a blockchain. *Healthcare*, *11*(2), 170. https://doi.org/10.3390/healthcare11020170

Foschini, L., Gavagna, A., Martuscelli, G., & Montanari, R. (2020, June). Hyperledger fabric blockchain: Chaincode performance analysis. In *ICC 2020-2020 IEEE International Conference on Communications (ICC)* (pp. 1-6). IEEE. https://doi.org/10.1109/ICC40277.2020.9149080

Hamdan, S., Ayyash, M., & Almajali, S. (2020). Edge-computing architectures for Internet of Things applications: A survey. *Sensors*, *20*(22), 6441. https://doi.org/10.3390/s20226441

Ismail, L., Materwala, H., & Sharaf, Y. (2020, October). Blockhr–a blockchain-based healthcare records management framework: Performance evaluation and comparison with client/server architecture. In *2020 International Symposium on Networks, Computers and Communications (ISNCC)* (pp. 1-8). IEEE. https://doi.org/10.1109/ISNCC49221.2020.9297216

Jamil, F., Ahmad, S., Iqbal, N., & Kim, D. H. (2020). Towards a remote monitoring of patient vital signs based on IoT-based blockchain integrity management platforms in smart hospitals. *Sensors*, *20*(8), 2195. https://doi.org/10.3390/s20082195

Kaur, H., Alam, M. A., Jameel, R., Mourya, A. K., & Chang, V. (2018). A proposed solution and future direction for blockchain-based heterogeneous medicare data in cloud environment. *Journal of Medical Systems*, *42*, 1-11. https://doi.org/10.1007/s10916-018-1007-5

Khatoon, A. (2020). A blockchain-based smart contract system for healthcare management. *Electronics*, *9*(1), 94. https://doi.org/10.3390/electronics9010094

Mayer, A. H., Rodrigues, V. F., da Costa, C. A., da Rosa Righi, R., Roehrs, A., & Antunes, R. S. (2021). Fogchain: A fog computing architecture integrating blockchain and Internet of Things for personal health records. *IEEE Access*, *9*, 122723–122737. https://doi.org/10.1109/ACCESS.2021.3109822

McGraw, D., & Mandl, K. D. (2021). Privacy protections to encourage use of health-relevant digital data in a learning health system. *NPJ Digital Medicine*, *4*(1), 2. https://doi.org/10.1038/s41746-020-00362-8

Memorial. (2023, July 3). Tokluk kan şekeri kaç olmalı? Memorial. (Accessed:21/01/2025) URL

Nasir, Q., Qasse, I. A., Abu Talib, M., & Nassif, A. B. (2018). Performance analysis of Hyperledger fabric platforms. *Security and Communication Networks*, *2018*(1), 3976093. https://doi.org/10.1155/2018/3976093

Ngabo, D., Wang, D., Iwendi, C., Anajemba, J. H., Ajao, L. A., & Biamba, C. (2021). Blockchain-based security mechanism for the medical data at fog computing architecture of Internet of Things. *Electronics*, *10*(17), 2110. https://doi.org/10.3390/electronics10172110

Nowrozy, R., Ahmed, K., Kayes, A. S. M., Wang, H., & McIntosh, T. R. (2024). Privacy preservation of electronic health records in the modern era: A systematic survey. *ACM Computing Surveys*, *56*(8), 1-37. https://doi.org/10.1145/3653297

Oladele, J. K., Ojugo, A. A., Odiakaose, C. C., Emordi, F. U., Abere, R. A., Nwozor, B., ... & Geteloma, V. O. (2024). BEHedas: A blockchain electronic health data system for secure medical records exchange. *Journal of Computing Theories and Applications*, *1*(3), 231-242. https://doi.org/10.62411/jcta.9509

Ranjan, R., & Sahana, B. C. (2024). A comprehensive roadmap for transforming healthcare from hospital-centric to patient-centric through healthcare Internet of Things (IoT). *Engineered Science*, *30*, 1175. https://doi.org/10.30919/es1175

Ratta, P., Kaur, A., Sharma, S., Shabaz, M., & Dhiman, G. (2021). Application of blockchain and internet of things in healthcare and medical sector: applications, challenges, and future perspectives. *Journal of Food Quality*, *2021*(1), 7608296. https://doi.org/10.1155/2021/7608296

Rayan, R. A., Tsagkaris, C., & Iryna, R. B. (2021). The Internet of Things for Healthcare: Applications, Selected Cases, and Challenges. In *IoT in Healthcare and Ambient Assisted Living* (pp. 1-15). Springer. https://doi.org/10.1007/978-981-15-9897-5_1

Said, H. E., Al Barghuthi, N. B., Badi, S. M., Hashim, F., & Girija, S. (2024, August). Developing a Decentralized Blockchain Framework with Hyperledger and NFTs for Secure and Transparent Patient Health Records. In *The International Conference on Innovations in Computing Research* (pp. 478-489). Cham: Springer Nature Switzerland. https://doi.org/10.1007/978-3-031-65522-7_42

Satamraju, K. P. (2020). Proof of concept of scalable integration of Internet of Things and blockchain in healthcare. *Sensors, 20*(5), 1389. https://doi.org/10.3390/s20051389

Shahnaz, A., Qamar, U., & Khalid, A. (2019). Using blockchain for electronic health records. *IEEE Access, 7*, 147782-147795. https://doi.org/10.1109/ACCESS.2019.2946373

Torğul, B., Şağbanşua, L., & Balo, F. B. (2016). Internet of Things: A survey. *International Journal of Applied Mathematics Electronics and Computers, Special Issue-1*, 104-110. https://doi.org/10.18100/ijamec.267197

Tripathi, G., Ahad, M. A., & Casalino, G. (2023). A comprehensive review of blockchain technology: Underlying principles and historical background with future challenges. *Decision Analytics Journal, 100344*. https://doi.org/10.1016/j.dajour.2023.100344

Xiang, D., & Cai, W. (2021). Privacy protection and secondary use of health data: Strategies and methods. *BioMed Research International, 2021*(1), 6967166. https://doi.org/10.1155/2021/6967166