Egyptian Informatics Journal 23 (2022) 187–196

Contents lists available at ScienceDirect

Egyptian Informatics Journal

journal homepage: www.sciencedirect.com

Full length article

A blockchain-based models for student information systems

Sura I. Mohammed Ali [a], Haitham Farouk [b], Hussien Sharaf [c]

[a] Department of Mathematics and Computer Application, Collage of Science, Al-Muthanna University, Iraq
[b] Department of Computer Science, Faculty of Computers and Information, Suez University, Suez, Egypt
[c] Department of Computer Science, Faculty of Computers and Information, Suez University, Suez, Egypt

ARTICLE INFO

Article history:
Received 24 May 2021
Revised 30 October 2021
Accepted 1 December 2021
Available online 23 December 2021

Keywords:
Blockchain
Student information system (SIS)
Decentralized ledger
Transactions
Permissioned and permissioned-less consensus network
Stateful and stateless data

ABSTRACT

Blockchain stores a series of transactions in form of a sequence of linked blocks. Hence, the concept of a single decentralized ledger is easily maintained. Transactions and interactions that take place among the participants accessing the distributed and decentralized but cooperative blockchain network are held through a single ledger. A student information system (SIS) can make use of a decentralized, reliable, and highly trusted ledger that stores vital information. Traditional education systems encounter problems such as centralized record keeping where fault tolerance depends on a single cloud provider; not to mention locally hosted databases. The implementation of blockchain in the education sector provides a new horizon for set of non-functional requirements including but not limited to: security, immutability, independence from the institution, immutability of official records and certificates. In addition, total trust in the accuracy and infallibility are all gathered in the decentralized ledgers of blockchain. The proposed models emphasize on the data availability; represented in students' ability to access all of their data at any time. This paper proposes three models for using blockchains to implement fully functional SIS that maintains transactions such as students' and faculty members' records, course registration records and student marks. In addition, avoiding the role of a super administrator or a centralized exposed store where data integrity is vulnerable. Using the proposed models pushes towards an electronic community where genuine certificates can be easily issued and published to the interested parties without the need for involving a centralized administration.

1. Introduction

Blockchain introduces a new horizon for achieving data integrity using decentralized and well-maintained data stores. Blockchain can be implemented within the automated management systems of individual higher education institutions or groups of educational institution [1].

Education is a core area where different stakeholders need to share and modify shared information. Modification of records can take place in different levels of security levels. The motivation of using blockchain in SIS comes from the growing need for acquiring high security and trust in such critical systems. In addition, a blockchain system emphasizes on making use of a decentralized, reliable, and highly trusted ledger that stores vital information.

Regardless of the marked enchantment in technology, there is a growing need for reliability and data integrity. Students would have full independence over their personal data in Blockchain. It will provide students with complete independence from the institution as well as complete control over their data while keeping integrity of immutable data. The blockchain stores data that is permanently recorded and encrypted using cryptographic technologies into decentralized blocks. Cryptographic procedures used in block generation and connecting blocks improve the security of each blockchain transaction, and the data recorded on the blockchain are immutable records whose states cannot be changed once they are generated. Security, resilience, and irreversibility are all connected to immutability. The blockchain's promise in education extends much farther, allowing students to add flavor of anonymity over their personal data, independence from the institution, immutability of records of official papers and certificates, and total

E-mail addresses: suraibraheem@mu.edu.iq (Sura I. Mohammed Ali), H.Farouk@Suezuni.edu.eg (H. Farouk), H.Sharaf@suezuni.edu.eg (H. Sharaf)

Peer review under responsibility of Faculty of Computers and Information, Cairo University.

Production and hosting by Elsevier

https://doi.org/10.1016/j.eij.2021.12.002

trust in the accuracy and infallibility owing to network design. The proposed models emphasize on the data availability; represented in student's ability to access all their data at any time. Blockchain technology enables an alternative approach to teaching. Many institutions, organizations, and businesses are establishing their own blockchain efforts to investigate the advantages and uses in education [2–5].

The motivation for this paper arises from the fact that; in education sector, students' information is critical and sensitive. The existence of a super administrator who can retrieve data of general administrative framework, learning and research may be seen as a big vulnerability [6]. In conventional education framework relying on a centralized system, there are some challenges regarding record keeping. The goal of the new this research is to find other alternatives for record keeping. Based on blockchain technology; the new model can provide a more protected and trusted archive of records which represents depending on time-stamps can be a big assist for all stakeholders [7].

This paper proposes an approach for using blockchain to implement fully functional SIS that maintains students' records, course registrations record and student marks. It provides significant, safe, and transparent methods for building a global system for educational learning.

2. Blockchain technology

Blockchain technology is one of the famous popular techniques in the latest few years [8]. One of the most important applications is securing data of student information systems (SIS). The features of blockchain are readiness of sharing and visibility that are essential for any SIS. The courses registrations and exam marks can be viewed as transactions like financial transitions; once committed to the system must never be removed. A transaction can be only reversed only by submitting another reverse transaction. The information hashed in the blockchains; when carried the change in content of one block will invalidate the entire chain of blocks [9]. Each block contains a list of transactions, a hash of cryptography for the transaction list, version, in addition to other values necessary for blockchain such as the hash value of the previous block. This implies that the hash value is used to evaluate the relation between two blocks.

The details will be calculated by checking the hash value in the context of blockchains, data is stored in the form sequence of blocks. Each block is linked to the next block of records and store all committed transactions using a public ledger [7]. Transactions and interactions that take place among the participants accessing the distributed and decentralized blockchain network are holding through the ledger.

The chain expands continuously as new blocks are appended to it. Each block includes a message, current hash, previous block hash, date; block id, etc. relying on the application [1]. The first block in the blockchain is called genesis block which has no parent, is shown in Fig. 1.

All regular transactions rely on the centralized consensus, depending on the party. This offers many interesting features such as decreasing transaction cost, efficiency, and security. To attain secure, fast and clean transactions, the idea of a linked list of blockchain's will be introduced. A blockchain system processes the transactions submitted by different users according to their security level without the need for third party intervention services. Different models have been presented in the domain of student information systems (SIS).

In the university context, a lot of efforts and costs are put into managing records for all educational transactions occurring since the first-time students submit their papers and register courses for the first semester until they graduate. In this context, security and data integrity challenges can be identified [10].

The proposed model adds more security via the use of hashing and data readily available with decentralized data storage. The information of blockchain that is provided by the university; is open to all parties that have interest. Immutability, sensitiveness, and deal with of storing the records in blockchain; altogether helped to get good implementation of student information system [11]. A linked list of blockchains can record all values that include birth certificates, social security cards, student loans, etc. [5] in addition to transactional data such as courses registration and exam marks.

2.1. Components of blockchain

A blockchain is a linked list of blocks that contains sets data. Each block has a cryptographic hash in its transaction list and another hash of the previous block. Transactions' hash is stored in a cryptographically secured data structure called a Merkle tree. The following represent the basic terminology for blockchain:

- *Node* — it represents a computer system connected to a blockchain that holds a full copy or a partial copy of the blockchain and it could have several combinations of SIS roles. On another categorization of nodes, a full node can hold the entire list of blockchain while a light-weight node holds a partial list of blocks. The more nodes a system has, the more decentralized it becomes.
- *Blockchain* — it is a distributed network of nodes maintaining a linked list of blocks that represent a decentralized ledger distributed/copied on all nodes according to the node's types. Each full node preserves a full valid copy of the ledger.
- *Block* — it is a list of records, including a set of transactions. It also contains a hash value based on current block state and another hash value based on the previous block state. A Genesis block is the first block in the blockchain, which has no parent. A block includes, but not limited to, the following:

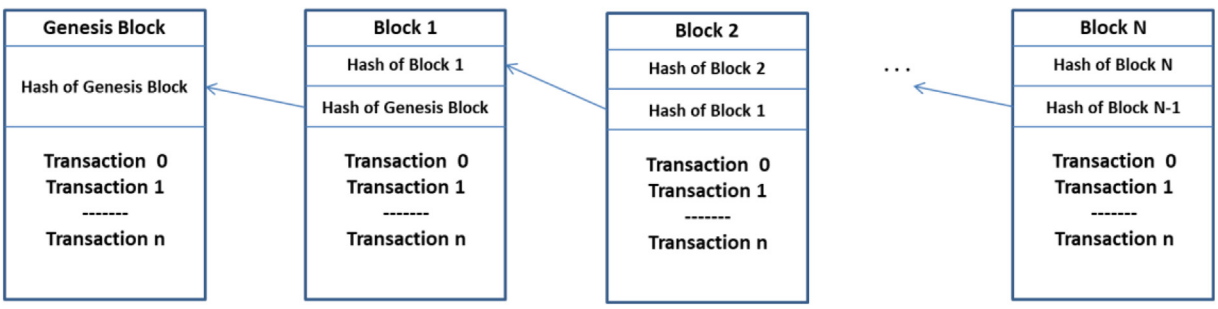

Fig. 1. Blockchain architecture.

a. *Version* — a number used in comparing two copies of a block to determine which one is more recent.
b. *Transaction List* — it is defined according to the domain being handed. In SIS, a transaction is either a set of stateless data about the student/faculty member or a tasteful set of data such as course registration of exam marks.
c. *State* — it is defined according to the domain. In SIS, a state could represent total, count and other aggregate data derived from the transaction list. A block state must change whenever a transaction is added to the block.

- *Merkle Tree* — A binary tree data structure where transaction tokens are added and hashed in pairs, then their result is accumulated then the resulting hash is concatenated together with another result coming from another pair and so on up to the tree root.
- *Consensus* — A protocol for accepting a newly created block using a cooperative set of nodes in a blockchain without the need for a central service. A Consensus mechanism is a vital activity to the functioning of any blockchain of adding a new block to the block chain.

There are several models that emerged even before blockchain came to existence. Some of them [12]: proof-of-work (POW), proof-of-Stake (POS), round-robin, N2N, leader-elected consensus, etc. All of them imposes that several participants are required to accept a transaction before permanently adding it to the blockchain. But they vary in percentage of nodes that should agree.

According to [12]; the various models of consensus could be categorized into:

a. Permissioned: where consensus takes place between nodes known to each other and authorized to read/write transactions, i.e., Federated consensus and N2N.
b. Permission-less: where anonymous nodes can request reading/writing transactions, i.e. (POW) and (POS).

- *Miners* — are nodes that keep trying to create new blocks. Each miner starts to form his block (i.e., a nominated informal block) which involves his selected transactions. Usually, a miner gets rewarded with a coin or a fraction of a coin. In SIS context, a safe choice is to exclude the existence of miners in a SIS blockchain. Miners can be discarded by allowing only known nodes to create blocks.

2.2. Background and related work

In the domain of education, several approaches to using blockchain technology were introduced. This section discusses a variety blockchain models for the university's domain. The authors introduced a case study where the university of Nicosia and Birmingham research center have used Blockcert system. A Blockcert system approves control to issue personal official documents, academic certificates, and private data. The Universities of Nicosia and Woolf primarily based on blockchain architecture, Blockcert system consist of many components that interconnections it of the educational environment.

The domain of universities makes use of hashed certificates in the blockchain for verifying students' certificate. An authenticated employer can verify students' certificate and issue a hard copy based on blockchain integrity.

The authors [13] discussed in their paper that twenty four universities in Jakarta have used existing McRhys model. McRhys is a model to record and integrate all activities in university using blockchain technology.

"Distributed Open Ledger" is a concept that is adopted by McRhys model. Information of blockchain can be accessed by all stakeholders who have interest. Stakeholder roles are identified and configured by the university administration. Any transaction is validated and verified before it is added into the immutable blockchain network. The objective of this model is to solve the concerns of validity, transparency, reliability of data and access of data with ease by the employer when the blockchain ledger is applied.

A Diploma certificate forgery on university is an important concern that attracted researchers' attention. Another challenge in education institutions; is recording marks and grades.

Recorded grades should be unchangeable and saved on a ledger-based system. Once a set of exam marks are recorded (added transaction) into the immutable blockchain ledger, no one can modify it. As a topic within academic domain, the authors in [9] proposed a new framework based on blockchain, records of education and components added to the system, and then posted to the blockchain to publish it for all interested parties.

The center of the framework includes individual full nodes, miners, and provider nodes. The proposed models contribute with provide the blockchain provide the privacy and security of data stored on it, and functions customized access to different parties that corresponding for the education.

According to [14], EduCTX is a global higher education credit platform that has been suggested. This platform is built on the European Credit Transfer and Accumulation System idea (ECTS). It is a worldwide trusted, decentralized higher education credit and grading system that may provide a globally united perspective for students and higher education institutions (HEIs), as well as other prospective stakeholders such as businesses, institutions, and organizations. They demonstrated a prototype implementation of the ecosystem based on the open-source Ark Blockchain Platform as a proof of concept. According to [14] provided insight on the use of Blockchain as a secure, distributed cyberinfrastructure for the future grid. First, the fundamental concepts of Blockchain and its current state-of-the-art technology were introduced. Then, a smart grid cyber-physical infrastructure architecture based on Blockchain is presented. Following that, several prospective Blockchain application fields in future grids are described. Following that, several potential difficulties are mentioned. According to [15], blockchain technology was predicted to transform the way transactions are done, influencing a wide range of possible application sectors. Because blockchain technology is based on a peer-to-peer network that allows diverse parties to collaborate, the service system is chosen as a unit analysis to assess its potential contribution. They discovered a set of qualities that promote trust and decentralization, making the development and coordination of a service system easier.

3. Proposed blockchain-based SIS

3.1. System models

The most essential requirements of an SIS are data integrity and the use immutable ledger in addition to the ability to share protected information securely between the interested parties. Features of blockchain's can meet those requirements using three models. In an SIS, data can be categorized as follows:

A. SIS stateless data: data that have no effect on a student's GPA and are entered once. SIS stateless data are relatively less critical for SIS. Typical SIS stateless data are course list, list of faculty members, list of students. For each list of stateless data, an authorized staff enters a basic set of fields, yet the concerned user might be required to enter more data about himself such as a faculty member or a student modifies his phone or address.

B. SIS stateful data: data that causes a student state to change. For example, a student enrolling/withdrawing a course. A faculty member enters course/exam marks.

Stateless data can still be seen critical, but for some systems it is satisfying to keep them into a traditional system with a super admin who signs confidentiality contract. Transactional data are highly critical in all aspects. A Transaction can be only reversed by submitting another reverse transaction.

In SIS, node types with respect to roles can be categorized into:

A. *Admin node*
   i. Reads/writes students' and faculty members' basic stateless information records.
   ii. Reads/writes curriculum.
   iii. Reads/writes faculty members' assigned course list.
   iv. Reads/writes university rules settings such as deadlines, etc. These rules settings govern the block chain.
   v. Creates a block.

A university can choose to categorize the admin roles into subcategories according to their organization.

B. *Student node*
   i. Reads/writes course registration records according to the university rules and within the deadlines,
   ii. Reads/writes **some** fields of the student's own stateless data such as phone and address according the university rules and within the deadlines.
   iii. Read only a certificate/student's transcript.

C. *Faculty Member (Professor) node*
   i. Reads/writes marks (Final exam, Midterm, course work).
   ii. Reads/writes **some** fields of his own stateless data such as phone and address according to the university rules and within the deadlines.
   iii. Read only a certificate/student's transcript.
   iv. Create a block.

D. *Guest node*
   i. Read only a certificate/student's transcript upon student approval.
   ii. Read only a curriculum upon admin approval.
   iii. Read only faculty members' assigned course list upon faculty member approval.

A guest node represents an external party who has interest in a set of the SIS information.

3.2. Proposed basic design

The models proposed are in form of theoretical prototypes. Each model provides a ledger-based system. Our scope focuses on transactional data being the most critical data.

A transaction life cycle starts in SIS, when a node creates a request to write a transaction. The sender node undergoes the following steps:

A. The sender node creates a transaction token (T) that represents the new transaction. It digitally signs the transaction using its assigned private key. Then send to a full node.

B. The full node authenticates the received transaction using the public key of the sender node and accepts if a valid data is found.

C. The full node having the full list of blocks, tries to add the transaction to the chosen block.

The hash code of a valid block is recalculated as in equation (1) [16].

$$\mathrm{BHC} = \mathrm{Hash}\left(\sum_{0}^{n} T_i + \mathrm{R} + \mathrm{P} + \mathrm{N\ \ once}\right), \tag{1}$$

where 'BHC' is the Block Hash Code, n is the total count of transactions in a block transaction list, 'T_i' is the transaction token of each transaction in the transaction list of a block, R is the Merkle tree root, 'P' is the hash code of the previous block in the chain, 'Nonce' (Number only once) is an arbitrary number that can be used only once for producing the corresponding BHC.

D. The block version is incremented.

The modified block is still unconfirmed until the consensus protocol completes its work. For example, if a transaction is sent by node A, it must be verified by the nearest full node according to the rules of an SIS. Node A will only accept a user with the right security privileges to alter marks according to its role type. Then the nearest full node will check against higher level rules such as the deadline of submitting marks. A verified block will then be published to other node peers to confirm and accept its validity if it accepts the block then the block is propagated to the blockchain.

Each transaction contains information that identifies the creating node and time of creation. Each block has a cryptographic hash of the previous block, and another has the transaction data. In this way, the data record posted on the blockchain cannot be tampered after a consensus is accomplished.

On registration of a new node a key pair is generated using a public encryption key algorithm for privacy considerations. Then the user can start adding transactions using his private key that is accepted by the blockchain. A false node/account can never add transactions as his private key will never match any of the public key stored in the blockchain. The new node can then request to publish transactions into the blockchain network. The private key can also be used as node identifying like an IP on a network.

Step 1: Faculty member or student registration key setup

In blockchain, an authorized user needs to setup his node with a private key to be accepted on a blockchain. Every time an authorized user wants to operate on block data, the registration private key is required to sign the outgoing transaction, this key as authorized to read, add or update data.

Step 2: A transaction can be accepted using the following protocol in the nearest full node

A. Decrypt the incoming transaction request using the public key for the sender node requesting the transaction. If a valid request is received, then continue.
B. Add the transaction to the list of transactions and calculate the new the new Merkle root that expresses all list of transactions in the current block. The transaction is added to the Merkel tree.
C. In an SIS an abstract transaction token could be course code being registered or marks being added.
D. The sender **light-weight node** does the first level verification by accepting only legal requests according to the security level rules embedded into it. For example, a faculty member is not authorized to change a students' registrations.

Step 3: A transaction can be added to a block nearest full node using the following protocol

Any node uses its private key to encrypt the transaction token, then the receiving full node decrypts it before calculating the hash and adding the hashed token into the Merkle tree. The plain text token gets added to the transaction list together with the sender node address, while the hashed token gets added to the Merkle tree. The procedure of adding a transaction token to the

Merkel tree involves concatenating a pair of transaction tokens, hashing them, then concatenating the resulting hash together with another result coming from another pair and so on until one hashed token is written in the Merkle tree root, Merkle tree is shown in Fig. 2.

A. A verification process would involve two steps:

1. Decrypting each transaction using the public key that belongs to its sender. For that purpose, a hash table that maps each node address to its public key is maintained in in the header of the transaction list. If the transaction, plain text token exists in the transaction list, then continue.
2. Use the decrypted transaction tokens to rebuild the Merkle tree. If the same value of the root is calculated, then a valid block is signaled and broadcasted.

B. If the coming request conforms to the global rules of the university such as submitting a registration within deadline, then continue.

C. Calculate the hash value of all the tokens of requests blocks in blockchain via *hash token* = h (*hash token*$_1$, *hash token*$_2$, . . ., *hash token*$_l$).

D. If hash value of requested blocks is equal to the hash value of requested blocks holds, it means the user can be authorized to access the queried data, otherwise, the full node should deny the access request.

E. The version of the current block is incremented.

Step 4: The result is a blockchain that gets validated and appended in the database

Blockchain model's implementation is conducted using a blockchain network and database as in Fig. 3 and Fig. 4. All transactions information stored in the blockchain system which will not be possible to change or delete them.

3.3. Three proposed models of SIS blockchain

Considering the above basic design definitions, to put our hands on several settings that can form various frameworks for an SIS. There are several dimensions that can form different models for blockchain in SIS.

A. Permissions consensus mechanism versus permission-less.

B. Transaction list definition and categorization of transactions into blocks.

C. Full nodes versus light-weight permissioned consensus network, with transactions that only accepts stateful data nodes.

Each model considers each of the above dimensions and builds up a blockchain accordingly.

Model 1: Permissioned consensus network, with transactions that only accepts stateful data

In the first model, the simplest choices are used to build the blockchain. It is most appropriate for small organizations. Model 1 considers a permissioned consensus network, with transactions that only accepts stateful data and all nodes are full nodes where each node keeps a full copy of the blockchain. The sender node that adds a transaction is responsible for adding it to the transaction list; which could typically be a SQLite database; and adding it to the Merkle tree. The sender node is also responsible for validating all the fields and verifying that the sender account has the authorization to create and send the transaction according to the SIS rules that are embedded into the genesis block of the blockchain during the setup. In this model the stateless information is kept in an ordinary system outside the scope of the blockchain.

By definition, a permissioned blockchain is a private network. No account can be created unless the new user (student or a pro-

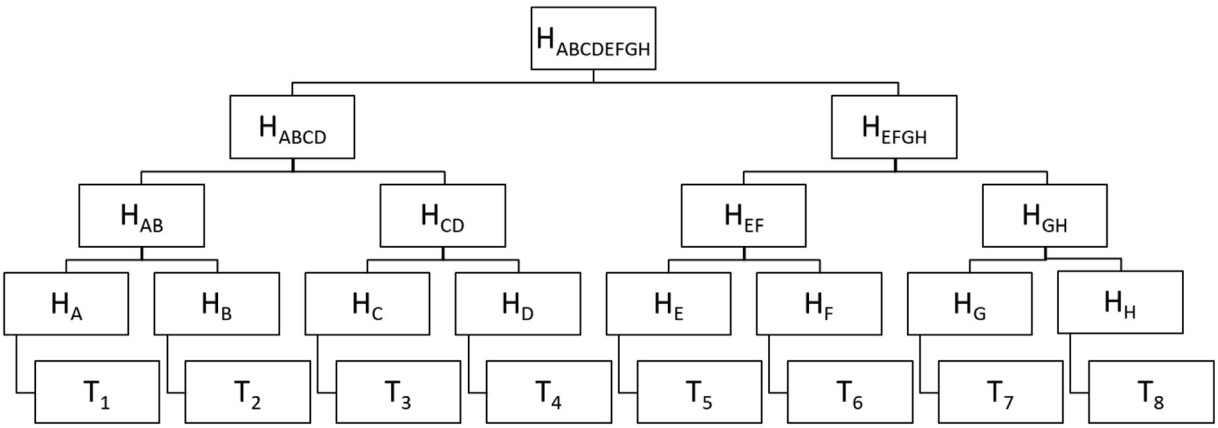

Fig. 2. Merkle Tree.

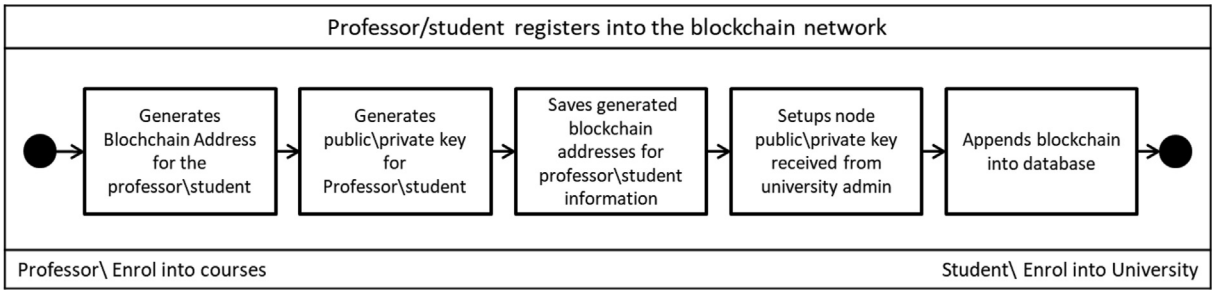

Fig. 3. Professor/student registers into the blockchain network.

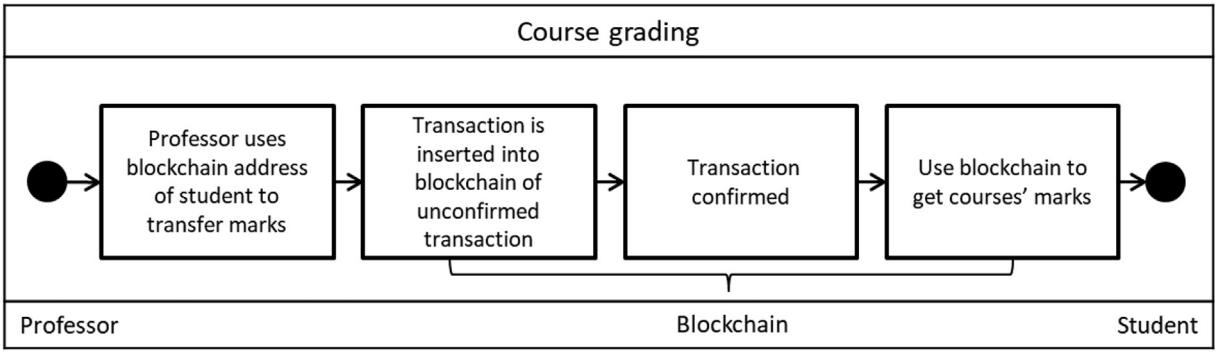

Fig. 4. Course grading of blockchain.

fessor) gets a private key from the administration (Admin) of the organization and uses it to setup a new node. A new node can only be created on a computer inside the organization campus; hence all nodes are well identified and highly trusted. The process of setting up a new node cannot start without a valid private key that has a counter-partner public key created by an admin node using a predefined algorithm. The private key is used to encrypt any transactions created and sent by any node. It is used as a signature that allows the blockchain to identify the owner of any transaction for audit trail. Step 3 in section 3.2 states the details of building a transaction token, adding it to a block and publishing it to the blockchain network. Please, refer to the activity and sequence diagrams of Fig. 5 and Fig. 8, respectively.

Model 2: Permissioned consensus network with full and light-weight nodes and transactions that accepts stateful and stateless data

In the second model, only full nodes keep a full copy of the blockchain, whereas light-weight nodes keep a temporary partial set of transactions in the form of individual temporary blocks. This model is most appropriate for small to medium organizations. The full nodes are nodes that are hosted physically inside the organization.

Model 2 uses a semi-public blockchain that allows light-weight nodes with read-only and limited write permissions. A light-weight node can be created with the same procedure as full nodes: a private key is received from the administration of the organization and the user uses it to setup a new node. If the node is created outside the campus network, then it defaults to a light-weight node hosted in the user's computer. If the node is created inside the campus network, then it defaults to a full node hosted in one of the organization servers. Only full nodes that physically exist inside the campus can create and send transactions since full nodes are well identified and highly trusted. A light-weight node can read data related to stateful transactions that the user has read privileges for. It also allows a user to modify stateless data using immutable transactions. For example, a student can send a transaction to

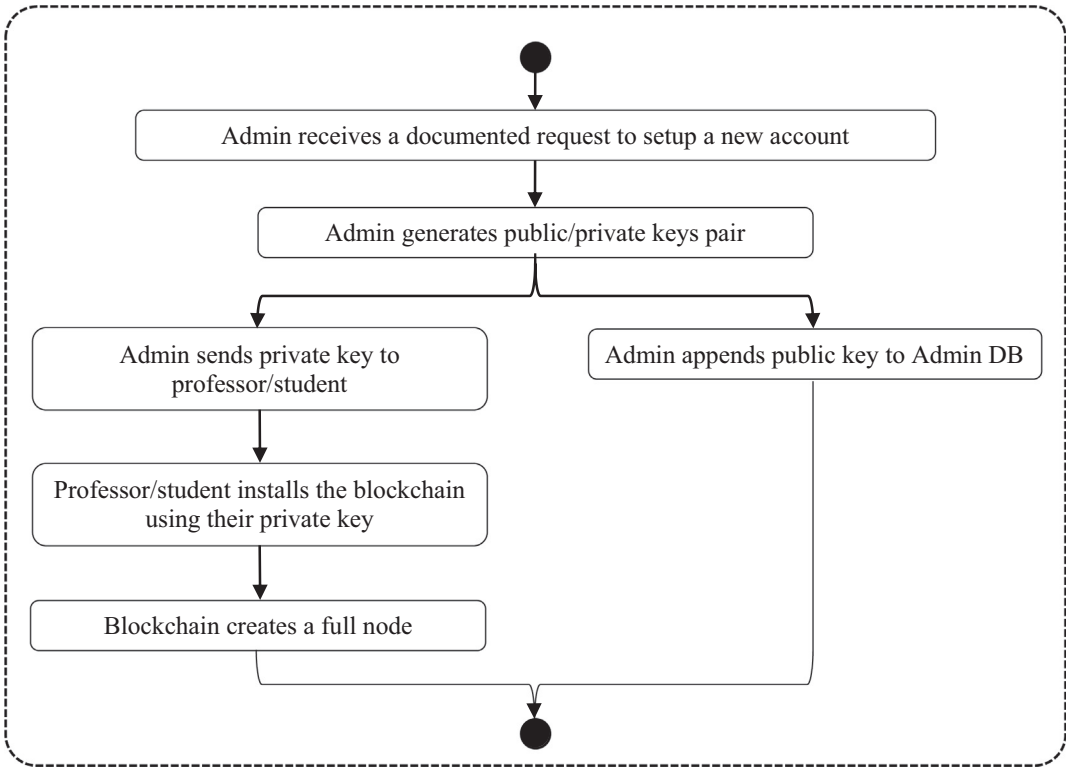

Fig. 5. Activity diagram for creating nodes in model 1.

modify his phone or address from outside the campus. However, professors can add course marks using their full nodes only inside the campus.

When a user creates a light-weight node and registers it to the blockchain it is originally empty. When a user requests to read a student profile or read a list of course marks, only this piece of information is sent to and hosted temporary into his light-weight node. All requested data are encrypted using the public key of the requesting node before data is sent to it. The requesting node decrypts the data using its private key and stores this data until the end of the web session then destroys it.

Two scenarios for light-weight node are presented:

A. In case of a read request: the light-weight node sends its IP address together with a request encrypted by its private key to confirm its identity. A full node that receives the request

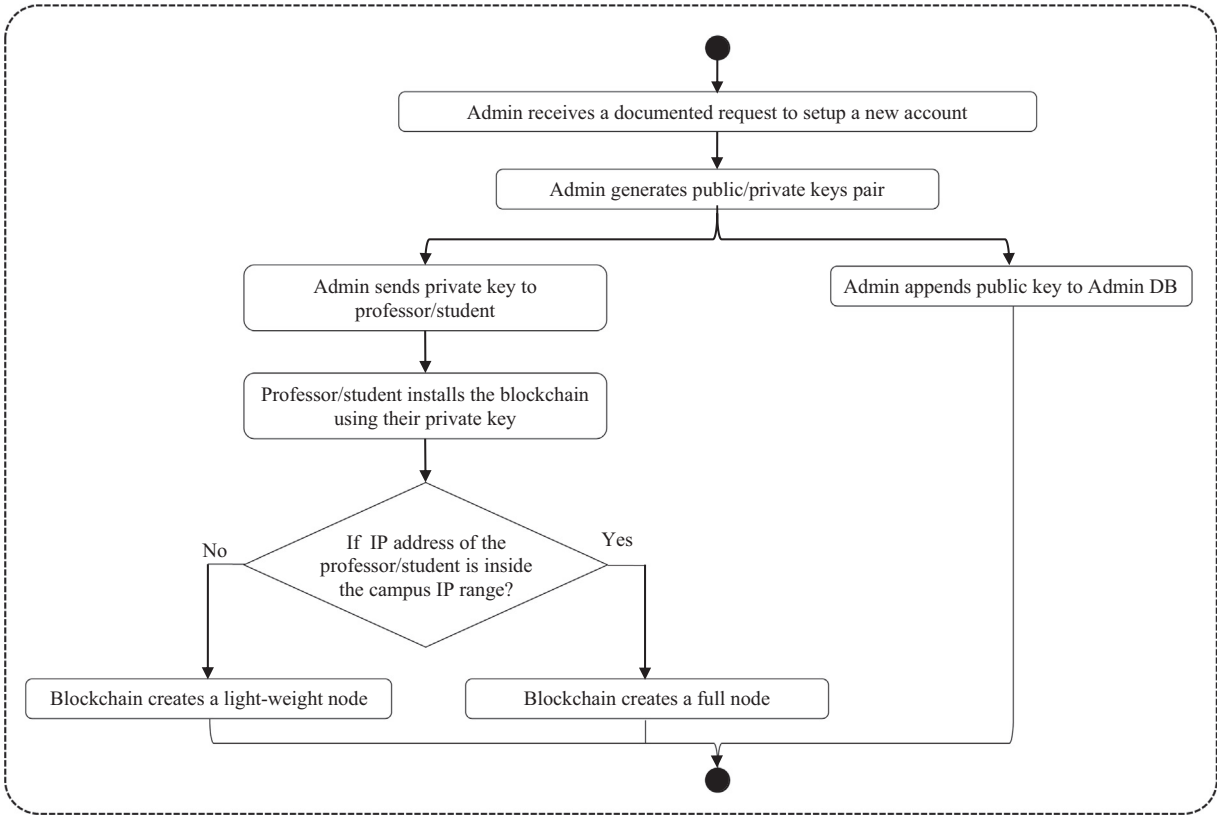

Fig. 6. Activity diagram for creating nodes in model 2.

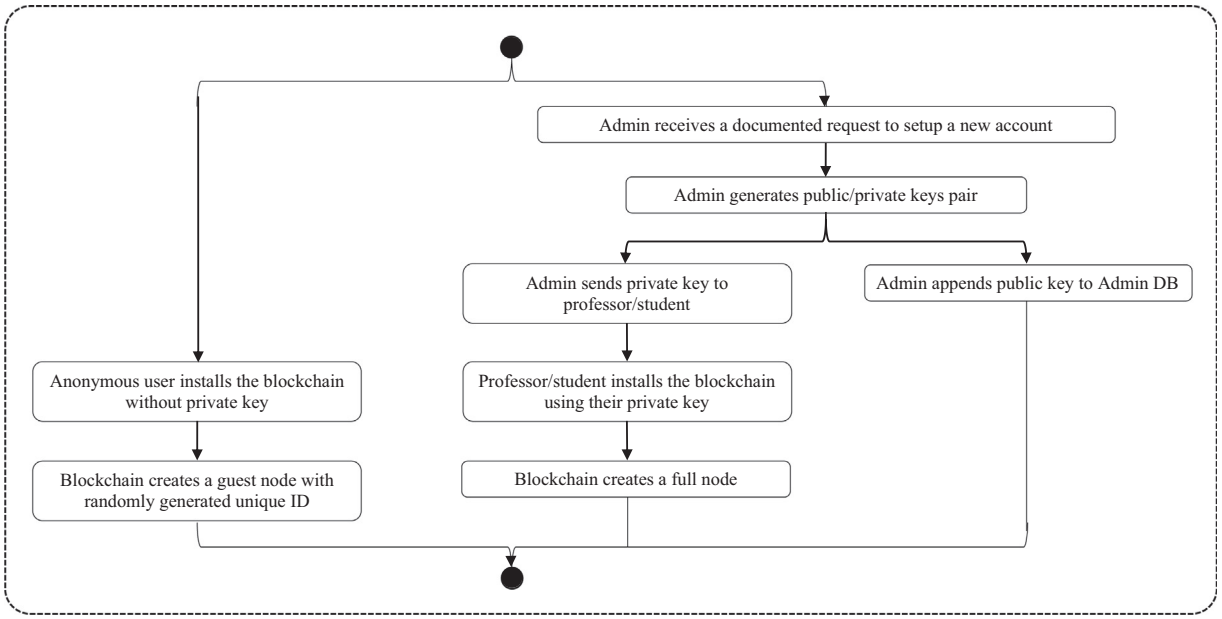

Fig. 7. Activity diagram for creating nodes in model 3.

checks first if the IP have the privilege to access the requested data. Then the full node decrypts the request using the public key the belongs to the IP of the sending node to confirm the identity. Next it sends back the requested report.

B. In case of a write request: the light-weight node requests to add a transaction. A full node identifies the requesting node and if it accepts the handshaking, it sends a block where the transaction can be added into. The light-weight node receives a whole block, and it follows the details described in Section 3.2 step 3. The block is sent back to the full node where it is validated and broadcasted to all other full nodes where it is validated according to the census protocol. Please, refer to the activity and sequence diagrams of Fig. 6 and Fig. 9, respectively.

Model 3: Permission-less consensus network

In the third model, there are two types of nodes: full nodes and guest nodes. Transactions accept stateful and stateless data. Only full nodes keep a full copy of the blockchain and can add transactions, whereas guest nodes have read-only permissions. Guest nodes keep temporary partial set of transactions in form of temporary blocks that get destroyed in the end of the session. Initially full nodes are hosted physically inside the organization.

Model 3 allows a new full node to be setup outside the organization campus network, using a private key received from the organization administration. Any new node that is setup without entering a private key is signed as a guest node and never has a private key.

In big organizations, the frequency of creating new transactions is much higher than medium and small organizations. One of the challenges is the flooding of too many transactions going to and coming from the full nodes leading to slower transaction acceptance rate. The categorization of transactions is vital. With no miner's incentive, the SIS blockchain depends solely on full nodes to accept and handle concurrent transactions. According to authors of [17] a contract can be used where the sender node can randomly address one of service providers to execute the contract code. Another scheme let each of the receiving full nodes receive requests from a list of predefined of nodes. So that only one full node executes the requests and add it to a block and then broadcasts it for other full nodes to verify and accept. Please, refer to the activity and sequence diagrams of Fig. 7 and Fig. 10, respectively.

4. Discussion

Every user gets a unique identifier in the form of a private key with a counter partner public key used to encrypt any transactions coming from his nodes. The node addresses and the public keys are

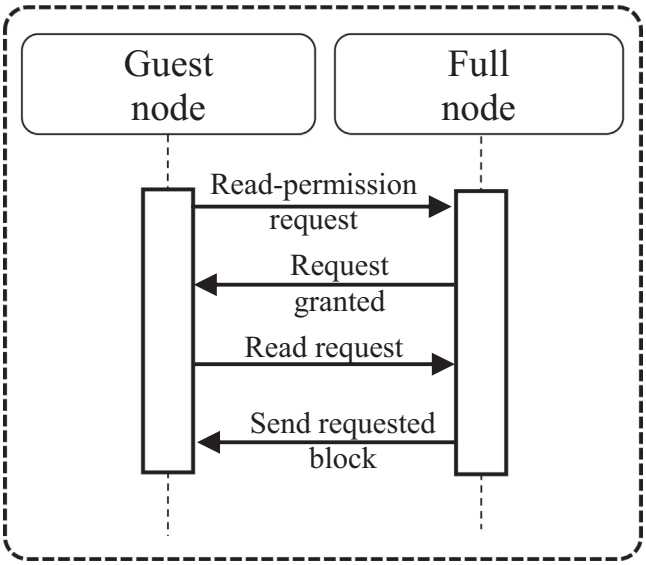

Fig. 10. Sequence diagram for guest node read requests in model 3.

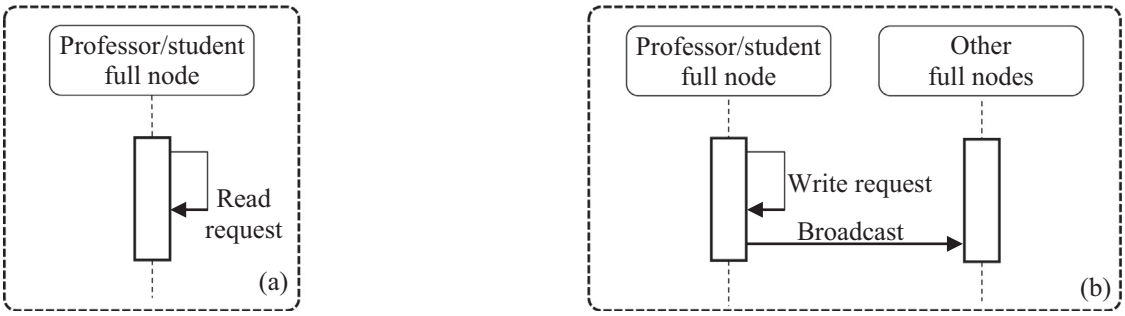

Fig. 8. Sequence diagram for full node requests in model 1, model 2 and model 3: (a) Read request and (b) Write request.

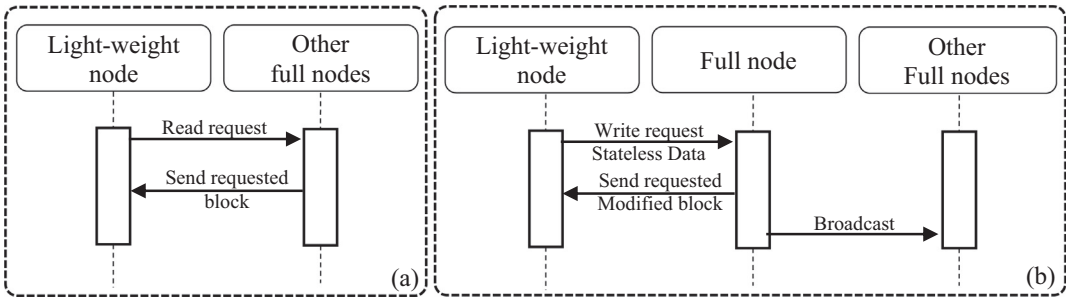

Fig. 9. Sequence diagram for light-weight node requests in model 2: (a) Read request and (b) Write request.

Table 1
Comparing of the three proposed models.

Features	Model 1	Model 2	Model 3
Organization size	Small, private organizations	Small to medium organizations with needs for access privileges.	Large organizations with various needs
Architecture	One type of nodes: full nodes are physically inside the organization.	Two types of nodes: 1- Full nodes are physically inside the organization. 2- light-weight node can be hosted outside the organization.	Two types of nodes: 1- Full nodes are physically hosted anywhere as long as the user has a valid private key. 2- Guest nodes are anonymous nodes for read only.
Performance	Least confirmation time.	More confirmation time because a greater number of blocks.	Requires longer confirmation time because of bigger number of full nodes.
Security	Flexible and secure because of their storage data is limited, so the data is more reliable.	Verified each transaction data is reliable.	Storage capacity, resilient, flexible, and secure. Resource sharing in a public.
Scalability	Tolerate smaller number transactions.	Tolerate stateless and stateful data in the form of transactions. But limited in tolerating simultaneous transactions	Tolerate bigger amounts of data and several transactions simultaneously.
Maintainability	Easier to maintain because of a smaller number of full nodes.	Still easy to maintain because of a moderate number of full nodes	Hardest Maintainability because of unlimited number of full nodes
Usability	Can only be used inside the organization campus.	Can be used inside the campus with unlimited privileges. However, limited privileges are outside the campus.	Unlimited usage inside and outside the campus. Data can be shared publicly with guest, anonymous users.

stored as a list of pairs in the genesis block under a root indicating the responsible node that should verify and publish any requests received from the nodes under its tree. The genesis block is copied into every instance of the blockchain.

All the transactions will be stored in the blockchain system in the form of a list transaction list, which could typically be a SQLite database and a Merkle tree. Having copies of the Merkle tree in each blockchain instance avoids any possibility to change or delete any transaction. The properties for each model are described in Table 1. As the previous section is shown, every model has its strengths and weakness; see the Table 1. The administration of the university controls of following:

- **User information**: There is information about a user (his public address name, username).
- **Authorization subject**: All subjects will be in administrators' accounts. The administrator will give permission for each faculty member accounts to use subjects from his (administrator) account and enter the username of the faculty members account and number of subjects that faculty members will be allowed to send.
- **User model** (including faculty member and students): The administrator will control all users and can add blockchain user information into a centralized database and save blockchain users' information in the database.

The faculty members at the university controls of following tabs:

- **User information**: There are information about the student (ID, username, email).
- **Courses:** A faculty member gets a list of the students who are registered in his subjects and mark each of them. Administrators allow the faculty members to purplish marks from their node so that the students can view their marks from their nodes. Students can access their accounts and see the (marks) for their subjects. Student account has the following tabs:
- **User information**: There are information about student (ID, username).
- **Courses**: Students have list of subjects and corresponding (marks).

The implementation of blockchain models is conducted using a blockchain network, and database, as in Fig. 3 and Fig. 4. For example, as a real test case in the proposed models, if 10 students has been involved from second stage and 20 students from third stage then l30 blockchain is created for students. A token is made for each 5 subjects for each stage. For second stage students there are 50 transactions (10 χ 5) to transact marks for students. For third stage students there are 100 transactions (20 χ 5) to transact marks for students. After sending the (marks) to the students. All the transaction information will be stored in the blockchain system. It will not be possible to change or delete them. The functions for each model are described, where every model has its strengths and weakness as reviewed in Table 1.

5. Conclusion and future vision

Using blockchain, the paper introduces three different blockchain models that can match different sizes of organizations. The three models avoid the need for miners because there is no incentive for miners. Instead, the consensus protocols allow one full node to verify and add a transaction, then broadcasts it to other full nodes. The three models depend on the fact that only trusted, and well-identified nodes can be created. Other types of nodes with limited privileges can exist. The infrastructure of model 3 is based on blockchains are well suited for large-scale data processing.

Universities can deal with a student information system (SIS) in a way that greatly achieves security. Storing and sharing data will be simpler with decentralized data storage and could be programmed into the block chain. Another feature, the ledger can represent as a valid guide to maintain SIS and can introduce reliable, highly trusted model for accessing and storing data.

Using UML class and interaction overview diagrams, the paper explained the core building blocks and functionalities of our models within the description. In future, the authors plan to test the proposed models using the smart contracts and the Ethereum network, based on activated the verification, authentication, and transparency of marks.

References

[1] Tolbatov AV, Agadzhanova SV, Tolbatov VA. Using blockchain technology for e-learning, *1*, no. 1, Art. no. 1, 2018.
[2] Liang X, Shetty S, Tosh D, Kamhoua C, Kwiat K, Njilla L. ProvChain: A Blockchain-Based Data Provenance Architecture in Cloud Environment with Enhanced Privacy and Availability, in *2017 17th IEEE/ACM International Symposium on Cluster, Cloud and Grid Computing (CCGRID)*, Madrid, Spain, May 2017, pp. 468–477. 10.1109/CCGRID.2017.8.

[3] Jha S, Koul S. *Application of block chain technology in higher education*. 16th AIMS Int'l Conference on Management (AIMS-16), 2019. Accessed: Oct. 29, 2021. [Online]. Available: http://dspace.jgu.edu.in:8080/jspui/handle/10739/2065
[4] Lacity MC. *Blockchain foundations: for the internet of value*; 2020.
[5] Tapscott D, Tapscott A. Blockchain revolution: how the technology behind bitcoin is changing money, business, and the world. New York: Portfolio / Penguin; 2016.
[6] Ezeudu Florence O, Eya Ngozi M, Nworgi Hope I. Application of blockchain-based technology in chemistry education students' data management. IJDTA 2018;11(2):11–22. doi: https://doi.org/10.14257/ijdta.2018.11.2.02.
[7] Kuppusamy P. Blockchain architecture to higher education systems. Int J Latest Technol Eng, Manage Appl Sci (IJLTEMAS) 2019;VIII(II):124–38.
[8] Lin I-C, Liao T-C. A survey of blockchain security issues and challenges. Int J Network Security 2017;19(5):653–9. doi: https://doi.org/10.6633/IJNS.201709.19(5).01.
[9] Turkanovic M, Holbl M, Kosic K, Hericko M, Kamisalic A. EduCTX: a blockchain-based higher education credit platform. IEEE Access 2018;6:5112–27. doi: https://doi.org/10.1109/ACCESS.2018.2789929.
[10] Meyliana M *et al.*, Defying the certification diploma forgery with blockchain platform: a proposed model. In: *Proceedings of the International Conferences ICT, Society, and Human Beings 2019; Connected Smart Cities 2019; and Web Based Communities and Social Media 2019*, Jul. 2019, pp. 63–71. 10.33965/ict2019_201908L008.
[11] Kumar SMKVP, Kumar KK, Krishna RS, Sri PSGA. Incorporation of blockchain in student management system. Int J Innov Technol Explor Eng (IJITEE) 2019;8(6):664–8.
[12] Seibold S, Samman G. Consensus Immutable agreement for the Internet of value. KPMG 2016;26:2001.
[13] Han M, Li Z, He J (Selena), Wu D, Xie Y, Baba A. A Novel Blockchain-based Education Records Verification Solution. In: *Proceedings of the 19th Annual SIG Conference on Information Technology Education*, Fort Lauderdale Florida USA, Sep. 2018, pp. 178–183. 10.1145/3241815.3241870.
[14] Dong Z, Luo F, Liang G. Blockchain: a secure, decentralized, trusted cyber infrastructure solution for future energy systems. J Mod Power Syst Clean Energy Sep. 2018;6(5):958–67. doi: https://doi.org/10.1007/s40565-018-0418-0.
[15] Seebacher S, Schüritz R. Blockchain Technology as an enabler of service systems: a structured literature review. In: *Exploring Services Science*, Cham, 2017, pp. 12–23. 10.1007/978-3-319-56925-3_2.
[16] Torky M, Gaber T, Hassanien AE. Blockchain in Space Industry: Challenges and Solutions, *arXiv:2002.12878 [eess]*, Feb. 2020, Accessed: Oct. 30, 2021. [Online]. Available: http://arxiv.org/abs/2002.12878.
[17] Wüst K, Matetic S, Egli S, Kostiainen K, Capkun S. ACE: asynchronous and concurrent execution of complex smart contracts. In: *Proceedings of the 2020 ACM SIGSAC Conference on Computer and Communications Security*, Virtual Event USA, Oct. 2020, pp. 587–600. 10.1145/3372297.3417243.