

Received November 8, 2021, accepted November 29, 2021, date of publication December 8, 2021, date of current version December 17, 2021.

Digital Object Identifier 10.1109/ACCESS.2021.3133953

Blockchain-Based Management of Blood Donation

DIANA HAWASHIN¹, DUNIA AMIN J. MAHBOOBEH²,
KHALED SALAH², (Senior Member, IEEE), RAJA JAYARAMAN¹,
IBRAR YAQOUB², (Senior Member, IEEE), MAZIN DEBE², AND SAMER ELLAHHAM³

¹Department of Industrial and Systems Engineering, Khalifa University of Science and Technology, Abu Dhabi, United Arab Emirates

²Department of Electrical Engineering and Computer Science, Khalifa University of Science and Technology, Abu Dhabi, United Arab Emirates

³Heart & Vascular Institute, Cleveland Clinic Abu Dhabi, Abu Dhabi, United Arab Emirates

Corresponding author: Ibrar Yaqoob (ibrar.yaqoob@ku.ac.ae)

This work was supported by the Khalifa University of Science and Technology under Award CIRA-2019-001.

ABSTRACT Today's a large number of blood donation management systems fall short in providing traceability, immutability, transparency, audit, privacy, and security features. Also, they are vulnerable to the single point of failure problem due to centralization. In this paper, we propose a private Ethereum blockchain-based solution to automate blood donation management in a manner that is decentralized, transparent, traceable, auditable, private, secure, and trustworthy. The proposed solution stores non-critical and large data off-chain using the decentralized storage of the InterPlanetary File System (IPFS). We present the system architecture, sequence diagrams, entity-relationship diagram, and algorithms to briefly explain the working principles of our blood donation management solution. We evaluate the performance of our solution in terms of efficiency and effectiveness through performing security analysis. We make our smart contract code publicly available on Github¹.

INDEX TERMS Blockchain, private Ethereum, blood donation management, blood bank, traceability, security.

I. INTRODUCTION

Blood is one of the most crucial fluids in the human body. It contributes in aiding the organs with the essential and valuable substances required for living. Since the demand for blood surpasses all other medical necessities, governments often educate their citizens on the importance of blood donation through organizing awareness programs. The number of donors in the years 2018-2019 were estimated to be 136,908 donors, contributing to a total of 216,639 donations [1], [2]. In general, every 56 days, mostly healthy individuals give blood donations [3]. The World Health Organization (WHO) estimates that the annual amount of blood donations collected is 112.5 million units which is approximately 50 million liters per year [4]. Yet, the shortage of blood donors has risen with the emergence of new diseases, raising the need to enable reliable and efficient blood donation management [5]. Patient Blood Management (PBM) is a vast and challenging task. The restrictions and gaps occurring with the current blood management system limit the efficient performance of the

supply chain. Hannon [6] reported that the blood component wastage rates usually run from 1% to 5%, and that the amount of disposal is not shared or visible to clarify the reason behind it. Thus, any improvement or development is a significant factor in providing effective healthcare worldwide.

Figure 1 illustrates a typical flow process of blood donation. First, donors have three options to donate blood. Option 1, through healthcare centers where blood units are transported to the nearest blood bank. Option 2, through mobile blood collection units. Option 3, directly through blood banks. After that, separation, testing, and storage are operated on each donated blood in the blood bank. The separation process is based on separating whole blood units into components of red cells, platelets, and plasma through centrifuges. Next, testing is performed to verify the blood type and indicate any infectious diseases. When test results are established, units proper for transfusion are then labeled and stored either in refrigerators and freezer lockers or in walk-in cool and freeze rooms [7]. According to the Food and Drug Administration (FDA) and the American Association of Blood Banks (AABB) standards, red blood cells are stored in refrigerators at 6°C and have an expiration date of 42 days.

The associate editor coordinating the review of this manuscript and approving it for publication was Pedro R. M. Inácio¹.

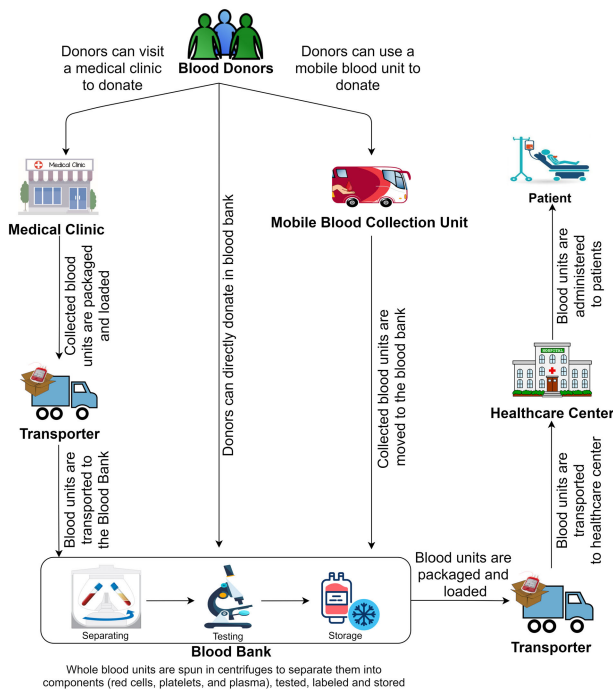


FIGURE 1. Flow diagram of the existing blood donation process.

As per the FDA requirements, plasma is frozen in freezers for up to one year, and platelets are stored for up to five days at room temperature [8], [9]. Subsequently, blood units are packaged and loaded to transporters based on doctors' orders for their patients' treatment. Finally, after healthcare centers receive blood units, they further transfuse to patients [10].

Blood-related information might range from blood type to blood state, to the donor's health record, when the donor donated the blood, and other related readings. Despite the benefit of interblood bank transfers, hospitals commonly fear the chance of receiving the wrong blood, or even worse, blood infected with hepatitis, HIV, or other similar diseases [11]. Several risks are associated with carrying infective donated blood in the supply chain. An epidemic in the late 80s occurred because polluted blood infected with HIV was carried out in the supply chain [12]. Nonetheless, counterfeiting and forgery of medicinal products are considered as another concern in the supply chain, where a falsified illegal copy of an original product can be replaced with the original product itself. This results in the possibility of replacing the blood with another type or attaching a false label to cover up the existing effective blood [13]. These are major obstacles in the supply chain management system which caught the attention of researchers and other interested parties.

Blockchain-based technology in the blood supply chain can assist in reducing the aforementioned risks. The emerging technology possesses several solutions for the verification of the origin of the donated blood. This can be made possible by tracing the source information of the donors in a trusted manner throughout the stages of the supply chain. Several

countries across the world have emphasized the importance of traceability and mandated its existence in healthcare supply chains. The Drug Supply Chain Security Act (DSCSA) in the United States, has obligated pharmaceutical industries to inherent electronic systems that will identify prescriptive drugs while being distributed across the country [14]. Similarly, China requires all stakeholders in healthcare supply chains to use a specialized IT system to be able to record the information of products whenever they are sent to or from their warehouses [15]. Subsequently, several supply chains integrated traceability as an important part of establishing authenticity across their chains. This paper proposes a blockchain-based blood donation chain management system to address the aforementioned issues. The main contributions of this paper are as follows:

- We propose a private Ethereum blockchain-based solution to automate blood donation management processes in a manner that is fully decentralized, traceable, transparent, auditable, private, secure, and trustworthy.
- We integrate the private Ethereum blockchain with the decentralized storage of the InterPlanetary File System (IPFS) to overcome the storage limitations.
- We develop two smart contracts along with algorithms to implement functionalities and define rules regarding blood donation management.
- We evaluate the proposed blockchain-based blood donation management solution and the developed smart contracts using the security analyses. Also, we compare our proposed approach with the existing solutions.
- Our proposed blockchain-based blood donation management solution is generic and can be customized to meet the needs of other industrial applications with minimal modifications and efforts.

The rest of the paper is structured as follows. Section II and III present the background and related work which is classified into two categories; namely, blockchain and non-blockchain based blood donation management solutions. Section IV describes the proposed blockchain-based solution for blood donation management. Section V and VI present the implementation, testing, and validation details of the proposed solution. In Section VII, we conduct the security analysis of the proposed approach. Section VIII summarizes our findings along with concluding remarks.¹

II. BLOCKCHAIN BACKGROUND

Blockchain has been implemented in many fields and disciplines. It is made up of a chain of blocks, where each block comprises information on several transactions. Blockchain could also be referred as a ledger [16]. The blockchain network consists of several nodes which are the participants of the chain, where each participant is granted the authority to make direct transactions in a decentralized manner. Transactions are authorized only when validated by all the nodes in the network. This procedure grants the network its com-

¹<https://github.com/BloodMang/PBMTraceability>

monly known immutability feature. Furthermore, blockchain participants tend to digitally sign transactions using public-key cryptography. In a cryptographic system, each participant in the blockchain network is provided with a private and a public key. Upon executing a transaction, a node digitally signs it with its private key and the recipient verifies it using the sender's public key. The transactions are then further validated by other nodes, validators, using a consensus algorithm. This method ensures the high security of the network and avoids illegal actions of any sort. Besides, blockchain is classified into three categories, public, private, and consortium. The three classes differ in the number of nodes that are allowed access in the network and other characteristics. The types of blockchains are addressed below.

In public blockchains, anyone can participate and act as a node in the ledger, hence the name public. As soon as the members are assigned as nodes, they are granted the authority to perform transactions within the network. Yet, a consensus protocol exists in public blockchains where transactions are required to be validated by the other participating nodes in the ledger. This excludes the need for a central authority to be responsible for transaction validation. However, public blockchains are known for being vulnerable to cyberattacks and susceptible to network security threats. The particular reason for the circumstance is the fact that the identity of the validators and other members as well as the details of their transactions on the blockchain are anonymous. This results in a highly possible leakage of data as a consequence of such features of public blockchains.

On the other hand, private blockchains are permissioned systems which are controlled by a single entity. Generally, the operation rate of private blockchains is known for being faster, as the mathematical operations required by the network are less complex than it is in public blockchains. The consensus algorithms used in private blockchains for transactions' approval are based on voting or multi-party, wherein such consensus algorithms transactions are approved in a very short time. This allows low energy consumption and hence faster transaction process. In a private blockchain, data protection and transparency are guaranteed for users, since the identity of the assigned validators and nodes is known to the central authority [17].

A consortium blockchain is simply a combination of the aforementioned two types of blockchains, public and private. It is considered as a permissioned blockchain however, unlike the private blockchain, the authority in a consortium is given to multiple groups. As a result, consortium blockchains are considered partially decentralized networks. The members in the consortium network are given the accessibility to read data from the ledger, however, only a few registered trustworthy users are given the authority to write data. In addition, the consortium is considerably faster than public blockchains, for the holding authorities are aware of the users' identities [18].

Moreover, the employment of blockchain technology in the medical domain is gradually boosting, and private companies in worldwide are using blockchain to store and interchange

medical records [5], [19]. Blockchain is an emerging technology which helps to avoid forgery of any sort, accomplishes transparency, and ensures the security of patients' identity and medical records in the blood cold supply chain. It works on procuring a trusted and untampered way for blood to be donated in the process flow from the collection, to storing until received by patients [20], [21].

III. RELATED WORK

Traceability is a well-known issue in several scientific and research areas. It is defined as the level to which one can trace specific items from their origin or production to their consumption. The supply chain is one of the main fields where the traceability of data is especially crucial due to the large number of involved stakeholders. The use case of managing the supply chain of blood unit donation is more complex than other types of supply chain networks due to the fact that the algorithms used can be relatively simple. However, there is a large number of stakeholders involved in extracting the blood units, transferring, storing, distributing, and providing them to patients. Moreover, the data in blood supply chain systems and any type of pharmaceutical supply chain is considered sensitive due to the importance of the origin of the items being transferred, the storage conditions, and the time period of transportation. Therefore, it is imperative to maximize the visibility of data to guarantee the safe delivery and distribution of blood units and establish trust in their quality. In this section, we review both blockchain and non-blockchain-based solutions proposed for visibility and traceability problems in the blood supply chain.

A. NON-BLOCKCHAIN-BASED SOLUTIONS

A large number of traceability solutions use a central server to handle the visibility and traceability problems. One popular solution for tracking and monitoring blood bags is using the radio frequency identification (RFID) technology. The RFID is a standing-out technology because of its high significance in supply chain management. In industrial applications, this technology is used to transfer data utilizing radio frequency waves and further identifies tagged items [22]. Davis *et al.* [23] proposed the RFID-based dynamic blood information management system to track blood products in blood centers. The RFID chip is added to the blood bags as part of the recognition procedure during the preparation period. When an order is issued from a healthcare center, the delivery operator picks the bag and encodes the RFID chip to its destination. The central information system keeps the track record of each blood bag using the RFID reader/encoder that automatically reads the product code registered on the chip and the blood donation number. This solution allows several tags to be read simultaneously, stores information on the chip, provides condition tracking sensors such as time and temperature, and facilitates automated detection and data collection [24].

To enhance productivity and safety, the RFID technology is integrated with supply chain processes. Although the integration has shown sufficient performance in healthcare supply

chains, it can also be used for the traceability and transportation of blood components. However, the RFID technology suffers from security and privacy problems. For example, tags can be read from a distance in a range of inches to some yards with a high-gain antenna reader, which can enable anyone to see the contents. Also, some tags can be turned off, and since the RFID-based systems use electromagnetic spectrum, they can be easily disrupted [25].

To provide food consumers with transparency, the authors in [26] presented an implementation of a microservices-oriented software architecture for middleware that collects, stores, and traces data origins in a centralized manner to provide data analysts with a resource for new insights into a whole supply chain. Moreover, a central blood donation management solution was proposed in [27]. It advances the efficiency of blood donation management by linking several systems involved in the blood donation process. It reduces the time required for blood donation by minimizing the amount of information collected from donors.

Eskandari-Khanghahi *et al.* [30] created a multi-period and multi-objective blood supply chain optimization model. The model accounts for uncertain data due to disaster and post-disaster rough conditions. The solution includes blood transfer between several blood facilities, transferring vehicles, blood centers, and hospitals. It also employs linear programming that considers location, allocation, and delivery of blood. The authors of [31] also proposed a supply chain solution to manage critical blood supply requests in crucial situations. Their model presented an algorithm to reduce the cost of operation for the blood supply chain. In their model, blood units are transferred among donors, blood collecting centers, ambulances, laboratories, and blood centers. The authors of [32] also introduced a centralized approach that comprises a network that implements an optimization model for blood supply chain management. The developed model aids in localizing blood facilities and allocating resources efficiently for critical times. The aforementioned solutions employ advanced optimization models in order to efficiently manage and orchestrate the blood supply chain and transferring of blood units. However, they suffer from some notable drawbacks that need to be discussed. The main issue is the lack of data visibility in the implemented system, which is what we are trying to achieve in this paper. While accounting for the traceability of the supply chain, utilizing a centralized system can compromise the transparency of data. The only information known about the blood units being transferred is what is provided by the central server. Therefore, any information that was not recorded or was obscured by the server will not be available to the rest of the members in the system, and most importantly, the doctors and their patients. Another major issue related to centralized servers is the single point of failure. Aggregating all of this crucial patient data into a single vulnerable data location imposes a huge risk on the entire supply of blood units in case of any data loss. This significantly affects the availability of the system, as any downtime of the central server implies that the system cannot

operate, which could lead to fatal consequences in the case of blood supply chain solutions.

B. BLOCKCHAIN-BASED SOLUTIONS

Traceability of blood units requires that stakeholders are aware of the status of the blood units at all times. Employing a decentralized technology, such as the blockchain, provides the required visibility as all data is clearly available to all members of the network. Data that is transacted to and stored on the blockchain is permanent, which means that in the case of the blood supply chain, when the blood units are transferred to another entity and recorded on the blockchain, none of those entities can deny transporting or receiving the blood units. This ensures the accountability of each member as all of its actions are recorded and can be accessed by any member of the blockchain network. Moreover, because of the decentralized nature of the blockchain, the same information is replicated on all the full nodes that are part of the network. This indicates that the data is tamper-proof and cannot be altered, which is crucial considering the significance of the data related to the blood units and donors. In addition, it was mentioned earlier that the algorithms used in the blood supply chain are generally not very complex unless a sophisticated probabilistic model has been used. Otherwise, fairly simplified algorithms can be utilized to achieve the goal of the blood supply chain. This makes the blockchain an ideal platform for such an application, as systems developed on the blockchain network are required to be cost-efficient and simplistic in nature. This is because the cost of operation significantly increases with more complex algorithms when using smart contracts on the blockchain. While blockchain applications are preferred to be cost-efficient, they are capable of handling a large number of requests and transactions, which is needed in the case of most supply chain networks. Therefore, we can conclude that blockchain technology can be used for a more efficient and effective way to employ a pharmaceutical supply chain, including the blood donation supply chain, as it overcomes the issues with centralized solutions related to visibility, transparency, availability, and data integrity. Avoiding such issues significantly improves the performance and reliability of tracking donated blood units from their origin to the patient recipients.

Kim *et al.* [5] proposed a blood cold supply chain system which is based on the Hyperledger Fabric. The proposed system assists in verifying the temperature-related transactions, and timestamps are recorded to specify when the temperature is confirmed. Moreover, it indicates the state of storing the blood units, where participants in the blood cold supply chain update the reason behind the status change. Besides, the authors addressed the case of an emergency and defined a business logic to allow blood to be exchanged between hospitals. This system is designed to share information as blood is moved, consumed, and discarded in real-time. Similarly, Lakshminarayanan *et al.* [11] proposed a system which is implemented in the Hyperledger Fabric framework. It ensures transparency of the donated blood by tracking the blood units

and providing a platform for the exchange between blood suppliers. Toyoda *et al.* [28] supported the RFID supply chain using EPC stored in the tag to distinguish items from one another. This integration assists reliability and avoids forgery through tracking products and checking their tags. Yet there are some limitations to the aforementioned solutions. For example, the verification of the system proposed in [5] is incomplete due to the lack of assessment analysis. As it is required to show its feasibility by incorporating the proposed design into the real blood information management system. Nonetheless, there is a need for a hacking prevention technique for patients' blood transactions. Moreover, the tracking solution proposed in [11] was only limited to the monitoring of blood bags, and it does not ensure the traceability of the blood components. Since different components of blood have different expiry dates and storage temperatures, it is essential to consider these aspects.

The authors in [33] discussed how the decentralized blockchain can benefit the pharmaceutical supply chain. In particular, they consider the plasma derivatives supply chain as a use case to present their findings. The major risks of managing the supply chain are highlighted as well, such as maintaining the origin of the plasma and identifying poor blood quality. Hence, the blockchain was proven to be a suitable solution to overcome these challenges. Moreover, the paper also explains how incentives can motivate donors to be more active. Another solution based on the Hypeledger was proposed in [19] to implement a blockchain-based blood supply chain management system. In addition to this solution being based on a private blockchain, it does not keep track of the identity of the donor and does not examine the components of the blood unit. The authors in [34] proposed a decentralized solution based on an Ethereum-like blockchain to facilitate blood tracking. In their design, certified blood donation centers (CBDC) are the only privileged members to have writing permissions. Donors are recognized via an identifier such as their social security number and passwords.

In summary, none of the aforementioned blockchain-based solutions ensure the privacy of data while providing a robust system to track and manage the supply chain of donated blood. In contrast, the solution offered in this paper overcomes the aforementioned performance issues of traditional solutions while simultaneously preserving the privacy and integrity of data by utilizing the private Ethereum blockchain. Our proposed solution captures different aspects of the blood donation system such as collecting, delivering, requesting, and transferring blood units. It ensures that all these aspects are captured in a decentralized, traceable, accountable, transparent, secure, and auditable manner. Our solution traces all the necessary phases of the donated blood unit cycle, starting from the donation process until consumption. Our solution includes all actors that contribute to the blood unit donation process to ensure trust and accountability. Another key feature of our solution is that we used a private instance of the Ethereum blockchain to ensure the privacy and confidentiality of the stored information of the involved entities.

The private Ethereum blockchain ensures that stored data can be only viewed by authorized entities. Finally, in comparison to other solutions, our solution is the only one that takes into consideration the security of the smart contract, which is extremely important to ensure that the stored information is safe at all times and that there are no vulnerabilities. Moreover, the implementation, testing, and validation of the smart contract were thoroughly investigated to ensure that it was functioning properly.

IV. BLOCKCHAIN-BASED DONATED BLOOD TRACEABILITY SYSTEM

In this section, we propose a blockchain-based approach for blood donation management. Our proposed system is based on the private Ethereum network, which organizations mainly develop to increase the confidentiality of the network. Only nodes with proper authorization can access this blockchain. Moreover, these nodes are not connected to the main network, and they can access this private blockchain only. Blood management is a sensitive area holding private information; hence, records of patients and other medical supplies must be accessible to specific participants and entities. Therefore, a private Ethereum blockchain network is more appropriate to be implemented in the suggested system as it adds privacy, confidentiality, and authorization.

Figure 2 demonstrates a high-level system architecture of the proposed solution. Our approach leverages the smart contract feature of the private Ethereum blockchain. There are two smart contracts in our proposed solution; namely, the production smart contract and the consumption smart contract. Frontend Decentralized Applications (DApps) are used by actors to access smart contracts, and an application programming interface (API) is used to link the software devices that access the DApps to the smart contracts. Some examples of APIs that can be used include Infura, JSON RPC, and Web3. Also, only pre-authorized functions can be executed by the appointed actor, who has restricted access to the smart contract functions. Furthermore, storage systems require the handling of media such as images. Besides, all transactions and details are stored securely on the ledger. The following are the details of the system components:

- **Actors** are the entities responsible for executing functions within the smart contracts. The actors can be split into two groups. One group has access to the functions within the production smart contract, including phlebotomist, transporter, and blood bank technician. The other group has access to the functions within the consumption smart contract, including the blood bank administration, doctor, and nurse.
- **Storage Systems** provide a way for storing large-sized data that would be costly to store on the Ethereum distributed ledger. The decentralized storage system named IPFS assists in storing large-sized files in a decentralized manner, and a hash corresponding to the files is uploaded and stored on the ledger.

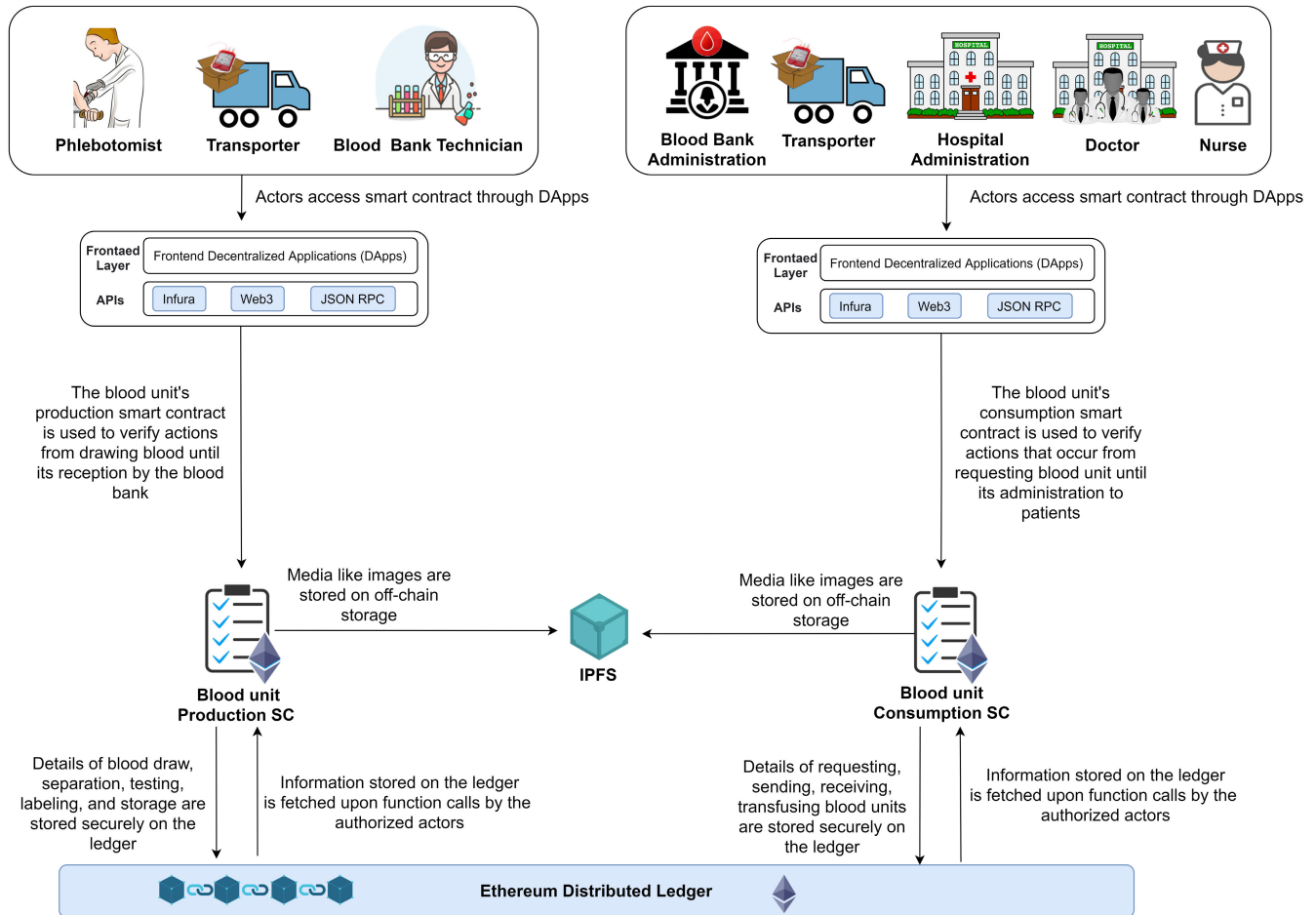


FIGURE 2. A high-level architecture of the proposed blockchain-based system for blood donation supply chain.

- **Ethereum Smart Contracts** define the functions of the different actors in the blood cold supply chain. Additionally, by using modifiers, access to these functions is given to the authorized actors. A modifier is essentially a way to customize a function by adding more features or imposing constraints.
- **Ethereum Distributed Ledger** is a database shared across the network and stores all details fetched upon function calls by actors. It ensures security and provides a high level of transparency.

The following subsection discusses sequence diagrams representing the interactions between the actors, smart contracts, and storage systems are represented.

A. SEQUENCE DIAGRAMS OF THE PROPOSED BLOCKCHAIN-BASED SOLUTION

Figure 3 shows the interactions among the different actors within the production smart contract that enable to collect, transport, and create blood units. First, the phlebotomist will deploy the contract and initiate the whole blood unit collection function. Once the event is emitted, the transporter will execute the start delivery function. An event will immediately

be sent; thus, the blood bank technician will prepare the separation and testing setups. The delivery will then end, and an event will be broadcasted to declare that the collected blood units have been delivered. Next, collected blood units' reception function will be run by the blood bank technician. Then, images of the ready blood components units will be uploaded to the IPFS after the separation and testing process. Finally, a blood unit creation function will be initiated by that technician, and an event will be triggered to announce that either the red cells, plasma, or platelets unit production process has ended successfully.

Figure 4 represents the interactions among actors involved in blood unit consumption within the consumption smart contract. The first step describes the interaction between the blood bank administration and the doctor. The former will initiate blood units' readiness for ordering function, and an event will be emitted depending on the blood component unit type. The latter will call a request function which places an order for blood component units. Later on, approval for one of the blood component unit request will be granted by the blood bank administration. Then, the transporter will execute the start delivery function when an event for the approval of the doctor's request is broadcasted. Then, once the blood units reach their destination, the delivery process

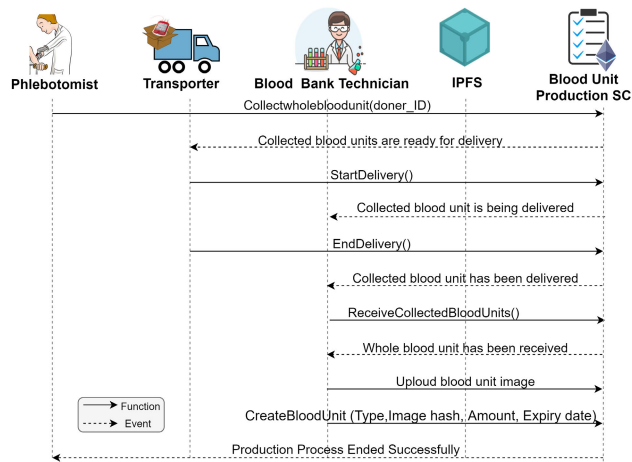


FIGURE 3. Sequence diagram for blood unit production smart contract.

will end, and an event will notify all entities. After that, the order's reception function will be run by the doctor and an event will be emitted. Next, a prescription for a needed blood component unit function will be executed by the doctor where the patient's name, blood component type, and amount are specified. Based on the doctor's prescription, the nurse will run the blood administration function. Lastly, an event announcing the end of the blood unit consumption process is emitted.

V. IMPLEMENTATION OF THE PROPOSED BLOOD DONATION MANAGEMENT SOLUTION

In this section, we present our developed algorithms along with the implementation details of the proposed blockchain-based solution for blood donation traceability. Moreover, we conduct the security analysis of the developed smart contracts.

The private Ethereum blockchain platform is used to develop the proposed blood donation management and traceability system. It enables the development and implementation of smart contracts and Distributed Applications (Apps) with no downtime, manipulation, regulation, or third-party interference. More specifically, it allows developers to develop and publish distributed applications as it can be both a framework and a programming language that runs on a blockchain [8]. Furthermore, Remix IDE is used to compile and test the smart contracts within the private Ethereum blockchain, as it is an online web-based programming platform where codes are written and executed. Solidity is the programming language used to create these smart contracts. In addition to that, Remix IDE gives users the privilege to debug and test their codes.

A. IMPLEMENTATION DETAILS

The production smart contract contains various types of variables. The phlebotomist and blood bank technician are the only entities that will have an Ethereum address assigned to them. Mapping of the transporters to ensure that only those who are permitted to transport the collected blood unit from

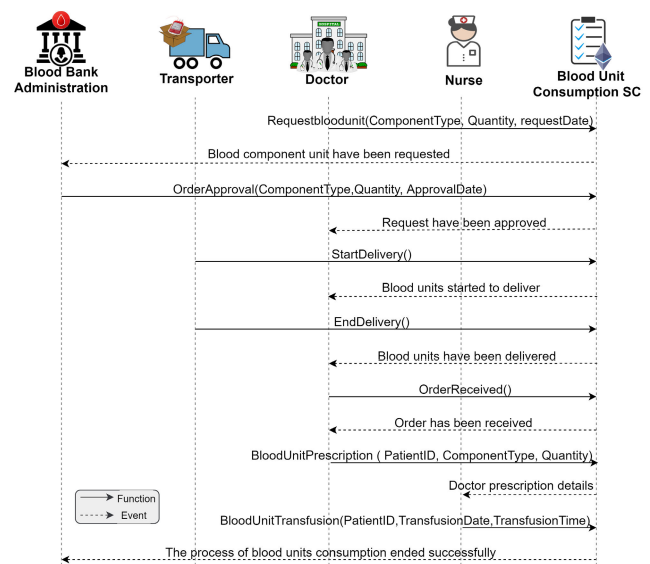


FIGURE 4. Sequence diagram for blood unit consumption smart contract.

the donation location to the blood bank are authorized. Furthermore, there is an enumerating variable named "wholebloodunitStatus" which contains the different states that the collected blood unit will go through such as "NotReady", "ReadyforDelivery", "StartDelivery", "on Track", "End-Delivery", and "wholebloodunitReceived". Besides, the enumerate variable "BloodcomponentType" takes an input in the form of uint8 where "0" represents "redcelltype", "1" represents "plasmatype", and "2" represents "platelet-type". The phlebotomist will deploy the smart contract that represents the collected blood unit. The smart contract will primarily define the owner of the smart contract, which is the phlebotomist in this case, the start time of deploying the smart contract, the Ethereum address of the blood bank technician, donor ID, and the initial state of the whole blood unit. Then, once the smart contract gets deployed, the tracing processes begin. Moreover, the trusted transporters will be assigned to allow them later to execute functions that need authorization.

The whole blood unit tracing process consists of five core steps. First, the collection of the donated whole blood unit where the phlebotomist will announce that the whole blood unit is collected and ready for delivery. Next, the authorized transporter will have to load and pick up the collected blood unit from the phlebotomist and announce the start of the delivery process. Then, the arrival of the collected blood unit at the blood bank states the end of the delivery process. After that, the state of receiving the whole blood unit will be announced by the blood bank technician. Finally, the creation of the suitable transferable blood unit will be done and announced by the blood bank technician including the blood component type, amount, and expiry date.

On the other hand, the consumption smart contract contains various types of variables. Ethereum address includes the blood bank administration and hospital administration.

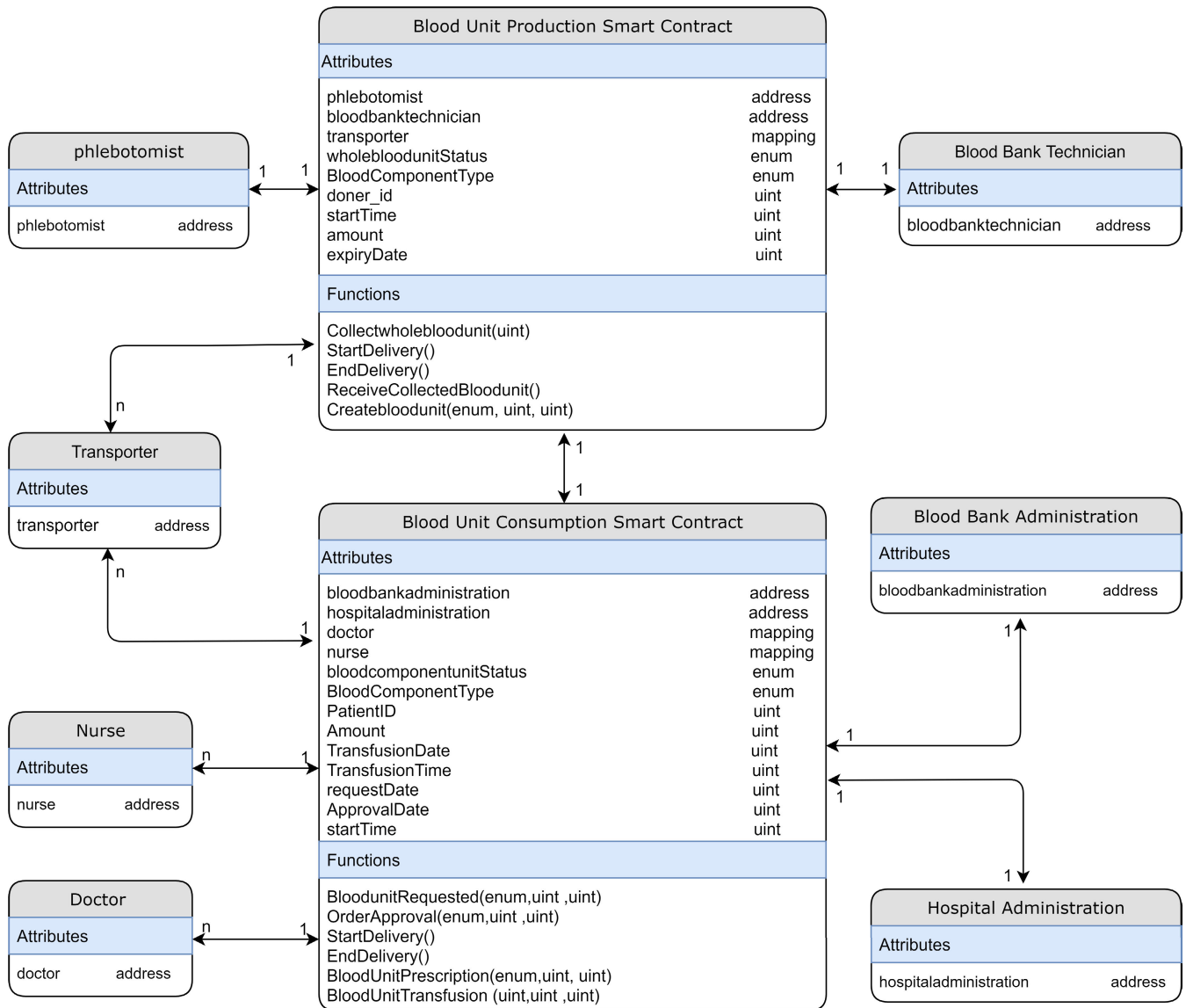


FIGURE 5. Entity-relationship diagram.

As well, it has mapping for transporters, doctors, and nurses. Only authorized transporters are allowed to transport the blood unit orders from the blood bank to the hospital. Besides, as the production smart contract, there are enumerate variables, the “bloodcomponentunitStatus” which contains the different states that the blood component unit will go through, such as “NotReady”, “ReadyforDelivery”, “StartDelivery”, “on Track”, “EndDelivery”, and “bloodcomponentunitReceived”. Also, the enumerate variable “BloodcomponentType” takes inputs in the form of uint8 where “0” represents “redcelltype”, “1” represents “plasmatype”, and “2” represents “platelettype”. The blood bank administration will deploy the smart contract and will be defined as the owner. In addition, the smart contract will also define the start time of deploying the smart contract,

the Ethereum address of the hospital administration, and the initial state of the blood component unit. Once the smart contract gets deployed, the consumption tracing process begins and the trusted transporters will be assigned by the blood bank administration. Besides, the trusted doctors and nurses will be assigned by the hospital administration. Moreover, there are six main steps through tracing the consumption process. First, blood component unit will be requested where the authorized doctor will announce that order request with its details such as blood type, quantity, and date of requesting. The requesting can be for either red cells, plasma or platelet unit. Second, the order approval will be announced by the blood bank administration. Third, the authorized transporter will have to pick up the blood component unit order from the blood bank and announce the start of the delivery process. Fourth,

the end of the delivery process will be announced when the order arrived at the hospital. Fifth, the blood unit prescription with the details of patient ID, blood component unit type, and quantity will be announced by the authorized doctor. Finally, in the sixth step, the transfusion of blood component unit will be done, where the authorized nurse will announce the ending of consumption process.

Figure 5 illustrates the entity-relationship diagram that stands out the smart contracts' attributes, functions, and relationships amongst the involved actors in each one. Firstly, the smart contract of the production process is created using the attributes such as phlebotomist, the blood bank technician, and mapping, which hold the lists of authorized transporters. Every production smart contract is created for only one collected whole blood unit. Thus, a 1:1 relationship between the phlebotomist and the smart contract. Moreover, a 1:1 relationship between the blood bank technician and the smart contract, while it is n:1 between the transporter and the smart contract. Secondly, the main actors in the consumption smart contract are the blood bank administration, hospital administration, the authorized transporters, doctors, and nurses. This smart contract can be associated with multiple transporters, doctors, and nurses. Meanwhile, only one blood bank administration and only one hospital administration can be associated with one consumption smart contract. Therefore, it is a 1:1 relationship between each administration and the smart contract. However, a production smart contract can interact with a consumption smart contract. Therefore, a 1:1 relationship between two smart contracts is shown in the figure.

The blood unit production process in our solution consists of three main parts: collecting the whole blood unit, the delivery process from the donation location to the blood bank, and creating a transferable blood unit. The pseudocodes for these processes are represented in algorithms 1, 2, and 3, respectively. These algorithms offer a high-level clarification of the smart contract code compared to the aforementioned solidity code.

Algorithm 1: Collecting Blood Units

Input: donorID, bloodbanktechnician address

```

1 bloodunitstate is a customized enumerate variable that
  illustrates different states of the collected blood unit.
Output: An event declaring that whole blood unit is
  ready for delivery.
2 if (caller == phlebotomist)  $\wedge$  (bloodunitstate ==
  NotReady) then
3   bloodunitstate = ReadyforDelivery
4   Emit an event declaring that the whole blood unit is
    ready for delivery
5 else
6   Revert the contract state and display an error
    message.
7 end

```

Algorithm 2: Delivery Process

```

1 bloodunitstate is a customized enumerate variable that
  illustrates different states of the collected blood unit.
2 assignedtransportes_list: A mapping of the transporters
  EAs
3 if caller == phlebotomist then
4   | assigningtransporter_list[transporter] = true.
5 else
6   | Revert.
7 end
  /* Assigned transporters list is ready */
8 if (caller ==  $\in$  assignedtransportes_list)  $\wedge$ 
  (bloodunitstate == ReadyforDelivery) then
9   | bloodunitstate = onTrack
10  | Emit an event declaring that the whole blood unit is
    ready for delivery
11 else
12  | Revert the contract state and display an error
    message.
13 end
  /* Delivery process starts by the assigned transporter. */
14 if (caller ==  $\in$  assignedtransportes_list)  $\wedge$ 
  (bloodunitstate == onTrack) then
15  | bloodunitstate = EndDelivery
16  | Emit an event declaring that the whole blood unit
    delivery procedure has ended
17 else
18  | Revert the contract state and display an error
    message.
19 end
  /* Delivery process ends by the assigned transporter. */
20 if (caller == bloodbanktechnician)  $\wedge$ 
  (bloodunitstate == EndDelivery) then
21  | bloodunitstate = wholebloodunitReceived
22  | Emit an event declaring that the collected blood unit
    has been received
23 else
24  | Revert the contract state and display an error
    message.
25 end

```

Algorithm 1 represents the collection of the whole blood unit. The phlebotomist deploys a production contract, inserts the donor ID and blood bank technician address. Moreover, the collection step can only start if the state of the blood unit is "NotReady," meaning that the blood unit has not been collected yet and not ready to be loaded with a transporter. Algorithm 2 illustrates the process of delivering a collected whole blood unit from the donation location to the blood bank. This process consists of four stages. First, assign a transporters list to restrict the authorized transporters. Then, start the delivery process by an assigned transporter. This stage requires the

Algorithm 3: Creating Component Blood Units

Input: *_BloodcomponentType*, *_A*, *_Expd*

- 1 *BloodcomponentType* is a customized enumerate variable that illustrates different type of the blood component unit. (*redcellstype*, *plasmatype*, *plateletstype*)
- 2 *_A* is the blood component amount in milliliter.
- 3 *_Expd* is the blood component Expiry date.

Output: An event declaring that a production process of blood component unit has been ended.

- 4 **if** *caller* == *bloodbanktechnician* **then**
- 5 **if** *BloodcomponentType* == *redcellstype* **then**
- 6 Production process of red cell unit is ended
- 7 **else**
- 8 **if** *BloodcomponentType* == *plasmatype* **then**
- 9 Production process of plasma unit is ended
- 10 **else**
- 11 *BloodcomponentType* == *plateletstype*
- 12 Production process of platelets unit is ended
- 12 **end**
- 13 **end**
- 14 **else**
- 15 Revert and show an error.
- 16 **end**

collected blood unit state to must be a “ReadyforDelivery.” However, when it is executed, it emits an event announcing that the delivery process has begun. After that, ending the delivery process by an assigned transporter. This step operates only when the whole blood unit state is “onTrack.” Once executed, it emits an event announcing that the delivery process is ended. Finally, receiving the whole blood unit can only be executed by the blood bank technician. Furthermore, it only runs if the state is “EndDelievry.” That means it cannot receive the whole blood unit before announcing the end of the delivery. Creating blood component units can only be run by the blood bank technician, as shown in Algorithm 3. This algorithm has inputs related to each blood component unit, such as type, amount, and expiry date. The created blood component unit can be either red cells, plasma or platelets. However, as an output, an event declaring that a production process of blood component unit has been ended.

On the other hand, the blood unit consumption process in our solution consists of requesting the order, approving the order, order delivery process, doctor’s prescription, and blood component unit transfusion. The pseudocodes for these processes are represented in algorithms 4, 5, 6, and 7, respectively.

Algorithm 4 shows that only doctors who belong to the assigned doctors list created by the hospital administration can request blood component units. An assigned doctor needs to insert some inputs for requesting, such as blood component unit type, quantity, and the requesting date. An event will be emitted depending on the blood component unit requested:

Algorithm 4: Requesting Order

Input: *_BloodcomponentType*, *A*, *ReqDate*

- 1 *BloodcomponentType* is a customized enumerate variable that illustrates different type of the blood component unit. (*redcellstype*, *plasmatype*, *plateletstype*)
- 2 *A* is the blood component amount in milliliter.
- 3 *ReqDate* is the date of requesting the component blood unit
- 4 *assigned_doctors_list*: A mapping of doctors *EAs*
- 5 **if** *caller* == *hospitaladministration* **then**
- 6 *assigned_doctors_list*[*doctor*] = *true*.
- 7 **else**
- 8 Revert.
- 9 **end**
- 10 /* Assigned doctors list is ready */
- 10 **if** *caller* == \in *assigned_doctors_list* **then**
- 11 **if** *BloodcomponentType* == *redcellstype* **then**
- 12 An event declaring that a red cell unit has been requested
- 13 **else**
- 14 **if** *BloodcomponentType* == *plasmatype* **then**
- 15 An event declaring that a plasma unit has been requested
- 16 **else**
- 17 *BloodcomponentType* == *plateletstype* An event declaring that a platelet unit has been requested
- 18 **end**
- 19 **end**
- 20 **else**
- 21 Revert and show an error.
- 22 **end**

red cells, plasma, or platelets. Then, Algorithm 5 represents the order approval which will be run only by the blood bank administration and emit an event that either red cells, plasma or platelet order with its details has been approved. The delivery process of an order from the blood bank to the hospital is represented in Algorithm 6. To start this process, a list of authorized transporters will be assigned by the blood bank administration. Moreover, starting the delivery process requires the order state to must be a “ReadyforDelivery.” However, it had no input, and when it is executed, it emits an event announcing that the delivery process has begun. Then, the end delivery step needs the order state to be “onTrack” to start. Finally, an authorized doctor will confirm the receiving step of the order requested. Algorithm 7 presents the prescription and transfusion process. An authorized doctor will insert the patient ID, the blood component type, and the unit’s quantity. Once the prescription function is executed, an event declaring that a component blood unit prescription has been assigned will be emitted. A prescription can include either red cells, plasma or platelet unit. Subsequently, an authorized

nurse will administer the blood component unit to the patient, and the consumption process will be ended.

Algorithm 5: Approving Order

Input: *_BloodcomponentType*, *A*, *AproDate*

- 1 *BloodcomponentType* is a customized enumerate variable that illustrates different type of the blood component unit. (*redcelltype*, *plasmatype*, *plateletstype*)
- 2 *A* is the blood component amount in milliliter.
- 3 *AproDate* is the date of order approval
- 4 **if** *caller* == *bloodbankadministration* **then**
- 5 **if** *BloodcomponentType* == *redcelltype* **then**
- 6 An event declaring that a red cells unit order with its details has been approved
- 7 **else**
- 8 **if** *BloodcomponentType* == *plasmatype* **then**
- 9 An event declaring that a plasma unit order with its details has been approved
- 10 **else**
- 11 *BloodcomponentType* == *plateletstype* An event declaring that a platelets unit order with its details has been approved
- 12 **end**
- 13 **end**
- 14 **else**
- 15 **end**
- 16 Revert and show an error.

B. TRACEABILITY ANALYSIS OF THE PROPOSED SOLUTION

Each blood unit has a unique Ethereum address generated specifically for it. Hence, logging the Ethereum address of each blood unit can be inconvenient, as errors may occur through such a prolonged fatiguing process. Therefore, a Quick Response (QR) code can be utilized for each blood unit. A QR code is a two-dimensional type of matrix barcode that can be easily read and scanned using smartphones. It is notorious for being a fast technique that encodes over 4000 characters [15]. Using an Ethereum QR code generator, an Ethereum address can be easily mapped to a QR code, in which a unique QR code is generated whenever an Ethereum address is passed. The generated QR code will contain information about the blood unit attached to it and map it to the corresponding Ethereum address whenever it gets scanned. As soon as the QR code gets attached to the blood unit, it will be ready for distribution to hospitals and other medical centers and further transfused to patients. The steps followed in verifying the authenticity of a blood unit are described below. First, the QR code attached to the blood unit is scanned using a DApp. Through web3j, the DApp interacts with the Ethereum local or remote node. Then, the DApp interacts with the Ethereum node, Infura, through JSON-RPC, for the success of mapping the QR code to its matching Ethereum address. The Ethereum node, also known

Algorithm 6: Order Delivery Process

- 1 *Orderstate*: Customized enumerate variable that illustrates different states of the order
- 2 *assignedtransportes_list*: A mapping of the transporters EAs
- 3 **if** *caller* == *bloodbankadministration* **then**
- 4 *assigningtransporter_list*[*transporter*] = *true*.
- 5 **else**
- 6 Revert.
- 7 **end**
- 8 /* Assigned transporters list is ready */
- 9 **if** (*caller* == \in *assignedtransportes_list*) \wedge (*Orderstate* == *ReadyforDelivery*) **then**
- 10 *bloodunitstate* = *onTrack*
- 11 Emit an event declaring that the order is ready for delivery
- 12 **else**
- 13 Revert the contract state and display an error message.
- 14 **end**
- 15 /* Order delivery process starts by the assigned transporter. */
- 16 **if** (*caller* == \in *assignedtransportes_list*) \wedge (*Orderstate* == *onTrack*) **then**
- 17 *Orderstate* = *EndDelivery*
- 18 Emit an event declaring that the order delivery procedure has ended
- 19 **else**
- 20 Revert the contract state and display an error message.
- 21 **end**
- 22 /* Order delivery process ends by the assigned transporter. */
- 23 **if** (*caller* == \in *assigned_doctors_list*) \wedge (*Orderstate* == *EndDelivery*) **then**
- 24 *Orderstate* = *OrderReceived*
- 25 Emit an event declaring that the order has been received
- 26 **else**
- 27 Revert the contract state and display an error message.
- 28 **end**

as gateway, will map the Ethereum address of the blood unit to the smart contract. By mapping the Ethereum address, the events of the different functions of the smart contract that are stored will be highlighted in the ledger.

The QR code assists in mitigating inefficient tracing and false identification of donated blood during the transportation process. The application use case actively demonstrates that our proposed solution is effective in terms of tracing blood units within a healthcare supply chain. The private Ethereum blockchain features such as web3j, Infura, and JSON-RPC

Algorithm 7: Prescription and transfusion

```

Input: _BloodcomponentType, ID, A, Date, Time
1 ID is the patient ID
2 BloodcomponentType is a customized enumerate
   variable that illustrates different type of the blood
   component unit. (redcellstype, plasmatype, plateletstype)
3 A is the blood component amount in milliliter.
4 Date is the transfusion date
5 Time is the transfusion time
6 assigned_nurses_list: A mapping of the nurses EAs
7 if caller == hospitaladministration then
8   | assigned_nurses_list[nurse] = true.
9 else
10  | Revert.
11 end
   /* Assigned nurses list is ready */
12 if caller ==  $\in$  assigned_doctors_list then
13   | if BloodcomponentType == redcellstype then
14     | An event declaring that a red cells unit
       | prescription has been assigned
15   | else
16     | if BloodcomponentType == plasmatype then
17       | An event declaring that a plasma unit
       | prescription has been assigned
18     | else
19       | BloodcomponentType == plateletstype An
       | event declaring that a platelets unit
       | prescription has been assigned
20     | end
21   | end
22 else
23   | Revert and show an error.
24 end
   /* Prescription has been assigned */
25 if caller ==  $\in$  assigned_nurses_list then
26   | An event declaring that a component blood unit has
       | been transfused and the consumption process has
       | been ended
27 else
28   | Revert.
29 end

```

are utilized in the proposed solution, resulting in automating the processes without the need for manual input.

VI. TESTING AND VALIDATION

In this section, the key functions of the developed smart contracts for the proposed approach are tested and validated. Remix IDE is used to implement the assessment stage.

To evaluate the functionality of our smart contracts, we deploy both the production smart contract and the consumption smart contract at “0x5B38Da6a701c568545dCfcB03FcB875f56beddC4” and “0x78731D3Ca6b7E34aC0F824c42a7cC18A495” addresses, respectively. Table 1 shows the Ethereum addresses of some actors in the smart contracts,

TABLE 1. The Ethereum address of each participant in the testing scenario.

	Ethereum Address
Phlebotomist	0x5B38Da6a701c568545dCfcB03FcB875f56beddC4
Blood bank technician	0x4B20993Bc481177ec7E8f571ceCaE8A9e22C02db
Doctor	0x617F2E2fD72FD9D5503197092aC168c91465E7f2
Nurse	0x17F6AD8Ef982297579C203069C1DbffE4348c372

```

[ { "from": "0xd2a5bC10698FD955D1Fe6cb468a17809A08fd005",
  "topic":
  "0x08cbec5a26aca275a89e5c51f8a7865870c44b2af3e0f267efcc123a325414
  2a", "event": "wholebloodunitReadyForDelivery", "args": { "0":
  "0x5B38Da6a701c568545dCfcB03FcB875f56beddC4", "1": "19970507",
  "phlebotomist": "0x5B38Da6a701c568545dCfcB03FcB875f56beddC4",
  "donor_ID": "19970507" } } ]

```

FIGURE 6. Successful execution of *Collectwholebloodunit* function.

as they are used during testing. In addition to that, the used inputs in the functions do not represent real data, they are just assumptions for testing purposes. The transactions and logs of the key smart contract’s functions are presented below.

- **Collectwholebloodunit:** In this function, it was tested whether or not the owner of the smart contract was able to add the donor ID. Successful execution of the function and its corresponding events and logs are displayed in Figure 6.
- **Createbloodunit:** The *Createbloodunit* function is the most important key function in this contract. It notifies the details of the produced blood component units. As we mentioned in the previous section, the enumerated variable “*BloodcomponentType*” takes inputs in the form of uint8 where “0” represents “*redcellstype*”, “1” represents “*plasmatype*”, and “2” represents “*plateletstype*”. Thus, a successful execution of creating a platelet unit is given in Figure 7.
- **BloodunitRequested:** The doctor sends a blood component unit request to the smart contract. This is done using the *BloodunitRequested()* function. Figure 8 shows the logs after the doctor has sent the request. The request includes blood component unit type, quantity, and requested date.
- **BloodUnitPrescription:** In this function, it was tested whether or not the doctor is able to insert the details of the prescription. Successful execution of the function and its corresponding events and logs are shown in Figure 9.
- **BloodUnitTransfusion:** As a final step in this process, the nurse administrates the blood component unit to the patient, and this is done using *BloodUnitTransfusion()* function. Figure 10 shows logs after the nurse has performed this step.

VII. DISCUSSION

This section discusses the generalization aspect of the proposed private Ethereum blockchain-based blood donation management system. Generally, Ethereum-based solutions


```
[ { "from": "0xd9145CCE52D386f254917e481eB44e9943F39138",
  "topic":
  "0x3c698bbc70a2e1f8302003fb662c209bef746413cf8a6531367b9347b60
  14dec", "event": "plateletsunitproductionend", "args": { "0":
  "0x4B20993Bc481177ec7E8f571ceCaE8A9e22C02db", "1": "2", "2":
  "100", "3": "5052021", "bloodbanktechnician":
  "0x4B20993Bc481177ec7E8f571ceCaE8A9e22C02db",
  "BloodcomponentType": "2", "plateletsamount": "100",
  "plateletsexpiryDate": "5052021" } } ]
```

FIGURE 7. Successful execution of Createbloodunit function.

```
[ { "from": "0x99CF4c4CAE3bA61754Abd22A8de7e8c7ba3C196d",
  "topic":
  "0x107c501810866f2cd6492df9d0308093c7b05fbc9840e41d1a8004c2422
  fced5", "event": "PlateletsRequest", "args": { "0":
  "0x617F2E2fD72FD9D5503197092aC168c91465E7f2", "1": "2", "2":
  "100", "3": "29042021", "doctor":
  "0x617F2E2fD72FD9D5503197092aC168c91465E7f2",
  "BloodcomponentType": "2", "Amount": "100", "requestDate":
  "29042021" } } ]
```

FIGURE 8. Successful execution of BloodunitRequested function.

```
[ { "from": "0x99CF4c4CAE3bA61754Abd22A8de7e8c7ba3C196d",
  "topic":
  "0xc753993f97543a862693abbfcdf478b7cb142a4b0fae2c0fa14e5567c7a
  1ee60", "event": "DoctorplateletsPrescription", "args": { "0":
  "0x617F2E2fD72FD9D5503197092aC168c91465E7f2", "1": "2", "2":
  "1998671", "3": "100", "doctor":
  "0x617F2E2fD72FD9D5503197092aC168c91465E7f2",
  "BloodcomponentType": "2", "PatientID": "1998671", "Amount":
  "100" } } ]
```

FIGURE 9. Successful execution of BloodUnitPrescription function.

```
[ { "from": "0x99CF4c4CAE3bA61754Abd22A8de7e8c7ba3C196d",
  "topic":
  "0x1ff7c306f212f697e31341698a2df391f811006c35be81b9a08307a2793
  02d25", "event": "consumptionend", "args": { "0":
  "0x17F6AD8Ef982297579C203069C1DbfFE4348c372", "1": "1998671",
  "2": "1052021", "3": "2330", "nures":
  "0x17F6AD8Ef982297579C203069C1DbfFE4348c372", "PatientID":
  "1998671", "Date": "1052021", "Time": "2330" } } ]
```

FIGURE 10. Successful execution of BloodUnitTransfusion function.

discuss and analyze the costs involved in the solution implementation and execution; however, because our solution is built on a private Ethereum blockchain, the gas price is set to zero; therefore, no costs are involved. We present the security analysis of the proposed approach. Finally, we conduct a comparison with the existing blockchain-based solutions.

A. GENERALIZATION

Our proposed blockchain-based solution demonstrates how the blockchain technology can be leveraged to trace the donated blood units during the delivery and distribution phase. Although the functions in the smart contract code were established for the donated blood delivery system, they can be employed in a way to fit other types of supply chain systems. The donated blood supply chain differs from others in terms of the type of product itself. The main difference, in this

case, is the temperature that must be sustained to maintain the product quality. Some items, such as pharmaceutical drugs require certain humidity and temperature levels as well as special storing conditions to be taken into consideration during the delivery and distribution phase. Besides, since this study focuses on traceability, tracing the origin of products like blood, pharmaceutical drugs, vaccines, and more will be similar, as it simply requires scanning the unique identification code attached to the product.

Figure 2 can be used as a reference to discuss the generalized application of the proposed solution in a different supply chain. The figure further elucidates the need for a few modifications, so it can be adopted in other supply chains, such as actors and registrations. Therefore, to accommodate the additional actors for the other proposed applications, the registration functions in the smart contract would require few alterations. Moreover, if the supply chain does not require storing and accessing large data files, then off-chain storage will not be needed. However, some applications might require payments and fund transfer; hence, a special setup in the code will be needed, and an on-chain storage would be more suitable to retain the transaction logs among stakeholders. Finally, simple adjustments can be made to the algorithms provided in this paper to further customize them to fit the functions needed in the new desired applications.

B. SECURITY ANALYSIS

- **Integrity:** The proposed private Ethereum blockchain-based solution provides tracing of donated blood units within the healthcare supply chain. The accessibility of donated blood-related information is implemented by recording all transactions on the ledger. Besides, the use of QR code to provide each blood bag with its distinct tag ensures the integrity of the data.
- **Accountability:** The Modifier features of the Ethereum smart contract assign to each participant a specific role. If any mistakes occur during transporting the donated blood units, the assigned and authorized distributor will be accountable. Hence, all participants are held accountable for their actions.
- **Authorization:** In the healthcare supply chain, the protection of the information on the ledger against forgery is of high importance. As it's a known feature of the blockchain that transactions once deployed cannot be modified or deleted, this results in having all transactions and activities recorded. Hence, with the implementation of Modifiers, as explained before, only authorized members are given accessibility to critical functions with limitations based on their role in the supply chain.
- **Availability:** The private Ethereum blockchain is known for its decentralized and distributed structure. All transaction data that are accessed and recorded by all participants cannot be lost even when one node fails, as they are stored at each participant node. This feature ensures the constant activity of the blockchain for critical industries such as healthcare.

```

INFO:root:contract remote_contract.sol:bloodunitproduction:
INFO:symExec: ===== Results =====
INFO:symExec: EVM Code Coverage: 99.1%
INFO:symExec: Integer Underflow: False
INFO:symExec: Integer Overflow: False
INFO:symExec: Parity Multisig Bug 2: False
INFO:symExec: Callstack Depth Attack Vulnerability: False
INFO:symExec: Transaction-Ordering Dependence (TOD): False
INFO:symExec: Timestamp Dependency: False
INFO:symExec: Re-Entrancy Vulnerability: False
INFO:symExec: ===== Analysis Completed =====

```

FIGURE 11. The production smart contract vulnerability analysis.

- **MITM Attacks:** All transactions in the ledger require the sender's signature by its private key. This will ensure that the blockchain cannot be tampered with by intruders or attackers; therefore, man-in-the-middle (MITM) attacks are impossible. This feature is substantial for improving the blockchain-based blood donation management and the overall healthcare supply chain. Since trusted entities only are provided with information and can perform the actions, this eliminates the possibility of counterfeit.

C. SECURITY ANALYSIS OF THE SMART CONTRACTS

Herein, we perform the security analysis of the developed smart contracts. Although Remix IDE provides code debugging as well as run-time error warnings, it is not sufficient enough to secure the smart contract. Hence, to test and validate the developed Ethereum smart contracts against vulnerabilities, a specialized tool named Oyente was used to improve the reliability of the smart contract. Oyente is a powerful tool which was designed to protect Ethereum smart contracts from any attacks. It establishes trust in the deployed smart contract robustness by making the code free of risks. Oyente runs on Linux and deeply analyzes the solidity code to check if it is susceptible to any exploitation and Cyber attacks.

Figure 11 presents the security results generated by Oyente on the *bloodunitproduction* smart contract and Figure 12 corresponds to the security analysis on the *bloodunitconsumption* smart contract. A high-level programming language named Solidity is used to code the smart contracts. It is compiled into Ethereum Virtual Machine (EVM) code, which is a low-level stack-based bytecode language [29]. The EVM code coverage in the security analysis was found to be 99.1% for the smart contract. The Ethereum Virtual Code coverage percentage represents the number of opcodes executed divided by the total number of opcodes. The Oyente tool showed "False" for all types of possible vulnerabilities to the developed smart contracts' code.

D. COMPARISON WITH THE EXISTING BLOCKCHAIN AND NON-BLOCKCHAIN BASED SOLUTIONS

Table 2 compares our proposed solution with the existing non-blockchain-based solutions based on certain parameters [23], [26], [27]. The most critical parameter is decentralization which requires multiple machines to store identical data simultaneously to ensure integrity. The com-

```

INFO:root:contract remote_contract.sol:bloodunitconsumption:
INFO:symExec: ===== Results =====
INFO:symExec: EVM Code Coverage: 99.1%
INFO:symExec: Integer Underflow: False
INFO:symExec: Integer Overflow: False
INFO:symExec: Parity Multisig Bug 2: False
INFO:symExec: Callstack Depth Attack Vulnerability: False
INFO:symExec: Transaction-Ordering Dependence (TOD): False
INFO:symExec: Timestamp Dependency: False
INFO:symExec: Re-Entrancy Vulnerability: False
INFO:symExec: ===== Analysis Completed =====

```

FIGURE 12. The consumption smart contract vulnerability analysis.

TABLE 2. Comparison of the proposed solution with the existing non-blockchain based solutions.

Features	Davis et al. [23]	Fawaz Alharbi [27]	Dimitri da Silva et al. [26]	Our Solution
Decentralization	✗	✗	✗	✓
Security	✗	✗	✗	✓
Integrity	✗	✗	✗	✓
Traceability	✓	✗	✓	✓
Accountability	✗	✗	✗	✓

TABLE 3. Comparison of the proposed solution with some of the existing blockchain-based solutions.

Features	Kim et al. [5]	Lakshmi narayanan et al. [11]	Toyoda et al. [28]	Our Solution
Blockchain Platform	Hyperledger	Hyperledger	Ethereum	Ethereum
Program	Chaincode	Chaincode	Smart Contract	Smart Contract
Mode of Operation	Private	Private	Public	Private
Currency	✗	✗	✓	✓
Traceability	✓	✓	✓	✓
Off-Chain Storage	✗	✗	✓	✓

parison shows that our solution is the only one that works in a decentralized manner to provide resilience against the single point of failure problem. Furthermore, security is another critical parameter that is used to distinguish our solution from other solutions. The centralized management of other solutions makes them vulnerable to attacks because it is easier to attack a single point than to attack multiple machines. Moreover, the solution proposed in [27] is the solution that does not offer the traceability feature which is an essential component for success in such a complex system. Finally, our solution ensures accountability as each action is stored permanently on an immutable ledger; whereas, other solutions do not capture the actions of all the entities, and thus they do not ensure accountability.

Table 3 compares our proposed solution with the existing blockchain based solutions based on important parameters, such as blockchain platform, program, mode of operation, currency, traceability, and data storage. Our proposed solution uses an Ethereum network in a private mode of operation;

whereas, the solution proposed in [28] uses the Ethereum network but in a public mode of operation. In contrast, the solutions proposed in [5] and [11] employed Hyperledger Fabric platform. The infrastructure cost for an Ethereum network is relatively low compared to a Hyperledger Fabric network. Additionally, Hyperledger Fabric has shown limited use cases compared to our proposed Ethereum blockchain-based solution. In general, the Ethereum programming language is much easier compared to the Hyperledger Fabric. The Chaincode is used in [5] and [11]; whereas, the smart contract is used in our solution. Moreover, the use of PoW as a consensus algorithm requires the Ethereum blockchain-based solutions to use the native cryptocurrency of Ethereum, named Ether, as an incentive for the miners. In contrast, Hyperledger Fabric uses different consensus algorithms that do not involve fee payment to the validating nodes; therefore, it has no native currency. The solutions proposed in [5] and [11] have no off-chain storage option, and storing large amounts of data on-chain can be quite costly; unlike them, our solution utilizes off-chain storage to store large-sized content.

VIII. CONCLUSION

In this paper, we have proposed a blockchain-based blood donation management system that traces the origin of the blood in a transparent, private, secure, trustworthy, auditable, and decentralized manner. The proposed solution employed the smart contract feature of the private Ethereum blockchain to record and log events automatically. We integrated the private Ethereum blockchain with the IPFS to deal with the limited storage issue. We tested and validated the functionality of our solution using the Remix IDE. Our developed smart contracts' code has been made available on the Github repository. We conducted the security analysis to show that the proposed blood donation management solution is robust and secure enough against major security vulnerabilities and attacks. In addition, we compared our proposed approach with the existing solutions. In the future, we aim to deploy and test our solution on the real Ethereum network and build an end-to-end DApp. Furthermore, violation monitoring will be added to further enhance the security of the blood cold supply chain.

REFERENCES

- [1] *Why Blood Donation Are So Important*. Accessed: Jan. 20, 2021. [Online]. Available: <https://www.orlandohealth.com/content-hub/why-blood-donations-are-so-important>
- [2] *FeelGood: A Blood Donation System Based on Smart Contracts*. Accessed: Mar. 20, 2021. [Online]. Available: <https://www.northeastern.edu/rise/presentations/feelgood-a-blood-donation-system-based-on-smart-contracts/>
- [3] *Dubai Health Authority*. Accessed: Apr. 20, 2021. [Online]. Available: <https://www.dha.gov.ae/en/SpecialtyCentres/DubaiBloodDonationCenter/Pages/FAQ.aspx>
- [4] Sensire. *Blood Cold Chain—The Challenges of Saving Lives*. Accessed: Apr. 24, 2021. [Online]. Available: <https://www.sensire.com/blog/blood-cold-chain-the-challenges-of-saving-lives>
- [5] S. Kim, J. Kim, and D. Kim, "Implementation of a blood cold chain system using blockchain technology," *Appl. Sci.*, vol. 10, no. 9, p. 3330, May 2020, doi: 10.3390/app10093330.
- [6] T. Hannon, "Waste not, want not," *Amer. J. Clin. Pathol.*, vol. 143, no. 3, pp. 318–319, Mar. 2015, doi: 10.1309/ajcpm8facvc0prg.
- [7] *Design Guidelines Centres—Who*. Accessed: Mar. 18, 2021. [Online]. Available: <https://tinyurl.com/mekrcr49e>
- [8] *How Are Blood Component Expiration Dates Determined?* Accessed: May 16, 2021. [Online]. Available: https://stanfordbloodcenter.org/how_are_blood_c/
- [9] *What Happens to Donated Blood*. Accessed: May 23, 2021. [Online]. Available: <https://www.redcrossblood.org/donate-blood/blood-donation-process/what-happens-to-donated-blood.html>
- [10] MED Alliance Group. (Aug. 4, 2020). *The Blood Supply Chain—Learning About the Points of Failure*. Accessed: Feb. 12, 2021. [Online]. Available: <https://www.medalliancegroup.com/the-blood-supply-chain-learning-about-the-points-of-failure/>
- [11] S. Lakshminarayanan, P. N. Kumar, and N. M. Dhanya, *Implementation of Blockchain-Based Blood Donation Framework* (IFIP Advances in Information and Communication Technology), vol. 578. Cham, Switzerland: Springer, 2020, pp. 276–290, doi: 10.1007/978-3-030-63467-4_22.
- [12] A. Farrugia, "Safety issues of plasma-derived products for treatment of inherited bleeding disorders," *Seminars Thrombosis Hemostasis*, vol. 42, no. 5, pp. 583–588, Jul. 2016.
- [13] S. Sadri, A. Shahzad, and K. Zhang, "Blockchain traceability in healthcare: Blood donation supply chain," in *Proc. 23rd Int. Conf. Adv. Commun. Technol. (ICACT)*, Feb. 2021, pp. 119–126, doi: 10.23919/icaict51234.2021.9370704.
- [14] U.S. Food and Drug Administration. *Drug Supply Chain Security Act*. Accessed: Jul. 16, 2021. [Online]. Available: <https://www.fda.gov/drugs/drug-supply-chain-integrity/drug-supply-chain-security-act-dscsa>
- [15] A. Musamih, K. Salah, R. Jayaraman, J. Arshad, M. Debe, Y. Al-Hammadi, and S. Ellahham, "A blockchain-based approach for drug traceability in healthcare supply chain," *IEEE Access*, vol. 9, pp. 9728–9743, 2021, doi: 10.1109/access.2021.3049920.
- [16] U. Hacıoglu, *Digital Business Strategies in Blockchain Ecosystems: Transformational Design and Future of Global Business*. Cham, Switzerland: Springer, 2020, p. 74.
- [17] T. T. A. Dinh, J. Wang, G. Chen, R. Liu, B. C. Ooi, and K.-L. Tan, "BLOCKBENCH: A framework for analyzing private blockchains," in *Proc. ACM Int. Conf. Manage. Data*, May 2017, pp. 1085–1100.
- [18] Z. Li, J. Kang, R. Yu, D. Ye, Q. Deng, and Y. Zhang, "Consortium blockchain for secure energy trading in industrial Internet of Things," *IEEE Trans. Ind. Informat.*, vol. 14, no. 8, pp. 3690–3700, Aug. 2018.
- [19] S. Kim and D. Kim, "Design of an innovative blood cold chain management system using blockchain technologies," *ICIC Exp. Lett. B, Appl.*, vol. 9, pp. 1067–1073, Oct. 2018, doi: 10.24507/iciclb.09.10.1067.
- [20] *How Blockchain is Helping Make Every Blood Donation More Effective?* Accessed: Feb. 22, 2021. [Online]. Available: https://www.ey.com/en_ae/better-begins-with-you/how-blockchain-could-ensure-every-drop-of-blood-is-tracked-and-every-outcome-is-measured
- [21] K. E. Kendall, "A decentralized information and control system for blood management," *J. Syst. Softw.*, vol. 1, pp. 299–306, Jan. 1979, doi: 10.1016/0164-1212(79)90031-1.
- [22] X. Zhu, S. K. Mukhopadhyay, and H. Kurata, "A review of RFID technology and its managerial applications in different industries," *J. Eng. Technol. Manage.*, vol. 29, no. 1, pp. 152–167, 2012, doi: 10.1016/j.jengtecman.2011.09.011.
- [23] R. Davis, B. Geiger, A. Gutierrez, J. Heaser, and D. Veeramani, "Tracking blood products in blood centres using radio frequency identification: A comprehensive assessment," *Vox Sanguinis*, vol. 97, no. 1, pp. 50–60, Jul. 2009, doi: 10.1111/j.1423-0410.2009.01174.x.
- [24] *B Medical Systems*. Accessed: Jul. 8, 2021. [Online]. Available: <https://www.bmedicalsystems.com/en/wp-content/uploads/sites/2/2020/12/BMS-Brochure-Blood-Management-Solutions-EN.pdf>
- [25] *Problems With RFID*. Accessed: Feb. 6, 2021. [Online]. Available: <http://www.technovelgy.com/ct/Technology-Article.asp?ArtNum=20>
- [26] D. da Silva, J. Costa, B. Assuncao, V. Kuprych, and C. Teixeira, "Microservice-based middleware for collaborative supply chain tracing," in *Proc. 15th Iberian Conf. Inf. Syst. Technol. (CISTI)*, Jun. 2020, pp. 1–6, doi: 10.23919/CISTI49556.2020.9141036.
- [27] F. Alharbi, "Progression towards an e-management centralized blood donation system in Saudi Arabia," in *Proc. Int. Conf. Adv. Emerg. Comput. Technol. (AECT)*, Feb. 2020, pp. 1–5, doi: 10.1109/AECT47998.2020.9194178.

- [28] K. Toyod, P. T. Mathiopoulou, I. Sasase, and T. Ohtsuk, "A novel blockchain-based product ownership management system (POMS) for anti-counterfeits in the post supply chain," *IEEE Access*, vol. 5, pp. 17465–17477, 2017, doi: [10.1109/access.2017.2720760](https://doi.org/10.1109/access.2017.2720760).
- [29] L. Luu, D.-H. Chu, H. Olickel, P. Saxena, and A. Hobor, "Making smart contracts smarter," in *Proc. ACM SIGSAC Conf. Comput. Commun. Secur.*, Oct. 2016, pp. 254–269, doi: [10.1145/2976749.2978309](https://doi.org/10.1145/2976749.2978309).
- [30] M. Eskandari-Khanghahi, R. Tavakkoli-Moghaddam, A. A. Taleizadeh, and S. H. Amin, "Designing and optimizing a sustainable supply chain network for a blood platelet bank under uncertainty," *Eng. Appl. Artif. Intell.*, vol. 71, pp. 236–250, May 2018.
- [31] N. Armaghan and M. Y. Pazani, "A model for designing a blood supply chain network to earthquake disasters (case study: Tehran city)," *Int. J. Quality Res.*, vol. 13, no. 3, pp. 605–624, Sep. 2019.
- [32] A. Jabbarzadeh, B. Fahimnia, and S. Seuring, "Dynamic supply chain network design for the supply of blood in disasters: A robust model with real world application," *Transp. Res. E, Logistics Transp. Rev.*, vol. 70, pp. 225–244, Oct. 2014.
- [33] T. Peltoniemi and J. Ihalainen, "Evaluating blockchain for the governance of the plasma derivatives supply chain: How distributed ledger technology can mitigate plasma supply chain risks," *Blockchain Healthcare Today*, 2019.
- [34] M. Çağliyangil, S. Erdem, and G. Özdağoglu, "A blockchain based framework for blood distribution," in *Digital Business Strategies in Blockchain Ecosystems*, U. Hacıoglu, Ed. Cham, Switzerland: Springer, 2020, pp. 63–82.



DIANA HAWASHIN received the B.S. degree in electrical engineering from United Arab Emirates University, United Arab Emirates, in 2020. She is currently pursuing the M.S. degree in engineering systems and management with Khalifa University. She is also a Research and a Teaching Assistant with the Department of Industrial Engineering, Khalifa University. Her research interests include blockchain, healthcare, and supply chains.



DUNIA AMIN J. MAHBOOBEH received the B.Sc. degree in electrical engineering and computer science from the Khalifa University of Science and Technology, United Arab Emirates, in 2018, where she is currently pursuing the M.Sc. degree in artificial intelligence, electrical engineering, and computer science. She is also a Research and a Teaching Assistant with the Department of Electrical Engineering and Computer Science, Khalifa University of Science and Technology. Her research interests include machine learning, healthcare, and web development (Python, Django).



KHALED SALAH (Senior Member, IEEE) received the B.S. degree in computer engineering with a minor in computer science from Iowa State University, USA, in 1990, and the M.S. degree in computer systems engineering and the Ph.D. degree in computer science from the Illinois Institute of Technology, USA, in 1994 and 2000, respectively. He is currently a Full Professor with the Department of Electrical and Computer Engineering, Khalifa University, United Arab Emirates.

He has over 220 publications and three U.S. patents, and he has been giving a number of international keynote speeches, invited talks, tutorials, and research seminars on the subjects of blockchain, the IoT, fog and cloud computing, and cybersecurity. He served as the Chair of the Track Chair of IEEE GLOBECOM 2018 on cloud computing. He is an Associate Editor of IEEE BLOCKCHAIN TECHNICAL BRIEFS and a member of IEEE Blockchain Education Committee. He is currently leading a number of projects on how to leverage blockchain for Healthcare, 5G networks, combating deepfake videos, supply chain management, and AI.



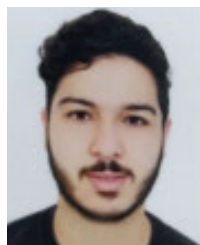
RAJA JAYARAMAN received the bachelor's and master's degrees in mathematics from India, the M.Sc. degree in industrial engineering from New Mexico State University, and the Ph.D. degree in industrial engineering from Texas Tech University. He is currently an Associate Professor with the Department of Industrial and Systems Engineering, Khalifa University, Abu Dhabi, United Arab Emirates. His expertise is in multi-criteria optimization techniques applied to diverse applications, including supply chain and logistics, healthcare, energy, environment, and sustainability.

His research interests include blockchain technology, systems engineering, and process optimization techniques to characterize, model, and analyze complex systems with applications to supply chains, maintenance operations planning, and healthcare delivery. His postdoctoral research was centered on technology adoption and implementation of innovative practices in the healthcare supply chains and service delivery. He has led several successful research projects and pilot implementations in the area of supply chain data standards adoption in the U.S. healthcare system. His research has appeared in top-rated journals, including *Annals of Operations Research*, *IIE Transactions*, *Energy Policy*, *Applied Energy*, *Knowledge-Based Systems*, *IEEE Access*, *Journal of Theoretical Biology*, and *Engineering Management Journal*.



IBRAR YAQOOB (Senior Member, IEEE) received the Ph.D. degree in computer science from the University of Malaya, Malaysia, in 2017. He is currently working with the Department of Electrical Engineering and Computer Science, Khalifa University, United Arab Emirates. Previously, he worked as a Research Professor with the Department of Computer Science and Engineering, Kyung Hee University, South Korea, where he completed his postdoctoral fellowship. He worked

as a Researcher and a Developer with the Centre for Mobile Cloud Computing Research (C4MCCR), University of Malaya. His research interests include blockchain, edge computing, mobile cloud computing, big data, the Internet of Things, and computer networks. His numerous research articles are very famous and among the most downloaded in top journals. He has been listed among top researchers by Thomson Reuters (Web of Science) based on the number of citations. He is currently serving/served as a guest/associate editor in various journals. He has been involved in a number of conferences and workshops in various capacities.



blockchain technology, the Internet of Things (IoT), fog computing, and supply chain applications.



MAZIN DEBE received the B.Sc. degree in computer engineering and the M.Sc. degree in electrical and computer engineering from the Khalifa University of Science and Technology, Abu Dhabi, United Arab Emirates. He is currently working with the Center for Cyber-Physical Systems, Khalifa University of Science and Technology, as a Research Associate. He has published multiple research articles in highly ranked IEEE conferences and journals. His research interests include

SAMER ELLAHHAM received the bachelor's degree in biology and the M.D. degree from The American University of Beirut, Beirut, Lebanon.

He is currently a Cleveland Clinic Caregiver, Cleveland, U.S. seconded as Senior Cardiovascular Consultant, and the Director of Accreditation in the Quality and Safety Institute, Cleveland Clinic Abu Dhabi. He is the Middle East Regional Chair, an ISQua Expert of Patient Safety Movement Foundation, a member of Aha Hospital

Accreditation Science Committee, a member of the European Society of Cardiology Heart Failure Writing Group, an ex-Middle East Representative of the JCI Standards Subcommittee, and a member of the American College of Cardiology Accreditation Foundation Board. He finished his internal medicine residency at Georgetown University Hospital—Washington Hospital Center and his fellowship in cardiology at the Virginia Commonwealth University Health System in the USA. He worked in Washington DC at Georgetown University Hospital—Washington Hospital Center and in several clinical and leadership positions before moving to United Arab Emirates, in 2008. He continues to be an active clinician. He demonstrates great skill and experience in the management of patients with heart failure, ischemic heart disease, and valvular heart disease and led a multi-disciplinary team in the care and delivery of advanced therapies to these patients. He has unique abilities to partner and engages local and regional referring providers. He can work in a highly matrixed environment, possess strong leadership and organizational skills, and have the experience of working effectively in a large health system. He led the First AHA GWTG Heart Failure Initiative outside the USA and was the recipient of the AHA GWTG Award in Washington DC. He is the champion of the AHA GWTG in the region. He has served as the Chief Quality Officer for SKMC, from 2009 to 2017. In his role, he has led the development of a quality and program that has been successful and visible and has been recognized internationally by several awards. As a Chief Quality Officer and Global Healthcare Leader, he had a focus on ensuring that the implementation of these best practices leads to breakthrough improvements in clinical quality, patient safety, patient experience, and risk management. He was the Executive SKMC sponsor of the American College of Surgeons National Surgical Quality Improvement Program (ACS NSQIP®) the leading U.S. validated, risk-adjusted, outcomes-based program to measure and improve the quality of surgical care. SKMC is the first multispecialty ACS NSQIP center outside the USA. He led the publication of, first in the region, annual SKMC outcome books, since 2011, and he is a strong believer in transparency in health care and external reporting. He was the Leader of the First Pilot International Robust Process Improvement (RPI) project by the Joint Commission Center for Transforming Healthcare and several other similar successful performance improvement projects at SKMC. He is American Board Certified in Internal Medicine, Cardiovascular Disease, Vascular Medicine, and American Board of Medical Quality. He was recently recertified in 2017 by the American Board of Cardiology (ABIM). He is a Certified Professional in Healthcare

Quality (CPHQ) by The National Association for Healthcare Quality (NAHQ), certified in Medical Quality (CMQ) by The American Board of Medical Quality (ABMQ), certified as EFQM Model Assessor and a Lead Trainer of TeamSTEPPS.

Dr. Ellahham is a fellow of the American College of Cardiology, American Heart Association, American College of Chest Physicians, American College of Physicians, and American College of Medical Quality. He is a fellow of the American College of Cardiology and a Key Member of Heart Failure and Transplant, Adult Congenital and Pediatric Cardiology, Cardio-Oncology, Innovation, Quality, and Peripheral Vascular Disease Sections. He is a Distinguished Fellow of the New Westminster College, BC, Canada, and an Advisory Board Member of the University of Wollongong in Dubai. He was the Middle East Representative of the JCI Standards Subcommittee and a member of the Editorial Advisory Board of the *Joint Commission Journal on Quality and Patient Safety*. He was a Reviewer of HCAC Cardiac Quality and Safety Standards. He has been a Champion and a Leader of the use of Lean, Six Sigma, and Change Management to improve healthcare quality and has numerous publications in this area. He is Lean Six Sigma Master Black Belt Certified. He is an American Society of Quality (ASQ) trainer in Lean and Six Sigma both green and black belt. He is also an ISQua Expert. He is a recognized innovative leader in quality, safety, patient experience, artificial intelligence, blockchain, telehealth, clinical cardiology, and the use of robust performance improvement in improving healthcare delivery. He serves on several U.S. and international prestigious committees and advisory bodies. He is the Middle East Regional Chair of the Patient Safety Movement Foundation. He received several research awards, including the DuPont Pharmaceuticals Research Award from ACCP 58th Annual Scientific Assembly, the Young Investigator Award, the Alfred Soffer Research Award from ACCP 58th Annual Scientific Assembly, a Finalist of the First Young Investigator Award 12th Annual Meeting of the Mediterranean Association of Cardiology and Cardiac Surgery, the American Heart Association Get with the Guidelines Award, the SKMC Infection Prevention Award, in 2011 and 2012, the Sheikh Khalifa Excellence Award, in 2014, the Quality Leadership Award from the World Quality Congress and Awards, the Business Leadership Excellence Award from World Leadership Congress, in 2015, one the nominees for Safe Care magazine Person of the Year in the United States, the Dubai Quality Award, in 2015, and the Sheikh Khalifa Excellence Golden Award, in 2015. He is an avid researcher; his research interests include heart failure, acute coronary syndromes, frailty, dyslipidemia, accreditation, second victim phenomenon, resilience, innovation, artificial intelligence, telehealth, blockchain, patient flow, patient experience and engagement, lean-six sigma, patient safety, bowtie risk management tool, and KPI management. He is a recognized world-leader in these fields. He is the Eminent Editor of the *Journal of Cardiology & Cardiovascular Therapy* and an Associate Editor of the *American Journal of Medical Quality*. In addition, he serves on the Editorial Board of *Journal of Thoracic Disease and Cardiothoracic Surgery*, *Developments in Clinical & Medical Pathology* (DCMP), *The Joint Commission Journal on Quality and Patient Safety*, *Telehealth and Medicine Today* (TMT), *Blockchain in Healthcare Today* (BHTY), *Medical Science*, *Cardiovascular Research—An Open Journal*, *UPI Journal of Pharmaceutical, Medical and Health Sciences* (UPI-JPMHS), *Open Access Research in Anatomy, Gerontology & Geriatrics Studies*, *Open Access Journal of Clinical Trials*, *Hypertension Today Journal*, *Focus on Hypertension Journal*, *Journal of Heart Health*, *Cardiovascular Pharmacology*, *Scientific Research and Community*, *International Journal of Surgery & Surgical Procedures*, *EC Cardiology*, *Journal of Cardiovascular Medicine*, *Journal of Pulmonary Medicine*, and *Canadian Journal of Biomedical Research*. He is also a Reviewer for several peer-reviewed journals, including *The Joint Commission Journal on Quality and Patient Safety*, *International Journal of Quality & Reliability Management*, *Journal of American College of Cardiology*, *American Heart Journal*, *Annals of Internal Medicine*, *Archives of Internal Medicine*, *Chest*, *Circulation*, *Clinical Cardiology*, *The Lancet*, *Diabetes Care*, *Endocrinology and Metabolism*, *European Journal of Heart Failure*, *Congestive Heart Failure*, *Journal of Nuclear Cardiology*, *Journal of Transplant Coordination*, *Journal of Cardiovascular Pharmacology*, *Southern Medical Journal*, *European Journal of Innovation Management*, *The Anatolian Journal of Cardiology*, and *npj Digital Medicine*.

...