IEEE Access
Multidisciplinary ⋮ Rapid Review ⋮ Open Access Journal

Received April 7, 2022, accepted May 30, 2022, date of publication June 3, 2022, date of current version June 10, 2022.

Digital Object Identifier 10.1109/ACCESS.2022.3180168

A Framework to Make Voting System Transparent Using Blockchain Technology

MUHAMMAD SHOAIB FAROOQ[1], USMAN IFTIKHAR[2], AND ADEL KHELIFI[3]

[1]Department of Computer Science, University of Management and Technology, Lahore 54000, Pakistan
[2]Department of Software Engineering, University of Management and Technology, Lahore 54000, Pakistan
[3]Department of Computer Science and Information Technology, Abu Dhabi University, Abu Dhabi, United Arab Emirates

Corresponding author: Muhammad Shoaib Farooq (shoaib.farooq@umt.edu.pk)

ABSTRACT A widespread mistrust towards the traditional voting system has made democratic voting in any country very critical. People have seen their fundamental rights being violated. Other digital voting systems have been challenged due to a lack of transparency. Most voting systems are not transparent enough; this makes it very difficult for the government to gain voters' trust. The reason behind the failure of the traditional and current digital voting system is that it can be easily exploited. The primary objective is to resolve problems of the traditional and digital voting system, which include any kind of mishap or injustice during the process of voting. Blockchain technology can be used in the voting system to have a fair election and reduce injustice. The physical voting systems have many flaws in it as well as the digital voting systems are not perfect enough to be implemented on large scale. This appraises the need for a solution to secure the democratic rights of the people. This article presents a platform based on modern technology blockchain that provides maximum transparency and reliability of the system to build a trustful relationship between voters and election authorities. The proposed platform provides a framework that can be implemented to conduct voting activity digitally through blockchain without involving any physical polling stations. Our proposed framework supports a scalable blockchain, by using flexible consensus algorithms. The Chain Security Algorithm applied in the voting system makes the voting transaction more secure. Smart contracts provide a secure connection between the user and the network while executing a transaction in the chain. The security of the blockchain based voting system has also been discussed. Additionally, encryption of transactions using cryptographic hash and prevention of attack 51% on the blockchain has also been elaborated. Furthermore, the methodology for carrying out blockchain transactions during the process of voting has been elaborated using Blockchain Finally, the performance evaluation of the proposed system shows that the system can be implemented in a large-scale population.

INDEX TERMS E-polling, voting system, blockchain application, blockchain voting, E-voting, electoral system, blockchain, cryptographic hash, secure voting.

I. INTRODUCTION

Blockchain can reduce a lot of efforts and resources invested in polling stations, specifying the areas, appointing staff, and preventing security risks at polling stations [1]. Holding a digital election through blockchain not only saves money but also reduces the risk of inequity in the voting process [2]. Modern technologies such as blockchain technology are very secure and beneficial if used carefully. It can make the voting system more transparent, reliable, and also enhance traceability of transactions [3]. In a traditional digital voting system, a voting machine has been used which is connected to a centralized database. This machine can be tampered with by a person who has physical access to it. It may cause single point of failure in the whole network of the voting system; on the other hand, an immutable blockchain would not be affected by an individual saboteur in the network [4]. In a blockchain, the data is being stored in a decentralized manner, which is constantly verified if the records are accurate. Therefore, in case of a malicious attack on a node, only that node would be affected and peer to peer network still provide all services. It makes blockchain technology a threat-proof and reliable system to be used as a private ledger in the voting system. Blockchain has offered a level of security

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

and trust more than the previously used technologies [5]. The resources can be saved by hiring less staff, security forces, and arranging polling stations for a traditional voting system can be given to the miners [6]. It does not require a physical module/voting machine which always has a chance of cyber-attack through tampering devices [1]. The most ambient part of this system is the transparency which make people guarantee that their votes are being used in the right means. Blockchain technology is more secure and transparent than any other database. It is being considered to be applied in critical systems like banking, medical fields, food safety and cryptocurrency exchange [6]. Therefore, using blockchain technology in the voting system assures that it will be more trustable, secure and consistent as compare to other digital voting systems. Blockchain technology can carry out a transparent voting process while maintaining voter privacy and trust.

In the traditional voting system, a voter first registers his voting status by sending an SMS to Identification Authorities (IA) [7]. IA verifies the voter and enlists his vote in the specified polling station. The voter goes to the polling station physically to cast the vote. Votes are then collected by the appointed staff at polling stations [8]. Staff is responsible for counting the votes and reporting them to higher authorities. This involvement of staff in the voting system increases the chances of human errors [3]. The authorities again make sure that the count is correct and then publish the result. This whole system has a lot of loopholes where tampering with votes is possible [9]. Either the staff can tamper with the votes or add fake votes while counting them. The authorities can announce wrong results of their own choice, under the influence of a third party [10]. There is no surety given to the voter that proves that his vote is being considered or not. The whole traditional voting system offers no transparency or guarantee of fair voting. The situation cannot be resolved by only digitalizing the system or by casting votes through digital voting machines [11]. These digital voting machines can be tampered with by hackers; which disrupts the voting process [12]. Hence, a decentralized system is required which can keep the data secure, even if a single node gets corrupt. The problems discussed above highlight the lack of transparency in traditional and digital voting systems [13]. The authorized organization gets blamed frequently for any human error or lack of system efficiency, even if they try to conduct fair voting in-country [7]. The **Fig. 1** shows the traditional voting system implemented in most countries.

The main focus of this manuscript is to discourage any illegal tampering in the voting system to gain the trust of voters by proposing a blockchain-based voting system. One of the core features of blockchain technology is that if a node is added to the chain it can never be erased. It is possible due to decentralized data storage in blockchain technology. The data is stored on multiple nodes in the chain and the data is copied on every single node of the network, which makes it impossible to erase it from every node simultaneously. It assures the reliability of the voting system. Moreover, the tallying computation is reduced because the data is computed on the fly, every node computes the data separately which is then stored as one. Voter verification is made possible through online verification with the assistance of Identifying Authorities. The concept of using blockchain in the voting system is offered to eliminate the vote tempering and the casting of ghost votes. Each ID can be used to cast vote once. By implementing the blockchain, the integrity of the votes stored on the system gets secured. Blockchain blocks any illegal breach to the system that tries to equip any role in the chain. It eliminates the dependency on middlemen in the voting system which helps to save resources and prevent human errors. Initially, the blockchain-based Voting Management System(VMS) will be implemented along with the manual traditional voting system, to ensure its validity. When VMS will show promising results, then the whole voting activity could be transferred to VMS only. A pilot implementation can be carried out in small elections of an area before large-scale implementation.

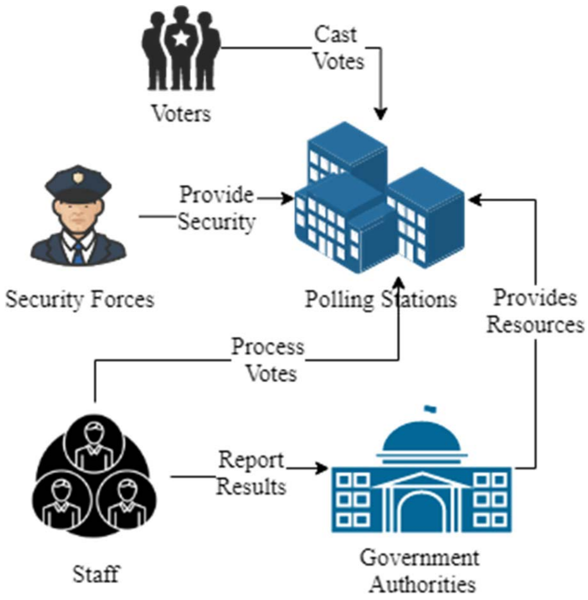


FIGURE 1. Traditional voting system.

The rest of the article is structured as follows: the related work has been presented in Section II, and the proposed framework has been discussed in Section III, while Section IV focuses on the implementation and evaluation of the proposed framework. Finally, the article has been concluded in Section V.

II. RELATED WORK

G. Rathee *et al.* [14] introduced a digital voting system based on blockchain that could be implemented in a technologically advanced environment. The system assumed that all the connected external entities are trustworthy. However, the security flaws in the system could be a huge risk, as intruders can enter the system to rig the votes. Whereas in our proposed voting system, the use of encryption and secure networks minimize the risk of intruders barging into the votes.

M. Pawlak *et al.* [15], proposed a system that does not require any operating entities. However, it could not secure a voter's identity, and also required complex computing. The system was able to collect votes from users but due to complex computation, upon higher user rate the latency became an issue. The identities of the voters became vulnerable. The system could not compute a large amount of data hence it has failed to be implemented at a large scale. Whereas in our proposed voting system the latency is managed by flexible use of consensus algorithms. The use of cryptographic hash in blockchain omits the risk of the vulnerability of a voter's identity.

A proposed system by D. Chaum *et al.* [16] improved the robustness and fair tallying of votes. End-to-end verification was made possible for voters to assure their count of the vote is integrated. Each voter was able to view if his vote was considered and recorded correctly. The voters were given a unique code that they could enter into the system to verify their vote. Whereas in our proposed voting system the verification of votes has been further simplified. Voters can verify their votes through registered phone numbers and email addresses. Verification of votes after the voting activity builds up the trust of voters.

The voting system without polling stations was discussed by P. Mccorry *et al.* [17]. He suggested that voting through the internet using blockchain can give good results if implemented correctly. They discussed some technical flaws in digital voting systems. The robustness of the system could not be controlled. The error of doubling users gets minimal by using end-to-end verification. These voting systems had low latency and did not secure the privacy of voters. This latency in the system is controlled by using a flexible consensus algorithm and smart contracts in the blockchain voting system being proposed.

The **Table 1** shows a detailed comparison of our proposed VMS with other voting systems based on blockchain technology. The state-of-the-art blockchain voting system was based on a single fixed consensus algorithm however our proposed system supports a flexible consensus algorithm at run time that helps to control the performance of the voting activity. We have also proposed a solution for the prevention of the 51% attack to prevent malicious activities during the voting process. Our system has further proposed the Chain Security Algorithm which automatically verifies the validity of the chain each time a new block is added to it or some unauthorized change occurs in block data. We have also proposed a mechanism of Unspent Transaction Output(UTXO) and Smart Contracts that allow the system to prevent any incomplete and malicious transaction in the blockchain.

The above discussion shows that there has been a lot of debate about making a secure, efficient, and transparent voting system in past but any comprehensive solution or system has not been proposed that could fulfill all requirements and key purposes of the voting system. The novelty of the proposed system besides implementing blockchain is to assure security, transparency, and integrity of results in a voting system that empowers the trust of voters. This article proposes a more efficient and better implementation of blockchain in digital as well as a traditional voting system that is already being used. It gives higher authorities more detailed insight into the system while keeping it decentralized and transparent for every voter.

TABLE 1. Comparison of current systems with proposed.

	VMS	[14]	[15]	[16]	[17]
Flexible Consensus Algorithm	✓	✗	✗	✗	✗
Attack 51%	✓	✗	✗	✗	✗
Chain Security Algorithm	✓	✗	✗	✗	✗
UTXO	✓	✗	✗	✗	✗
Smart Contract	✓	✗	✗	✗	✓

III. PROPOSED FRAMEWORK OF VOTING SYSTEM

Blockchain is mutable, unlike other programming structures where an admin can add, delete or update the data. If such a system is used for voting, then anyone having access can tamper with the system and update or delete the votes. This is not the case with blockchain technology. Once a node is added to the chain, it cannot be deleted or updated under any condition. If a node is attacked by an intruder, the corresponding nodes detects it and rebuild the damaged node, hence the chain becomes immutable. The blockchain is decentralized which makes the voting system independent of any single computing node. The voting activity remains operational even if any one or more nodes get attacked or become unavailable. It assures reliability under any severe condition. The main stakeholders of the proposed framework are voters, Identification Authorities (IA), and Administration Authority (AA) of election commission.

A. VOTING SYSTEM ARCHITECTURE

In **Fig. 2** a high-level architecture of the proposed system has been presented. It shows how the main stakeholders; Voters, VMS, AA, and IA work together to perform certain voting tasks. All voters are connected to VMS directly through dAPP; it is either a mobile application or a web portal. The identification authority verifies voters registering in the system. Any voter who is verified and eligible to vote is allowed in the application to take part in voting.

The process of the whole system includes certain parts; the first one is the user interface of the application, which also requires front-end security. It is critical because the user enters his credentials on that interface, so it should be secure

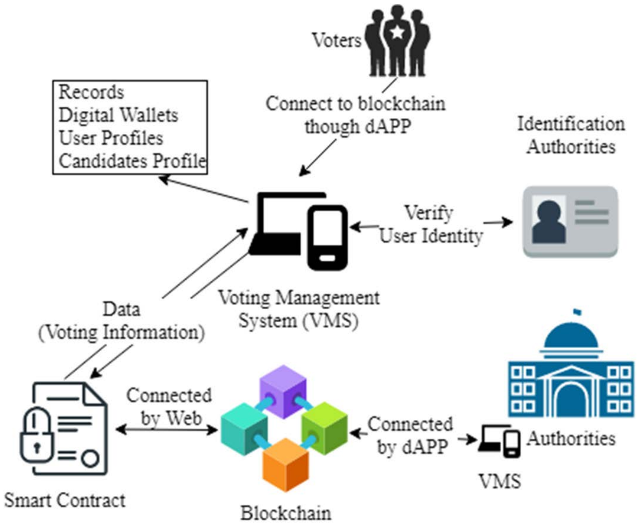


FIGURE 2. Voting system architecture.

and simple. The system provides full and fair access to every user during voting activity. It also provides traceability after casting of vote. The voter registers in the system by his credentials. VMS uses the ID details of voters and verifies them with online records of IA to register the voter in the system. The user receives a unique OTP to log in to the system. An OTP is generated each time the voter wants to login into VMS. All the detail of the voter is saved in VMS. After successfully registering in the system, One Voting Coin (VC) is added to the wallet of each voter. To prevent voters from voting twice, each voter is given only one VC.

B. WORKFLOW OF PROPOSED MODEL

The voter after completing the verification is registered into the Voting Management System. A single chain system is implemented on the blockchain. The national database of the country is also integrated with the system to keep the voter's voting integrity. For every vote, a transaction is being generated against the voter's National ID. The transaction is then mined by the minors and saved in the blockchain. When the voter casts the vote, his Vote Coin in his/her wallet is also being utilized. The voter cannot cast another vote after one vote coin is utilized. As the voter will sign-in through his/her credentials then the voter has been redirected to the election interface where all the candidates who are contesting in his constituency are shown to the voter. Upon the voter's request to cast vote, VMS verifies the voting status of the voter from the blockchain by checking all transactions hash that already exists against his/her computerized National ID. If a transaction hash is found against the voter's computerized National ID then VMS declined the request and logout the voter from the system. If a voter has not voted yet, the request is transferred to the miner to add the node. The voter selects the desired candidate and casts his vote. The transaction is monitored with the help of a transaction hash and carried out by the miner. The node is then added to the chain for balloting.

Voters must have access to any smartphone or web browser to take part in voting. The voter's interface would be provided in multi-languages to make it easy to use for all users. The proposed system can contain a large number of voters at the time of voting. A decentralized blockchain system enables a voter to vote from any part of the world. A person can take part in voting from anywhere, even if he is in a foreign country, in this way his/her computerized National ID is verified from the national database so he can cast the vote.

Voting transactions are sent to a pool from which miners analyze them and remove the malicious request by taking the consensus from the other nodes before adding it to the chain. The votes are fully secured using a cryptographic hash. Each vote cast adds a new block in the chain. System also make sure that only one vote can be cast by one user by using the vote coin. Even if due to some technical fault, the balance of the voting coin does not get updated, hence system ensures that no double votes have casted by a voter. By checking whether a transaction hash is generated against voter computerized National ID or not, if any node or request of a voter is malicious then miner automatically rejects it. When the transaction completes and a node is successfully added to Vote Chain, the voter of that particular voting transaction is notified through an SMS to his registered phone-no and email. The voter has provided with a unique transaction hash by which he can verify his vote through a web portal and upon successfully completion of transaction the vote has been counted in the whole voting activity. The voter wallet has then contained zero vote coins, when a voter successfully casts the vote.

C. LAYERED STRUCTURE OF THE PROPOSED VMS

The proposed framework has ben presented in a layered structure. System services have been bifurcated the process into five layers, it has the following layers presented as shown in **Fig. 3.** The interface layer contains all the dAAPs developed for voters and administration. These are the distribute APPs through which any stakeholder can connect to VMS. The goal of this layer is to provide an interface for interacting with the system.

Application Layer provides a user verification system by involving external sources. It is the front-end interface of the whole voting system. It encapsulates the data of the voting system in online databases. All the blockchain transactions are also handled in this layer. The user is verified by his/her National ID, that was eligible to take part in this voting activity.

The Trust layer is the most important part of the whole framework. It ensures consensus is made correctly and the data is transferred securely through smart contracts. It verifies every new block added to the chain.

The basic information related to the blockchain is stored in the blockchain layer; it keeps track of any faulty node in the chain to ensure the immutability of the chain. All the public and private keys, and transaction data are stored in this layer.

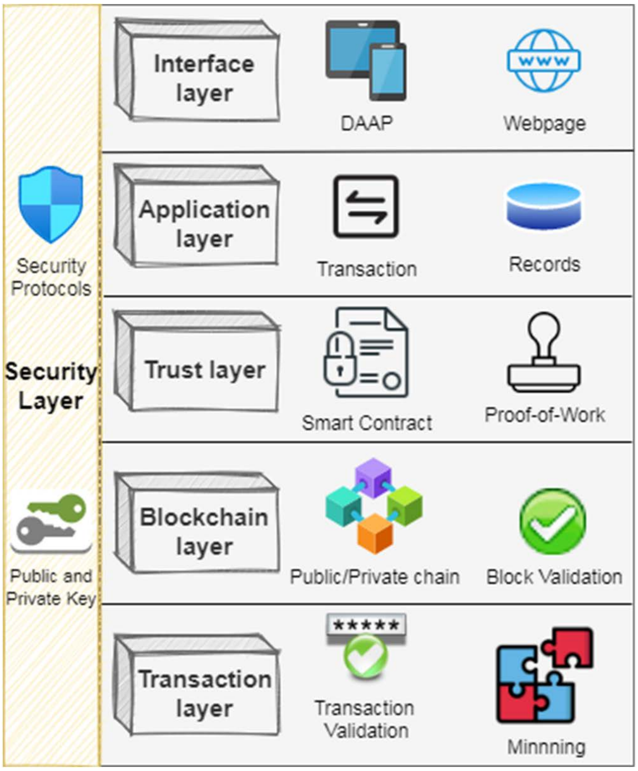


FIGURE 3. Layered structure of the framework.

The transaction layer encompasses all the transactions that are made by using smart contracts between VMS and voters. Mining of all the transactions takes place in the transaction layer.

The most critical layer is the security layer by which the blockchain is protected from any attack. Any attempt of attack is defended by using algorithms and basic rules that make it difficult for external entities to harm the chain. The security protocols are implemented on the entire chain. The private and public keys make data encrypted and secure throughout the system.

D. FLEXIBLE CONSENSUS ALGORITHM

The proposed framework supports scalable blockchain. The system provides the support of plug-and-play consensus algorithms. By default, the Proof of Work consensus (PoW) algorithm has been proposed to keep the blockchain efficient throughout the voting activity, but the framework also supports other available consensus algorithms that can be chosen at the deployment time of the blockchain. Proof of Work states that a certain amount of computational power has been spent on a transaction. The framework is flexible enough to incorporate the change in consensus algorithms at run time. It not only helps to keep the blockchain secure but also keeps the voting activity undisrupted.

Choosing a suitable consensus algorithm at deployment time helps the blockchain to give maximum performance at run time. Different consensus algorithms can be implemented in Voting such as Ripple, Proof of Vote, Proof of Trust, and Proof of Stake. Here, Proof of Stake tells the computing power required to the system by validating random nodes. If a node is validating a false transaction, then it immediately loses its stake in the voting chain. The transaction from such nodes is not accepted.

IV. DESIGN AND IMPLEMENTATION

The process of voting is run by maintaining our system that is backed up by the blockchain. The hashes of transaction for every voter has stored on the chain and all the results of the election are also stored on the blockchain and from there the result of the election can be viewed on the resulting dashboard of the users. The system first verifies whether the voter is the country nationality holder and it also checks whether the voter has already voted or not if he still has a vote coin, the system allows him to cast vote. After verifying the voting details i.e. voter identifier, vote, and timestamp was stored in the chain which saves vote details. The whole process is elaborated in **Fig. 4**.

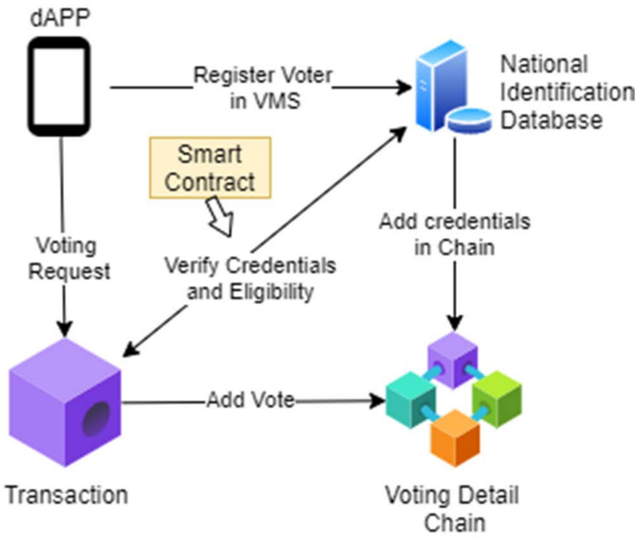


FIGURE 4. Smart contract used in the proposed VMS.

A. dAPP SETUP

Voting Management System consists of different components discussed in this section. It has a user interface for secure interaction of voters with the system, which also includes front-end security. A dAPP interface has been implemented front end of the VMS. The dAPP is a decentralized application based on blockchain technology. It runs on a P2P blockchain network. The user identification is critical because the user enters his/her credentials on that interface, so it should be tamper-free and simple. The system provides full fair access to every voter and provides traceability after casting the vote. The voter login the system by his/her credentials. System uses the ID details of the user and verify them with the Database to register the user in the system.

The user gets a unique OTP to log in to the system. The OTP has generated each time the voter login into VMS. The purpose of using the dAPP system is to ensure the reliability of VMS; as decentralization makes processing efficient at all nodes. If one node of the system during the voting system gets vulnerable, all the other nodes are not harmed. The node which gets vulnerable is reinstate by other nodes.

B. UNIQUENESS OF VOTERS

The uniqueness of users can be established by using their computerized National ID. When voters enter their details, those are then verified with the assistance of identifying authorities. This makes sure that a person's identity is not being used by another voter. By taking credentials, the authorities verify, if the person is eligible to cast a vote in the election or not and whether any transaction hash has already been assigned to his/her National ID. If voter is eligible, the system registers the voter and one Voting Coin is awarded to every voter wallet. Verifying the user by this method also enables the system to judge user either fulfills the requirements imposed by the law and cast the vote. When the voter casts the vote, a transaction hash is assigned against the National ID of the voter. Furthermore, the wallet balance of voters is also updated to zero voting coin, which eliminates the possibility of doubling a vote from the same user. When a voter casts a vote, the blockchain updates and saves the vote of the voter, which means that the user is not be able to cast a vote again unless a new VC is issued.

C. ELECTION AS A SMART CONTRACT

Smart contracts are providing a secure connection between the user and the network while executing a transaction in the chain. These are the rules that are implemented on the entire blockchain and cannot be neglected under any condition. All the nodes have to follow the smart contracts to save the vote in the system successfully.

The first smart contract is for user verification between IA and the VMS; it uses the Can-Cast-Vote function which checks the requirements of the system to make sure the specified voter can vote. After verification, it enters the voter details record for further use. The voter is being connected with a voting smart contract that specifies which candidates would be shown to him/her. If the consensus between the node and chain agrees then voting is allowed. The smart contact to cast vote in the system is defined in **Fig. 5.** It checks the Vote Coin in the wallet of the voter, whether he/she has eligibility tocast vote or not. A function Cast vote is defined which takes voters' National ID and the wallet address as input and checks if the user voting coin is available. If the voter has a voting coin then the smart contract allows them to cast vote otherwise it has rejected the vote request.

Every vote is stored in the transaction as shown in **Table 2** and each voter gets a Transaction ID for his/her vote. All the information is stored in the transaction is highly encrypted by using the cryptographic hash. If the consensus does not match the voter is preceded to a new screen to help him get the right instructions for voting. Either the ID would be wrong or the user is not eligible to cast vote due to any legal reason. Each voter has a wallet given by the authorities which contains only one Voting Coin. After a successful transaction, the wallet is reduced by one and goes empty. It assures that voter is not being able to vote again. The transaction on the chain is shown in **Table 2**. Here the TX Hash is the transaction ID of a particular transaction, the block in which the transaction is being performed is under the Block column; it is where the transaction is sent. The value of the transaction is the information; to whom the vote was cast. For example, "Candidate A and E" is voted in the first transaction in Block 101. In the case of voting for multiple candidates in an election activity, the data value stores information about both candidates. The voter is still given only one vote coin. The voter selects the desired candidates and uses the vote coin once, which saves information of multiple candidates in that block.

The function proceeds in case of candidate's address is not null, otherwise VMS shows an error and asks to re-select

Algorithm 1: Smart Contract: Casting Vote in VMS

```
Require: Initialization of parameters
Initialize voter_Coin = this voter_Coin
Initialize vote = this vote
Initialize casted_vote = this casted _vote
Func (Cast Vote)
Input: reciever_address
Require: voter_Coin =1
Select Candidate1,2,...,n
If voter_Coin = 0
then casted_Vote = Vote
else revert to reciever_address
End Func
End Smart Contract
```

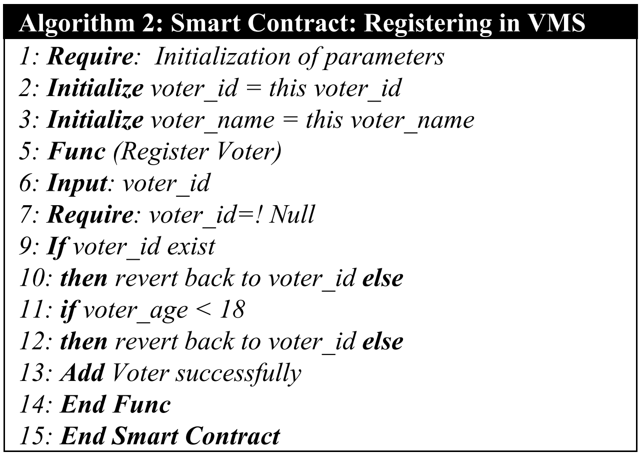
FIGURE 5. Smart contract casting vote in VMS.

TABLE 2. Transaction in VMS.

TxHash	Block	From	To	Value
0xG244e…	1011	0xIUHiu…	T1SC	Voter Info (NIN, Wallet, etc.) Candidate A Candidate E
0xL1345…	1012	0xU98hi…	T2SC	Voter Info (NIN, Wallet, etc.) Candidate B Candidate F
0xOpl21…	1013	0xIpda4…	T3SC	Voter Info (NIN, Wallet, etc.) Candidate A Candidate D

the candidate. By applying this check in smart contract, any unnecessary transaction does not carry out on the blocks of our chain, which contains voting details. However, no incomplete transaction is carried out by smart contract. The transaction is sent to memPool only if all the conditions come true. A memPool is a replicated staging area where every node temporarily keeps the unconfirmed new transactions. After pushing the transaction to memPool the voter is notified via his registered phone number. The miner mines the block of transaction into the blockchain. The system supports permissioned blockchain and if due to any reason the transaction is not added to the blockchain, it has rolled backed to memPool and waiting again for mining process. System guarantees to commit every transaction from memPool without voter interaction. The system allows the miner to mine this rolled-back transaction.

In **Fig. 6**, a smart contract while registering a new user into the system is presented. This smart contract has to make sure that a voter never registered in the voting system twice. For each new registration request, the system verifies through smart contract that specific voter should not already exist in chain. If a National ID does not exist, the system registers the new voter in the system and adds one voting coin (VC) in his wallet. This smart contract also verifies the age of the user in another check so it would not populate the chain with excessive useless registration. The age is verified with the assistance of Identification Authorities through his National ID.

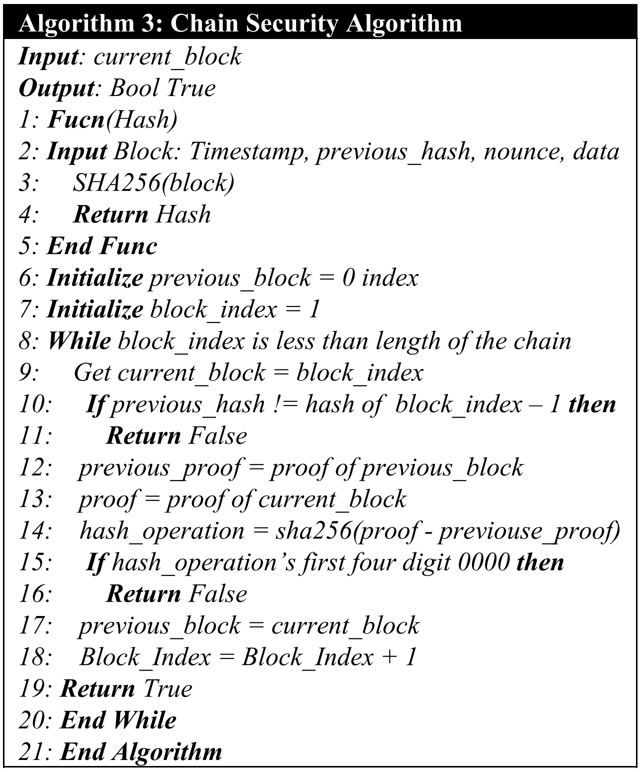


Algorithm 2: Smart Contract: Registering in VMS

```
Require:  Initialization of parameters
Initialize voter_id = this voter_id
Initialize voter_name = this voter_name
Func (Register Voter)
Input: voter_id
Require: voter_id=! Null
If voter_id exist
then revert back to voter_id else
if voter_age < 18
then revert back to voter_id else
Add Voter successfully
End Func
End Smart Contract
```

FIGURE 6. Smart contract registering in VMS.

The Chain Security Algorithm in **Fig. 7** automatically validates the chain, each time a new block is added. It verifies the block to be secure and valid before replicating it to all the peers in the network. The algorithm compares the hash values of the new and previous blocks. If the chain is valid then Chain Security Algorithm allows it to replicate on all nodes in peer to peer network. In case of malicious activity was detected by the algorithm, it declares the chain invalid and also informs all peers on network. This algorithm assures the security of the blockchain.



Algorithm 3: Chain Security Algorithm

```
Input: current_block
Output: Bool True
Fucn(Hash)
Input Block: Timestamp, previous_hash, nounce, data
    SHA256(block)
    Return Hash
End Func
Initialize previous_block = 0 index
Initialize block_index = 1
While block_index is less than length of the chain
    Get current_block = block_index
    If previous_hash != hash of  block_index – 1 then
        Return False
    previous_proof = proof of previous_block
    proof = proof of current_block
    hash_operation = sha256(proof - previouse_proof)
    If hash_operation's first four digit 0000 then
        Return False
    previous_block = current_block
    Block_Index = Block_Index + 1
Return True
End While
End Algorithm
```

FIGURE 7. Chain security algorithm.

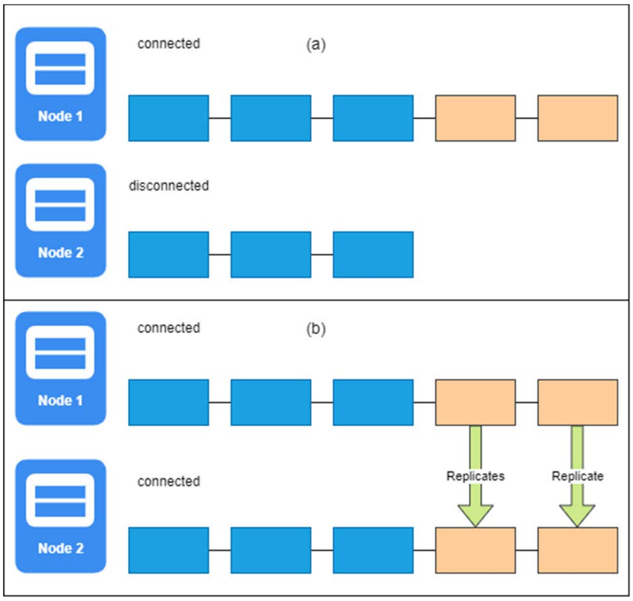


FIGURE 8. Chain recovery during partial failure of the network (a): Node 2 is disconnected from network, while Node 1 is mining normally. (b) Node 2 joins network and replicate the blocks from Node 1, as the Node 1 is the longest chain in network.

In case of any technical failure, such as internet breakdown or power shutdown in an area. The system manages partial failure of the network as shown in **Fig. 8**. The affected Node 2 is unable to continue the mining process due disconnection from network and during disconnection all other connected nodes keep mining the blocks as shown in **Fig. 8 a**). When the affected Node 2 is connected again with the network, the longer chain wins as per blockchain rule i.e. Node 1 is considered the valid chain and replicated on the affected Node 2.

D. TRANSACTIONS IN VMS (UTXO)

This section explains the concept of Unspend Transaction Outputs (UTXO) in the proposed system. As discussed in Section IV part c, each voter is given one VC in his wallet at the time of registering. Voters can spend it only once while voting for candidates. Voting transactions in VMS are performed using the UTXO mechanism. In **Fig. 9** there are several transactions by different voters to candidates. While performing each transaction 1 VC is spent. Miner gets this VC as a reward. There is no transaction fee in the voting system as system keep the rights of all voters equal. Voter1 spends 1 VC to cast the vote for Candidate2. This transaction is recorded and forwarded to the memPool of transactions. From there, miners pick up multiple transactions and start to mine. Here the value of UTXO after each transaction is updated to zero, ensuring that a voter cannot vote again.

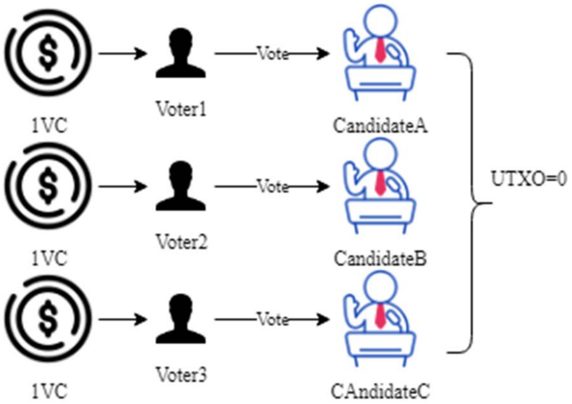


FIGURE 9. **Transaction in VMS. (UTXO.)**

E. CRYPTOGRAPHIC HASH

Cryptographic hash keeps the data hidden from any intruder in the system; it has multiple benefits that include the privacy of the user's identity. Cryptographic hash uses encryption to keep the transaction secure when it is transmitted on the network to be added to a node, only the authorized owner of the transaction can decrypt the transaction and view the content using his private key. The tracking of the user's vote is made possible by providing the voter with the address of his transaction; as soon as the vote is cast the voter is notified through SMS and email. Voters can track their vote in blockchain through the hash value of the transaction that is provided on the registered phone number. The voting data includes all the information saved while casting the vote. The data remains secured, unharmed, and hidden. The only person with the tracking information is the voter, who can view and verify his voting information. The transaction is saved in a block and locked using the public key of the voter. While tracking the vote, the node is identified by the voter's public key. The voter uses his private key to view the transaction made by his wallet. Voters can only view the vote; they can never change or delete the vote once it is cast. Any user information being transferred in a transaction is encrypted by cryptography. The process of casting vote in VMS is further elaborated in Fig. 10. It shows that while casting vote the voter adds a digital signature to the transaction. This digital signature keeps the transaction secure.

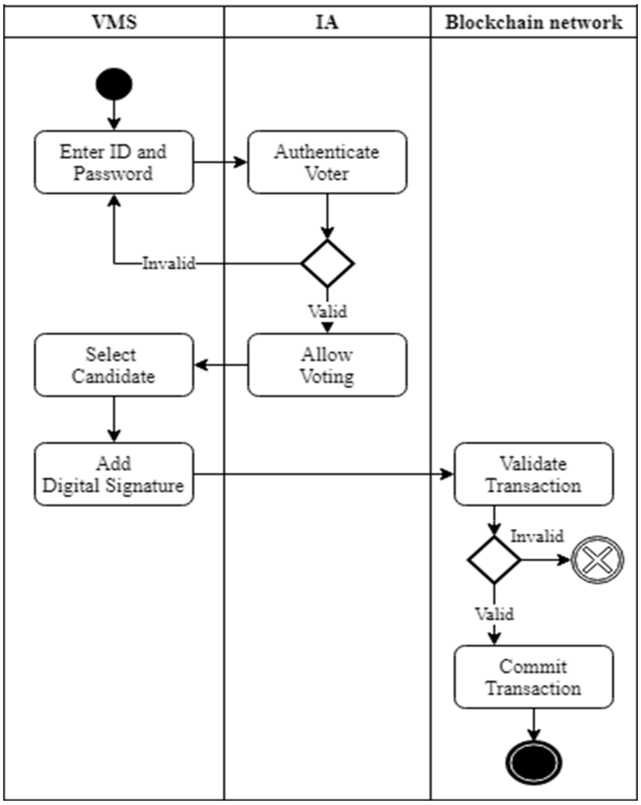


FIGURE 10. **Process of transaction.**

A miner in blockchain can only change the nonce value of the transaction, as all the data is encrypted by using a hash. **Fig. 11** shows an example of a block added to the chain in VMS. All the voting transactions are placed in chain. The nonce value is the one that the miner can change, to mine the block in the chain. All other values are not editable by a miner. **Fig. 11** elaborates the block data in detail but in a real block, this data is stored in the form of hexadecimal. The hash value is the tracking address of this block given to the voter to verify his vote. Miner tries to reach the golden nonce in order to add the block in chain.

Integrity and confidentiality are key features of blockchain technology. The system uses signatures to lock and unlock the messages in the blockchain. Hence, only the voter is authenticated to view the message. The private key and message of the user create a signature. A function called signing algorithm takes the private key and Vote (message) of the voter to create

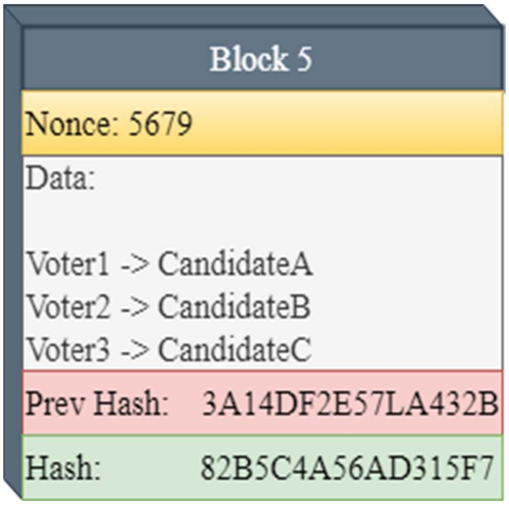


FIGURE 11. A block in VMS.

a signature of that voter. The signature along with the message and public key of the voter is stored in the vote chain as shown in **Fig. 12**.

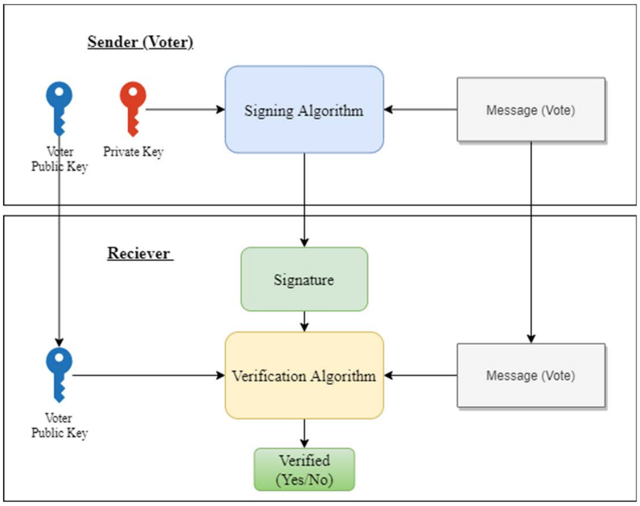


FIGURE 12. Signature.

F. CHAIN DEPLOYMENT

This section elaborates the chain deployment. The voter casts the vote in the system, the voting transaction is mined into the blockchain. Each successful transaction in the blockchain is notified to the voter via registered phone number. The voter can verify his vote by using the hash value of the transaction. As this method of changing the status of vote transactions on chain overcomes the issue of the partial process transactions, the chain only computes the voting result by processing the successful transactions, as only those votes are counted. The delay of 10 minutes has been decided to tackle the transactions load on blockchain. After voting the election authorities can view the votes result on the dashboard of the VMS. The dashboard shows the details of registered voters, qualified candidates, and election statistics. A flaw of the system can be the low literacy rate of any country, but it can be resolved by providing guidelines through media and by keeping the interface as simple as possible. The system also defines multi-language interface that can easily be used by native language voters. The voting system is secured and the only risk is 51% monopoly of hash rate by a miner or group of miners of the voting system.

G. PREVENTING ATTACK 51%

The 51% attack means monopolizing the 51% hash rate of the blockchain. To achieve such a level of hash rate very high computational power is required. In VMS 51% of attack is prevented by two solutions. Firstly, the hash rate of each miner is pre-recorded in the system. Hash rates are monitored to check if any miner does not equip a 51% hash rate in a blockchain system. If such a user is detected, he is not allowed to mine during the voting process. Secondly, miners are pre-selected for the voting process. Miners are hired to mine during the voting process, under the supervision of authorities. Any external group of miners is not0 allowed during voting as elaborated in **Fig. 13**. The system is being monitored constantly while voting activity, the network blocks any malicious miner to mine block in the chain. If pre-selected miners leave the mem-pool while voting activity, the system detects the activity and does not allow those miners to re-join the chain. To keep it fully secure and tamper-proof, if any malicious activity occurs or a miner tries to add a false block in the chain that is not verified, a 51% attack is declared and the process of mining activity is stopped. Nodes with larger chains win and monopolize the network. The blockchain is permissioned and nodes cannot join the network without the permission of the Election Authority. At the run time a group of malicious nodes cannot join without verification of credentials and permission. Hence in proposed solution 51% attack is not possible.

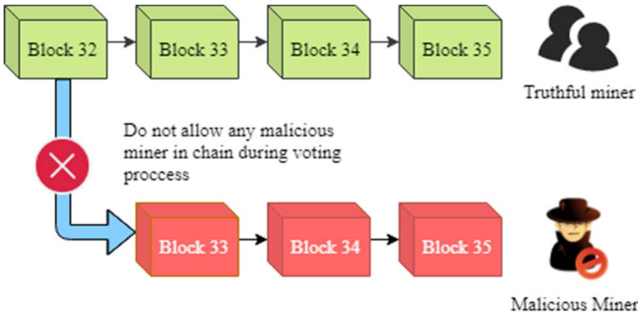


FIGURE 13. Prevention of 51% attack in VMS.

V. PERFORMANCE EVALUATION

This part contains the performance evaluation of the proposed framework; data is taken with real-life scenarios using Remix a browser-based Blockchain tool. Solidity is used as the programming language in the experimentation of the proposed model.

A. RESPONSE TIME OF VMS

The response time of VMS can be determined by calculating the time taken to execute each transaction in the blockchain. **Fig. 14** shows the response time of the system. It shows the increase in response time of the VMS as the votes are increased. The blockchain is decentralized hence each node responds according to its commuting power.

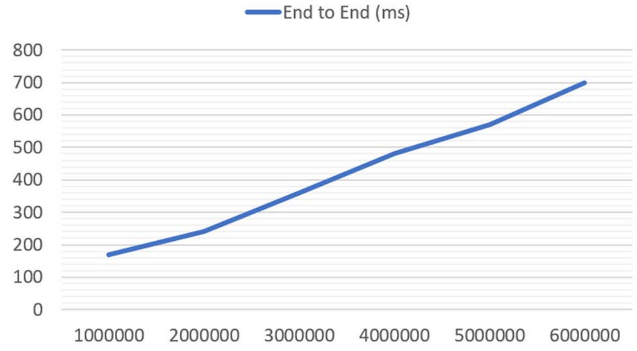


FIGURE 14. Response time of VMS.

The increasing number of votes in the system shows a uniform increment in the size of the blockchain. Response time is increased gradually as more nodes are waiting in the memPool of miners. Flexible Consensus algorithms are applied here to keep the efficiency of the chain uniform.

B. SIZE OF CHAIN

The chain size is around 0.004MB to 0.2MB for every 500 Blocks. The size keeps on increasing with the addition of Blocks in the chain. In **Fig. 15** uniform increment in the size of the chain is shown as the blocks are being added over time.

The chain consists of zero blocks at initially, but eventually, it increases in size because the process of voting is usually 1-2 days. Hence latency can be an issue for a very readily growing chain. The size of the chain increases slowly when the voting process starts; after the intense voter traffic of transactions in memPool the voters start to mine the transactions in the blockchain, once the mining process starts, the size of the chain increases gradually. After successfully mining all votes in blockchain the tallying process starts.

C. LATENCY

The latency of the system is analyzed by measuring the time taken to execute the transaction request. The transaction request is triggered at a respected time that gives the latency rate of the system. As shown in **Table 3** It may increase by an increase in number. of requests per second. Tallying of votes is being held on different nodes of the system during the voting process, it makes it very efficient and reduces any further processing at a singular node. Intruder nodes are blocked by using smart contract algorithms, these algorithms don't allow any incomplete transaction to hit the blockchain. The voter is only allowed to vote when all the relevant smart contracts are fulfilled. Hence, the traffic on-chain is controlled during the voting process. Latency in the chain depends on the computing power of nodes; flexible consensus algorithms keep the blockchain efficient throughout the voting activity. The transaction inclusion time in the blockchain is fixed, it reduces the system's latency.

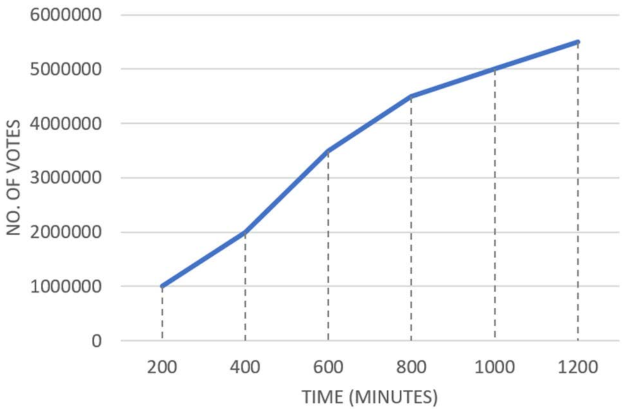


FIGURE 15. Behavior of size of chain.

TABLE 3. Latency in VMS.

Number of Blocks	Latency (ms)
1000000	170
2000000	254
3000000	545
4000000	670
5000000	701
6000000	852

VI. CONCLUSION

The purpose of proposing a blockchain-based solution for the voting system was to build trust between government and voters to make-believe that their voting integrity is kept safe. The blockchain-based voting is also make the voting process transparent and trustworthy. The amount of money spent on voting activity in any country is very high for the traditional voting system, whereas the proposed solution for using the blockchain voting systems to make the voting process cheaper, faster and trustworthy. It helps to enhance people's relations with their democratic state, as they get a transparent system on which they can rely and trust. The framework elaborates on the feature, services and role of official authorities using blockchain in the voting system which is highly in need to improve the level of the electoral system and its reliability, traceability and trust. The verification of each vote makes it immutable. The use of hash assures the privacy of voters and the concept of public and private keys allows the authorities to control the process precisely. The traceability of the voting system assists in preventing hackers from modifying or viewing the voting information. It assures that one voter only votes one vote. The usability of this system performs well by using the more effective approach of implementing a flexible consensus algorithm to reduce extensive computing resources in the blockchain. This transparent behavior of the system tends to be promising for voters to rely and trust. The Chain Security Algorithm is also added, which automatically verifies the validity of the chain each time a new block is added to it. Smart Contracts play an important role to prevent any incomplete and malicious transactions in the blockchain voting system.

The proposed system is a secure, transparent, and reliable platform for the authorities, and voters. The proposed framework has a promising output based on the performance evaluation of blockchain technology in VMS. The experiment shows that the system keeps processing efficiently while processing a large number of transactions in the blockchain.

REFERENCES

[1] S. S. Hossain, S. A. Arani, M. T. Rahman, T. Bhuiyan, D. Alam, and M. Zaman, ''E-voting system using blockchain technology,'' in *Proc. 2nd Int. Conf. Blockchain Technol. Appl.*, Dec. 2019, pp. 113–117, doi: 10.1145/3376044.3376062.

[2] B. Shahzad and J. Crowcroft, ''Trustworthy electronic voting using adjusted blockchain technology,'' *IEEE Access*, vol. 7, pp. 24477–24488, 2019, doi: 10.1109/ACCESS.2019.2895670.

[3] F. P. Hjálmarsson, G. K. Hreiòarsson, M. Hamdaqa, and G. Hjálmtýsson, ''Blockchain-based E-Voting system,'' in *Proc. IEEE 11th Int. Conf. Cloud Comput. (CLOUD)*, Jul. 2018, pp. 983–986.

[4] M. S. Farooq, M. Khan, and A. Abid, "A framework to make charity collection transparent and auditable using blockchain technology," *Comput. Electr. Eng.*, vol. 83, May 2020, Art. no. 106588, doi: 10.1016/j.compeleceng.2020.106588.

[5] N. M. Crosby, P. Pattanayak, S. Verma, and V. Kalyanaraman, ''Blockchain technology—Beyond bitcoin,'' Sutardja Center Entrepreneurship Technol., Univ. California, Berkeley, CA, USA, Tech. Rep., Oct. 2015. Accessed: Jan. 24, 2018. [Online]. Available: http://scet.berkeley.edu/wp-content/uploads/BlockchainPaper.pdf

[6] T. Dimitriou, ''Efficient, coercion-free and universally verifiable blockchain-based voting,'' *Comput. Netw.*, vol. 174, Jun. 2020, Art. no. 107234, doi: 10.1016/j.comnet.2020.107234.

[7] S. Shah, Q. Kanchwala, and H. Mi. (2016). *Block Chain Voting System*. Economist. [Online]. Available: https://www.economist.com/sites/default/files/northeastern.pdf

[8] S. Park, M. Specter, N. Narula, and R. L. Rivest, "Going from bad to worse: From internet voting to blockchain voting," *J. Cybersecurity*, vol. 7, no. 1, pp. 1–15, Feb. 2021, doi: 10.1093/cybsec/tyaa025.

[9] K. M. Khan, J. Arshad, and M. M. Khan, ''Secure digital voting system based on blockchain technology,'' *Int. J. Electron. Government Res.*, vol. 14, no. 1, pp. 53–62, Jan. 2018, doi: 10.4018/IJEGR.2018010103.

[10] C. K. Adiputra, R. Hjort, and H. Sato, "A proposal of blockchain-based electronic voting system," in *Proc. 2nd World Conf. Smart Trends Syst., Secur. Sustainability (WorldS)*, Oct. 2018, pp. 22–27, doi: 10.1109/WorldS4.2018.8611593.

[11] A. Barnes, C. Brake, and T. Perry. *Digital Voting with the use of Blockchain Technology Team Plymouth Pioneers-Plymouth University*. Accessed: Feb. 14, 2022. [Online]. Available: https://www.economist.com/sites/default/files/plymouth.pdf

[12] J. Huang, D. He, M. S. Obaidat, P. Vijayakumar, M. Luo, and K.-K.-R. Choo, ''The application of the blockchain technology in voting systems: A review,'' *ACM Comput. Surv.*, vol. 54, no. 3, pp. 1–28, Apr. 2022, doi: 10.1145/3439725.

[13] F. Fusco, M. I. Lunesu, F. E. Pani, and A. Pinna, "Crypto-voting, a blockchain based e-voting system," in *Proc. 10th Int. Joint Conf. Knowl. Discovery, Knowl. Eng. Knowl. Manage.*, 2018, pp. 223–227, doi: 10.5220/0006962102230227.

[14] G. Rathee, R. Iqbal, O. Waqar, and A. K. Bashir, "On the design and implementation of a blockchain enabled E-Voting application within IoT-oriented smart cities," *IEEE Access*, vol. 9, pp. 34165–34176, 2021, doi: 10.1109/ACCESS.2021.3061411.

[15] M. Pawlak, A. Poniszewska-Marańda, and N. Kryvinska, ''Towards the intelligent agents for blockchain e-voting system,'' *Proc. Comput. Sci.*, vol. 141, pp. 239–246, Jan. 2018, doi: 10.1016/j.procs.2018.10.177.

[16] D. Chaum, A. Essex, R. Carback, J. Clark, S. Popoveniuc, A. Sherman, and P. Vora, ''E-voting 40 scantegrity: End-to-end voter-verifiable optical-scan voting,'' *IEEE Secur. Privacy*, vol. 6, no. 3, pp. 40–46, May 2008. Accessed: Feb. 14, 2021. [Online]. Available: https://www.computer.org/security/

[17] P. McCorry, S. Shahandashti, and F. Hao, "A smart contract for boardroom voting with maximum voter privacy," in *Financial Cryptography and Data Security*. Sliema, Malta: Springer, 2017, pp. 357–375, doi: 10.1007/978-3-319-70972-7_20.