IEEE Access
Multidisciplinary ⋮ Rapid Review ⋮ Open Access Journal

Received 4 May 2023, accepted 20 May 2023, date of publication 24 May 2023, date of current version 1 June 2023.

Digital Object Identifier 10.1109/ACCESS.2023.3279724

RESEARCH ARTICLE

Distributed Ledger Technology Based Integrated Healthcare Solution for Bangladesh

MD. ARIFUL ISLAM[1], (Member, IEEE), MD. ANTONIN ISLAM[1], MD. AMZAD HOSSAIN JACKY[2], MD. AL-AMIN[2], MD. SAEF ULLAH MIAH[2], (Member, IEEE), MD. MUHIDUL ISLAM KHAN[3], AND MD. IQBAL HOSSAIN[4]

[1]Brain Station 23 Ltd., Dhaka 1212, Bangladesh
[2]Department of Computer Science, Faculty of Science and Technology, American International University-Bangladesh, Dhaka 1229, Bangladesh
[3]Department of Electronics and Computer Science, University of Stavanger, 4021 Stavanger, Norway
[4]Department of Paediatrics, Chakaria Unique Hospital, Chattogram 4349, Bangladesh

Corresponding author: Md. Muhidul Islam Khan (md.m.khan@uis.no)

This work was supported by the Norwegian Research Council through the 5G Management and Orchestration for Data and Network Integration (5G-MODaNeI) Project under Grant 308909.

ABSTRACT Healthcare data is highly sensitive and must be safeguarded. Personal and sensitive data, such as names and addresses, is stored in Encrypted Electronic Health Records (EHRs). This paper proposes a Blockchain-based distributed application platform for Bangladesh's public and private healthcare service providers. The proposed application framework enables users to create secure digital agreements for commerce or collaboration by leveraging data immutability and smart contracts. As a result, all stakeholders can collaborate securely over the same Blockchain network, taking advantage of their data's openness and read/write nature. The proposed application is made up of various application interfaces for various stakeholders. The proposed solution employs Hyperledger Fabric and Blockchain to ensure data integrity, privacy, permissions, and service availability. In the application portal, each user has a profile. The creation of a unique identity for each user, as well as the establishment of digital information centers across the country, has greatly aided the process. This application collects health data from each user in a systematic manner, which is useful for research institutes and healthcare-related organizations. For this application, a national data warehouse in Bangladesh is feasible, and various healthcare-related analyses can be performed using the collected data, assisting the strategy and planning department in making informed decisions regarding the healthcare sector in Bangladesh. Because Bangladesh has both public and private healthcare providers, a simple digital strategy is essential for all organizations to accomplish their services. This study proposes a solution to achieve this goal.

INDEX TERMS Distributed ledger technology, integrated healthcare, blockchain, smart contract, hyperledger fabric, blockchain in healthcare.

I. INTRODUCTION

The healthcare industry is an integral part of a country's economy and includes many medical products and services that are also necessities [1]. This sector includes an extensive chain of services to which the government contributes in a big way. For example, the government of the People's Republic of Bangladesh has at least 104,659 human resources for health services [2] for which the government budgeted $2,174 million [2] in 2007, and the budget is increasing day by day according to public demand. The People's Republic of Bangladesh offers a comprehensive Essential Health Service Package (ESP), which includes many free medicines of various generic drugs and many low-cost medical tests, as presented in Table 1. However, as these services are provided in an area of 148,460 sq km, densely populated with 163 million people (2019, World Bank), many accuse the country of mismanagement, imbalance, and a lack of service distribution. As a result, the national public health service cannot correctly meet public satisfaction [3].

The associate editor coordinating the review of this manuscript and approving it for publication was Mueen Uddin.

TABLE 1. Minimum standards and extra services by facility level, which are provided by the government of the people's Republic of Bangladesh [11].

Community Clinic	Union Health and Family Welfare Centre	Upazila Health Complex	Maternal and Child Welfare Centre	District Hospital
				Trauma Care
				Ophthalm. Surgery
		General Surgery		General Surgery
		Obstetric Fistula		Obstetric Fistula
		CEmONC		CEmONC
		Severe cases		Severe cases
	BEmONC	BEmONC	CEmONC	BEmONC
Normal Newborn	Normal Newborn	Normal Newborn	Pre-term NB	Normal Newborn
N.V. Deliveries	N.V. Deliveries	N.V. Deliveries	Newborn Sepsis	N.V. Deliveries
NCD Screening	NCD Screening	NCD Screening	Normal Newborn	NCD Screening
SBCC	SBCC	SBCC	N.V. Deliveries	SBCC
EPI/IMCI	EPI/IMCI	EPI/IMCI	SBCC	EPI/IMCI
FP Short Acting	FP Short Acting	FP Short Acting	EPI/IMCI	FP Short Acting
Growth Monitoring	GM, SAM mngmt	GM, SAM mngmt	GM, SAM mngmt	GM, SAM mngmt
ANC/PNC	ANC/PNC	ANC/PNC	FP all methods	ANC/PNC
Lim. curative care	Lim. curative care	Lim. curative care	ANC/PNC	Lim. curative care

🟧	Minimum Standards by facility level
□	Extra Services

To address all these shortcomings, the government is trying to find technology-based solutions, and many third-party companies are trying to solve these problems on behalf of the government. However, since it is a nationwide service with a large number of administrators, the government should move the entire administrative system to a digital platform where every action (only the official one) of each health official is counted as data and all data can be authenticated, stored, and analyzed for monitoring and management. In this way, Bangladesh can take another step towards digitization.

But with this great philosophy of digitizing a nation comes responsibility for digital security (data, identity, and assets) and cybersecurity (information and information technology (IT) infrastructure). After all, many security measures can be taken when an organization or institution deals with physical documents [4]. The global monetary system is an excellent example of this. These can have many factual printing patterns that can be easily distinguished from other counterfeit copies. But when it comes to digital data, tampering can be done with the central database without leaving a trace. This can lead to a national or international threat. The Kaseya Ransomware attack [5] is a good example of this. It affected over 1500 companies and over 1 million locked computers. At the same time, the ongoing trial is fraught with many allegations. Some are aimed directly at officials for lack of proper oversight and system gaps, and most are due to an attempt to serve a populous nation with traditional monolithic management.

Adopting a digital platform in the ministerial office environment is not a new challenge for Bangladesh. Central and state-owned banks are good examples of this. They work with global banking systems and, at the same time, with locally managed file transfer (MFT) and payment gateway services. However, the issue is digital data's security, traceability, and immutability. Therefore, the importance of the new, cutting-edge technology known as Blockchain cannot be overlooked. To introduce this emerging technology at a national level, we must promote it early. Therefore, the solution proposed in this article could be a breakthrough in the movement of the world's fourth industrial revolution [6], [7]. The proposed solution proposes a digital information management system that ensures the public health service's status, information transparency, and security.

More specifically, the project is a complete solution designed to provide a single service to the general population of Bangladesh. The back end will be protected with Blockchain-based distributed ledger technology (DLT). So, there will be no dependence on third parties for digital healthcare. Due to technical limitations, public and private medical complexes or hospitals will be kept in service delivery. All healthcare providers will be connected as peers to a single digital health service at the national level. So competition between medical facilities for good services will continue, but the population will access these services from a single source through their citizen health portal. In addition to solving all these current problems of civilian service, our project could

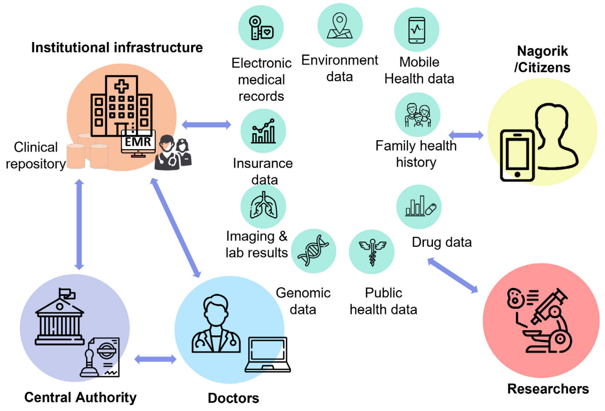


FIGURE 1. Different types of EHR components can be kept inside a secure Blockchain Network to facilitate cross-domain research.

be an essential tool to improve civilian life in more diverse ways.

A decentralized Blockchain service might be the best choice for several governmental and non-governmental organizations involved in clinical or biomedical research. Using this technology, different organizations can collaborate with other institutions or organizations to share and analyze data without relinquishing control. Each institution can maintain complete control over its computing resources while collaborating with other institutions. This way, a robust data set can be formed for each clinical trial [8], [9]. Fig. 1 provides an overview of this idea of secure collaboration. Patient-generated data becomes available to researchers. As a result, a decent environment, as shown in Fig. 1, can be created where academia, the healthcare industry, and healthcare professionals can collaborate. In the case of security and privacy, private Blockchain can be an excellent option to control data's transparency, security, and immutability. If anonymized and then tracked in the research process with a timestamp, this secondary data source would enable millions of individuals, healthcare providers, healthcare institutions, and medical researchers to share vast amounts of genetic, nutritional, lifestyle, environmental, and health data with guaranteed security and privacy [8], [10].

The key contributions of this study are:

- We propose a DLT-based healthcare platform allowing providers to securely store and share patient data. The platform uses a permissioned Blockchain to ensure only authorized users can access patient data while maintaining data privacy and security.
- Our proposed platform allows healthcare providers to share patient data across different healthcare organizations, which improves data sharing and interoperability. The platform also allows patients to access their medical records, which promotes patient engagement and empowerment.
- Our proposed platform considers the consensus mechanism for fault tolerance and better exception handling compared with the state-of-the-art in research.
- The platform enables efficient management of healthcare resources, such as medical supplies and personnel, by providing real-time data on the availability and utilization of resources. This allows healthcare providers to make informed decisions and optimize the use of resources.
- Our proposed method ensures the integrity of healthcare data and prevents fraud and errors. The platform also enables real-time tracking and monitoring of healthcare services, which can help detect and prevent fraudulent activities.

The paper is structured as follows: related work is discussed in Section II; the problem statement is outlined in Section III; a proposed solution based on the problem statement is presented in Section IV; the detailed system design for the proposed solution is discussed in Section V; technical details of the system implementation are described in Section VI; a discussion on the proposed solution is presented in Section VII; and the paper concludes with future directions for this work in Section VIII.

II. RELATED WORK

When our proposed solution becomes operational, it will deal with highly sensitive data that must be managed securely. Since the electronic health record (EHR) [12], [13] is a part of this system, it will contain a lot of personal data, most of which is considered sensitive. This is because all this data will be stored here, from names, addresses, ID card numbers, and insurance numbers to medical histories. Moreover, this medical data will be updated and shared regularly, provided the patient has given consent.

Almost all first-world countries operate their health services through various digital platforms [14]. There are several approaches where EHR systems are enhanced with Blockchain-based services [15], [16]. Various tech giants also offer cloud-based *Software As A Service* (SaaS) solutions [17], [18]. However, the real concerns in this case are *privacy-protecting*, *General Data Protection Regulation* (GDPR) [19], and *performance or scalability* [20], [21]. Comparing some related work in this area, we can see that each service has had to compromise in some cases. Table 2 presents a comparison based on different features of existing systems.

The Distributed Personal Health Record System (PHR) [22] proposed by Roehrs et al. is a decentralized system that functions as a centralized system among participating devices. To describe the concept in more detail, we can use the term *'peer-to-peer'*. The way the author has presented the architecture, particularly scalability, can be a plus point for this system. But when it comes to the concept of a PHR system, it should be controlled by the patients. But the author has also mentioned that different organizations can adopt this system for practice. This raises concerns about security and privacy. Also, in some cases, it violates the GDPR. Bocek et al. [23]

TABLE 2. Comparison of related works.

Reference	Framework	Type	Privacy-Preserving	GDPR	Performance/ Scalability	CIA Triad			AAA Security Model		
						Confidentiality	Integrity	Availability	Authentication	Authorization	Accounting
[22]	Peer-to-peer	Private	✗	✗	✓	✓	✓	✗	✓	✓	✗
[23]	Ethereum	Private	✗	✗	✗	✗	✓	✓	✓	✓	✓
[24]	Hyperledger Fabric	Private	✗	✗	✓	✗	✓	✗	✓	✓	✓
[25]	Ethereum	Private	✗	✗	✓	✗	✓	✗	✓	✓	✓
[8]	Hyperledger Fabric	Private	✗	✓	✗	✗	✓	✗	✓	✓	✓
[26]	Ethereum	Public	✗	✗	✗	✓	✓	✗	✓	✓	✓
[27]	Ethereum	Public	✓	✗	✓	✓	✓	✓	✓	✓	✓
[28]	Custom	Private	✓	✗	✓	✓	✓	✓	✓	✓	✗
[29]	Custom	Private	✓	✓	✗	✓	✓	✓	✗	✗	✗
[30]	Custom	Public	✗	✗	✓	✗	✗	✓	✓	✓	✗
Proposed solution	Hyperledger Fabric	Private	✓	✓	✓	✓	✓	✓	✓	✓	✓

have proposed an Ethereum network-based *proof-of-concept* (PoC) on the pharmaceutical supply chain. However, it is mentioned here that various IoT devices can be used to communicate directly with a Blockchain node server via an HTTP protocol and store this data in PostgreSQL databases. At this point, the lack of decentralization is noted. This is because if attackers breach this single node's database or another cyberattack occurs, data privacy may be compromised. Since there is a risk of data being exposed, most of the requirements of the GDPR are not met. However, this paper covers two types of audit approaches. The solution of Liang et el. [24] is based on Hyperledger Fabric. This has reasonable control over data transparency and can store data in an immutable ledger. But in the architecture, end-user privacy is not the primary concern. While reviewing this paper, some gaps were found in the technical details. The paper makes extensive use of many wearable devices. But how these third-party components share data with their Blockchain network has yet to be adequately described. So, there are many questions left unanswered regarding GDPR guidelines. On the other hand, this solution was developed on Hyperledger Fabric version 0.5, where the private data collection feature was unavailable, and the authors did not propose any custom solutions. So this is a disadvantage for privacy protection. However, since the whole system is cloud-based, there is an advantage in scalability. MediBchain, proposed by Omar et el. [25], is an Ethereum-based solution built on a cloud-based server infrastructure. This is a plus point for the scalability of the system. The cloud infrastructure provides some scalability, but since the stakeholders' data is encrypted, new collaborations and linking of different organizations can be a problem. The appropriate authority requires security and proper control of data transparency. A permissioned Blockchain solution that collects health data from mobile devices was proposed by Ichikawa et al. [8]. This solution is also based on an older version of Hyperledger Fabric, which faces the same problem of private data collection. This thwarts some privacy standards. Moreover, this solution relies on the PBFT algorithm for the consensus mechanism. Thus, with Hyperledger, if an attacker manages to attack the principal of the Blockchain network for more than (N-1)/3 at a time, there is a possibility that the entire Blockchain service can be disabled [31]. MedRec by Azaria et al. [26] is an Ethereum-based solution for decentralized EHR management. This solution has some scalability issues. MedRec does not take care of the security of individual databases, which a local administrator must manage. It also needs to solve the problem of digital rights management. Since it is an open-source project, the author would like others to contribute to its further development and help solve the current issues. The work of Nchinda et al. [27] is a follow-up to Azaria et al. [26]. Both projects are from the same maintainers and have the same name (MedRec). This work [27] is mainly about an EHR management system from MIT. The earlier work [26] is primarily about improving medical record tracking. Scalability issues are solved here. However, patients have more control over their information, even if the network type is public. But *proof-of-work* is used as the consensus algorithm has already been identified as an energy-inefficient process. So the cost rate is high, and as a service, it also depends on third-party miners. Another problem is that personal data is stored outside the chain. Therefore, some users need help with the authenticity of the data. This also violates the basic data protection regulation in some cases.

BCHealth [28] is a custom private Blockchain framework with compelling arguments in the CIA triad as well as for authentication and authorization. But in the case of accountability, their paper has no strong arguments. While there is a brief discussion on health data auditing, there needs to be a detailed explanation for auditing their framework. Privacy preservation was well described in BCHealth and BCPrivacy-Preserving [29] studies. However, BCHealth does not ensure its privacy-preserving policy meets the GDPR standard. On the other hand, the BCPrivacy-Preserving study lacks information on the three A's of security. In particular, performance and scalability have yet to be demonstrated compared to the other studies in Table of 2. HealthBlockEdge [30] is also a custom framework that is also built on public Blockchain. They have demonstrated their performance efficiency and scalability, and their fault tolerance and service availability are well guaranteed. However, the factors of privacy protection, GDPR standards, confidentiality, integrity, and accountability or auditing need to be better described.

HealthCards consist of a membership certificate generated by a Membership Service provider. In our case, we used Fabric CA. And this is one of the requirements for any Blockchain write operations in the system. Another requirement is a temporary token generated via the client app when valid credentials are used on the login page, which expires after some time. These two layers of security make it very robust. This way, we can protect the user's privacy and access more thoroughly. The European General Data Protection Regulation (GDPR) consists of seven principles described in article 5.1-2 of the general data protection regulation [19]. The primary scopes based on these principles are as follows: When a system automatically or manually processes data, it should be for legitimate purposes. Our proposed solution works specifically for the healthcare domain and aims to advance healthcare research and service. GDPR requires that data processing be lawful, fair, and transparent to the data subject [32]. As we design this Blockchain-based solution, maintaining transparency based on the type of organization is the forte of this cutting-edge technology. Our system collects only a citizen's medical data for data minimization and accuracy. Due to its design, it dynamically updates all user data, thus adhering to GDPR principles. The proposed system is an excellent solution to the GDPR's integrity and confidentiality principles. Because the entire system runs on a private Blockchain network, its administration and governance mechanisms are well-established. All the information blocks contain encrypted data at the same time. This also meets the Healthcare Security Regulations (HIPAA) requirements for administrative safeguards [33].

III. PROBLEM STATEMENT

Keeping documents and their preservation for administration are not new in human civilization. For this reason, in the past, we can observe the trend of data storage from cave walls to papers; in modern times, it is servers in digital form. However, nowadays, the main focus is on digital data because of the increasing dependencies and reliability of digital platforms. Any tampering with that copy of the data can be detected when the stored data is in physical form. Even on the printed surface, many security measures, such as passports and currencies, can be taken to detect the document's validity. Both have so many factual printing patterns that they can be easily distinguished from other counterfeit copies. But in the case of digital data, the most significant challenge lies in the fact that it can be altered or deleted without a trace. Suppose an intruder can manipulate the data from the central database. In that case, this application will print a new copy with fake data for other users, which may cause national or international threats.

In Bangladesh, the government has a precise administrative structure and chain of command for maintaining public medical care and health services. The central administration also allocates an estimated yearly budget for maintaining this system. But like all other sectors, this one also has its downsides. Sometimes there are various allegations against some of the system's officials, leading to public suffering in the medical industry. The department has its own quality assurance policy. Since the department is extensive and various organizations are working here, many consumers need help even to approach the higher authorities with their complaints. There is a massive communication gap between the higher authorities and the consumers. Most consumers need to learn that the services they want are among the rights already granted by the department. Nevertheless, when the authorities receive a complaint, sometimes no legal action can be taken because proper documents and evidence are unavailable. As a result, some unethical individuals take advantage of these benefits to earn more money in an unethical manner. Observing these scenarios, the following problems are identified:

- Doctors influenced by private drug distributors.
- Medications provided for free by the government are being sold.
- Medical records are not properly kept. In addition, the records that are kept are difficult to find.
- The procedures for feedback and complaints are very critical.
- As the medical staff does not care about the quality assurance of the services, consumers often have to suffer a lot in hospitals.
- Due to the lack of monitoring, some unauthorized doctors take advantage of the benefits.
- Anyone can easily falsify their qualification status to increase their value.
- Most patients do not know their medical records. As a result, doctors cannot properly investigate their illnesses. Item Most importantly, the consumer's interaction with the service provider and its authority could be more pleasant. Hence, most people switch to private services, and those who cannot afford them must endure countless hours of suffering in public service.

IV. PROPOSED SOLUTION

The government is already familiar with keeping and working with ledgers in paper form. Distributed Ledger Technology (DLT) [34], [35] can be a great solution that can be quickly adopted in this sector as a digital service. Conventional technology may be enough to do normal office work. But we have described the scale of this sector and other problems in the above sections. This sector requires much smarter transparency, correctness, validity, authenticity, and reliability. Most importantly, this service must be more secure at the national level to prevent information manipulation. For this reason, it can be claimed that this sector should be introduced with a new technology commonly known as Blockchain [36], a form of DLT. This technology can be recommended because Blockchain has gained the most attention among other new solutions in industry, government, and academia [37]. It has some specific development standards and a strict set of rules by which it can ensure its security from the beginning. After its functional operation, it generates and maintains a series of blocks by which it can form a ledger with these successive blocks [37]. Even still, there are some examples of financial topics in this article. We want to clarify that neither the payment system service nor any other cryptocurrency [38] related concerns are covered by our suggested solution. Our primary goal in this research is to propose a digital information management system that assures public health care security, transparency, and status.

The solution we have proposed corresponds directly to the national demand in Bangladesh. More specifically, the government of Bangladesh has already published the ''National Blockchain Strategy,'' in which the needs of the health application area have been specified [39]. The government has mentioned the current system's limitations [39]. The importance and demand for a privacy-friendly system are cited as the perfect solutions in this area. The collection, storage, and retrieval of health data while maintaining privacy is the crucial point of the demand raised by the government [39]. In our solution, we provide the same elements, and in our prototype, these features are also functional with the Blockchain solution. Most importantly, Bangladesh has a well-specified and strict law under the ''Bangladesh Digital Security Act'', due to which the government of Bangladesh cannot subscribe to public cloud services on Blockchain. Although these public Blockchain cloud services are simple to manage, they are not suitable for storing sensitive national data. For this reason, Bangladesh is already convinced of the importance of a ''National Blockchain Platform'' hosted by the relevant agency of the Government of Bangladesh [39]. So, we have developed a prototype as a Proof of Concept (PoC) employing a reputed framework, ''Hyperledger Fabric'', hosted on our server and globally redirected through a domain of our DNS server [40], [41], and followed by the use case Health Application Domain. Our proposed solution could be a potential step toward adopting Blockchain technology for the nation. Our proposed system can be set up on our local server and hosted globally with a domain name. So this system can be easily set up in the national data center.

With several government and non-government organizations involved in clinical and biomedical research, a decentralized Blockchain service may be the best choice regarding secure and traceable collaboration with resource management features. With the help of this technology, various organizations can collaborate with other institutions or organizations to share and analyze data without relinquishing control. Each institution can control its computing resources while collaborating with other institutions. This way, a solid data set can be formed for each clinical trial [33], [42]. Fig. 1 represents an overview of this idea of secure collaboration. In this way, patient-generated data becomes available to researchers. As a result, a decent environment, as shown in Fig. 1, can be created where academia, the healthcare industry, and healthcare professionals can collaborate.

In terms of security and privacy, private Blockchain can be an excellent option to control data's transparency, security, and immutability. If anonymized and then tracked in the research process with a timestamp, this secondary data source would enable millions of individuals, healthcare providers, healthcare institutions, and medical researchers to share vast amounts of genetic, nutritional, lifestyle, environmental, and health data with guaranteed security and privacy [33].

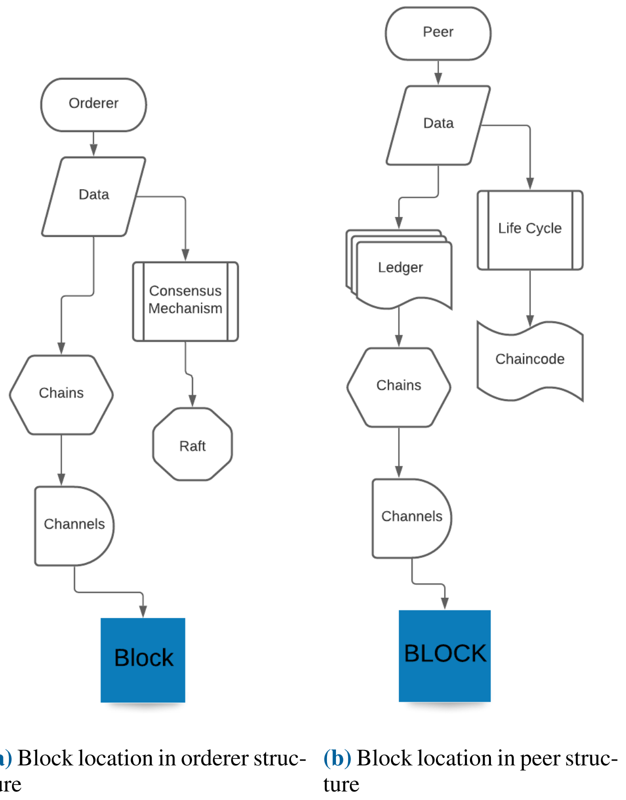
V. SYSTEM DESIGN

The proposed solution utilizes the private Blockchain network, also known as Permissioned Ledger Technology [37]. To demonstrate the proof of concept (PoC) of the proposed solution, a prototype is developed with the renowned Blockchain framework ''Hyperledger Fabric'' [43], [44], which is well suited to the scenario of the proposed solution. The proposed solution includes licensed medications, administrative services, official health data, patient medical histories, and licensed pharmaceutical data, which must be stored securely in the system. Using government data makes it challenging to control data transparency. The service can be simply maintained, and problems with data transparency and control can be handled without a laborious process with a private Blockchain network. In this network, an administrator registers each user and all members of an organization. No malicious actor or anonymous user can join and launch an attack on the system. In the public ledger, anyone can join or leave at any time and is prone to vulnerability. Some instances, such as the one described in [37] and [45], raise privacy concerns in this public ledger. The majority of public ledger systems rely on miners. Dependence on outside entities is detrimental to governmental institutions. Hence, the proposed system utilizes the private ledger to maintain privacy.

The proposed solution is developed with enterprise-grade permissioned distributed ledger technology to ensure that various government agencies and business organizations can take advantage of a DLT system across multiple use cases [37]. This permission ledger technology is primarily known as immune ledger technology [46]. Additionally, this framework holds a record of 3000 transactions per second, dispelling the traditional notion of Blockchain's slow speed and long processing time [43]. The associated design components of this system, including consensus mechanisms, data storage, smart contracts, system stakeholders, system architecture, governance, and privacy concerns, are discussed in the following sections.

A. CONSENSUS MECHANISM

The mining concept is widely used in public Blockchain technology, where miners validate transactions. Instead of using mining, commonly used in public Blockchain technology, the concept of ordering peers is used. These peers are selected to validate transactions and can be one or more in a single channel, with the government organization



(a) Block location in orderer structure **(b)** Block location in peer structure

FIGURE 2. Location of a Block inside the hyper ledger fabric framework in the orderer and peer structure.

serving as the direct owner and authority of the system. The private Blockchain network uses Hyperledger Fabric and Docker to host the peers as a service, making the system more manageable, scalable, and efficient. The consensus mechanism's validation is performed by the Raft [47] algorithm, known for its fault tolerance and quick exception handling. If the leader peer does not respond in time, Raft elects a new ordering peer to serve as a leader and validate transactions.

B. DATA STORAGE

The proposed system is based on distributed ledger technology and uses a special mapping for data storage. The system has two types of data: "on-chain data," which is stored in blocks with a size of 1 megabyte and contains only transactional data [48], and "off-chain data," which is non-transactional and contains more significant amounts of data such as user pictures, doctor certificates, and scanned documents. Fig: 2a and 2b depict the "on-chain data," and Fig. 3 represents the off-chain data structure. For the prototype, off-chain data is kept in a Node.js-based centralized web application but can be managed with IPFS [49] at the production level. The transactions are stored in blocks, and the system uses CouchDB [50] as the World State to quickly invoke chain code or smart contracts for a better user experience.

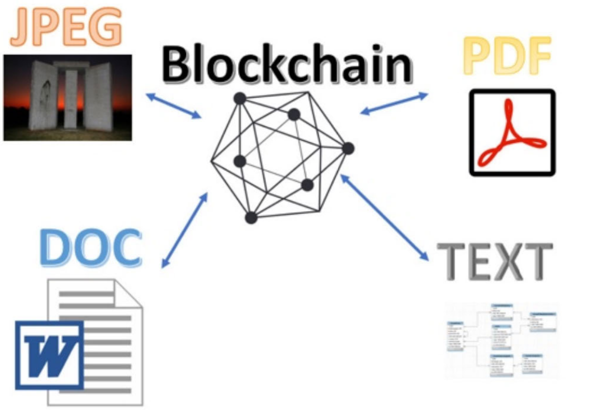


FIGURE 3. Example of Off-Chain data [48].

C. SMART CONTRACT

A smart contract, also known as chain code, is an executable script that contains the system's business logic and runs during execution to control the capability of information to be read and written [36], [46]. It allows setting conditions and logic in the computer code according to system requirements. In Hyperledger Fabric, chain code or smart contracts can be written in JavaScript using provided SDKs. Each supporting peer of an organization has the same chain code, and when a request appears, the chain code checks the conditions and executes changes in the world state (DB) if needs are met. The chain code operation is depicted in Fig. 4. The visual representation of the nationwide Blockchain service can be seen in Fig. 4. This diagram showcases all public and private medical institutions acting as Blockchain nodes and being numerically identified for clarity. Here, each node is a private Blockchain node, and when they work together as a national health service, they are considered a consortium network. Different types of medical infrastructure, such as government and non-government hospitals, health complex districts, and division-level health administration offices, are labeled with numbers 2 to 6. Technically, each node can either be an Endorser, Committing Peer, or both.

Infrastructures such as government and non-government hospitals or health complex districts, division-level health administration offices, etc., can hold their own IT infrastructure and host a node in the Blockchain. This helps establish trust in data immutability and creates a consortium health network layer where all healthcare institutions can have a single, transparent, and open distributed ledger for business and research collaboration. The entity marked as number 1 in the figure represents the administration and is responsible for conducting the consensus process. However, entities marked 2 to 6 can also host their orderer node to participate in the consensus mechanism if they choose to bear the cost of processing power. However, it is not mandatory. The national administration directly maintains the administration node. The number 7 entity is notable for its participation in the consensus mechanism at the leader peer of the number 1 orderer

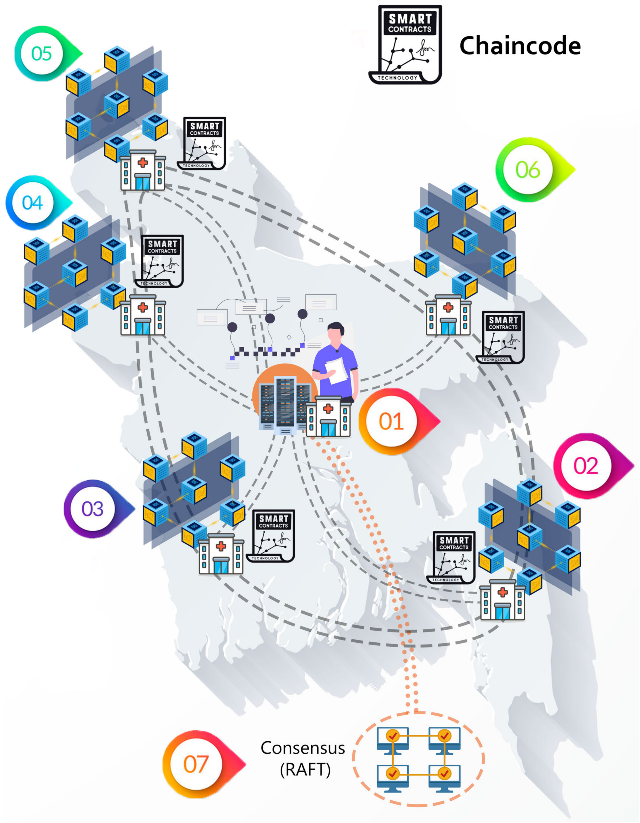


FIGURE 4. A visual representation of national scale Blockchain service where every public / private medical institutions are acting as Blockchain Nodes in consortium network.

node. In the prototype, the consensus mechanism used is the Raft Algorithm, marked as number 7. Steps 2 and 3 in Fig. 5 depict the data flow and governance process within the Blockchain network. The overall data flow and governance are visually represented in Fig. 5. In this diagram, the 'Endorser' and the 'Committing Peer' are shown differently to clarify the idea. However, in our implementation, the 'Endorser' and the 'Committer' are the same peer for each organization. The data flow steps shown in this diagram are as follows:

Initially, the client application sends a request to the SDK server. In our proposed solution, the SDK server is a Node.js server that acts as a shim and can convert the HTTP request into a gRPC request for the private Blockchain network.

1) The converted request with the health card (wallet) signature is then sent to the associated organization's anchor peer, which acts as the endorser.
2) In the next step, the endorser starts the MSP operation to check user validation. If the request is from a valid user, the endorser executes the chain code, and the read-write set is generated against the world state.
3) The endorser then sends the proposed transaction, the executed read and write record of its chain code, and the endorser-signed digital signature as a response to the client application of the SDK server.
4) Then, the SDK server compares the client's proposed transaction with the endorser's signed response body to validate the proposed transaction. If both match, it forwards the endorsed response to the ordering peer.
5) The ordering peer executes all the endorsed requests chronologically, according to the consensus mechanism. After the successful execution of the consensus mechanism, a newly updated block is created.
6) Finally, when the ordering peer confirms the new block, all committing peers add the block to their Blockchain. The header of the new block is created with the combination of the hash of the newly created block's data and the previous block's hash. The exact same process is performed for the next new block. From here, each block is actually linked to a chain of sequences in the Blockchain. So no information can be manipulated at any point in the chain.

The smart contract plays a crucial role in the backend operations of the blockchain service, managing various triggers to improve user experience and speed. The JavaScript SDK was used to develop the smart contract, and the website was developed using the Express.js [51] framework. The system is adaptable and scalable and can be enhanced to gather sensor data from IoT devices for the healthcare industry if privacy and transparency are guaranteed [52], [53]. The proposed system's EHR and smart contracts for different stakeholders are described in Algorithm 1, 2, 3.

D. STAKEHOLDERS

There are three types of users or stakeholders in the proposed system. The application is available for each type of user on different platforms, including web, mobile, and desktop platforms. The stakeholders in the system are as follows:

- Central Authority (In the PoC, BMDC plays this role).
- Doctor.
- Citizen or Patient (In the PoC, this type of user is known as "Nagorik").

1) CENTRAL AUTHORITY

Among these three types of users, The Central Authority can be any organization or ministry of the central government. For example, "Ministry of Health and Family Welfare." This type of user can perform the following tasks from the web portal:

- Authorize Doctors, Medicines.
- Analyze health data with privacy (Anonymous).
- Review complaints submitted by the patients.
- Can verify and approve the update of any doctors' information.
- Can check the government-authorized medicine or drug list.
- Can store data on medicine distribution for future investigation.
- Can store data to analyze doctors' drug prescribing tendencies for the future.

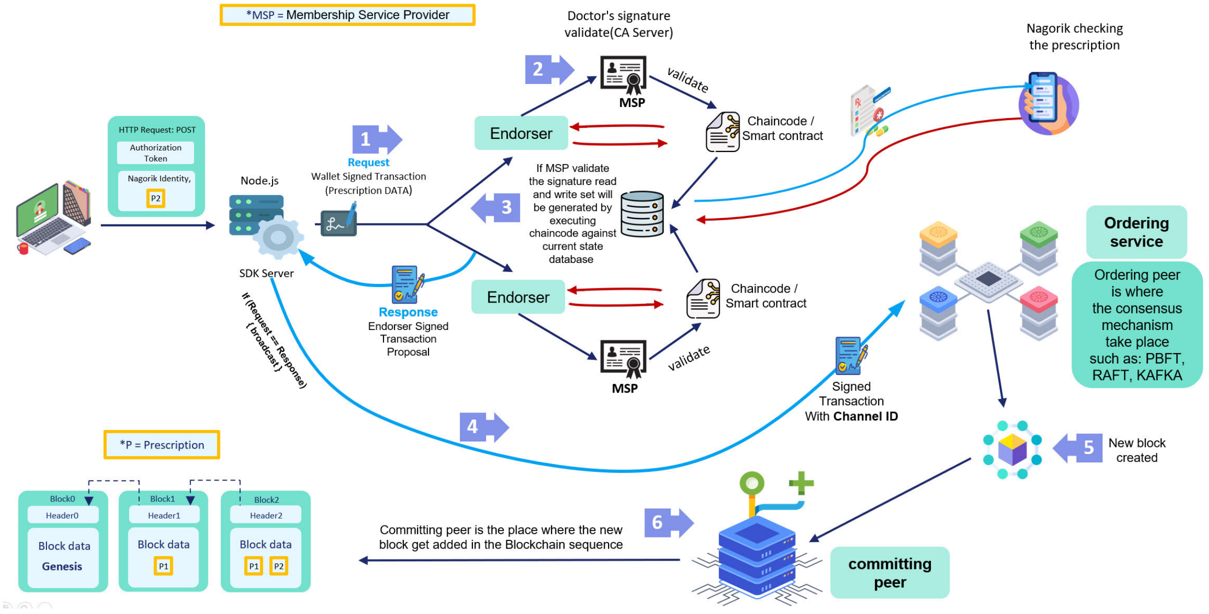

*MSP = Membership Service Provider
Doctor's signature validate(CA Server)
2
MSP
validate
Nagorik checking the prescription
HTTP Request: POST
Authorization Token
Nagorik Identity,
P2
Node.js
1
Request
Wallet Signed Transaction (Prescription DATA)
Endorser
Chaincode / Smart contract
If MSP validate the signature read and write set will be generated by executing chaincode against current state database
3
SDK Server
Response
Endorser Signed Transaction Proposal
Endorser
Chaincode / Smart contract
MSP
validate
Ordering service
Ordering peer is where the consensus mechanism take place such as: PBFT, RAFT, KAFKA
if (Request == Response) { broadcast }
Signed Transaction With Channel ID
4
*P = Prescription
5
New block created
Block0
Header0
Block data
Genesis
Block1
Header1
Block data
P1
Block2
Header2
Block data
P1 P2
Committing peer is the place where the new block get added in the Blockchain sequence
6
committing peer

FIGURE 5. Step by step process of Data flow and Governance with three different peer types inside the Blockchain Network.

Algorithm 1 Pseudo Code of Smart Contracts for Central Authority

```
Smart Contract 1: Adding new doctor to Blockchain
▷ Smart contract for adding a doctor entity
to the BC Network
procedure addNewDoctor(healthCard)
    connPath ← Network configuration path
    gateway ← Network gateway object
    if gateway==“success” then
        inputs ← Inputs of doctor info
        contract ← getContractObj(healthCard)
        json ← doctor data in json
        result ← contract.evaluateTrx(‘qs’, json)
        return result
    else
        return “Invalid Path or Credential”

Smart Contract 2: Approve doctor application
▷ Smart contract for approving a doctor entity
to the BC Network
procedure approveDoctorApplication()
    applicationList ← List of doctor application
    selected ← Select an application from list
    if Approved then
        json ← converted doctor data
        result ← contract.evaluateTrx(‘qs’, json)
        return “Application approved”
    else
        return “Application rejected”
```

- Can view the official news feed for every verified health item to keep the user updated.

Algorithm 2 Pseudo Code of Smart Contracts for Doctor

```
Smart Contract 1: Apply for authorization
▷ Authorization application for doctor entity
procedure doctorApply()
    applicationInputs ← Inputs of doctor information
    if applicationInputs==“valid” then
        result ← Adds application to DB
        if result = ‘success’ then
            return Application submitted successfully
        else
            return “Invalid Contract”
    else
        return “Invalid inputs”

Smart Contract 2: Adding a prescription to Blockchain
▷ Procedure for adding a prescription to the BC Network
procedure addPrescription(healthCard, data)
    healthCard ← Identity Certificate
    contract ← getContractObj(healthCard)
    json ← converted prescription data
    if contract != null then
        result ← contract.evaluateTrx(‘qs’, json)
        return result
    else
        return “Invalid Contract”
```

2) DOCTORS

The central authority user of this system must authorize “Doctor” users. Authorized physicians will be provided with a desktop application. This software can work both online and offline. With this software, the system can keep all the data to

Algorithm 3 Pseudo Code of Smart Contracts for Nagorik/Citizen

```
Smart Contract 1: Doctor Appointment ▷ Smart contract for taking appointment of a doctor by citizen entity
procedure DoctorAppointment()
    SearchDoctor(type, location)
    if Doctor == found then
        slot $\leftarrow$ Select Slot
        if slot == "available" then
            return Slot Appointed
        else
            return Slot is not available
    else
        return "No suitable Doctor found"

Smart Contract 2: Adding new complain to Blockchain ▷ Smart contract for adding new complain to the BC network
procedure addComplain(healthCard, data)
    healthCard $\leftarrow$ Identity Certificate
    contract $\leftarrow$ getContractObj(healthCard)
    json $\leftarrow$ converted complain data
    if contract != null then
        result $\leftarrow$ contract.evaluateTrx('qs', json)
        return result
    else
        return "Invalid Contract"
```

help the doctor in case studies of patients. As the software keeps all the data, the activity of any doctor, also known as a health service provider, can also be analyzed. During general operations, using the software, doctors can perform the following tasks:

- Create or print a prescription and get automatic assistance for suggesting proper medicine.
- Alert notification if any drug or medicine can threaten the current patient.
- Store all general information efficiently to assist the doctor during poor network connection.
- Get fast identification of the patient through their NID card or birth certificate.
- Doctor can update their educational or experience status from the software, and after verifying the validity of that information, it will get published.

3) NAGORIK

"Nagorik" users are the general population of Bangladesh. They are the patients who will seek medical services from the system. Their national identification numbers can easily identify them. Each identified user has an account on the web portal, through which they can access national health services online. The services are:

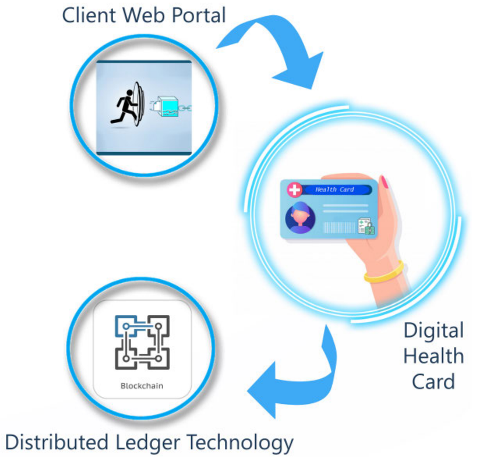


FIGURE 6. The connection between the user and DLT where user can access the Blockchain environment with a dedicated health-card.

- Patients or citizens can log in and log out in their own "Nagorik Portal".
- Can see all their medical records (medical tests, prescriptions) online with full privacy.
- Can check the authorized medicine or drug list and the free medicines and tests.
- Can place a complaint if they are unsatisfied with government medical services.
- Can check their complaint priority status whenever they want.
- Can find a specific specialist doctor according to the patient's problem (verified specialty).
- Can get easy appointments from home so that the doctor and the patient can maintain a convenient schedule.

This suggested approach meets the majority of the needs for healthcare and medical services. It can gather all the information pertinent to this service. Therefore, if necessary, this information can be utilized as proof to look into complaints made against a service provider. All parties can be certain they are upholding their duties because the data is kept impartial. No user can use this system to accuse a service provider falsely.

E. ARCHITECTURE

The proposed Blockchain application has a two-part architecture consisting of the "SDK" part and the "Decentralized Ledger Technology (DLT)" part. The "SDK" part is the connection point between the user and the DLT solution. As Node.js and the Hyperledger Fabric framework have abundant developer resources, we chose to use the Express JS framework in the "SDK" part. This was done as these frameworks and languages are widely used in Bangladesh and can provide a well-maintained, interactive service to connect users to this advanced technology. The flexible design of the "SDK" part allows for a Human-Computer Interaction (HCI) oriented application. Fig. 6 depicts the approach used in the architecture.

In the proof of concept (PoC), a public website was created using conventional web technology and easily accessible through a public domain name. For regular visitors, the website operates like any other. Still, for citizens of Bangladesh with a National Identification Number (NID), the site's backend provides access to a secure national online health service using DLT technology. The portal was developed using Node.js and the Express.js framework. As the proposed solution is private DLT, only authorized users can join or leave the system, which is verified through a special DLT organization. Once registered, users can access the DLT services through the SDK portal using a digital health card. Each request from the user is verified and executed if it follows the smart contract, and the information is then archived in a sequential form of blocks, which is the concept of Blockchain. Fig. 4 provides an overview of this process.

In Fig. 7a, a sequence diagram of our system is shown. First, the doctor submits a registration request to the system, which is then sent to the Blockchain. The Central Authority validates the registration and sends back a confirmation to the doctor. Upon successful access, the doctor can log into the system via a Desktop application, while a Nagorik user can log into the system via a web portal. After logging in, a Nagorik user can request an appointment with a doctor, which is then sent to the Blockchain and returned to the requester as appointment data. The Nagorik user then confirms the appointment, which is updated on the Blockchain. A doctor can request a patient's information and medical history, provided they have obtained their consent. This information is retrieved from the Blockchain and sent back to the doctor through a desktop application. The doctor can also request validation of the medicine and, upon receiving permission, be authorized for market supply. The doctor can prescribe medicine for a patient, and the prescription information is sent to and updated on the Blockchain. Nagorik can also make complaints through the web portal, which are sent to and added to the Blockchain if valid.

The data flow in the proposed system's process is illustrated in Fig. 7b. The process begins with the doctor installing the desktop software, which prompts for credentials. The doctor enters their credentials, and the client SDK sends the request to the endorser peer. If the request is invalid, the client SDK receives a message indicating the invalid request. On the other hand, if the request is valid, the world state is sent back to the client SDK for matching. The desktop software then receives the matching result. If the result matches, the doctor is logged into the system. Otherwise, the request is rejected.

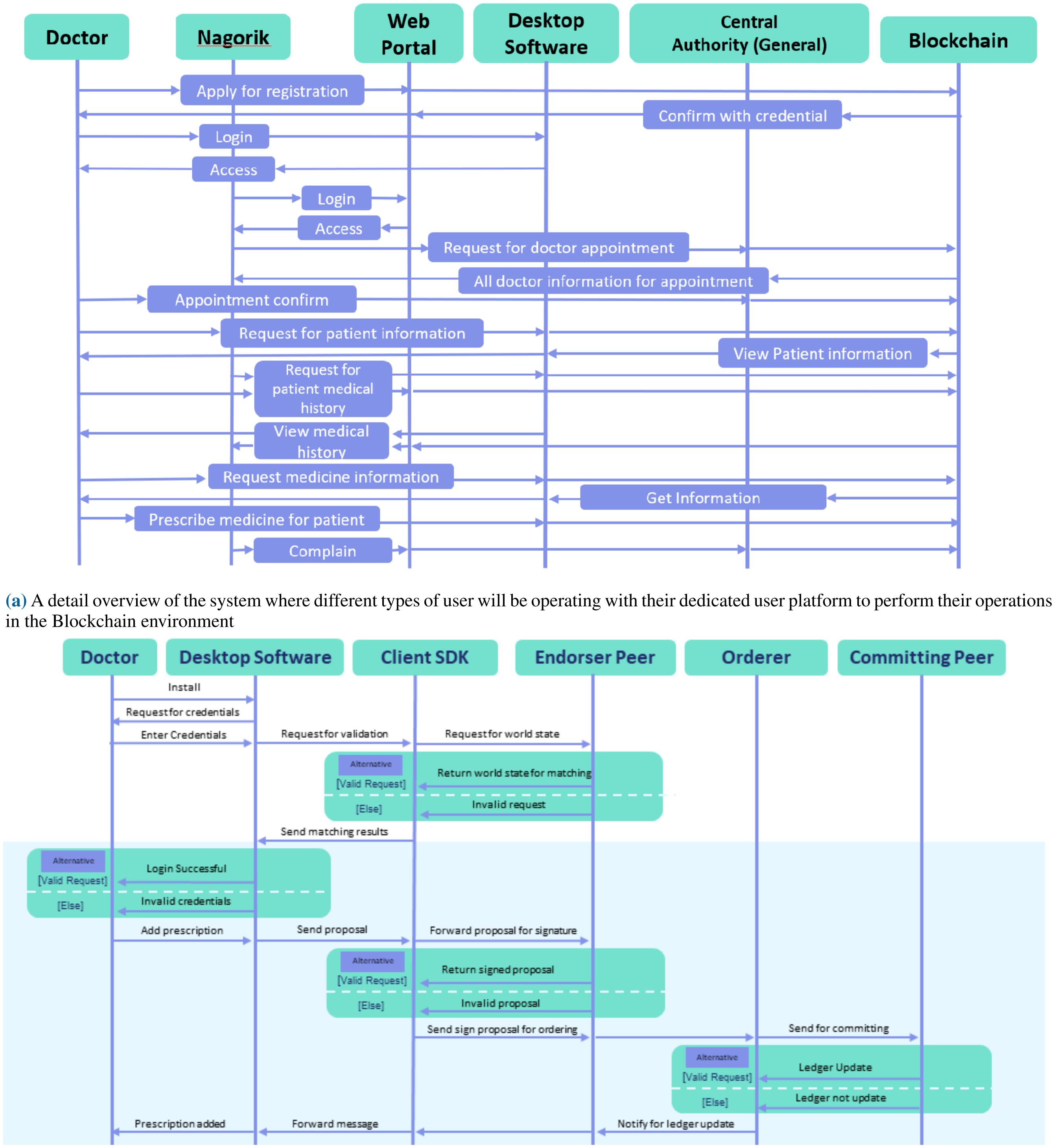
The proposed system has a transparent chain of command to maintain its governance. The basic infrastructure is shown in Fig. 8, which illustrates the example in a virtual machine environment. The figure displays multiple peers holding different system parts, including two "orderer" nodes that use the consensus mechanism (Raft) to create valid blocks in the Blockchain sequence. There are three separate organizations (ports 7051, 9051, and 5051) managing the individual user activities of the system. Each user type can have its own governance organization. These three organizations, known as "Anchor Peers," handle all user or client requests and hold the chain code, executing it on request. If necessary, extra peers can be hosted as "Gossip Peers" to verify the integrity of the system node through peer-to-peer block gossip [46]. Each of the three Anchor Peered organizations has an administrator responsible for registering new users, adding them to the network, and creating a health card for each registered user. Only registered users can access the portal to use the DLT service and perform organizational actions. All user actions are verified with their health cards, and if the verification is successful, the ordering peer verifies it using its consensus mechanism. This is the basic structure of the system's governance, with all organizations identified by MSP, a "Membership Service Provider" that maintains its operation through the use of "Certificate Authority (CA) servers". Fig. 8 shows nodes containing ports 9054, 7054, 8054, and 5054, each with its own dedicated CA server. The orderer also has its own CA server, and all servers can perform the identification process managed by the MSP, including tools for crucial management and node registration [46].

F. GOVERNANCE

The architecture of the proposed system has a well-structured governance structure. Each process in the system has a designated governance role for each node. The administrators of each organization can manage users' registration process and create a health card for each user, with the digital signature distributed by the CA server (MSP). The confirming peers validate the user's identification (MSP) and the operation requested by the user (chain code). If both checks are successful, the "orderer" authorizes the data and generates a new block. Afterward, each peer in the DLT has a dedicated component to commit the block. From Fig. 2a and Fig. 2b, it is evident that each peer and "orderer" maintains an up-to-date block. Fig. 11 shows that users are added to each organization in the private DLT through a strict governance structure. This results in a decentralized ledger technology that operates with a strong, technically automated, and trustworthy governance body where all control is neutral.

G. PRIVACY AND ACCESS

Blockchain-based healthcare services could prevent patients from acquiring copies of their health records or transferring them to another healthcare service provider. Records can be verified after they are signed by the source and added to the Blockchain. This service can guarantee the immutability of patient records using the critical pair concept of the asymmetric key algorithm, for example, RSA. Encrypted data in the Blockchain can only be read using the patient's private key, which allows patients to control access to their sensitive data. This is in line with the European General Data Protection Regulation (GDPR) [54], Bangladesh Digital Security Act [39], and other healthcare security regulations



(a) A detail overview of the system where different types of user will be operating with their dedicated user platform to perform their operations in the Blockchain environment

(b) A sequence diagram of Doctor's Desktop application communicating with three different types of peers of Blockchain Network through SDK based application

FIGURE 7. Platform overview and Doctor's application sequence diagram.

(HIPAA) [33]. The concept of health data exchange between healthcare service providers is not yet implemented in the proposed PoC application. However, it will be implemented in the future, and the system architecture will follow the RSA algorithm for data exchange between stakeholders. In the current PoC, the communication between the nodes is secured by the SSL implementation, and for the text and other media files, SHA-256 hash algorithms are used.

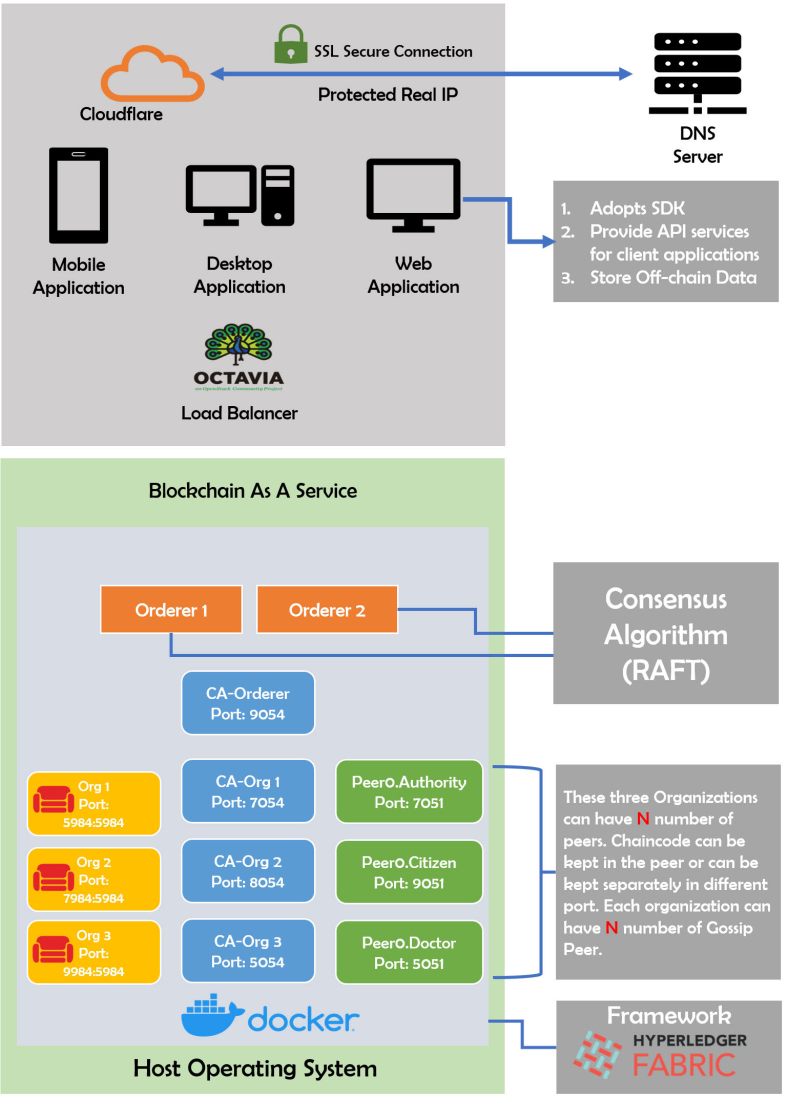


FIGURE 8. System Architecture (DLT represented with Docker Containers).

The proposed system utilized the Hyperledger Fabric framework, which covers CIA, AAA triads, trust, transparency, and other relevant privacy and data integrity issues. These issues are briefly discussed in the following sections.

1) CIA

Confidentiality, Integrity, and Authentication these three can be found at the core fundamentals of the proposed solution.

a: CONFIDENTIALITY

A private Blockchain creates a private network that all actors cannot access or leave at will. And all member organizations in this network must register each user in this private network through this organization's particular certificate authority server. When an actor accesses the network after successful registration, they have limited visibility into the data on the network. Only when an organization is added to a specific channel in this network and all other member organizations of this channel agree to a particular smart contract a user of this organization will have access to this data. There is another concept that two organizations can bridge their private communications. This is called "private data collection" [55]. In this way, this framework can ensure confidentiality at the entire network layer and also at the channel layer. So there is no possibility of unauthorized network access or disclosure.

b: INTEGRITY

Integrity is the most important property of the Blockchain application, and our proposed solution manages it at two levels. Whenever a transaction takes place, the current state of this record can be stored in Worldstate. The transaction record is stored in Blockchain blocks as a distributed ledger at the second level. These values and messages are encoded into each block with the help of hashing algorithm. We use

Fabric as our standard Blockchain framework, so SHA256 is the default for hashing these values [56]. SHA256 is the algorithm that ensures the following three properties of cryptography [57].

- Preimage resistance
- Second preimage
- Collision resistance

The differential properties of this hash function make it so computationally tricky that it is nearly impossible to attack compared to any other algorithm currently in use. So, there is no risk of data tampering.

c: AVAILABILITY

The proposed solution is a collection of multiple peers. Here, all member organizations can have several N peers. And each of these peers has a ledger. In the case of Order Peers, the proposed solution uses Raft as the consensus algorithm. If there are multiple Order Peers in the network, each of them can ensure consensus in the network. Thus, if one organization peer or order peer in the network fails, the others can successfully operate and execute a transaction. So in terms of fault tolerance, the proposed solution ensures service availability.

In the private network of our proposed solution, each organization utilizes a world state that is established using CouchDB. Given that CouchDB guarantees all ACID properties, it can be stated that our proposed solution also offers Atomicity, Consistency, Isolation, and Durability (ACID). If we were to migrate our system to Amazon Web Service, the EC2 instance template of their private Blockchain would support PostgreSQL as the Worldstate [58]. This relational database can also be a suitable alternative to ensuring the ACID property in our solution.

2) AAA

Authentication, authorization, and accounting are the three components that make up the AAA of security. These three A's are also found in our proposed private Blockchain network, which enforces cybersecurity policies according to global standards. Each of these components is explained below:

a: AUTHENTICATION

In our solution, there are two phases of authentication. The first phase is a basic authentication process using a username or identity and a password verified by the Web 2 layer (SDK server). The second phase involves authentication by the CA server of our private network, also known as the Membership Service Provider (MSP). Anonymous access is impossible as each member of the private network is identifiable by their organization's special CA server (Certificate Authority).

b: AUTHORIZATION

After gaining authorization, the user can enter the private network. It's worth noting that each user/member is affiliated with an organization in the network. The channels outline the extent of an organization's activities. Hence, users/members are only permitted to function in regions where their organization is part of a channel and where a smart contract enables their organization to carry out operations. Our proposed system can handle authorization from the Blockchain layer and validate it through the client service of the Web 2 layer, resulting in dual layers of authorization based on the user's or member's role, group, and category.

c: ACCOUNTABILITY

As we mentioned, the proposed solution can perform authentication and authorization from the Web2 (SDK server) and Web3 (private Blockchain network) layers. Our system can also provide accountability across these two different layers. We used the Node.js SDK in our Web2 layer. It already uses an established framework and database. For auditing the Web2 layer, we can use the system log, which stores all the actions performed by the user and can be retrieved during an audit.

And for the Web 3 layer, our private blockchain network, we can follow the Best Practices for Smart Contract Security-Hyperledger Fabric 2.0 from Cloud Security Alliance 2022 [59] before moving our prototype to production. In particular, from the chapter '6.2 Smart Contract Audit' and '6.3 Secure Smart Contract Lifecycle (SSCDL)'. In the initial phase, these five steps can be followed to ensure accountability through an audit.

1) **Expert Code Analysis:** Have a smart contract expert perform a review. This is a common practice in today's software industry. Managing a version control system with a group of people, managing multiple branches, and having experienced developers review the code before it is merged is already an established process.
2) **Control Flow Analysis:** Create a diagram of the program's control flow and look for anomalies. Currently, there are many popular automation tools that can also check coding style best practices and conventions.
3) **Docker-Bench-Security:** The Docker Bench for Security is a script that reviews dozens of common best practices for deploying Docker containers in production.
4) **Test Coverage:** Ensure that unit tests cover all smart contract codes.
5) **Comments:** Smart contracts should include clear comments on functionality. If developers can comply with appropriate comments, business owner representatives can review the smart contract before it is installed on the channels.

These five steps can ensure first-level accountability for our proposed solution. Even in most cases of CI/CD pipelines, clean code with meaningful naming conventions can also be helpful for review by the business owner's technical auditor. Appropriate comments on a code can help the commercial auditor confirm the technical auditor's review. With all of these steps, our proposed solution can ensure accountability.

Then, to meet all accountability requirements in our system, we can choose any SIEM (Security Information and Event Management) system on a production server from a trusted vendor like IBM or Microsoft.

VI. SYSTEM IMPLEMENTATION

This section describes the implementation of the proposed system. This section provides details on the implementation of the infrastructure, security, network setup, and application layer. This system also discusses Blockchain as a service in the context of the proposed solution.

A. INFRASTRUCTURE

The implementation of the proposed physical infrastructure necessitates several specific components. The establishment of the system necessitates a reliable Internet connection, an Optical Network Unit (ONU), a 2.4 GHz Wi-Fi router, and a standard desktop computer to serve as the hosting server. The specifications for the hosting server utilized in the proposed solution are outlined as follows:

- Random Access Memory (RAM): 16 GB
- Storage device: 256 GB SSD
- CPU: Intel Core i7 (7th generation, 4 Cores processor with 8 threads, 3.90GHz Frequency with 8MB Cache)
- Power Supply Unit (PSU): 650 Watt

The current prototype requires no additional cooling system due to its limited number of users and traffic. A public IP is assigned to the router for hosting the proposed solution, with port forwarding configured for external port 80 for the full service. To protect the public IP and port number from the internet, the prototype uses Cloudflare as a CDN service [41], [60]. Cloudflare serves as a security measure, providing SSL security protocol [61], [62] for an encrypted connection between the web server and the client browser, as depicted in Fig. 8. Additionally, Cloudflare has a feature to monitor and protect the server from DDoS attacks [63]. If abnormal traffic is detected, "Under Attack Mode" is enabled, rendering all API services inaccessible and requiring Web clients to pass a CAPTCHA test [64] to access the website.

In the event of a production-level deployment, the proposed system includes various security measures, which are discussed in Section VI-B entitled "Security Analysis."

B. SECURITY ANALYSIS

In the rapidly evolving landscape of Blockchain technology, security remains a critical concern for its widespread adoption and implementation. As Blockchain applications have the potential to store sensitive and valuable data, it is imperative to ensure that they are secure and protected against various threats and vulnerabilities. This section aims to provide a comprehensive analysis of the security aspects of the proposed Blockchain application solution, covering the various security concerns and the measures taken to mitigate them. This security analysis aims to ensure the robustness and reliability of the Blockchain system and instill confidence in its users.

1) HYPERLEDGER FABRIC's SECURITY MODEL

The consensus mechanism of HyperLedger Fabric is highly customizable, allowing it to be adapted to the specific trust requirements of a deployment or solution. This flexible architecture allows for the utilization of established CFT (crash fault-tolerant) or BFT (byzantine fault-tolerant) ordering tools. In addition to consensus, HyperLedger Fabric has a robust security model unique to a permissioned Blockchain, where entry is restricted, and actors must have a digital identity verified by a Certificate Authority and issued in the form of an X.509 digital certificate. These identities, managed by a trusted membership service provider (MSP), are crucial to security as they dictate the permissions and access actors have within the Blockchain network. The network also employs policies, managed through configurable agreements, as a means of infrastructure management, determining the acceptance or rejection of changes to the network, a channel, or a smart contract. In addition to the security provided by the web3 infrastructure, the proposed solution also includes a web2 client app for users to access the Blockchain network as needed.

2) WHY RAFT CONSENSUS MECHANISM IS CHOSEN

The Hyperledger Fabric (HLF) is a permissioned platform that prioritizes confidentiality through its channel architecture and private data capability. Transactions are ordered using the consensus algorithm, which operates independently from the peers responsible for executing transactions and maintaining the ledger. The modular design of the consensus algorithm enables customization to meet the trust requirements of a specific deployment or solution. As a result, the platform can utilize well-established toolkits for crash fault-tolerant (CFT) or Byzantine fault-tolerant (BFT) ordering. To ensure fast transaction speed, scalability, and compatibility with large consensus groups, a suitable consensus algorithm was selected, considering the framework's support and features. A comparison with some popular consensus models is presented in the following text.

- The PoW (proof of work) algorithm, though widely utilized, is not a requirement for permissioned blockchain systems and is energy-intensive in nature [65].
- Proof of Stake (PoS) addresses many issues present in Proof of Work (PoW), but it is more fitting for permissionless blockchain systems [66].
- The heavy communication among nodes in pBFT (practical byzantine fault tolerance) limits scalability and restricts its suitability to small consensus groups [67].
- Proof of Importance (PoI) boasts high speed and low computational requirements, but its unequal distribution of nodes creates the potential for influential nodes to impact the network negatively [67].

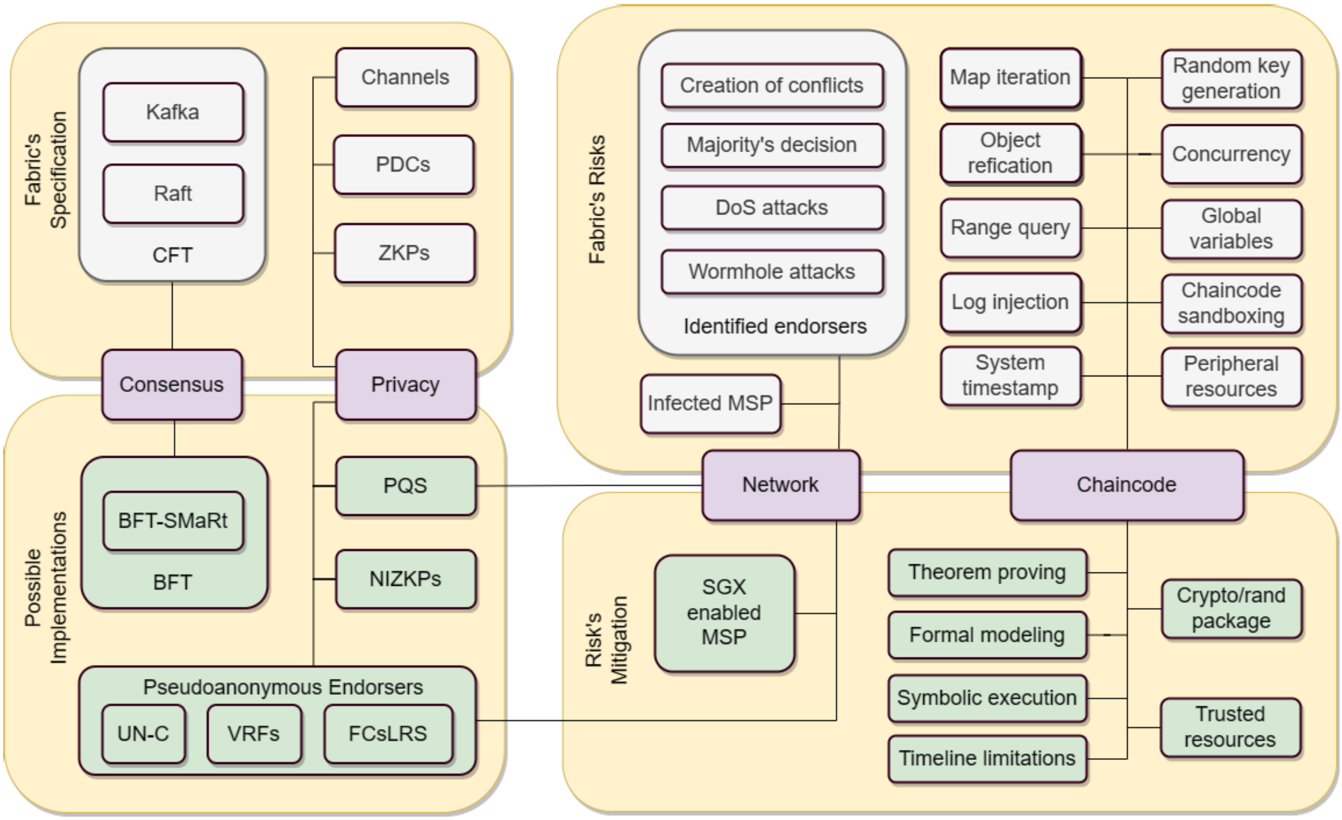


FIGURE 9. Security Architecture of Hyperledger Fabric [69].

Finally, we selected Raft as the consensus algorithm for our permissioned blockchain network due to its high transaction per second (TPS) rate, scalability, suitability for large consensus groups, and support from Hyperledger Fabric, which endorses its use.

3) THREAT MODELING AND SECURITY ANALYSIS

The Hyperledger Fabric (HLF) is widely regarded as the most deployable distributed ledger, according to the Block Data report from October 2022 [68]. It stands out as a superior option for enterprise-based environments compared to other permissioned Blockchain solutions [69]. The security architecture of the enterprise-grade framework can be seen in Fig. 9. Key attributes are outlined below.

a: CONSENSUS CHALLENGES AND OPEN ISSUES

The Hyperledger Fabric (HLF) framework offers a pluggable ordering service that determines the order of transactions and groups them into blocks. During the research, Fabric adopted the Crash Fault Tolerant (CFT) based Raft protocol as its core ordering service and updated its Ordering Service Node (OSN). The Hyperledger Foundation itself encouraged using Raft [70].

In a permissioned network, where each peer is accountable for its behavior, there is an incentive for nodes to follow the protocol. Despite the tolerance provided by the consensus protocol (CFT or BFT), Fabric's design can mitigate common and sophisticated consensus-oriented attacks, such as double spending, in an enterprise environment [69].

Until 2021, there was no dedicated production-grade Byzantine Fault Tolerant (BFT) ordering service for Fabric. However, in 2021, IBM announced its development of a dedicated BFT Ordering service for Fabric, presented at the IEEE International Conference on Blockchain and Cryptocurrency (ICBC) [71].

In the Raft-based consensus, only the elected Node (Leader Peer) receives the transaction proposal as ordered. This is the default mechanism. However, the Client SDK can be modified to submit the transaction to all Orderers, preventing censorship attacks by malicious Orderers. This approach was tested by the IBM research team at both Local Area Network (LAN) and Wide Area Network (WAN) levels [71].

This modification can also minimize the threat of Byzantine Attack [72]. Suppose a fully trusted follower or Candidate Node turns rogue and has passed all authentication and verification processes (Membership Service Provider - MSP). In that case, there is a low chance of it being elected in the Raft's Leader Election segment.

This work focuses on proposing a solution for a national-level health administration and citizens to overcome practical challenges through a neutral, decentralized governance system and is not concerned with research on the topic of consensus challenges and open issues.

Earlier, we mentioned that HLF provides a customizable ordering service. For our research and proof-of-concept implementation, we have selected Raft as the ordering service, and if this proof-of-concept is approved for production, we can easily switch to a new BFT-based ordering service with minimal effort. As shown in Fig. 8, the Blockchain nodes are containerized using Hyperledger's docker images, making it simple to update to the latest version in the production environment if the IBM team or the Hyperledger Foundation commits any core changes. This allows us to adopt new changes at the protocol level while keeping the smart contract and its business logic unchanged.

b: SMART CONTRACT SECURITY

Smart contracts in Fabric are not written in domain-specific programming languages like Ethereum's Solidity, so the risks and vulnerabilities may differ from those associated with general-purpose programming languages. For more information on the specific threats and their solutions regarding smart contracts or chaincode, references can be made to papers such as [73], [74] and [75]. Key topics of smart contract security concerns include random key generation, object reification, system timestamp and concurrency, and chaincode sandboxing. By focusing on these topics, we can minimize potential threats to the security of our smart contracts.

c: NETWORK THREATS AND FABRIC SPECIFICATION

In a permissioned Blockchain network, administrative permissions are handled so that the risk of anonymous attackers posing as trustworthy is greatly reduced. The potential threats that may arise from a compromised MSP were addressed in a recent study through the use of Intel Software Guard Extensions (SGX) [76]. The SGX remote attestation techniques and the certain execution features offered by this method can identify each system entity as a trusted node.

d: PRIVACY TECHNIQUES AND FABRIC SPECIFICATION

According to the Fabric specification, there are three general types of privacy managing techniques, namely, Channels, Private Data Collection (PDCs), and Zero-Knowledge Proofs (ZKPs).

Channel: A set of peers manages each channel, and it is associated with some policies that provide access to the corresponding resources. When a peer registers for a channel that is characterized by a unique identifier, the corresponding ledger is created and run on this peer, allowing it to manage an identical and consistent data store with the rest of the channel's peers [77].

Private Data Collection (PDCs): The PDC is created to provide peers with the capability of endorsing, committing, or queering private data without being forced to create a new channel. Private data collections can be defined as a subset of organizations on a channel [55].

A collection is the combination of two elements, one is the actual private data, and another is a hash of that data. The actual private data is stored in a private state database on the peers of authorized organizations, which can be accessed from the chaincode on these authorized peers. When a transaction is endorsed, ordered, and written to the ledgers of every channel's peer, the generated hash acts as evidence. The hash is used for state validation and can be used for audit purposes.

Zero-Knowledge Proofs (ZKPs): It is a method aiming at asset management along with audit support, and it is not considered in our Proposed solution.

C. NETWORK SETUP

To set up the personal server, two things need to be configured. One is a *router network configuration*, and the other is a *virtual box network configuration*. Multiple virtual machines need to be able to connect to physical and virtual networks through their *virtual network adapters* in the Virtual Box GUI. In this process, Virtual Box connects to the network through various processes. Virtual Box provides many different types of networking modes, as follows:

- NAT Network [78]
- Bridged Adapter [79], [80]
- Internal Network
- Host-Only Adapter
- Generic driver

Here, a bridge adapter is used to configure the network for application access.

Bridge Adapter: A bridge adapter is used to connect the virtual network adapter of a VM to a physical network to which the physical network adapter of the VirtualBox host machine is connected. The host machine is used to connect to the WiFi network. The following steps describe the process to enable the bridge network.

1) First need to select the settings option from the selected VM.
2) Then select the network option from the GUI.
3) After that, attach the adapter option of the bridge adapter network mode.
4) Then, the wireless network adapter option needs to be selected through the bridge adapter option.

As a result, the guest computer has an IP address that is the same as the host computer's. This IP address is assigned via the router configuration. The router is connected to a public IP, like 103.96.37.122. Now the router has assigned multiple private IP addresses to multiple machines. Here, the IP 192.168.0.109 with port 80 is assigned to the guest machine (Linux) through the bridge adapter in network mode. However, no one can access the application with a public IP as NAT (Network Address Translation) is not configured on the router to reach the virtual machine.

To configure the router, it needs to access the router's interface. This interface is accessed via the IP 192.168.0.1. After passing the login page, the GUI of the router configuration appears in front of the user. Then, select the NAT forwarding option. Next, select Virtual Server and specify the external port and the internal port with the private IP of the guest machine. This completes the port forwarding mechanism. Now the personal server setup is complete, and the application can be accessed globally using the public IP.

D. APPLICATION LAYER

At the top level of Fig. 8, three different platforms are mentioned, namely mobile, desktop, and web applications. This system supports cross-platform capability by using an SDK and an API gateway. As mentioned earlier, the proposed system uses the Hyperledger Fabric Framework and JavaScript SDK while maintaining Object Oriented Programming (OOP) standards. *Inheritance* is largely used here for adopting SDK components into the NodeJS environment. **"fabric-contract-api"** [81], **"fabric-ca-client"** [82] and **"fabric-network"** [83]; these three modules of Hyperledger Fabric are mostly used in the implementation of the prototype system.

1) FABRIC-CONTRACT-API

This module is used when a smart contract needs to be installed in a peer or for initializing the smart contract or chain code.

2) FABRIC-NETWORK

This is one of the most important modules that connects the digital *healthcard* (known as a wallet in Fabric) to the application layer of the proposed system. Fig. 6 depicts the overview of this module. From logging a user into the private Blockchain network to creating a *healthcard* for that user, this module is widely used. It is also responsible for verifying the *healthcard*. Fig. 11 represents the complete workflow of this user registration operation. This module also has another object called *gateway*. This object connects a user to the Blockchain network, which returns a *contract* object. It helps the client or user perform read and write operations in the smart contract. The authentication process for user login is also controlled from here. Fig. 12 exhibits this process step-by-step for better understanding.

3) FABRIC-CA-CLIENT

This module is responsible for establishing communication with the CA server. Whenever **fabric-network** is used over the *healthcard*, the module **fabric-ca-client** is called each time to authenticate the user and his request. All these processes are managed by the Node.js environment running on internal port 3000. In the Node.js environment, the main web application that actually works with the SDK is developed using the **Express** framework, which is actually an MVC-based web application framework. This web application acts as a portal between client applications and the Blockchain service. This web application has its own client interface and also an API service. This is the actual web service that is redirected by the router to the external port 80. For patient flexibility, a mobile application for our Blockchain service has also been developed for the Android and iOS platforms. For this purpose, flutter [84] is used. This mobile application uses the API services provided by the Express web application [85]. The physicians' software is developed using C#, a Windows-based desktop application software that uses the same API services provided by the Express-based web application to access the Blockchain services.

E. BLOCKCHAIN AS A SERVICE

This section describes the architecture and implementation of the Blockchain As A Service part of the proposed system. Fig 8 presents the overview of the proposed system.

1) FRAMEWORK

For robust privacy and security features with support for granular access control, private channels, and good documentation, we chose Hyperledger Fabric to implement this system [43]. The framework version v2.1.0 is utilized for the system's development. This framework has pre-built *docker images* [86] to support the development process and ensure all Blockchain standards are adhered to. Pre-built docker images are read-only templates that create docker containers with different configurations already implemented in those containers. Fig. 8 shows the architecture of the Fabric framework, which contains the entire Blockchain service with many Docker containers.

2) VIRTUAL MACHINE ENVIRONMENT

It is easier to run the Blockchain application using a Linux-based operating system. The Hyperledger framework is also configured based on the Linux OS. So the first task is to create a Linux-based environment. A virtual machine environment is required to run a Linux-based guest operating system from a Windows-based host operating system. In the proposed system, *Oracle Virtual box* [87] virtual machine platform is used to set up the Linux operating system on the Windows-based machine. Then, all the needed programs, like Docker and Hyperledger Fabric Framework, are set up on the Linux-based OS running in the virtual machine environment. It is true that all these clients have their own domain names. However, since this implementation is a prototype, the individual domain names are identical only in the Docker environment. These Blockchain peers are virtually decentralized on a single physical machine, and the entire Blockchain service is hosted live under a single domain name, using the router's external port 80. The VM environment implementation of the "Blockchain As A Service" concept is shown in Fig. 8.

3) DOCKER CONTAINER

Within the VM, Docker is used to automating the deployment of the application as a lightweight container, allowing the application to work efficiently in different environments. It provides a variety of processes with a virtualization system architecture. This makes it easy to deploy an application in a decentralized architecture on a prototype-level Blockchain service.

4) CA SERVER

The most important part of this permission-based private blockchain network is the CA server, also called a Certificate Authority Server. This particular service is responsible for identifying and certifying each "orderer", organization, and user. For the implementation of this system, these services are also hosted in different Docker containers on a single physical machine. In Fig. 8, we can see that there are four containers hosting these servers in four different ports (9054, 7054, 8054, 5054). It can be seen that there is a dedicated CA server (port: 9054) for certifying the "orderder" peers, and the other three organizations also have their own CA server. Each peer admin and the user has to get approved and verified through the dedicated CA server they belong to.

5) WORLD STATE

In Fig. 8, the Docker containers highlighted in yellow are the World State of the proposed system. These World states also run on Docker containers on ports (5984, 7984, and 9984) identified as Org1, Org2, and Org3. Only the organization's peers have their World State for fast queries and client support.

6) ORDERER

In the case of "Orderer" in our private blockchain network, an 'N' number of "Orderer" can be created. Because after each transaction, one of the peer orderers is the actual private blockchain authority that automates the verification process in the system. In this system implementation, we have created only two (2) orderers inside our system. In Fig. 8, these are marked with orange (Orderer 1 and Orderer 2). These two orderers run on ports 7050 and 8050.

7) USER ENROLLMENT PROCESS

In the proposed system, user enrollment can take two forms: adding a new organization to the Blockchain service and adding a new user to an organization. The process for both scenarios is briefly described in the following sections.

a: ENROLLING AN ORGANIZATION IN THE BLOCKCHAIN SERVICE

In the proposed solution, scalability is well designed, as the entire solution is based on the Hyperledger Fabric framework. In the future, if more entities need to be added to the system, they can simply be added to the private network as an organization peer, which is also an anchor peer. For this procedure, the system authority only needs to set up another certificate authority server for this particular organization. This CA server will handle the rest of this new entity's authentication and authorization operations in the private network. On the other hand, when the new organization is added to a private network channel, it can also interoperate with other peer organizations according to the smart contract business logic. The entire organization enrollment process is shown in Fig. 10.

b: ENROLLING USERS INTO AN ORGANIZATION

The implementation process for registering new clients is a very interesting part. With a private Blockchain network, it is very important to prevent bad actors and keep all actors and visitors identical. The implementation process is shown in Fig. 11 ensuring the standards are followed. Here, *CA server* acts as a "Membership Service Provider (MSP)". Each organization that is part of this Blockchain service has a **Default Admin**. This role, known as **Admin**, is responsible for authorizing a new user in the corresponding organization. And when a user is added, he/she must have a special *Digital Health Card* for continued existence in the system. An **admin** can create both an **user** and an **admin** user in the system. But an **user** can only perform its general operations in the system. From Fig. 11, it can be seen that the **X** organization has an **Admin** that signs up a new **User**, with *CA Server* serving as *MSP*. This collaboration has created a new *health-card* for a new user. This mechanism implies that an actor cannot gain access to or leave a private Blockchain network of its own volition. The entire process can be expressed with Algorithm 4.

8) LOGIN AUTHENTICATION PROCESS

User login authentication is the phase where the implementation process takes a new turn. Although the Hyperledger Fabric has excellent documentation. But merging a Model View Controller (MVC) framework with a completely new technology (Blockchain) for a better integration experience, there were so many difficulties in the development process. In the absence of proper documentation, this area was mostly developed by ourselves.

The whole process of login and authentication is presented in Algorithm 5 and Fig. 12 with a flowchart containing the following steps:

1) Start
2) Get credentials (identity and password) from the user as inputs.
3) check if identity exists in health card if "NO" then print "Invalid Identity"
   else
   get password from World-state using identity.
4) check if password matches if "YES" then grant access otherwise print "Invalid Password"
5) end

This whole operation is managed using the SDK, where *Health Card* is the key to accessing the Blockchain network.

VII. DISCUSSION

The National ID Card of Bangladesh is a perfect example to prove that we are ready to adopt private Blockchain applications at national level. As we have seen, the whole nation already has a unique identity. At least those who do not have

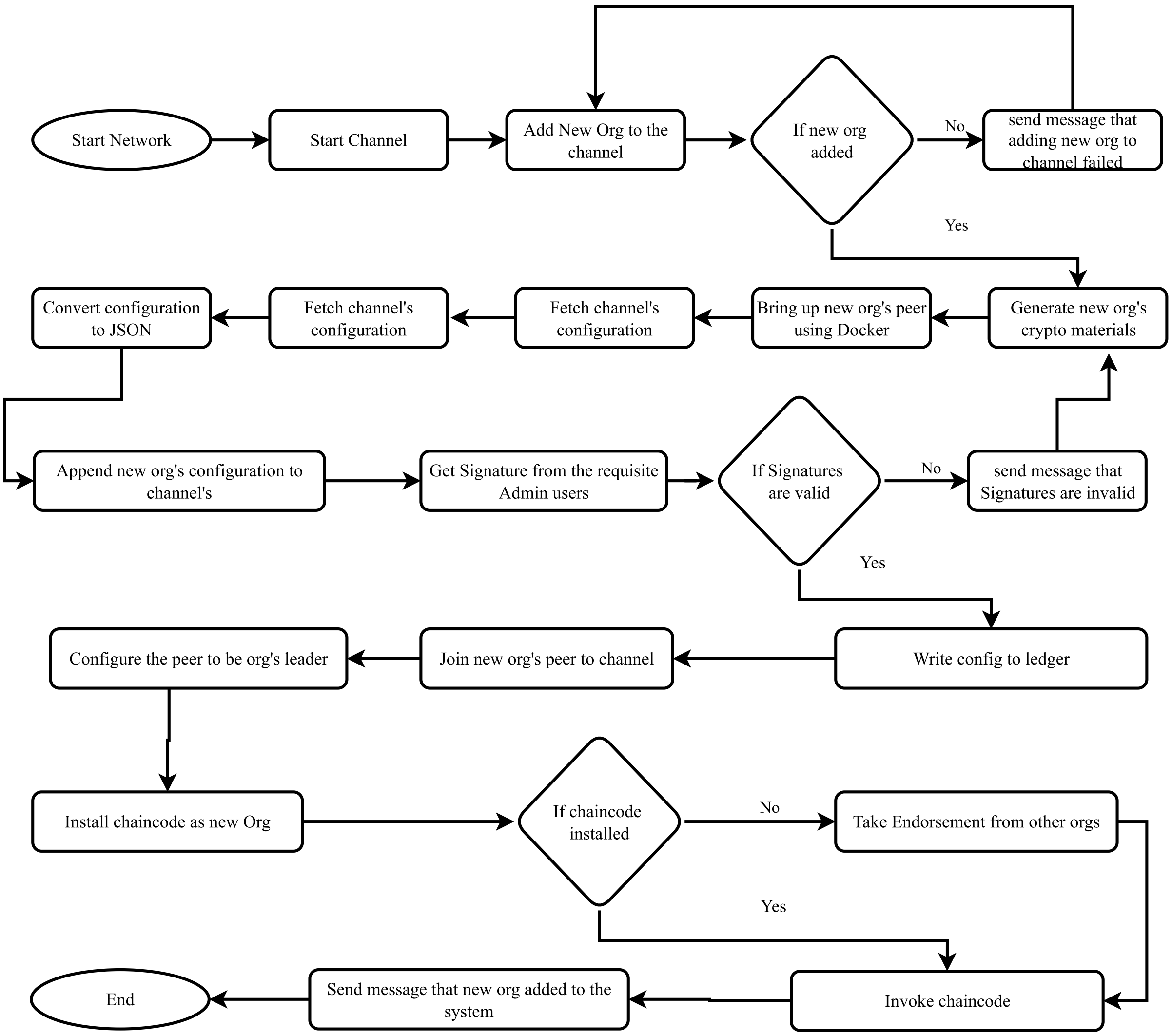

Start Network
Start Channel
Add New Org to the channel
If new org added
No
send message that adding new org to channel failed
Yes
Generate new org's crypto materials
Bring up new org's peer using Docker
Fetch channel's configuration
Fetch channel's configuration
Convert configuration to JSON
Append new org's configuration to channel's
Get Signature from the requisite Admin users
If Signatures are valid
No
send message that Signatures are invalid
Yes
Write config to ledger
Join new org's peer to channel
Configure the peer to be org's leader
Install chaincode as new Org
If chaincode installed
No
Take Endorsement from other orgs
Yes
Invoke chaincode
Send message that new org added to the system
End

FIGURE 10. **Organization enrollment process in the Blockchain network.**

their NID can be uniquely identified by their birth certificate. And as a practical example, all banks in Bangladesh are already working with their customers with one of these two documents. The government provides its web services to verify these documents online for financial institutions. In some cases, APIs are also provided by the government for these services. So, we can say that one of the major challenges in implementing permissioned Blockchain has already been solved. On the bright side, we no longer need to invest in registering every citizen in our system. Simply working with the "National Identity Wing" can save both investment and time. In this way, we will be able to include not only health officials but the whole nation in our health network.

At the moment, there is no competitor to our proposed solution in Bangladesh. But as the demand for DLT is increasing day by day, the ministry may need this kind of technology for its office work in the future. So if the ministry considers the proposed solution as an underdeveloped product and waits for a well-known third-party company to provide a standard enterprise service, it could be a competition for us. But that could be a risk, a breach of our national data security policy. Because then all the information of government health officials and citizens' medical records will end up in the hands of third-party vendors. But our proposed solution and PoC is not about competition and business. It is about the national demand to advance technology for humanity. By applying more R&D to this PoC, the government of Bangladesh can build its own DLT-based healthcare service that can be set up and maintained in the fourth-tier national data center, which could be a step towards the cloud-based national blockchain platform program [39] of Bangladesh.

The proposed application uses the Hyperledger Fabric framework. Nowadays, there are many different Blokchain frameworks or platforms. Among them, the Hyperledger Fabric Framework was chosen for several reasons, including the distribution nature of the framework, security, scalability,

Algorithm 4 Pseudo Code for User Registration in an Organization

```
procedure registerUser(orgName, identity) ▷ Procedure for registering an user into an organization
    orgMSP ← Load MSP ▷ Based on Orgranization
    connPath ← Network configuration path
    ca ← new CA client
    admin ← Admin's identity certificate
    healthCard ← user's identity certificate
    if healthCard == exist then
        return "heathCard Already Exists"
    else
        provider ← Admin's Identity Type
        secret ← Generate from CA
        enrollment ← From CA with secret and identity
        x509Identity ← x509Identity object
        newHealthCard ← Generate new Health Card
        if newHealthCard == "created" then
            return newHealthCard
        else
            return "Failed to register User"
```

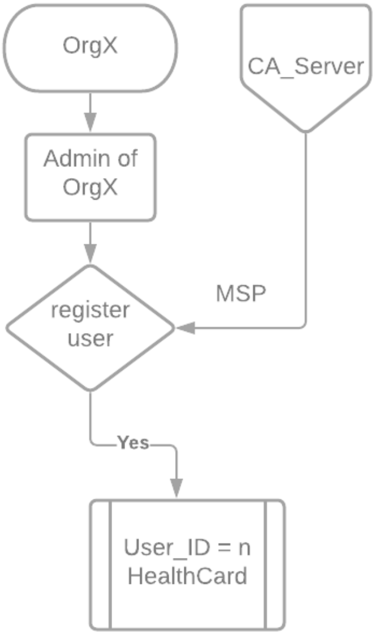


FIGURE 11. User Enrollment process in an existing organization.

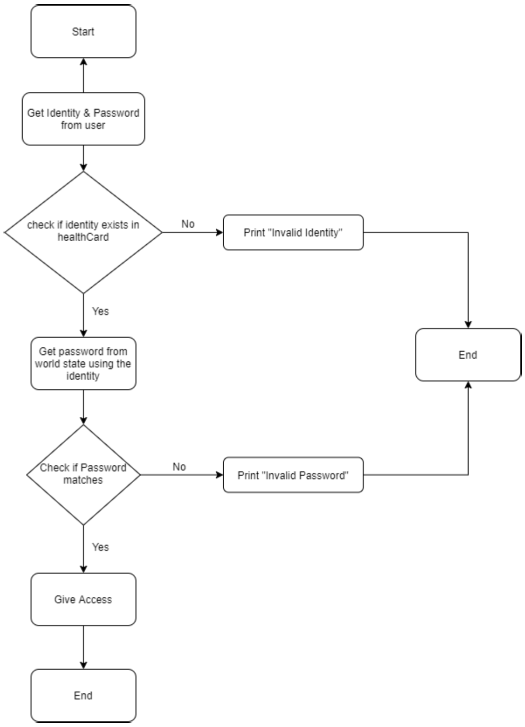

Start
Get Identity & Password from user
check if identity exists in healthCard
No
Print "Invalid Identity"
Yes
Get password from world state using the identity
End
Check if Password matches
No
Print "Invalid Password"
Yes
Give Access
End

FIGURE 12. User login authentication process diagram (SDK).

transaction speed, and transaction cost. The distribution nature of the chosen framework is open source and supports all modern security measures with the option of scalability. The transaction speed of the chosen Hyperledger Fabric framework is much higher than the existing framework, namely 3500+ transactions per second, without the need for an incentive in the form of cryptocurrencies. The summary of the comparison between the Ethereum, Hyperledger Fabric, Quorum and Multichain platforms is shown in table 3.

The proposed Blockchain application is tested using a load testing approach. The testing and analysis followed the APDEX (Application Performance Index), an open standard developed by an alliance of companies to measure the performance of software applications in the computing domain. With the aid of JMeter, we have done the load testing to measure our proposed system's throughput. For this testing process, 100 threads have been created. where the ramp-up period is 10 seconds. The system was therefore receiving 10 threads concurrently every second. As we have used

TABLE 3. Summary of the comparison of Blockchain frameworks for developing healthcare applications.

Platform	Open-Source	Support	Security	Scalability	Speed	Transaction Cost	Incentive
Ethereum [88]	✓	Ethereum Foundation, Hyperledger, Nethermind, OpenEthereum, EthereumJS	✓	✓	process 15 transactions per second	High	Cryptocurrency required.
Hyperledger Fabric [46]	✓	Linux Foundation	✓	✓	handle 3500+ transactions per second.	Low	No cryptocurrency required.
Quorum [89]	✓	Ethereum Developers and JPMorgan Chase	✓	✗	process more then 100 transactions per second	Low	No cryptocurrency required. (Free Gas Network)
Multichain [90]	✓	Coin Sciences Ltd.	✓	✓	process around 2000 transactions per second	Low	No cryptocurrency required

100 threads, each user from their own dedicated thread could submit several requests. After 10 seconds, the system will be dealing with a total of 100 threads.

The entire feature overview flow of our system is covered by this load-test. There were three different operational user types across the 100 threads. Those are doctors, BMDC (super-admin of the system) and Nagorik (citizen). Each of these users was visiting the landing page, logging onto their accounts, and using their authorized URLs to do CRUD actions.

During the course of this load-testing, it has covered the GET, PUT, POST, and DELETE operations. Furthermore, both Web2 and Web3 layers might use these actions. As we have already mentioned, our proposed system architecture has both on-chain and off-chain forms of data. Because of this, the request's latency was imperceptible in a decentralized system.

Our system can handle a maximum of 142 successful transactions per second, as shown in Fig. 13 (10 threads at a time), where the maximum transaction failure rate per second is just 41. Here, we must take into account the fact that our system was hosted on a local server and that the entire decentralized system (multiple containers) was running inside a single virtual machine with just 4GB of RAM and a single-core CPU. Because of that, a 3.66 percent failure rate can be easily considered (Fig. 14).

The graph in Fig. 13 shows the granularity of 10 seconds of total user transactions per second. We can also see that there are only two points in a second where the number of transaction failures is highest. However, the number of failures does not exceed 41 transactions, while the successful transaction rate is 142 per second. So the failure rate of the overall result is very low compared to the success rate.

Fig. 14 shows the result of summarizing the requests, whether they are in order or not. The graph shows that the results are mostly positive, with 96.34 percent of requests being positive and only 3.66 percent of requests being negative.

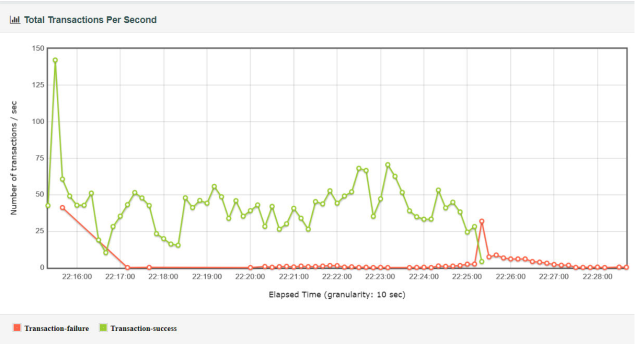


FIGURE 13. Success and Failure rate of Total Transactions Per Second in granularity of 10 seconds.

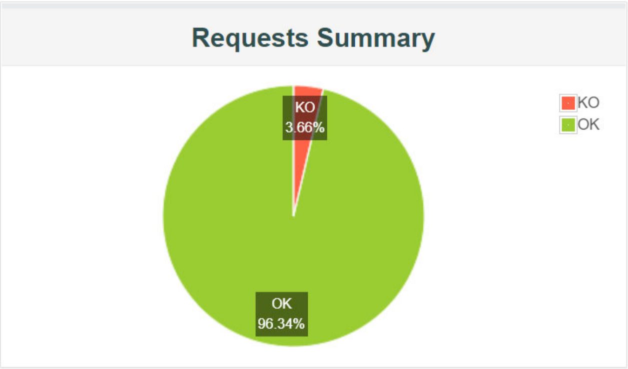


FIGURE 14. Performance test request summary.

Analyzing the4 data from the given Fig. 15 we find that, a graphical representation of our experimental data shows the number of requests per second (req/sec) from 0 to 140 horizontally and milliseconds (ms) from 0 to 5000 vertically. We can see that the maximum success rate is 133 req/sec, and the median latency is 331 ms. At the same time, comparatively, our system is dealing with the same amount (133 req/sec) of failed requests with only 0 ms median latency. Though our system has a record of handling

Algorithm 5 Pseudo Code for Authentication Process From the SDK Server to Blockchain Network

```
Web 2.0 Verification
▷ Procedure for the authentication between SDK server and BC network
procedure verifyWebToken(token)
    userToken ← get user token from session)
    isValid ← verifyToken(token, userToken)
    if isValid then
        return true
    else
        return false
▷ Get the contract obj to connect with blockchain
procedure getContractObj(healthCard)
    connPath ← Network configuration path
    gateway ← Network gateway object with connPath & healthCard
    if gateway==“success” then
        network ← Network object of given channel using gateway
        if network==“success” then
            contract ← Contract object for invoking smart contracts
            return contract
        else
            return “Channel Not Found”
    else
        return “Invalid Path or Credential”
procedure userLogin(orgName, identity, password)
    healthCardPath ← The Health Card path
    healthCard ← Identity Certificate based on Identity
    if healthCard == exist then
        authUser ← validateCredentials(orgName, identity, password)
        if authUser== true then
            session ← Create a session in the SDK server application
            redirect “baseURL/orgName/home”
        else
            return “Invalid Credentials”
    else
        return “Invalid Identity”
```

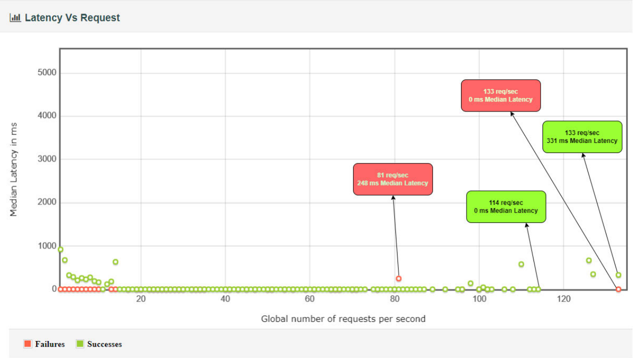


FIGURE 15. Median Latency of milliseconds (ms) for number of requests per second (req/sec).

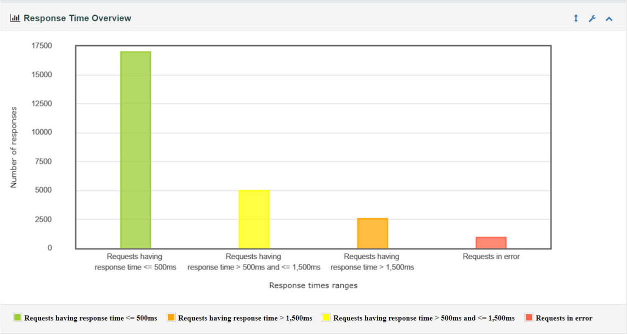


FIGURE 16. Response time overview.

114 requests per second in 0 ms median latency in the same ratio. Here, the maximum latency of a failure request is 81, with a median latency of 248 ms. And in the targeted ratio, this delay has happened only once. But comparatively, the success rate is very high according to the failure rate (green and red dots in Fig. 15).

In conclusion, it is clear that our suggested system can handle a high volume of requests with lower latency and throughput.

The diagram in Fig. 16 illustrates the overview of the response times of the requests generated by the users. It can be seen that the majority of the requests have a response time of less than 500 ms, namely 17020 out of 25,509 requests; 4987 requests have a response time of more than 500 ms and less than 1500 ms; 2568 requests have a response time of more than 1500 ms; and very few requests failed, namely 934 out of a total of 25,509 requests. Overall, the results are very positive.

It should be noted that these analyses should be considered hypothetical, as our Blockchain application is developed and installed on our own configured server. It is therefore subject to hardware and network limitations.

In the context of transaction throughput, Hyperledger Fabric exhibits a considerably higher TPS in comparison to Bitcoin and Ethereum. Specifically, Bitcoin demonstrates a TPS of 3-7, Ripple 1500, and Ethereum 15 - 30, whereas Hyperledger Fabric records a TPS of 3,500. Such a significant difference in TPS allows Hyperledger Fabric to efficiently process a greater number of transactions per second compared to Bitcoin, Ripple, and Ethereum, which is pivotal for use cases that require high volumes of transactions. Additionally, Hyperledger Fabric showcases lower latency than most other blockchain platforms. The latency of Hyperledger Fabric is less than 1 second, which is notably lower than Bitcoin’s (10 minutes) and Ethereum’s (13-15 seconds). This reduced latency contributes to faster transaction confirmations and can considerably enhance the overall

TABLE 4. Comparison of transaction throughput, latency in popular Blockchain platforms.

Platform	Transaction Throughput (TPS)	Latency (Seconds)
Bitcoin [91] [92]	3-7	600
Ethereum [93] [94]	15-30	13-15
Ripple [95]	1500	3-5
Hyperledger Fabric [96] [97]	3000	<1

TABLE 5. Human resource and cost required for the initial development phase.

<table>
<tr><td colspan="6">Human Resource For converting prototype level to Production level work</td></tr>
<tr><td colspan="6">Sprint: 20 | Working days in a week: 5</td></tr>
<tr><td>Resource
Type</td><td>Person
Count</td><td>Experience
Level</td><td>Salary
(BDT/mon)</td><td>Contact
Duration</td><td>Total Cost
(BDT)</td></tr>
<tr><td>HR Executive</td><td>1</td><td>Mid</td><td>50,000</td><td>9</td><td>450,000</td></tr>
<tr><td>Project Manager</td><td>1</td><td>Senior</td><td>150,000</td><td>9</td><td>1,350,000</td></tr>
<tr><td>UI/UX Designer</td><td>1</td><td>Mid</td><td>50,000</td><td>3</td><td>150,000</td></tr>
<tr><td rowspan="2">Software Engineer</td><td>1</td><td>Senior</td><td>150,000</td><td>9</td><td>1,350,000</td></tr>
<tr><td>2</td><td>Mid</td><td>73,000</td><td>9</td><td>1,314,000</td></tr>
<tr><td rowspan="2">Software Developer</td><td>2</td><td>Mid</td><td>55,000</td><td>9</td><td>990,000</td></tr>
<tr><td>5</td><td>Associate</td><td>35,000</td><td>9</td><td>1,575,000</td></tr>
<tr><td rowspan="2">SQA Engineer</td><td>1</td><td>Senior</td><td>70,000</td><td>9</td><td>630,000</td></tr>
<tr><td>2</td><td>Associate</td><td>35,000</td><td>4</td><td>280,000</td></tr>
<tr><td>Automation
QA Engineer</td><td>1</td><td>Mid</td><td>55,000</td><td>4</td><td>220,000</td></tr>
<tr><td rowspan="2">DevOPs Engineer</td><td>1</td><td>Senior</td><td>170,000</td><td>4</td><td>680,000</td></tr>
<tr><td>1</td><td>Associate</td><td>35,000</td><td>4</td><td>140,000</td></tr>
<tr><td>Total human resource</td><td colspan="2">19</td><td colspan="2">Total Sprint Cost</td><td>9,129,000</td></tr>
</table>

TABLE 6. Human resource and cost required for the maintenance phase.

<table>
<tr><td colspan="5">Human Resource for Maintenance</td></tr>
<tr><td colspan="5">Working days in a week: 5 | Yearly Bonus: 2</td></tr>
<tr><td>Resource
Type</td><td>Person
Count</td><td>Experience
Level</td><td>Salary
(BDT/mon)</td><td>Total Cost
(BDT)</td></tr>
<tr><td>HR Executive</td><td>1</td><td>Mid</td><td>50,000</td><td>50,000</td></tr>
<tr><td>Business
Administration</td><td>1</td><td>Senior</td><td>65,000</td><td>65,000</td></tr>
<tr><td>Digital Marketer</td><td>1</td><td>Mid</td><td>35,000</td><td>35,000</td></tr>
<tr><td>Graphics Designer</td><td>1</td><td>Mid</td><td>40,000</td><td>40,000</td></tr>
<tr><td>Project Manager</td><td>1</td><td>Mid</td><td>120,000</td><td>120,000</td></tr>
<tr><td rowspan="2">Software Engineer</td><td>1</td><td>Senior</td><td>150,000</td><td>150,000</td></tr>
<tr><td>1</td><td>Mid</td><td>73,000</td><td>73,000</td></tr>
<tr><td rowspan="2">Software Developer</td><td>1</td><td>Mid</td><td>70,000</td><td>70,000</td></tr>
<tr><td>2</td><td>Associate</td><td>35,000</td><td>70,000</td></tr>
<tr><td>SQA Engineer</td><td>1</td><td>Mid</td><td>55,000</td><td>55,000</td></tr>
<tr><td>DevOPs Engineer</td><td>1</td><td>Senior</td><td>170,000</td><td>170,000</td></tr>
<tr><td>Total human resource</td><td colspan="2">12</td><td>Total
Monthly Cost</td><td>898,000</td></tr>
</table>

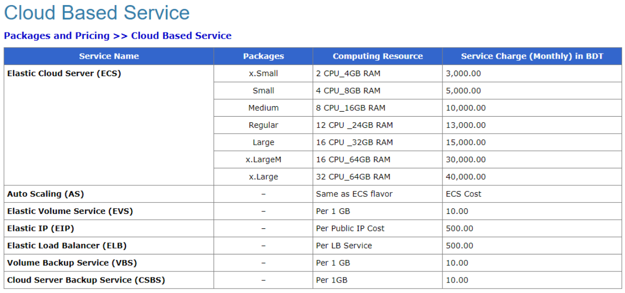
user experience. Furthermore, Hyperledger Fabric provides more adaptable permissions and privacy features than Bitcoin and Ethereum. Fabric facilitates the establishment of private channels between particular participants and enables a range of consensus algorithms to be employed based on the network's requirements. The platform-specific comparison is shown in Table 4. The high transaction throughput, low latency, and flexible permissions and privacy features of Hyperledger Fabric render it a strong contender for Blockchain applications at the enterprise level, where scalability and privacy are crucial factors.

The communication complexity of Hyperledger Fabric is determined by the amount of communication that is required between nodes in the network to reach consensus on the state of the ledger. Hyperledger Fabric uses a modular architecture that allows for flexible consensus mechanisms and allows for a variety of ordering service implementations. The communication complexity in Hyperledger Fabric is reduced through its use of a private data collection mechanism, which enables private transactions between specified parties without being visible to other participants on the network. This mechanism helps to reduce the amount of unnecessary communication between nodes, improving overall network performance. Additionally, Hyperledger Fabric uses a gossip protocol to disseminate blocks to peers, allowing for efficient and scalable distribution of data. The gossip protocol ensures that peers receive all necessary information in a timely and efficient manner, while also reducing the amount of communication required between nodes.

Overall, the communication complexity of Hyperledger Fabric is low and optimized for high performance and scalability, making it a strong choice for enterprise-level blockchain solutions.

In our proposed algorithm 1, AddNewDoctor procedure's time complexity is O(1) and ApproveDoctorApplication procedure's time complexity is O(1). So the total time complexity of algorithm 1 is $O(1) + O(1) = O(1)$. Next algorithm 2, DoctorApply procedure's time complexity is O(1) and AddPrescription procedure's time complexity is O(1). So the total time complexity of algorithm 2 is $O(1) + O(1) = O(1)$. After that algorithm 3, DoctorAppointment procedure's time complexity is O(n) and AddComplain procedure's time complexity is O(1). So the total time complexity of algorithm 3 is $O(n) + O(1) = O(n)$. Next algorithm 4 RegisterUser procedure's time complexity is O(1). So the total time complexity of algorithm 4 O(1). Finally, algorithm 5, VerifyWebToken procedure's time complexity is O(1), getContractObj procedure's time complexity is O(1) and userLogin procedure's time complexity is O(1). So the total time complexity of algorithm 5 is $O(1) + O(1) + O(1) = O(1)$.

As stated in this article, we have developed a Blockchain-based application for our proposed solution that is fully functional, and ensures all requirements and standards for a Blockchain application. Therefore, in Table 5, Blockchain infrastructure is not emphasized separately. Here, we believe it is prudent to invest primarily in client-side applications. This is due to the fact that, despite the fact that the entirety of the research in this article is devoted to Blockchain, no detailed work has been conducted on all the necessary features of its user organizations, including the user experience of their users. Therefore, if action is required at the production



Cloud Based Service

Packages and Pricing >> Cloud Based Service

Service Name	Packages	Computing Resource	Service Charge (Monthly) in BDT
Elastic Cloud Server (ECS)	x.Small	2 CPU_4GB RAM	3,000.00
	Small	4 CPU_8GB RAM	5,000.00
	Medium	8 CPU_16GB RAM	10,000.00
	Regular	12 CPU _24GB RAM	13,000.00
	Large	16 CPU _32GB RAM	15,000.00
	x.LargeM	16 CPU_64GB RAM	30,000.00
	x.Large	32 CPU_64GB RAM	40,000.00
Auto Scaling (AS)	–	Same as ECS flavor	ECS Cost
Elastic Volume Service (EVS)	–	Per 1 GB	10.00
Elastic IP (EIP)	–	Per Public IP Cost	500.00
Elastic Load Balancer (ELB)	–	Per LB Service	500.00
Volume Backup Service (VBS)	–	Per 1 GB	10.00
Cloud Server Backup Service (CSBS)	–	Per 1GB	10.00

FIGURE 17. Cloud service packages.

level, it will be in each of these sectors. Therefore, Blockchain Expert is not allocated a specific budget in our future cost plan.

The Table 5 is for the initial approach to the development of the project at the production level. Therefore, there is no permanent position listed in this table. This table depicts an estimate of a 20-sprint project plan in which various human resources will be added at varying intervals. For example, a UI/UX designer will be added to the sprint from the very beginning in order to create an accurate visualization of the features and a demo of their user experience. However, he or she will focus on the sprint for the initial three months. In contrast, a DevOps engineer will be added to the sprint for the last four months if we take it into account, as they are primarily responsible for server-side work and project deployment. For the duration of the sprint, however, there will be a dedicated management team, developer team, and testing team. Since these teams are available during the sprint, it is easy to choose any software development life cycle (SDLC) model for the development of this project.

After the proposed solution has been implemented on the production server, a full-time technical team is required, which is presented in Table 6 with proper cost estimation. Here, we've adjusted the budget for the technical team based on business needs, such as the need for a business administrator, digital marketer, and graphic designer to create content that will bring a lot of people to this system.

As previously mentioned, the use of third-party cloud services developed specifically for government use is not permitted. To comply with this requirement, the National Data Center (NDC) has been identified as the appropriate service provider [98]. The most appropriate package for our service, as depicted in Fig. 17, is the X.Large package, which is intended for national-scale use. The Elastic Cloud Service (ECS) included in this package is well-suited for private Blockchains. This package allows us to add new organizations to our network at any time and scale them within 5–6 minutes from the server-side. By purchasing a single public IP and subnet masking these IPs, we can scale any number of organization peers in the ECS environment using private IPs. The load balancing service included in this package is also very affordable.

The entire cost estimate was based on the Bangladeshi Taka in May 2022, when 1 USD was worth 86.07 BDT.

The *ICT Ministry* and *Bangladesh Computer Council* (BCC) is already working on various type of projects to digitalize government services to save time and money. *BanglaGovNet* is one of them. *BanglaGovNet* Stands for *Bangladesh Government wide Network*. It is a Public Network to connect all the Government entities throughout the country under a single Network [99].

The present study proposes an integrated solution to facilitate the provision of healthcare services. Specifically, this solution can be integrated into the existing infrastructure of the National Information and Communications Technology Center (NICTC). As of the time of this writing, 10% of the NICTC infrastructure is already in active use (Fig. 18). Notably, with just this 10% of infrastructure, digitalization of 66% of government services nationwide has been achieved, encompassing six different ministries, by the year 2022, according to the findings reported in Digitalization of Government Services [100].

The investigative department, *Implementation Monitoring and Evaluation Division* (IMED) of Bangladesh government (http://www.imed.gov.bd/) has already disclosed the following information in the national media that how these online services are playing a role in the administrative work of the country in terms of speed, time and money [100].

1) 70% were relieved of the hassle they faced at government offices.
2) 92% were effective in money saving.
3) 96% saved time.

The IMED also interviewed the service provider and the study finds that [100]:

1) 30% could focus more at work for digitalization.
2) 11% said their lives have become simpler.
3) 1% could find time for digitalization.

The country has made substantial progress in terms of internet infrastructure and IT training. Specifically, 80% of the population has access to the internet, of which 52% is supported by optical fiber technology. In addition, 39% of the population has access to intranet, and 38% has received IT training. As depicted in Fig. 18, 52% of the services are currently supported by website/web-portal technology. Given this existing infrastructure, we anticipate that our proposed solution can be deployed across the country in a timely manner, thereby contributing to the welfare of the general population.

The project titled "Development of National ICT Infra-Network for Bangladesh Government (BanglaGovNet) (1st Revised)" has been evaluated in detail by the IMED, as documented in their report. The report, available through [101], contains favorable findings that bode well for the anticipated efficacy and acceptance of our proposed solution.

Given that our proposed solution can be integrated within the existing framework of *BanglaGovNet*, we emphasize its potential to enhance service quality while saving time and

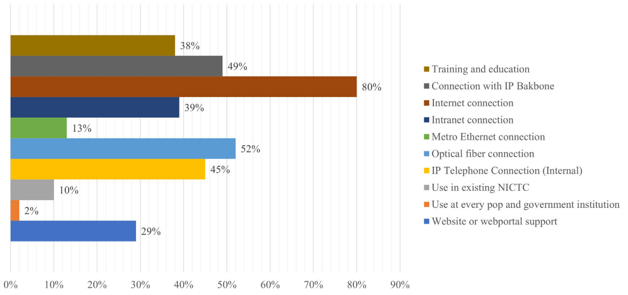


FIGURE 18. Infrastructure of existing government online services and acceptance of these services [101].

TABLE 7. Number and type of people surveyed by IMED [101].

Information Provider Type	Number of Information Provider
Active consumer of service	356
Trainee	20
Focus Group Discussion (FGD)	80
Officials involved in the project (KII)	25
Total = 481	

money. Our proposal is in line with the findings of government surveys that highlight the benefits of providing online access to nationwide government services. Notably, the sample size for these surveys was determined using the statistical formula outlined in [101].

$$n = \frac{Z^2 p(1-p)}{e^2}$$

Here, n = Sample size, p = Target proportion, Z = The value of standard variate at a given confidence level, and e = Margin of error.

For the sample, confidence level was considered as *95%* where the $Z = 1.96$. Target proportion was taken as *5%* and error level was also considered as *5%*.

After statistical calculation, $n = 356$ people were selected for interview according to the result of the calculation among those who were consuming the Government Online services [101].

Beyond that, *Focus Group Discussion* (FGD) were conducted by interviewing field level officials from different ministries and departments. Higher officials were considered as part of *Key Informant Interview* (KII). The detailed data of this statistical calculation is given in Table 7.

In order to develop and implement such an online service-based project and to maintain the trend of providing long-term service through it, the following people have also been surveyed by IMED.

According to the Ministry of Health and Family Welfare of Bangladesh has at least 104,659 human resources for health services [2] for which the government budgeted $2,174 million [2] in 2007.

So if we can use our proposed solution according to the above discussion, we can save both our money and time by ensuring good service with this limited human resource across the country. The graphical representation is shown at Fig. 19 which is highlighting the possibility of improving the quality of life of common people and how office work can be strengthened in the implementation of our proposed solution. The survey for developing the Fig. 19 is based on the individuals specified in Table 7.

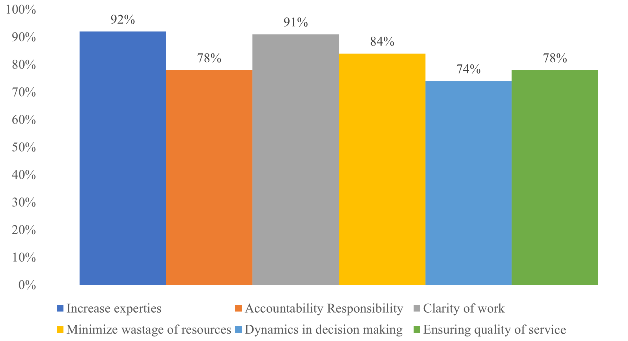


FIGURE 19. Effects of using online technology in government services to improve administrative work.

VIII. CONCLUSION

When it comes to health care in Bangladesh, both government and non-government organizations work together. Although the quality of services in the private sector is improving day by day and a competitive environment has been created to deliver good services, public hospitals are slowing down for a number of reasons. In many cases, corruption is responsible. Even though the government takes different steps every year to give the people of the country different health services, there is no way to make sure that they get them. Because our health care system operates on a national level, the ministry still monitors it in a traditional way. As a result, many wrongdoers get through here easily and without any irregularities. On the other hand, the complaints of the harmed people do not reach the policymakers. So, this study creates a healthcare application framework based on distributed ledger technology that can stop corruption and set up a good system for providing and keeping track of healthcare for the people of Bangladesh. The proposed system is a simple and reliable digital process for all organizations where all agreements and information are guaranteed to be immutable.

REFERENCES

[1] S. Gaudin, W. Raza, J. Skordis, A. Soucat, K. Stenberg, and A. Alwan, "Using costing to facilitate policy making towards universal health coverage: Findings and recommendations from country-level experiences," *BMJ Global Health*, vol. 8, no. 1, Jan. 2023, Art. no. e010735.

[2] *Government of the People's Republic of Bangladesh Ministry of Health and Family Welfare Human Resources Development Unit HRD Data Sheet-2011 B*, Ministry Health, Family Welfare, Dhaka, Bangladesh, 2011, pp. 2005–2008.

[3] T. Joarder, T. Z. Chaudhury, and I. Mannan, "Universal health coverage in Bangladesh: Activities, challenges, and suggestions," *Adv. Public Health*, vol. 2019, pp. 1–12, Mar. 2019.

[4] E. Jove, J. Aveleira-Mata, H. Alaiz-Moretón, J.-L. Casteleiro-Roca, D. Y. M. del Blanco, F. Zayas-Gato, H. Quintián, and J. L. Calvo-Rolle, "Intelligent one-class classifiers for the development of an intrusion detection system: The MQTT case study," *Electronics*, vol. 11, no. 3, p. 422, Jan. 2022. [Online]. Available: https://www.mdpi.com/2079-9292/11/3/422

[5] *Kaseya Ransomware Attack Underlines Supply Chain Risks*, Emerald Expert Briefings, Oxford Analytica, FiscalNote, CA, USA, 2021.

[6] M. S. U. Miah, M. S. Tahsin, S. Azad, G. Rabby, M. S. Islam, S. Uddin, and M. Masuduzzaman, "A geofencing-based recent trends identification from Twitter data," *IOP Conf. Ser., Mater. Sci. Eng.*, vol. 769, no. 1, Feb. 2020, Art. no. 012008.

[7] A. A. Khan, A. A. Laghari, A. A. Shaikh, M. A. Dootio, V. V. Estrela, and R. T. Lopes, "A blockchain security module for brain–computer interface (BCI) with multimedia life cycle framework (MLCF)," *Neurosci. Informat.*, vol. 2, no. 1, Mar. 2022, Art. no. 100030.

[8] D. Ichikawa, M. Kashiyama, and T. Ueno, "Tamper-resistant mobile health using blockchain technology," *JMIR mHealth uHealth*, vol. 5, no. 7, p. e7938, 2017.

[9] M. S. U. Miah, T. B. Sarwar, S. S. Islam, M. S. Haque, M. Masuduzzaman, and A. Bhowmik, "An adaptive medical cyber-physical system for post diagnosis patient care using cloud computing and machine learning approach," in *Proc. 3rd Int. Conf. Emerg. Technol. (INCET)*, May 2022, pp. 1–6.

[10] S. S. Islam, M. S. Haque, M. S. U. Miah, T. B. Sarwar, and R. Nugraha, "Application of machine learning algorithms to predict the thyroid disease risk: An experimental comparative study," *PeerJ Comput. Sci.*, vol. 8, p. e898, Mar. 2022.

[11] *Bangladesh Essential Health Service Package (ESP) Ministry of Health and FamilyWelfare Government of the People's Republic of Bangladesh Bangladesh Essential Health Service Package (ESP)*, Ministry Health Family Welfare, Dhaka, Bangladesh, 2016.

[12] T. Heart, O. Ben-Assuli, and I. Shabtai, "A review of PHR, EMR and EHR integration: A more personalized healthcare and public health policy," *Health Policy Technol.*, vol. 6, no. 1, pp. 20–25, Mar. 2017.

[13] A. P. Singh, N. R. Pradhan, A. K. Luhach, S. Agnihotri, N. Z. Jhanjhi, S. Verma, Kavita, U. Ghosh, and D. S. Roy, "A novel patient-centric architectural framework for blockchain-enabled healthcare applications," *IEEE Trans. Ind. Informat.*, vol. 17, no. 8, pp. 5779–5789, Aug. 2021.

[14] Q. Zhang, "The Internet hospital: How to combine with traditional healthcare model," *Hepatobiliary Surgery Nutrition*, vol. 11, no. 2, pp. 273–275, Apr. 2022.

[15] A. A. Mamun, S. Azam, and C. Gritti, "Blockchain-based electronic health records management: A comprehensive review and future research direction," *IEEE Access*, vol. 10, pp. 5768–5789, 2022.

[16] F. A. Reegu, H. Abas, Z. Hakami, S. Tiwari, R. Akmam, I. Muda, H. A. Almashqbeh, and R. Jain, "Systematic assessment of the interoperability requirements and challenges of secure blockchain-based electronic health records," *Secur. Commun. Netw.*, vol. 2022, pp. 1–12, Jul. 2022.

[17] S. Ghosh and R. Dasgupta, "Cloud computing infrastructure in healthcare industry," in *Machine Learning in Biological Sciences: Updates and Future Prospects*. Berlin, Germany: Springer, 2022, pp. 169–176.

[18] C. K. Rogers, M. Parulekar, F. Malik, and C. A. Torres, "A local perspective into electronic health record design, integration, and implementation of screening and referral for social determinants of health," *Perspect. Health Inf. Manage.*, vol. 19, no. 1, pp. 1–19, Jan. 2022.

[19] *GDPR Archives*. Accessed: Mar. 1, 2023. [Online]. Available: https://gdpr.eu/tag/gdpr/

[20] T. White, E. Blok, and V. D. Calhoun, "Data sharing and privacy issues in neuroimaging research: Opportunities, obstacles, challenges, and monsters under the bed," *Hum. Brain Mapping*, vol. 43, no. 1, pp. 278–291, Jan. 2022.

[21] A. J. Perez and S. Zeadally, "Secure and privacy-preserving crowdsensing using smart contracts: Issues and solutions," *Comput. Sci. Rev.*, vol. 43, Feb. 2022, Art. no. 100450.

[22] A. Roehrs, C. A. da Costa, and R. da Rosa Righi, "OmniPHR: A distributed architecture model to integrate personal health records," *J. Biomed. Informat.*, vol. 71, pp. 70–81, Jul. 2017.

[23] T. Bocek, B. B. Rodrigues, T. Strasser, and B. Stiller, "Blockchains everywhere—A use-case of blockchains in the pharma supply-chain," in *Proc. IFIP/IEEE Symp. Integr. Netw. Service Manage. (IM)*, May 2017, pp. 772–777.

[24] X. Liang, J. Zhao, S. Shetty, J. Liu, and D. Li, "Integrating blockchain for data sharing and collaboration in mobile healthcare applications," in *Proc. IEEE 28th Annu. Int. Symp. Pers., Indoor, Mobile Radio Commun. (PIMRC)*, Oct. 2017, pp. 1–5.

[25] A. Al Omar, M. S. Rahman, A. Basu, and S. Kiyomoto, "MediBchain: A blockchain based privacy preserving platform for healthcare data," in *Proc. Int. Conf. Secur., Privacy Anonymity Comput., Commun. Storage*. Berlin, Germany: Springer, 2017, pp. 534–543.

[26] A. Azaria, A. Ekblaw, T. Vieira, and A. Lippman, "MedRec: Using blockchain for medical data access and permission management," in *Proc. 2nd Int. Conf. Open Big Data (OBD)*, Aug. 2016, pp. 25–30.

[27] N. Nchinda, A. Cameron, K. Retzepi, and A. Lippman, "MedRec: A network for personal information distribution," in *Proc. Int. Conf. Comput., Netw. Commun. (ICNC)*, Feb. 2019, pp. 637–641.

[28] K. M. Hossein, M. E. Esmaeili, T. Dargahi, A. Khonsari, and M. Conti, "BCHealth: A novel blockchain-based privacy-preserving architecture for IoT healthcare applications," *Comput. Commun.*, vol. 180, pp. 31–47, Dec. 2021.

[29] K. M. Hossein, M. E. Esmaeili, T. Dargahi, and A. khonsari, "Blockchain-based privacy-preserving healthcare architecture," in *Proc. IEEE Can. Conf. Electr. Comput. Eng. (CCECE)*, May 2019, pp. 1–4.

[30] M. Ejaz, T. Kumar, I. Kovacevic, M. Ylianttila, and E. Harjula, "Health-BlockEdge: Blockchain-edge framework for reliable low-latency digital healthcare applications," *Sensors*, vol. 21, no. 7, p. 2502, Apr. 2021.

[31] M. Swan, *Blockchain: Blueprint for a New Economy*. Sebastopol, CA, USA: O'Reilly Media, 2015.

[32] R. Belen-Saglam, E. Altuncu, Y. Lu, and S. Li, "A systematic literature review of the tension between the GDPR and public blockchain systems," *Blockchain, Res. Appl.*, vol. 1, Jan. 2023, Art. no. 100129.

[33] N. Badr, "Blockchain or distributed ledger technology what is in it for the healthcare industry?" in *Proc. 11th Int. Joint Conf. Knowl. Discovery, Knowl. Eng. Knowl. Manage.*, 2019, pp. 277–284.

[34] A. Sunyaev, "Distributed ledger technology," in *Internet Computing*. Berlin, Germany: Springer, 2020, pp. 265–299.

[35] A. A. Laghari, A. A. Khan, R. Alkanhel, H. Elmannai, and S. Bourouis, "Lightweight-BIoV: Blockchain distributed ledger technology (BDLT) for Internet of Vehicles (IoVs)," *Electronics*, vol. 12, no. 3, p. 677, Jan. 2023.

[36] M. S. U. Miah, M. Rahman, M. S. Hosain, and A. A. Ahsan, "Introduction to blockchain," in *Blockchain in Data Analytics*, 1st ed. Newcastle upon Tyne, U.K.: Cambridge Scholars Publishing, 2020, pp. 1–23. [Online]. Available: https://www.cambridgescholars.com/product/978-1-5275-4429-1

[37] M. J. M. Chowdhury, M. D. S. Ferdous, K. Biswas, N. Chowdhury, A. S. M. Kayes, M. Alazab, and P. Watters, "A comparative analysis of distributed ledger technology platforms," *IEEE Access*, vol. 7, pp. 167930–167943, 2019.

[38] A. Narayanan, J. Bonneau, E. Felten, A. Miller, and S. Goldfeder, *Bitcoin and Cryptocurrency Technologies: A Comprehensive Introduction*. Princeton, NJ, USA: Princeton Univ. Press, 2016.

[39] *National Blockchain Strategy: Bangladesh Information and Communication Technology Division*, Inf. Commun. Technol. Division, Bangladesh, Dhaka, Bangladesh, 2020.

[40] Facebook, Twitter, and LinkedIn. *What You Need to Know About Network DNS Servers*. Accessed: Mar. 1, 2023. [Online]. Available: https://www.lifewire.com/what-is-a-dns-server-2625854

[41] M. Siduzzaman, M. M. Hossan, R. Alom, T. B. Sarwar, and M. S. U. Miah, "Performance comparison of HTTP/2 for common E-commerce web frameworks with traditional HTTP," *J. Phys., Conf. Ser.*, vol. 1529, no. 5, May 2020, Art. no. 052023.

[42] S. Hussain, S. S. Ullah, M. Shorfuzzaman, M. Uddin, and M. Kaosar, "Cryptanalysis of an online/offline certificateless signature scheme for Internet of Health Things," *Intell. Autom. Soft Comput.*, vol. 30, no. 3, pp. 983–993, 2021.

[43] C. C. Agbo and Q. H. Mahmoud, "Comparison of blockchain frameworks for healthcare applications," *Internet Technol. Lett.*, vol. 2, no. 5, p. e122, Sep. 2019.

[44] M. Uddin, M. S. Memon, I. Memon, I. Ali, J. Memon, M. Abdelhaq, and R. Alsaqour, "Hyperledger fabric blockchain: Secure and efficient solution for electronic health records," *CMC Comput., Mater. Continua*, vol. 68, no. 2, pp. 2377–2397, 2021.

[45] E. S. Babu, B. V. R. N. Yadav, A. K. Nikhath, S. R. Nayak, and W. Alnumay, "MediBlocks: Secure exchanging of electronic health records (EHRs) using trust-based blockchain network with privacy concerns," *Cluster Comput.*, vol. 1, pp. 1–28, Aug. 2022.

[46] E. Androulaki et al., "Hyperledger fabric: A distributed operating system for permissioned blockchains," in *Proc. 13th EuroSys Conf.*, 2018, pp. 1–15.

[47] J. Hu and K. Liu, "Raft consensus mechanism and the applications," *J. Phys., Conf. Ser.*, vol. 1544, no. 1, May 2020, Art. no. 012079.

[48] *Why New Off-Chain Storage is Required for Blockchains*, Int. Bus. Mach. Corp., Armonk, NY, USA, 2018, p. 11.

[49] R. Norvill, B. B. F. Pontiveros, R. State, and A. Cullen, "IPFS for reduction of chain size in Et4hereum," in *Proc. IEEE Int. Conf. Internet Things (iThings), IEEE Green Comput. Commun. (GreenCom), IEEE Cyber, Phys. Social Comput. (CPSCom), IEEE Smart Data (SmartData)*, Jul. 2018, pp. 1121–1128.

[50] *Overview—Apache CouchDB 3.2 Documentation*. Accessed: Mar. 1, 2023. [Online]. Available: https://docs.couchdb.org/en/stable/

[51] *Express Js*. Accessed: Mar. 1, 2023. [Online]. Available: https://expressjs.com/en/starter/installing.html

[52] A. A. Laghari, K. Wu, R. A. Laghari, M. Ali, and A. A. Khan, "A review and state of art of Internet of Things (IoT)," *Arch. Comput. Methods Eng.*, vol. 29, pp. 1–19, Jul. 2021.

[53] M. A. Islam, M. A. Islam, M. S. U. Miah, and A. Bhowmik, "An automated monitoring and environmental control system for laboratory-scale cultivation of oyster mushrooms using the Internet of Agricultural Thing (IoAT)," in *Proc. 2nd Int. Conf. Comput. Advancements (ICCA)*. New York, NY, USA: Association for Computing Machinery, 2022, pp. 207–212, doi: 10.1145/3542954.3542985.

[54] P. Voigt and A. Von dem Bussche, "The EU general data protection regulation (GDPR)," *A Practical Guide*, 1st ed. Cham, Switzerland: Springer, vol. 10, 2017, Art. no. 3152676.

[55] Hyperledger. (2022). *Private Data—Hyperledger-Fabricdocs Main Documentation*. [Online]. Available: https://hyperledgerfabric.readthedocs.io/en/release-2.2/private-data/private-data.html

[56] *Updating a Channel Configuration—Hyperledger-Fabricdocs Main Documentation*. Accessed: Mar. 1, 2023. [Online]. Available: https://hyperledger-fabric.readthedocs.io/en/release-2.0/config_update.html#more-about-these-parameters

[57] A. L. Selvakumar and C. S. Ganadhas, "The evaluation report of SHA-256 crypt analysis hash function," in *Proc. Int. Conf. Commun. Softw. Netw.*, 2009, pp. 588–592.

[58] Amazon Web Service. (2022). *Using the AWS Blockchain Template for Hyperledger Fabric—AWS Blockchain Templates*. [Online]. Available: https://docs.aws.amazon.com/blockchain-templates/latest/developerguide/blockchain-templates-hyperledger.html

[59] Cloud Security Alliance. (2022). *Best Practices for Smart Contract Security Hyperledger Fabric*. [Online]. Available: https://cloudsecurityalliance.org/artifacts/cybersecurity-best-practices-smart-contract-overview/

[60] E. J. Dewi, U. Rusydi, and R. Imam, "Implementation of cloudflare hosting for speeds and protection on the website," Ph.D. dissertation, Dept. Inform. Eng., Univ. Ahmad Dahlan, Yogyakarta, Indonesia, 2019.

[61] K. Kant, R. Iyer, and P. Mohapatra, "Architectural impact of secure socket layer on internet servers," in *Proc. Int. Conf. Comput. Design*, 2000, pp. 7–14.

[62] A. U. Rahman, S. U. Miah, M. A. Fahad, and D. Karmaker, "SHIMPG: Simple human interaction with machine using physical gesture," in *Proc. 13th Int. Conf. Control Autom. Robot. Vis. (ICARCV)*, Dec. 2014, pp. 301–305.

[63] F. Lau, S. H. Rubin, M. H. Smith, and L. Trajkovic, "Distributed denial of service attacks," in *Proc. IEEE Int. Conf. Syst., Man Cybern. Cybern. Evolving Syst., Humans, Org., Their Complex Interact.*, vol. 3, Oct. 2000, pp. 2275–2280.

[64] L. Von Ahn, M. Blum, N. J. Hopper, and J. Langford, "CAPTCHA: Using hard AI problems for security," in *Proc. Int. Conf. Theory Appl. Cryptograph. Techn.* Berlin, Germany: Springer, 2003, pp. 294–311.

[65] Ethereum. (2023). *Proof-of-Work Energy-Usage*. [Online]. Available: https://ethereum.org/en/developers/docs/consensus-mechanisms/pow/

[66] Consencys. (2023). *What is Proof of Stake?* [Online]. Available: https://consensys.net/blog/blockchain-explained/what-is-proof-of-stake/

[67] D. Batmunkh. (2023). *Private Blockchain Consensus Mechanisms*. [Online]. Available: https://medium.com/@arigatodl/private-blockchain-consensus-mechanisms-8e6fc48c8fb

[68] Tech Blockdata. (2022). *The State of Enterprise Blockchain in 2022*. [Online]. Available: https://www.blockdata.tech/blog/general/the-state-of-enterprise-blockchain-in-2022

[69] S. Brotsis, N. Kolokotronis, K. Limniotis, G. Bendiab, and S. Shiaeles, "On the security and privacy of hyperledger fabric: Challenges and open issues," in *Proc. IEEE World Congr. Services (SERVICES)*, Oct. 2020, pp. 197–204.

[70] Hyperledger. (2022). *The Ordering Service*. [Online]. Available: https://hyperledger-fabric.readthedocs.io/en/latest/orderer/orderingservice.html#the-ordering-service

[71] A. Barger, Y. Manevich, H. Meir, and Y. Tock, "A Byzantine fault-tolerant consensus library for hyperledger fabric," in *Proc. IEEE Int. Conf. Blockchain Cryptocurrency (ICBC)*, May 2021, pp. 1–9.

[72] J. Soryal and T. Saadawi, "Byzantine attack isolation in IEEE 802.11 wireless ad-hoc networks," in *Proc. IEEE 9th Int. Conf. Mobile Ad-Hoc Sensor Syst. (MASS)*, Oct. 2012, pp. 1–5.

[73] K. Yamashita, Y. Nomura, E. Zhou, B. Pi, and S. Jun, "Potential risks of hyperledger fabric smart contracts," in *Proc. IEEE Int. Workshop Blockchain Oriented Softw. Eng. (IWBOSE)*, Feb. 2019, pp. 1–10.

[74] H. Hasanova, U.-J. Baek, M.-G. Shin, K. Cho, and M.-S. Kim, "A survey on blockchain cybersecurity vulnerabilities and possible countermeasures," *Int. J. Netw. Manage.*, vol. 29, no. 2, p. e2060, Mar. 2019. [Online]. Available: https://onlinelibrary.wiley.com/doi/abs/10.1002/nem.2060

[75] T. Krupa, M. Ries, I. Kotuliak, K. Kostal, and R. Bencel, "Security issues of smart contracts in Ethereum platforms," in *Proc. 28th Conf. Open Innov. Assoc. (FRUCT)*. Moscow, Russia: IEEE, 2020.

[76] M. Fang, X. Zhou, Z. Zhang, C. Jin, and A. Zhou, "SEFrame: An SGX-enhanced smart contract execution framework for permissioned blockchain," in *Proc. IEEE 38th Int. Conf. Data Eng. (ICDE)*, May 2022, pp. 3166–3169.

[77] Hyperledger. (2022). *Hyperledger Fabric Channel Capabilities*. [Online]. Available: https://hyperledger-fabric.readthedocs.io/en/latest/capabilitiesconcept.html

[78] A. Bhowmik, S. U. Miah, and Mohaimen-Bin-Noor, "IoT (Internet of Things)-based smart garbage management system," *AIUB J. Sci. Eng.*, vol. 19, no. 1, pp. 33–40, 2020.

[79] I. Ali and N. Meghanathan, "Virtual machines and networks–installation, performance study, advantages and virtualization options," 2011, *arXiv:1105.0061*.

[80] M. S. U. Miah, A. Bhowmik, and R. T. Anannya, "Location, context and device aware framework (LCDF): A unified framework for mobile data management," in *Proc. Int. Conf. Comput. Advancements*, Jan. 2020, pp. 1–5.

[81] *Fabric Contract APIs and Application APIs—Hyperledger-Fabricdocs Main Documentation*. Accessed: Mar. 1, 2023. [Online]. Available: https://hyperledger-fabric.readthedocs.io/en/latest/sdk_chaincode.html

[82] *Fabric-CA Commands—Hyperledger-Fabricdocs Main Documentation*. Accessed: Mar. 1, 2023. [Online]. Available: https://hyperledger-fabric.readthedocs.io/en/latest/commands/fabric-ca-commands.html

[83] *How Fabric Networks are Structured—Hyperledger-Fabricdocs Main Documentation*. Accessed: Mar. 1, 2023. [Online]. Available: https://hyperledger-fabric.readthedocs.io/en/latest/network/network.html

[84] *Flutter Documentation*. Accessed: Mar. 1, 2023. [Online]. Available: https://docs.flutter.dev/

[85] E. Brown, *Web Development With Node and Express: Leveraging the JavaScript Stack*. Sebastopol, CA, USA: O'Reilly Media, 2019.

[86] G. Volpe, A. M. Mangini, and M. P. Fanti, "An architecture combining blockchain, Docker and cloud storage for improving digital processes in cloud manufacturing," *IEEE Access*, vol. 10, pp. 79141–79151, 2022.

[87] S. Scott, "Introducing Docker and oracle," in *Oracle on Docker: Running Oracle Databases in Linux Containers*. Berlin, Germany: Springer, 2023, pp. 3–18.

[88] A. Urquhart, "Under the hood of the Ethereum blockchain," *Finance Res. Lett.*, vol. 47, Jun. 2022, Art. no. 102628.

[89] G. A. F. Rebello, G. F. Camilo, L. C. B. Guimarães, L. A. C. de Souza, and O. C. M. B. Duarte, "Security and performance analysis of quorum-based blockchain consensus protocols," in *Proc. 6th Cyber Secur. Netw. Conf. (CSNet)*, Oct. 2022, pp. 1–7.

[90] A. Ismailisufi, T. Popović, N. Gligorić, S. Radonjic, and S. Šandi, "A private blockchain implementation using multichain open source platform," in *Proc. 24th Int. Conf. Inf. Technol. (IT)*, Feb. 2020, pp. 1–4.

[91] Blockchair. (2023). *Bitcoin Transactions per Second Chart*. [Online]. Available: https://www.hyperledger.org/blog/2023/02/16/benchmarking-hyperledger-fabric-2-5-performance

[92] Bitcoin. (2023). *Bitcoin Faq*. [Online]. Available: https://bitcoin.org/en/faq#transactions

[93] Blockchair. (2023). *Ethereum Transactions Per Second Chart*. [Online]. Available: https://blockchair.com/ethereum/charts/transactions-per-second

[94] Ethereum. (2023). *Blocks*. [Online]. Available: https://ethereum.org/en/developers/docs/blocks/

[95] Ripple. (2023). *XRP's Unmatched Benefits*. [Online]. Available: https://ripple.com/xrp/

[96] Hyperledger Foundation. (2023). *Benchmarking Hyperledger Fabric 2.5 Performance*. [Online]. Available: https://www.hyperledger.org/blog/2023/02/16/benchmarking-hyperledger-fabric-2-5-performance

[97] M. Kuzlu, M. Pipattanasomporn, L. Gurses, and S. Rahman, "Performance analysis of a hyperledger fabric blockchain framework: Throughput, latency and scalability," in *Proc. IEEE Int. Conf. Blockchain (Blockchain)*, Jul. 2019, pp. 536–540.

[98] M. A. H. Khan, A. K. Azad, and V. de Oliveira Cruz, "Bangladesh's digital health journey: reflections on a decade of quiet revolution," *WHO South-East Asia J. Public Health*, vol. 8, no. 2, pp. 71–76, 2019.

[99] Bangladesh Computer Council (BCC). (2023). *Bangladesh Government Wide Network: Banglagovnet Project*. [Online]. Available: https://bcc.gov.bd/site/page/a05da8a7-89d5-4191-bc26-c179f9e33456/BanglaGovNet-Project

[100] D. B. Rejaul and K. Byron, "Digitalisation in govt services saved time, money: Study finds," Daily Star, Dhaka, Bangladesh, Tech. Rep. 6, 2022. [Online]. Available: https://www.thedailystar.net/news/bangladesh/news/digitalisation-govt-services-saved-time-money-and-hassle-3064771

[101] "Development of national ICT infra-network for Bangladesh government (BanglaGovNet) (1st revised)," ICT Division Bangladesh, Dhaka, Bangladesh, Tech. Rep. 8, 2017, pp. 11–35. [Online]. Available: https://imed.portal.gov.bd/ and https://imed.portal.gov.bd/sites/default/files/files/imed.portal.gov.bd/page/e773d5bf_182e_4fc5_a856_dfd3c8d05ced/BanglaGov_Full_June_Report.pdf

MD. ARIFUL ISLAM (Member, IEEE) received the bachelor's degree in software engineering from American International University-Bangladesh (AIUB), in 2021. After three months of internship with EXIM Bank's IT Division, he started his professional career with "Robust Research and Development" (RRAD) for Bangladesh Customs and Vat mobile application project (on contract). He also volunteered at one of the international client meetings of Datasoft Manufacturing and Assembly Inc., Ltd., as a Consultant of private-blockchain technology. He is currently an Associate Automation Engineer with Brain Station 23 Ltd. After publishing his first work at the Google Play store, he got the opportunity from RRAD to design and develop an entire import entitlement system for the Bangladesh Customs Bond Commissionerate. In February 2021, he was among the top 40 National Finalist at the Blockchain Olympiad Bangladesh (BCOLBD), and in its order, he was among the top 12 of the National Teams of Bangladesh at the International Blockchain Olympiad 2021 (IBCOL 2021), where he secured the Merit Award on behalf of Bangladesh.

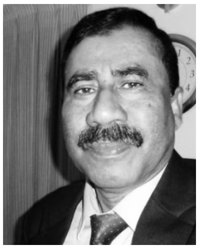
MD. ANTONIN ISLAM received the bachelor's degree in software engineering from American International University-Bangladesh (AIUB), in 2021. After several contract works with domestic and foreign clients, he started his professional career with "Robust Research and Development" (RRAD). Before that, he also volunteered at one of the international client meetings of Datasoft Manufacturing and Assembly Inc., Ltd.., as a Consultant of private-blockchain technology. He is currently an Associate Software Engineer with Brain Station 23 Ltd. His research interests include blockchain, the Internet of Things, and cloud computing. In February 2021, his team was among the Top 40 National Finalists at the Blockchain Olympiad Bangladesh (BCOLBD). He also received the Merit Award from the International Blockchain Olympiad 2021 (IBCOL 2021) where his team represented his country as one of the Top 12 National Teams of Bangladesh. His first paper "An automated monitoring and environmental control system for laboratory-scale cultivation of oyster mushrooms using the Internet of Agricultural Things (IoAT)" was presented at the International Conference on Computing Advancement (ICCA 2022).

MD. AMZAD HOSSAIN JACKY received the bachelor's degree in software engineering (SE) from American International University—Bangladesh (AIUB), Dhaka, Bangladesh, in 2021. He has done research in the field of human–computer interaction (HCI) named "Ball Game Controller: A Tangible User Interface." His research interests include blockchain technology, Web 3.0, and cloud computing. He was among the top 40 National Finalists at the Blockchain Olympiad Banglades (BCOLBD) 2021. After that, he received the Merit Award from the International Blockchain Olympiad 2021, where he worked extensively on server and network configuration and administration.

MD. AL-AMIN received the bachelor's degree in software engineering and the master's degree (magna cum laude) in computer science and engineering (CSE) from American International University-Bangladesh (AIUB), Dhaka, Bangladesh, in 2015 and 2017 respectively. He is currently a Lecturer with the Computer Science Department, AIUB. Besides his teaching profession, he is actively doing research and development projects on freelance platforms for different clients across the globe. He is also supervising research and development teams and providing technical consultancy. His research interests include distributed ledger technology (DLT), blockchain technology, Web 3.0, distributed computing, the Web of Things, information security, web assembly, and knowledge base systems. He has also served as a Registration Committee Member for the International Conference on Computing Advancements (ICCA 2020). He is a member of the Bangladesh Computer Society. During his master's degree, he received the academic distinction Magna Cum Laude (Silver Medal) Award for his academic results. He achieved ICT Fellowship Awards (twice: 2015–1016, 2016–2017) from the Ministry of ICT, Government of Bangladesh.

MD. SAEF ULLAH MIAH (Member, IEEE) received the B.Sc. and M.Sc. degrees from American International University-Bangladesh (AIUB). He is currently an Assistant Professor with the Department of Computer Science, AIUB. He is also engaged in research and teaching activities and has practical experience in software development and project management. In addition to his professional activities, he is passionate about working on various open-source projects. His main research interests include data and text mining, natural language processing, machine learning, material informatics, and blockchain applications.

MD. MUHIDUL ISLAM KHAN received the bachelor's degree in computer science and engineering from the Khulna University of Engineering and Technology (KUET), in 2007, the master's degree from the Bangladesh University of Engineering and Technology (BUET), in 2009, and the Ph.D. degree under the Erasmus Mundus Grant from the European Commission, in 2014. He was with Klagenfurt University, Austria, and the University of Genova, Italy. He was an Assistant Professor with BRAC University, Bangladesh, for one year. After that, he completed his one-year postdoctoral training with the Hebei University of Technology, Tianjin, China. He was a Research Scientist with the Electronics Department, Tallinn University of Technology, Tallinn, Estonia. He was a Senior Lecturer with the School of Information Technologies, Tallinn University of Technology. He is currently an Associate Researcher with the University of Stavanger, Norway. He participated in the ''eLINK''-project with the Corvinus University of Budapest, Hungary, from September 2009 to July 2010 (funded by the European Union). His research interests include wireless sensor networks, networked embedded systems, pervasive computing, machine learning/AI-based resource allocation in wireless networks, and blockchain-based technologies.



MD. IQBAL HOSSAIN received the M.B.B.S. degree from the Chittagong Medical College, Chittagong University, in 1981, and the master's degree in pediatrics from IPGMR, Dhaka, and the Bangladesh College of Physician and Surgeon, in 1994. He was a Freedom Fighter of the Bangladesh Liberation War, in 1971. He was a Doctor. He started his career with Rural Health Complex, from 1983 to 1989, as a Medical Officer. Since that time, he was working for the health of poor rural people of Bangladesh. He also took part in the EPI Program. From 1995 to 1999, he was a Pediatric Consultant with Rangamati General Hospital. From 1999 to 2000, he was an Assistant Professor of pediatrics with the Mymensing Medical College. After that, he went to Saudi Arabia and was with Al-Jouf Maternity and Children Hospital, till 2005. In 2005, he returned to Bangladesh and joined the Faridpur Medical College, as an Assistant Professor. During this time, he provided consultancy services and teaching on pediatrics to the undergraduate medical student. He was there till 2010 and then he left Bangladesh again and was a Pediatrician with Maldives health service, till 2015. He is currently serving the people of his own country (due to the pandemic caused by COVID-19) at a private hospital in Chakaria, Cox's Bazar, Bangladesh.