IJNRD.ORG ISSN : 2456-4184
INTERNATIONAL JOURNAL OF NOVEL RESEARCH AND DEVELOPMENT (IJNRD) | IJNRD.ORG
An International Open Access, Peer-reviewed, Refereed Journal

Using Blockchain Method For Transparent E-Voting Systems

Vinod Kumar Gupta
Research Scholar, Computer Science, Govt R.G.P.G. College, Ambikapur, India

ABSTRACT: Voting is considered one of the most critical actions for proper decision making over mental entities, boards of directors, and the financial sector. Many researchers proposed E-voting systems where the voting process is done online or through secure voting stations with high levels of trust for recording and counting the votes. Moreover, with the recent pandemic highlighting the need for remote voting, transitioning to E-voting is becoming even more critical. One way to build such technology is through Blockchain, which can be employed to guarantee the voting system requirements such as reliability, anonymity, decentralization, and privacy. This work proposesa blockchain-based E-voting system that consists of a distributed architecture for the voter, intermediate servers, and blockchain network components. The intermediate servers are mainly used to balance the work load between the voters and the blockchain servers. The system servers apply a scheduling algorithm to distribute the workload amongst themselves. In addition to the distributed architecture, a new algorithm for storing the blocks within each server's data base is introduced. Emphasis is further placed on how these block sare broadcast to the other servers.Simulation results show a clear difference in execution time when comparing the proposed distributed architecture with the centralized system. In addition, the new proposed blockchain algorithm shows better results in executing the vote-counting task and identifying any corrupted blocks.

KEYWORDS: Blockchain, Electronic, Voting, Distributed, Centralized,

1. INTRODUCTION

Voting for essential aspect of the modern world, where leader- ship positions are filled based on the choice of the voting community. The traditional form of voting is done manually by visiting the election centers and choosing the preferred candidate using election forms to identify the selected candidate. However,intoday'slife,manyfieldsof E-government have been developed, including voting. Because of there- centsituation that arose from the COVID-19 pandemic and it simpacton the world, many service fields have moved toward remote services, such as education, business, and banking sectors. In addition, some countries worldwide were forced to postpone their elections to avoid spreading the virus [1].

E-voting introduces a considerable shift in running elections due to eliminating human errors, reducing the time to count the votes, and its ease to the constituents in tending to vote. On the other hand, protective measures are needed to limit the chance of voter fraud or miscalculation of the votes. One

approach is to use blockchain in ensuring the proper implementation of e-voting systems [2].
Preserving the privacy of votes is quite critical when dealing with E-voting systems[3].E-voting systems are developed as online, web,desktop, or smartphone applications. Many systems were proposed to solve the challenges that may be found, such as keeping the identity of the voters anonymous, avoiding data tampering, and voters' ability to note their votes after casting them to guarantee the truthfulness of the system. The system must also guarantee different security metrics such as confidentiality, integrity, availability, transparency, and privacy [4]. Estonia was the first country to use an online voting system in 2005[5].
Satoshi Nakamoto first in vented block chain technology in 2008[6]. This technology eliminates network centralization that causes the sin- gle point of failure problem. A blockchain network consists of a set of distributed nodes connected to each other. The technology principle is summarized in storing participating nodes' data in consecutive blocks as a chain that forms a distributed ledger. The added blocks could be available to any participant but cannot be tampered with.

A block chain starts with the genesis block with no parent, and the next blocks are connected using the previous block hash value. Thekey characteristics that blockchain technology provides are decentralization, immutability, persistence, anonymity, and audibility. In addition, because of the diverse world of applications, different types of blockchain systems exist: public, private, and hybrid. These types differ in allowed permissions, system efficiency, and ownership .The applications are built based on each one's needs and goals. Many application fields were produced based on blockchain technology and its features, including finance, IoT, security and privacy applications, reputation sys- tems, and public and social services such as education and voting sys- tems[4].
E-voting systems are pronetomany cyber threats,such as data tampering, Distributed Denial Of Service (DDoS) attacks, spoofing, identity theft, etc. As possible solutions for these threats, developers proposed systems with security capabilities in mind, such as encrypting and hashing the data, using secure databases to check the voters, applying finger- prints within the system, or, more recently, building blockchain-based E-voting systems [7].
Protecting the exchanged data between two parties needs either a secure channel, which may not always be available, or applying dif- ferent security algorithms on the data to make it hard to be known. These algorithms include encryption, hashing, signing, and authentication. Encryption algorithm shave two forms:symmetric and asymmetric. In symmetric schemes, the encryption key is the same for the sender and receiver, such as the Advanced Encryption Standard(AES) and Data Encryption Standard(DES) algorithms. Asymmetric schemes, on the other hand, depend on public and private key pairs such as RSA and Elliptic Curve Cryptography(ECC). The public key isused to encrypt the data, and the private key isused to decryptit. Symmetric algorithms are faster than asymmetric, but they suffer from repudiation problems. Asymmetric algorithms are slower because they need complicated computations and are normally used for key exchange [8].
Hash function sare one-way functions such as SHA-256andareused to generate the Message Authentication Code(MAC) that cannot be altered or fabricated. Th evalue of the MAC depends on the original message and the secret key between the communicating parties, which results in a unique hash value.The MAC is sent with the original data to guarantee integrity by recomputing the hash value and comparing it with the received one. Signing algorithms add the sender's signature tothesentdatatoauthenticatehim/her.Datasigningisdoneusing
thesender'sprivatekey,andtheverificationisdoneusingthesender's public key, such as Elliptic Curve Digital Signature Algorithm (ECDSA), Digital Signature Algorithm (DSA), and RSA [9].
Load balancers distribute and forward then etwork traffic among different hardware or software resources, mainly decreasing the response time and increasing the throughput. Many load-balancing

algorithms, such as round-robin or a few connections, could be applied to achieve the balancing requirements. This technique helps improve system per- formance and increase scalability, availability, and other benefits [10]. AlthoughintegratingblockchainwithE-votingsystemssolvesmanyis- sues, as mentioned earlier, it still needs limitations and shortcomings, such as the scalability problem, represented by how many voters can participateorhowmanyvotescanbehandledpromptly.Moreover,the problem of the single point of failure exists due to the use of central management or the controller server,which is solely responsible forhan- dling the entire load.If this central server failsor is attacked,the entire systemwillberendereduseless.ThisriskisshowntheE-votingsys- tem in [11], where the authors used an intermediate server to manage the connection and authentication between the users and the blockchain network. Similarly, the system in[12]includesanelection administrator to control the overall process, and the system in [10]includes a man- agement server and counting server to manage the election and votes countingprocesses,inadditiontomanyothersystemsthatuseacentral server for management process.

Blockchain applications can bebuiltonpublicplatforms,suchas Ethereum, or private (permissioned) platforms, such as Hyperledger. Some disadvantages were discovered in Ethereum, such as its lack of clear documentation, complicated communication and interaction with the platform, and a vulnerability of missing, incomplete, or imperfect applications. Moreover, it has been shown that the platform has some centralizedfunctionsorelements[1].RegardingHyperledger,theauthors of [13]proposed a static analysis of this platform to discover its risks. Most of these risks were because of the programming languages, suchastheriskofhandlingtheerrorsandcheckingtheinputargument, in addition to other risks mentioned in their work. Another drawbackof the permission type is compromised security if a set of participants agree to alter the stored data, which compromises the whole network's integrity and the built system's controllability, censorship, and regula- tion levels.

ThisresearchaimstodesignanE-votingsystemthatconsistsofdis- tributed intermediate servers instead of central servers, as in existing systems [13–15]. This model's benefit is eliminating any central point of failure and preventing DoS/DDoS attacks. Another aim of this work is to improve the system's scalability and performance, including the blockchain network side. Moreover, the proposed model guarantees load balancing between the existing servers in the system.

The proposed distributed approach uses multiple intermediate servers to divide the system workload among them without a single point of failure. These servers are located between the voters' clients and the blockchain servers. Using these servers improves the system's scalability,whereitcanserveadditionalusersinatimeunit.Itimproves the throughput and decreases the service latency. The proposed system isbuiltusingthePythonprogramminglanguage,whichissimple,open-source, and portable.

The rest of the paper is organized as follows. Section 2presents a review of the existing research efforts related to both centralized and distributed E-voting systems. A discussion of the proposed E-voting sys-temofthisworkfollowsthis.Next,apresentationanddiscussionofthe results of implementing the system are given in section 4. A security analysis of the system and the legal issues of the E-voting system are giveninsection5,whichalsoprovidesacomparisonwiththecurrently existing systems. The paper is concluded in section 6.

2. RELATED WORK

This section discusses as to E-voting systems with a centralized ar- chitecture, including those with a centralized intermediate server, cen- tralized authority entity, or a management or controller entity. Theworkof[14]proposed a voting system mode lt oprovidea secure environment using a blockchain-distributedledger.Thesystemconsists of three main parts: a user web application, a blockchain network, and an intermediate server to manage the connection and authentication between the users and the blockchain network. The server includes a data base to storetheuserswhomayvote,anadminwillelectandpubli- cizesomeneededinformationforthevotingprocess,andtheblockchain nodesweredesignedtobemutuallyauthenticatedtoeachotherandthe server. Blockchain networks consist of validator nodes and participant nodes, where the validator nodes are used to validate the participant nodes and manage their joining the network. To participate in the vot- ing process, the user creates an account and casts the vote, each in a separate transaction. The vote is rejected if the selected candidate does not exist and no new block is created. The voting process is controlled within a specific duration, after which the final result can be extracted from any random blockchain node. However, the intermediate server forms a single point of failure since it controls the overall management and may be exposed by DOS attacks.

The authors of [16]proposed a blockchain E-voting system frame- worktoprovideliquiddemocracy.Theyfurtherintroducedsomeessen- tial requirements that should exist in any voting system. They used the permissioned blockchain, which depends on proof of authority (POA). Thesystemincludesanelectionadministratortocontroltheoverallpro- cess, voters who can vote, district nodes to verify the vote data to be added to the chain, and a bootnode to improve the network connectiv- ity by helping the district nodes discover each other. The transaction does not include a æfromg field to maintain identity anonymity in vot- ing. Two-factor authentication is used to identify the voters based on theirIDandPINnumber.Theauthorsusedthreeframeworkstocom pare their work: Exonum, Quorum, and Geth. One interesting featureof the system is that users can print their vote ID as a QR code. In this system, the boot node forms a single point of failure since the network connectivity depends on it.

Ablockchainvotingsystemwasproposedin[17].Thisworkadded confidentiality by encrypting the data content to increase the privacy andsecurityofthevoters.Thesystemconsistsofamanagementserver, acountingserver,andthevoter'sapplication.Eachvoterhaspublicand private keys to sign and verify their transactions. The counting server has publicand private keys for encrypting and decrypting transactions. Themanagementserverdirectlyconnectswithvotersandisresponsible forvoters'andcandidates'authentication.Thisisdonebyrunningareg- istrationphaseandexchangingencryptedandsignedmessagestocheck security metrics. The votes are encrypted using the counting server's publickeyandcollectedasblocktransactions.Afterblockcreationand validation,itisaddedtothechain.Whenthevotingtimeends,theman- agement server sends the voting data to the counting server to decrypt andcountit.Theproposedsystemachievedcompleteness,confidential- ity, privacy, and fairness. However, the management and the counting serversformasinglepointoffailuresincethewholesystemdependson these two servers and their effect on scalability.

AuditableBlockchainVotingSystem(ABVS)[18]wasproposedas anE- votingsystem.Themodelconsistsofpollingstations,super-nodes, andtrustednodes.Thepollingstationsworkasvotingapplications.The super-node is the main node in the network and is used to count the votes and save them in the main chain. This chain is copied to other

trustednodestoberedundantincaseofsystemfailure.Thesystemdoes not work remotely since voters must visit polling stations to vote. In addition, the system has been updated to work with multiple agents, suchasthetaskauthorization-configurationagentandthevotingagent, to reduce the single point of failure and distribute the load during the voting phases.

The authors of [19]proposed a blockchain-based voting system model(BBVS)builtmainlytosatisfythevotingneedsinJordanwithin private and centralized environments. The voting process works at two levels:thevoterinthefirstlevelvotestoaspecificgroupandthenvotes in the second level to the candidates he wants from within that group. Thesystembuildstwoblockchains,onetosavethevoterswiththepriv- ilegetovoteandtheothertosavethecandidates.Loggingintothesys- tem uses a special signature related to each voter after signing up. The system has a centralized machine to count and verify the voting results by checking the number of votes at each candidate block. The system was tested with different multi- threading scenarios and sorting algo- rithms.Thebestperformance resultwasforusingmulti-threadingwith synchronization over districts with an insertion sort algorithm. How- ever, the centralized environment and counting machine form a single point of failure and affect the system's scalability.

Theworkof[20]proposedadistributedblockchainE-votingsystem withaGUItointeractwiththesystem.Itfurtherallowsinteractionwith theapplicationserver,arbitrationserver,authenticationserver,andcen- tral database to store voters' information. The authors used the multi- chain blockchain, an open-source blockchain platform, to speed up the blockchain applications and solve some limitations [21]. The system achievesimmutability,transparency,andverifiability.Onedrawbackis that it depends on the full trust of miners and the central authority.This is in addition to using centralized single servers.

Aweb-basedE-voting systemusing the Ethereumplatform waspre- sented in [22]. The system comprises a web server, an intermediate databaseserver,andtheblockchainpart.Accessofthevoters(ordenial of)totheblock chain networkismanagedbyaspecificnodewithinthe blockchainnetworkcalledtheOrangePersonnode.Thesystemachieves auditing and transparency by connecting to the given node. However, the intermediate database servers form a single point of failure.

TrustVote[23], a blockchain-based E-voting, was proposed using two implementations; one using the Ethereum public blockchain plat- formandtheotherwiththeHyperledgerprivateblockchainplatform.

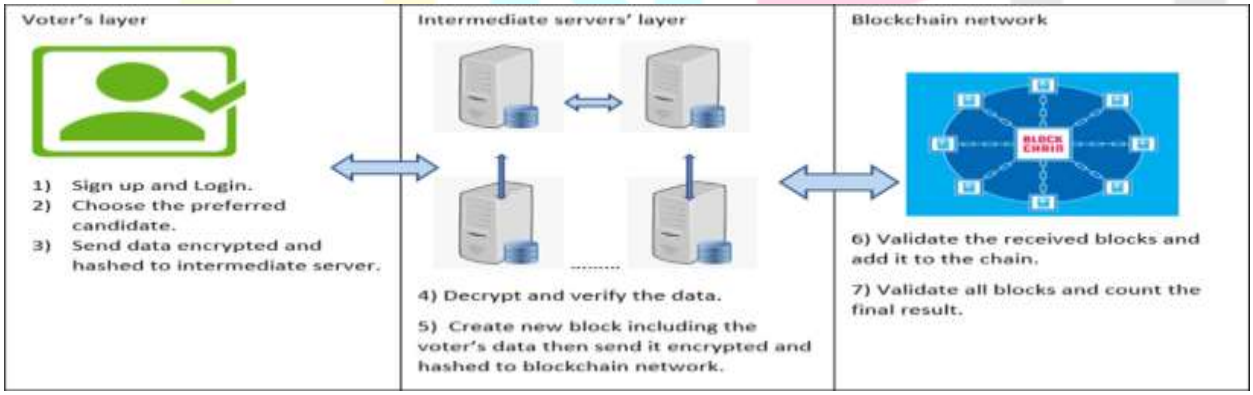


Fig.1. System description.

Thesystemincludesanadministratortoinitiatethesystemprocessiden- tification service to authenticate the users and the blockchain network. The voting is done through district nodes, and verifying and tallyingthe results are done through the managerial nodes. The system sup- portsliquiddemocracy,andHyperledgersupportsbetterperformancein theevaluationstage.However,theidentificationservice formsasingle point of failure, in addition to the limited scalability of this system.

The previous few references are those of a centralized nature. In what follows, we cover some distributed systems. The classification is done based on the system architecture, where it is built as tiers or a hi- erarchical structure, or the client side communicates directly with the blockchain servers without any centralized entity.

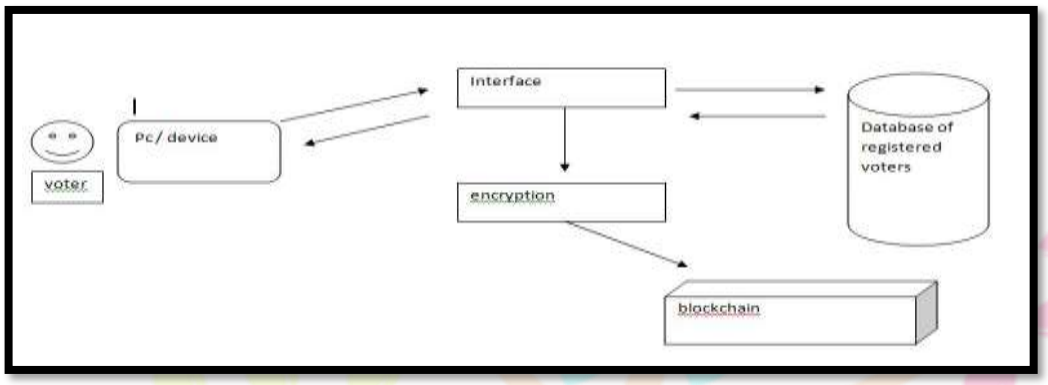


Fig.2 : Blockchain Based Electronic Voting System

In the blockchain E-voting system of [5], the authors designed the working network as tiers, where the voter is restricted from connecting tooneconstituencynodebasedonlocation.Eachconstituencynodehas one public-private key pair that local nodes use for data sharing. How- ever,thewholeconstituencynode'sdataarecompromisedifsuchakey is compromised. Although the system uses tiers, it still has a weakness in the single point of failure at the constituency level.

Inthecaseoftheworkin[24],thevotingnetworkwasdesigned in a hierarchical structure of levels, where the lowest level includes the voting centers and the highest level is used for data mining and adding totheblockchain.Thevotingcentersareintegratedasclustersandsyn- chronizedtogether.Eachclusterofthelowestlevelisconnectedtoone node of the blockchain network, and these clusters are independent. In this work, the helper servers are not used, resulting in less workload distribution and bringing the service closer to the users.

The work of [3]proposed a web application voting system within blockchain technology. To use the application, voters have to pre- registerinthesystem,andtheycanvoteonce.Thevoteissignedusing thevoter'sprivatekey,andtheycancheckiftheirvoteiscounted.The voting network is public; miner nodes create the block for each trans- action(vote)usingproof-of-work(POW),andnormalnodesvalidateit. The system contains a database to store information about the voters and a flag parameter to check when the user has voted. However, the systemdoesnothaveintermediateservers,whichcanaffectitsscalabil- ity.

A TeV framework (TrustedEVoting) was proposed by [2]. In this sys- tem, the vote is

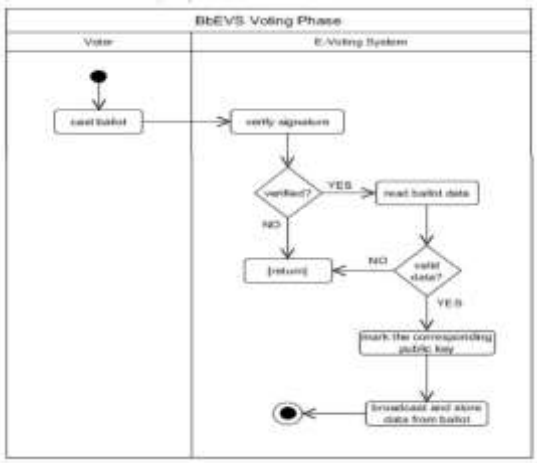


Fig.3 : Diagram of vote casting phase

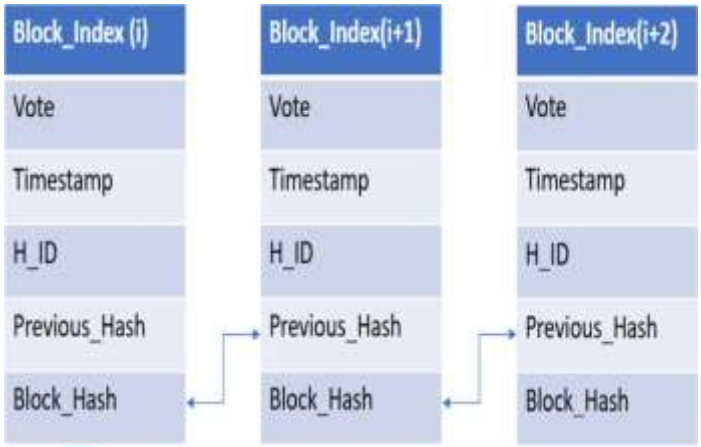


Fig.4. Blocks structure in the proposed model

encrypted using an election public key and signed usingthe voter's private key. Each district has a unique election key; a QR code could be used instead of the user's private key. The voter ID is computed using a hash equation with different parameters to maintain anonymity. The system was built based on (consortium) permissioned blockchain,andithandlesthecaseifthevotervotesmultipletimesandhowtoget the final result for that. The system provides integrity, anonymity, and publicconfidence.However,iftheuniqueelectionkeyatthedistrict level is compromised, then the voting process will fail.

The work in [25]proposed an E-voting system integrated with blockchain technology and uses double-layer encryption to ensure users' privacy and fairness. The system was implemented to work with a private network and Ethereum platform without intermediate servers. The authors applied homomorphic encryption, where the data could be analyzed without decrypting to maintain data privacy. The generated data is divided into fragments and distributed among the nodes in the distributed network. To rebuild the original form, a set of nodes are gatheredtogether.Thesystemachievesreliability,privacy,uniqueness, integrity, and verifiability. However, the absence of the intermediate servers reduces the system's scalability.

Another system was proposed by [26]to be applied in a university environment (SecEVS). The whole environment is divided into zones with unique voting blockchains. All blockchains are joined together to formaglobalbl ockchain.Thevote sareencryptedusingtheuniversity's publickeyandsignedusingthevoter'sprivatekey.Thesystemachieves voters'privacy,confidentiali ty,andeligibility.Inthissystem,allvoting data will be compromised if a universityiscompromised.Thesystemis centralized at the zone level.

3. PROPOSED SYSTEM (METHODOLOGY)

The design of the proposed system is intended to be as simple as possible,makingthedeploymentofthevotingsystemastraightforward matter.Wepresenttheimplementationdetailsofourframeworkinthis section.

1.1. *Systemdesign*

The proposed system consists of the voter client, intermediate servers,andblockchainnetworkservers'sides,asshowninFig.1.The voters sign up and log in with their credentials to the system to choose their preferred candidate. The system encrypts, hashes, signs the gen- erated data, and then sends it to one of the intermediate servers. Note that the round-robin schedule algorithm is used for intermediate server selection. A round-robin algorithm aims to achieve equal chances and balance and distribute the workload between these servers.

The intermediate server adds a new voter entry during the sign up processandthendistributesthisentrytotheotherintermediateservers. It checks for voter credentials while logging in, receives the vote, and verifiesthatthevotecamefromtherightperson.Then,itpreparesanew block carrying the received votes and other specified data and sends it to one of the chosen blockchain servers, which is also chosen using a round-robin schedule algorithm.

The chosen blockchain server receives the newly generated block and checks if it is unique using the hash value stored in the block to ensurethevoterhasnotvotedyet.Itvalidatestheblockbyrecomputing the block hash, comparing it with the stored value, and comparing the previous has hvalues. If verified correctly, it add sthe block to the chain data base and the ndistributes it to otherblockchainserverstoensureall servershavethesameexactdata.Otherwise,theblockmustberejected. Allmessagesbetweentwopartiesinthesystemareencryptedandsigned before sending.

Fig. 2shows the block contents; it includes the block index for the blocksorder,thechosenvote,thetimestampoftheblockgeneration time,H_ID,whichisthehashvalueofthevoterID,previous_hash,which isthehashvalueofthepreviousblock,andBlock_Hashwhichincludes the current block hash value.

Blockchainserverscancountthevotingresultsbyretrievingthedata from their databases. The data entries are validated first to ensure the data integrity by recomputing the block hash and rechecking the previ- oushashforallblocks.Ifallisdonesuccessfully,thevotesarecounted, and the results are given.

The proposed system design introduces a new algorithm for blockchain database creation and synchronization, as shown in Algo- rithm1.Thisalgorithmtakesthenewblock,theblockchainservers,and the receiver blockchain server as input. It then creates a new database with new tables for each blockchain server (if it already exists). Before inserting the new block into the database, it checks its validation using (Block_validation) and then broadcasts it to the other servers after the insertion.

Each server has a table related to it in every database in this algo- rithm.Forexample,server_1willhavesynchronizedtable_1indatabases1⋯n.Thetableincludesanorderedblock,ass howninFig.2,andthe databaseincludesthesetoftablesforeachserver.Whenanewblockis received,theserverchecksthe uniquenessoftheH_IDvalueintable_1 to table_n in its database. If it is unique, the same addition process will be done in table_1, and the same thing will be applied to other servers. This algorithm is simple and better for synchronization; for each ta-ble, just one server adds data, and the others can verify. To the best of our knowledge, this is the first research that has addressed the subject of blockchain database architecture and synchronization and provided differentscenariosforimplementingit.Itmustbementioned,however, that proper synchronization between the servers is critical for the suc- cessofthisoperation.Moreover,theschemeprovidesproperreplayat- tack protection due to the built-in checks for previous blocks of votes.

On the other hand, the conventional algorithm for block-chain databasesynchronizationusesonechainoronetableforallblocksfrom differentservers.Thisalgorithmcreatesonecommontablecopyateach server,andwhenanewblockisreceived,itvalidatesandaddsit.Ithas to recreate a new version of the block with a new previous_hash value and new block_hash value to be compatible with the other tables in the

other servers.

1.2. *Systemimplementation*

A simple and user-friendly graphical interface was built on the Voter side. Through this interface, the voter scan sign upto create anew entry in the voters' database, and then they can log in and choose their preferred candidate.

In the proposed system, we use the SQLite3 database to store the voters' information within the intermediate servers and the voting blockchainwithintheblockchainnetwork.Itishashedwithasaltvalue to keep the user password safe from being known or attacked.

TheproposedsystememploysECCasanasymmetriccryptographic algorithm. It is a public-key-based algorithm that depends on elliptic curves over finite fields. It provides a relatively small key size with a similarsecurityleveltonon-ECCalgorithms.Also,AES-128-CBCisusedasthesymmetricencryptionmethod.Thesecretsharedkeyisexchanged using the Elliptic Curve Diffie-Hellman (ECDH) algorithm from the gen-erated ECC public and private key pairs. This key is used by both the sender and receiver for encryption and decryption, respectively. SHA- 256 was adopted as the hashing algorithm to generate the message di- gest of 256-bit to guarantee data integrity over the insecure channels. Furthermore,theEllipticCurveDigitalSignatureAlgorithm(ECDSA) wasusedtosignandverifytheexchangedmessages.ECDSAisadigital signaturealgorithmthatusesthekeysgeneratedfromtheECCalgorithm [27].

Twoalgorithmsweredevelopedforblockcreationandvalidationin this work. Algorithm 2 shows the block creation steps. This algorithm takes the block contents as input. First, it concatenates these contentsasP1,andthenitcomputesthehashvalueofP1togettheBlock_Hash value; then it builds the NewBlock by putting the block index(i), the Vote, the block timestamp, the H_ID, the previous block hash, and the generated block hash.

Algorithm 3 shows the block validation steps. This algorithm takes the block to be validated as input. First, it checks the uniqueness of the H_IDvaluetoensuretheyoterhasnotvotedyet.Then,itrecomputesthe blockhashvalueandcomparesitwiththestoredone.Finally,itchecks thestoredpreviushashvaluewiththeblockhashofthepreviousblock in the chain. If all these conditions are done correctly, the validation is successful, and the block is added to the chain.

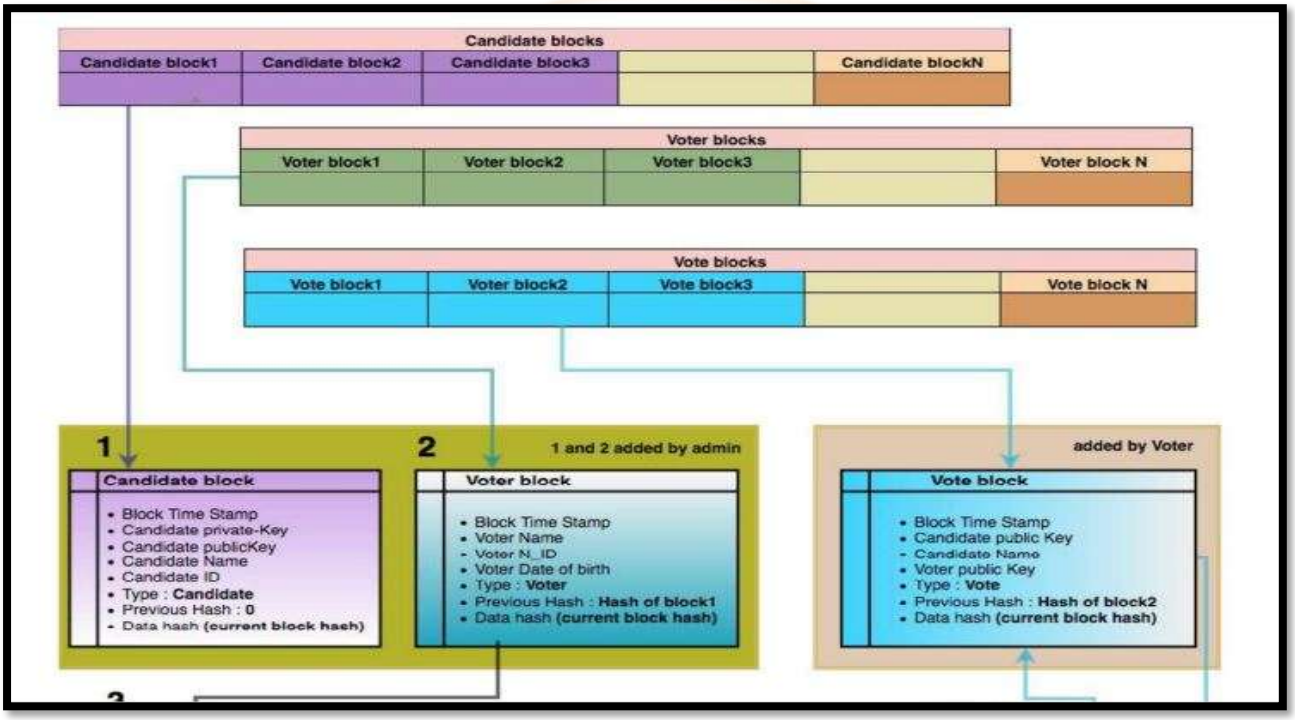


Fig.6 Architecture of the proposed e – voting system

4. RESULTS AND EVALUATION

ThissectionpresentstheevaluationresultsoftheproposedE-voting system. The system was implemented as a software application using Python version 3 under CentOS 8 Linux machines. Each server in the systemwasrunonaseparatePCwithidenticalcharacteristics.EachPC has4Gigabytememoryand2.7GHzIntelcorei7CPU.Fig.3illustrates the topology of the proposed distributed E-voting system.

Different scenarios were implemented for performance testing and to improve the system evaluation. The scenarios are shown next and are classified into performance testing, which includes self-evaluation under different cases, and the second part includes security analysis and legal issues.

1.1. *Performance evaluation*

In all testing scenarios, the system consisted of four intermediate servers and four blockchain servers, and the number of voter clients was taken as a variable performance indicator.

1.1.1. *Centralized vs distributed signup*

The first scenario for evaluation is to compare the time needed for thesignupprocess(i.e.,registeringthevoters)anditsimpactonthein-

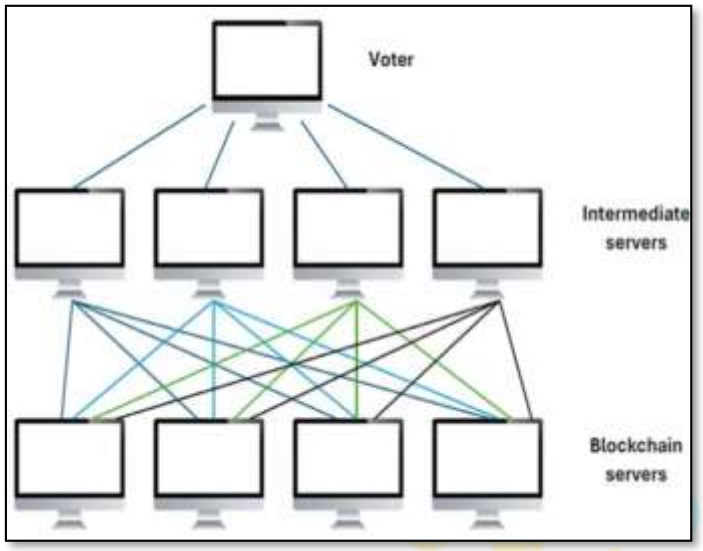
Fig. 4shows the system execution time (user and kernel levels) in seconds from the beginning to the end of the sign up process using a single centralized server and multiple distributed intermediate servers for a variable number of voters/clients connected to the system. The resultsshowacleardifferencebetweenthem,wherethecentralizedim- plementation needs much longer to execute the process than the dis- tributed implementation.

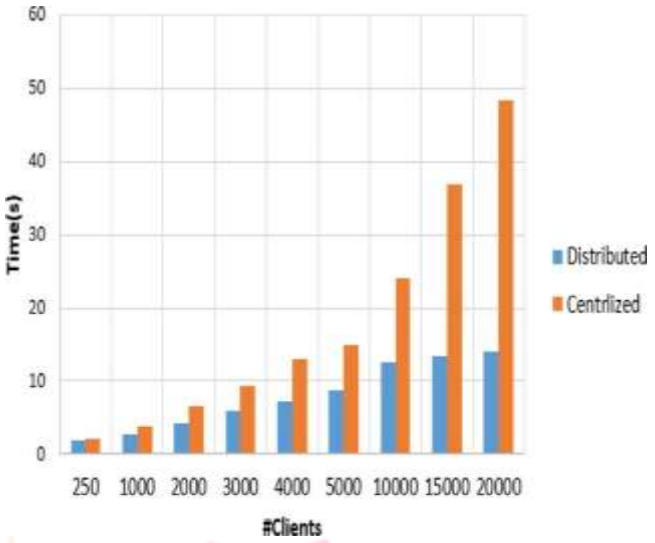
Thetimewasco mputedfordiffer entn umbers ofclient sintherange between 250 and 20,000 clients. As shown, in the centralized case, the timeisi ncrea singex ponentiallyd uetothein creasingnum berofclients. The time needed to use the distributed model is a small fraction of the timenee dedwhenus ingasingl eserver.Thisalsoincludesth eadditional task of distributing the new entry to the other intermediate servers.

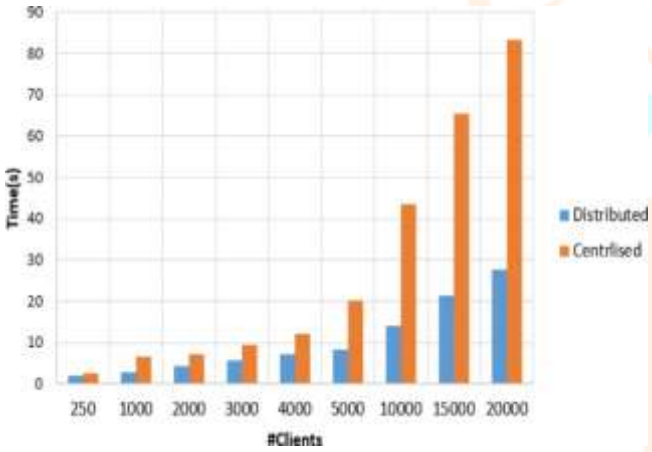
1.1.2. *Centralizedvsdistributedloginandvote*

The second scenario for evaluation is testing the system for login and voting to get the system execution time needed at the intermedi-ate servers. The comparison is done in the case of centralized and dis- tributed intermediate servers.

Fig. 5shows the system execution time (user and kernel levels) for loginandvotingusingdistributed andcentralizedimplementations. The resultsshowthatthedistributedimplementationrequireslesstimethan the centralized implementation. The results are related to the load at each server, wherein, in the centralized case, all the load is handled byasingleserver. Ontheotherhand, in the case of distributed servers,the







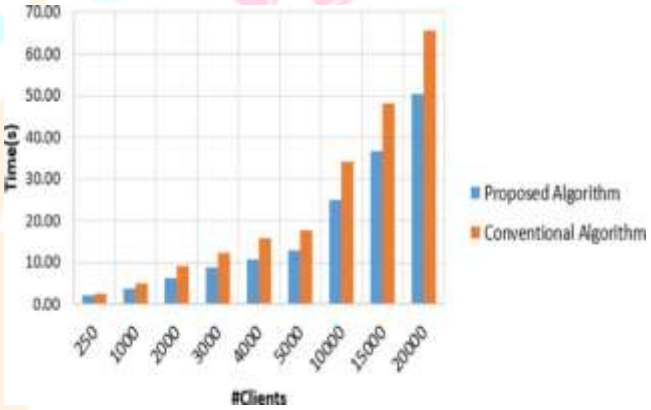


Fig.7 .Topology of the proposed distributed E-voting system
Fig.8 .Centralized versus distributed signup time

Fig.9. Centralized versus distributed login and voting time.
Fig.10. Time needed for block validation and broadcast

Load is divided among all four servers. As shown, in the centralized case, the time increases exponentially with the number of clients, where the base time value is at one client, then this value increases linearly with the number of clients.

1.1.1. *Blockchain data base synchronization*

The third scenario is testing the system to get the time the block chain servers need for blocks' validation and broadcasting process using the proposed algorithm (Algorithm 1) versus the conventional algorithm that uses one chain for all servers at each database. Recall that the proposedalgorithmbuildsseparatetablesforeachserverineachdatabase, while the conventional algorithm builds a single table in each database at each server.

Fig.6showsthenumberofprocessedvotesateachtestsystemversus the execution time needed in seconds to complete block verification, insertion, and broadcast.

The results show that there is no big difference between the two algorithms. However, the synchronization between the servers and pro- cessesiseasierusingtheproposedalgorithm.Theproposedalgorithm's

required timeis mainlyfor validatingandinsertingthe incomingblock in the other databases. As for the conventional algorithm, on the other hand,thetimeisforvalidatingtheincomingvote,insertingitwithinthe local database, regenerating a new block, and inserting it in the other databases.Theconsumedtimeincreaseswiththenumberofvotestobe validated and broadcast.

1.1.1. *Counting results*

Thefourthscenariowasimplementedtoshowthedifferenceinsys- tem execution time needed to count the voting final results using the proposed and conventional algorithms. Fig. 7shows the system time needed to count the results regarding the number of votes. The process includes retrieving the data from the database, validating all block sequences by checking the previous hash value in each block, computing the block hash again, and counting the votes to each candidate. The times were computed for processing 250 votes up to 20,000 votes. The results show that the proposed algorithm needs less time than the conventional algorithm. The difference becomes clearer when thenumberofvotesincreases.Theresultsarerelatedtohowtheblocks are validated in each database architecture, where in the proposed al- gorithm the database is divided into multiple tables so we can run the algorithm to work at each table in a separate process in parallel. Yet,in the conventional case, the process cannot be separated into multiple processes where each chain sequence is being validated at one time.

1.1.2. *Discoveringerrors*

This scenario was implemented to get the time needed to discovera block error found at different indexes. Fig. 8shows the system time neededtodiscovertheblockerroratindexes(500,1000,2000,4000, 8000,10000,16000,and20000).Asshown,theconventionalalgorithm consumes more time than the proposed algorithm. The reason for thatis the same as that in the fourth scenario.

5. SECURITY ANALYSIS AND LEGAL ISSUES

Regarding security analysis, Table 1compares the proposed sys- tem and other E-voting systems that use blockchain technology. The table compares security features such as privacy, authentication, in- tegrity, confidentiality, anonymity, and the block's verifiability. The proposed system achieves all these features. Although systems in [14,17,28]achieve all these features, they suffer from shortcomings, such as the existence of a single point of failure in [14,17]. The system of[28]focused on applying fuzzylogic ratherthan the system architecture issues.

Moreover, the proposed system resists DDoS Attacks because of the distributed and decentralized architecture. In general, a DDoS would affect the resources of a system completely. However, when the dis- tributed architecture is used, the attack plain becomes wider and the probability of success would be greatly reduced due to the redundancy in the system.

The proposed system is also resistant to the 51% attack because of the permissioned network and balanced load (the unlikely event that a groupwillacquiremorethan50%ofthehashingpowerofablockchainnetwork).Decentralizationlimitstheabili tyofaanysinglepartytohave a majority in the rate of hash computations. Furthermore, the redun- dancy in the existing databases serves to offer consensus to mitigate such an attack.

Finally, the SQL injection attack does not succeed since the system rejects the malicious SQL queries that steal the users' information.

As for the legal issues, Table2 shows acomparison between the pro- posed system and other E-voting systems that use block chain technology. Thetableshowsfeaturesrelatedtothearchitectureofthesystems, such as the case of the intermediate servers, the use of load balancing, theexistenceofasinglepointoffailure,thelevelofsystemscalability, user eligibility, and the case of working remotely.

As presented in the table, the proposed system achieves a fully distributed architecture with no single point of failure. The proposed sys- tem applies a load-balancing algorithm and provides strong scalability becauseofthedistributedarchitecture.Inthecaseoftheothersystems, someemployasingleintermediateserverorcompletelyremoveitfrom the architecture, affecting the system scalability and the single point of failure.

Most reviewed systems do not have the load balancing feature except the propose done and the system in[18]because of the multi-agents that help balance the load.

Table 1
Security features comparison.

Feature	[14]	[16]	[17]	[5]	[24]	[28]	[3]	[2]	[18]	Proposed system
Privacy	✓	X	✓	✓	✓	✓	✓	X	X	✓
Authentication	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
Integrity	✓	X	✓	X	X	✓	X	✓	X	✓
Confidentiality	✓	X	✓	✓	X	✓	X	✓	X	✓
Anonymity	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
Verifiability of blocks	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓

Table 2
Legal issues comparison.

Feature	[14]	[16]	[17]	[5]	[24]	[28]	[3]	[2]	[18]	Proposed system
Intermediate servers	Centralized	Not used	Centralized	Partially distributed	Not used	Not used	Not used	Partially distributed	Partially distributed	Fully distributed
Load balancing	X	X	X	X	X	X	X	X	✓	✓
No single point of failure	X	X	X	X	X	X	✓	X	X	✓
Scalability	Limited	Limited	Limited	Medium	Medium	Limited	Limited	Medium	Medium	Strong
Eligibility	✓	✓	✓	✓	✓	✓	X	✓	✓	✓
Works remotely	✓	X	✓	✓	X	✓	✓	✓	X	✓

Table 1 &2
Security features comparison.

The system includes an administrator to initiate the system process identification service to authenticate the users and the blockchain network. The voting is done through district nodes, and verifying and tallyingthe results are done through the managerial nodes. The system sup- portsliquiddemocracy,andHyperledgersupportsbetterperformancein theevaluationstage.However,theidentificationservice formsasingle point of failure, in addition to the limited scalability of this system.

The previous few references are those of a centralized nature. In what follows, we cover some distributed systems. The classification is done based on the system architecture, where it is built as tiers or a hi- erarchical structure, or the client side communicates directly with the blockchain servers without any centralized entity.

In the blockchain E-voting system of [5], the authors designed the working network as tiers, where the voter is restricted from connecting tooneconstituencynodebasedonlocation.Eachconstituencynodehas one public-private key pair that local nodes use for data sharing. How- ever,thewholeconstituencynode'sdataarecompromisedifsuchakey is compromised. Although the system uses tiers, it still has a weakness in the single point of failure at the constituency level.

Inthecaseoftheworkin[24],thevotingnetworkwasdesigned in a hierarchical structure of levels, where the lowest level includes the voting centers and the highest level is used for data mining and adding totheblockchain.Thevotingcentersareintegratedasclustersandsyn-

chronizedtogether.Eachclusterofthelowestlevelisconnectedtoone node of the blockchain network, and these clusters are independent. In this work, the helper servers are not used, resulting in less workload distribution and bringing the service closer to the users.

The work of [3]proposed a web application voting system within blockchain technology. To use the application, voters have to pre- registerinthesystem,andtheycanvoteonce.Thevoteissignedusing thevoter'sprivatekey,andtheycancheckiftheirvoteiscounted.The voting network is public; miner nodes create the block for each trans- action(vote)usingproof-of-work(POW),andnormalnodesvalidateit. The system contains a database to store information about the voters and a flag parameter to check when the user has voted. However, the systemdoesnothaveintermediateservers,whichcanaffectitsscalabil- ity.

A TeV framework (TrustedEVoting) was proposed by [2]. In this sys- tem, the vote is encrypted using an election public key and signed using the voter's private key. Each district has a unique election key; a QR code could be used instead of the user's private key. The voter ID is computed using a hash equation with different parameters to maintain anonymity. The system was built based on (consortium) permissioned blockchain,andithandlesthecaseifthevotervotesmultipletimesandhowtoget the final result for that. The system provides integrity, anonymity, and publicconfidence.However,iftheuniqueelectionkeyatthedistrict level is compromised, then the voting process will fail.

The work in [25]proposed an E-voting system integrated with blockchain technology and uses double-layer encryption to ensure users' privacy and fairness. The system was implemented to work with a private network and Ethereum platform without intermediate servers. The authors applied homomorphic encryption, where the data could be analyzed without decrypting to maintain data privacy. The generated data is divided into fragments and distributed among the nodes in the distributed network. To rebuild the original form, a set of nodes are gatheredtogether.Thesystemachievesreliability,privacy,uniqueness, integrity, and verifiability. However, the absence of the intermediate servers reduces the system's scalability.

ANOTHER SYSTEM WAS PROPOSED BY [26]TO BE APPLIED IN A UNIVERSITY ENVIRONMENT (SECEVS). THE WHOLE ENVIRONMENT IS DIVIDED INTO ZONES WITH UNIQUE VOTING BLOCKCHAINS. ALL BLOCKCHAINS ARE JOINED TOGETHER TO FOR MA GLOBAL BLOCKCHAIN.THE VOTES AREENCRYPTED USING THE UNIVERSITY'S PUBLIC KEY AND SIGNE DUSING THE VOTER'SPRIVATEKEY. THE SYSTE MACHIEVES VOTERS'PRIVACY,CONFIDENTIALITY, AND ELIGIBILITY. IN THIS SYSTEM, ALL VOTING DATA WILL BECOM PROMISED IF A UNIVERSITY IS COMPROMISED.THE SYSTEMIS CENTRALIZED AT THE ZONE LEVEL.

6. CONCLUSION AND FUTURE WORK

Securing a country's important event like elections will always be a concern trying to prove the success of such systems for the aim of trust and satisfying citizens for a clean reliable direct online results and privacy. All these features can be covered by using Distributed Blockchain technology that was implemented initially by Satoshi Nakamoto in 2008 for cryptocurrency application known as bitcoin in which electronic exchange of money between two sides is done with no centralization (directly no third party is involved). Since then, Blockchain as a distributed ledger technology took a massive part in the revolution of technology to be applied in many fields of science such as Banking industry, healthcare, contract management , reputable system, digital content distributed system and voting system**.** All meant to achieve the

security it offers due to the hash algorithm that chains block of transactions to the blockchain database and store that data in many places that makes it a hard challenge to break. The mentioned ideas of proposing evoting

systems are still under research as some details may be a future view to deal with details not mentioned such as the size of the ledger that holds the results and how it plays an important role in speeding the throughput time for revealing the final results, also how the huge population may affect the computing process of blockchain.

With the extraordinary growth in the use of blockchain technologies, a number of initiatives have been made to explore the feasibility of using blockchain to aid an effective solution to e-voting. This paper has presented one such effort which leverages benefits of blockchain such as cryptographic foundations and transparency to achieve an effective solution to e-voting. The proposed approach has been implemented with Multichain and in- depth evaluation of approach highlights its effectiveness with respect to achieving fundamental requirements for an e-voting scheme.

In continuation of this work, we are focused at improving the resistance of blockchain technology to 'double spending' problem which will translate as 'double voting' for e-voting systems. Although blockchain technology achieves significant success in detection of malleable change in a transaction however successful demonstration of such events have been achieve which motivates us to investigate it further. To this end, we believe an effective model to establish trustworthy provenance for e-voting systems will be crucial to achieve an end-to-end verifiable e-voting scheme. The work to achieve this is underway in the form of an additional provenance layer to aid the existing block chain based infrastructure

7. ACKNOWLEDGMENT

In the digital era where hacking and bypassing a system is easy, tampering of data is always possible leading to bad situations. Blockchain is used to store data which is near impossible to change or tamper with as it is very secure in nature. Voting as a process in any nation is an essential event and if votes get miscalculated by any external source it will be harmful. To avoid such kinds of situations and making it more comfortable blockchain technology comes in acknowledgment. This paper proposes a decentralized national e-voting system based on blockchain technology. It includes an admin panel to schedule the voting, manage candidates and declare the results. The web application will provide the users with an interface to enter their Aadhar card ID (text input) and a photo of themselves at the time of voting. The eligibility of the voter will be checked at the time they enter their Aadhar card ID. Eligible voter's phone numbers will be verified via One Time Password (OTP). After voter verification, individual voters will be considered eligible for voting. During voting, voters will be monitored through a webcam/front camera. The votes will be stored in a blockchain and any tampering would be detected easily. The address and the corresponding constituency will be checked in the backend. Voting results will be declared on a specified date and will be handled by the admin. The results will be displayed graphically with various options to choose from and will also include past results and statistics.

8. REFERENCES

[1] Jonatan Abraham, Vania Ceccato [2022] 'Assessing the market potential of electric bicycles and ICT for low carbon school travel: a case study in the smart city of Águeda ',Published by Elsevier B.V.(2022) 119–130

[2] Carolina Pontones-Rosa Et. And Al.[2022] 'Integrating water, waste, energy, transport and ICT aspects into the smart city concept'Published by Elsevier Ltd. This is an open access article under the CC BY-NC-ND license (2022) 186 (2017) 609 – 616

[3] Jesús E. Argente García, Alfonso P. Ramallo-González, Jaime Bernardeau-Esteller [2023]

‘ Ipsum – An Approach to Smart Volatile ICT-Infrastructures for Smart Cities and Communities’*Anne Håkansson / Procedia Computer Science 126 (2018) 2107–2116*

[4]Walid Chatti[2021] ‘A study about the relationship between internet usage and social isolation among Iranian students’*Procedia Social and Behavioral Sciences 15 (2011) 394–398*
[5]Doina Stratu-Strelet Et. And Al.[2021] ‘Challenges and Benefits of an Open ICT Architecture for UrbanWater Management’, Procedia Engineering 89 (2014) 1073 – 1079

[6]Francesco Longo Et. And Al.[2023] ‘Society of Knowledge and ICT skills in Universities ’, *Pr ocediaC om puterS cience21 0 (2022) 363–367 rScienc e00(2022)000–00 0*
[7] He GuanLin Et. And Al.[2022] ‘The interaction between technology, business environment, society, and regulation in ICT industries’ , IIMB Management Review (2022) 34, 103–115

[8] Nutteerat Pheeraphan [2021] ‘ ICT industry innovation: Knowledge structure and research agenda’, Technological Forecasting & Social Change 189 (2023) 122361

[9]Daniel Ayisi Nyarko Et. And Al.[2022] ‘Adults Use of ICT in Healthcare: The Persuasive Impact of Children’, Procedia Computer Science 98 (2016) 236 – 242

[10] Elisabete Arsenio, Joana V. Dias and Et. Al.[2016] ‘ ICT-based public policies and depopulation in hollowed-out Spain: A survey analysis on the digital divide and citizen satisfaction’, Technological Forecasting & Social Change 169 (2021) 120811