A Secure Decentralized E-Voting with Blockchain & Smart Contracts

Rohit Kumar
CSE Department
ABES Engineering College
Ghaziabad, UP (India)
rohit.19b101051@abes.ac.in

Lavi Badwal
CSE Department
ABES Engineering College
Ghaziabad, UP (India)
lavi.19b101163@abes.ac.in

Sandhya Avasthi
CSE Department
ABES Engineering College
Ghaziabad, UP (India)
sandhya_avasthi@yahoo.com

Ayushi Prakash
CSE Department
ABES Engineering College
Ghaziabad, UP (India)
ayushi5edu@gmail.com

2023 13th International Conference on Cloud Computing, Data Science & Engineering (Confluence) | 978-1-6654-6263-1/23/$31.00 ©2023 IEEE | DOI: 10.1109/Confluence56041.2023.10048871

Abstract— A democratic election is a crucial act in each nation, as it determines the country's future for a specific term. Some of the older voting methods, such as Ballot Paper and EVM (Electronic Voting Machine), have disadvantages such as lack of transparency, poor voter turnout, vote rigging, and many others. Using Blockchain technology and Smart Contracts, it is simple to circumvent the flaws of the Ballot system and EVM. Electronic Voting Powered by Blockchain and Smart Contracts outperforms these antiquated voting methods by delivering secure results in less time and at a lower cost. With E-Voting utilizing Blockchain, prices can be lowered, the necessity for Polling stations and the consumption of resources such as EVMs and Ballot Papers may be decreased, and security can be improved by offering End-to-End Encryption and authenticity. This blockchain-powered e-voting can readily acquire trust due to the transaction's transparency, immutability, and difficulty of modification once hosted, as a result of smart contracts. Using OTP Verification and face verification, the suggested solution is a MERN-based web application with a multitude of upgraded authentication and permission techniques. To improve security, this voting data is saved as a transaction in a Blockchain-based distributed ledger using smart contracts.

Keywords— Blockchain, Electronic Voting, EVM, End to End Encryption, smart contracts, SHA, Smart Contracts, MVC.

I. INTRODUCTION

In a huge democratic country like India, elections play a vital role. In India, where a major portion of the underprivileged population is illiterate or uneducated, election authorities must examine signatures or thumb impressions on paper ballots to determine the legitimacy of votes. Because they are plagued with mistakes, votes from disadvantaged people are effectively discarded. EVM technology guarantees that these groups participate in elections and that their votes are accurately counted. But EVM faces obstacles. This issue emerges due to Votes Manipulation, Polling Booth Capture, EVM Hacking, and Votes Tempering [1,2,3]. These problems were captured in the traditional way of voting and by the means of this advanced System, we tried to take meals over them. Online Voting is the latest trend comprised of the conduction of election or poll voting that makes the work of voting easier and fast.

When utilized in elections, electronic voting methods must be legal, accurate, safe, and convenient. However, adoption may be hampered by potential issues with computerized voting systems. To solve these concerns, blockchain technology was developed, which includes decentralized nodes for electronic voting. It is used to create electronic voting systems due to the benefits of end-to-end verification. This system is an excellent solution for traditional electronic voting methods due to its distribution, non-repudiation, and security properties. This E-Voting powered by Blockchain enables to cast votes online with the power Blockchain which enhances the security, authenticity, and end-to-end encryption of voting records such that Nobody can change or temper the records[4,5]. These records are stored in a decentralized manner such that all the information is shared with each node connected in the network and if any changes occur in data that this information is shared with every node. Our Tool or Application enables Citizens to cast votes authorized by Admin without going to polling Booths which reduces election costs and increases voting percentage.

1.1 PROBLEM BACKGROUND

Elections are conducted from ancient times when kings were chosen by voting from the People and the Ministers of the King come to vote for a decision. But in the present time the two most commonly used mediums of voting are:

- Ballot Paper
- E V M (Electronic Voting Machine) Voting

The *Ballot Paper Mode of Election* has drawbacks like Votes Tampering/Manipulation, Polling Booth Capture, and requires physical presence and need large amounts of funds for the conduction of the election. On the other side, the *EVM Mode of Election* can be Hacked and tampered with easily due to this it does not gain voters' trust. Fig 1 shows the differences between EVM-based voting and Blockchain-based voting processes.

After going through the drawbacks of old mediums of voting we proposed an E-Voting System which reduces these drawbacks.

1.2 PROBLEM STATEMENT

To develop a trustworthy Secure Electronic Voting Solution that hides the drawbacks of traditional voting mediums and should be cost-effective and free from any type of amendments and highly secure.

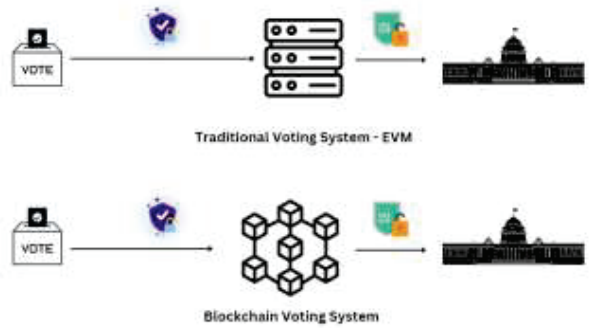

Fig 1: Comparison between EVM and Blockchain Mode of Voting

1.3 Research Objective

The main objectives of this study are to make a step forward in direction of online voting by providing ways that compensate lack ness of old voting mediums and provide an isolated way free from any type of dangers. As you all know elections like Govt. Elections, Polls, and Society Elections play a crucial role in judging a person based on the opinion of another person. In this case, online election systems are very useful with the help of this system users can cast their votes online which is immutable, highly secure and does not require additional setup, and saves money too.

II. Literature Review

A recent study discovered that the traditional voting procedure was not sanitary, raising questions about justice, and equality, and the people's will not be sufficiently defined and comprehended in the structure of democracy [6-8]. *Follow my vote* is a Decentralized E-Voting using Blockchain System but it does not provide immediate results (provides results after 48hrs of completion of the election) Voters have no Unique ID Card to vote and it is costly and has no central authority and it has also a limitation i.e. Once votes are voted incorrectly then it is counted as Invalid and this process is called as "kill-switch" [9]. As per this Paper, a new hashing Algorithm was introduced which increases user security and it also introduces some concepts of Block-sealing and block creation.

New voting procedures have been developed by engineers around the globe that protect against fraud while maintaining the integrity of the voting process. New electronic voting methods and procedures have been made possible by technology, and these are crucial and have caused serious problems for the democratic system [9]. Compared to human polling, electronic voting increases the reliability of elections. Comparing it to conventional voting methods, it has improved the voting process's efficiency and integrity [10]. Electronic voting is often utilized in a range of decisions because it is flexible, simple to use, and inexpensive compared to general elections [11]. Despite this, modern electronic voting methods have limitations in terms of basic voting fairness, privacy, secrecy, anonymity, and transparency due to their susceptibility to abuse of power and manipulated details. A framework is suggested in this System which used hashing method [10]. In this paper, Security analysis has been done on real India EVM (Electronic Voting Machine). As per the result, EVM can be tempered in many ways such as tampering with software before CPU Manufactures that violate votes [11]. The proposed voting system has no requirements for Hardware usage thus eliminating the disadvantages of EVM.

III. Proposed Methodology

A system with high security and accessibility is proposed, that is a MongoDB, ExpressJS, ReactJS, NodeJS (MERN) Based Web Application where the Voter first Signs up itself using the Sign-Up Form. Registration is confirmed after OTP Verification through E-Mail. After Completing registration voter will receive Welcome Mail after that voter can login into their account as it is a first-time voter then the System will tell the voter for 1st Time Voter Registration the voter can fill it a registration form and submit it then the user will receive an Application ID once the ID is approved by Admin then Voter ID no is shown into its profile and Voter can also download their Voter Card. Fig 2 describes the step-by-step process of the proposed system.

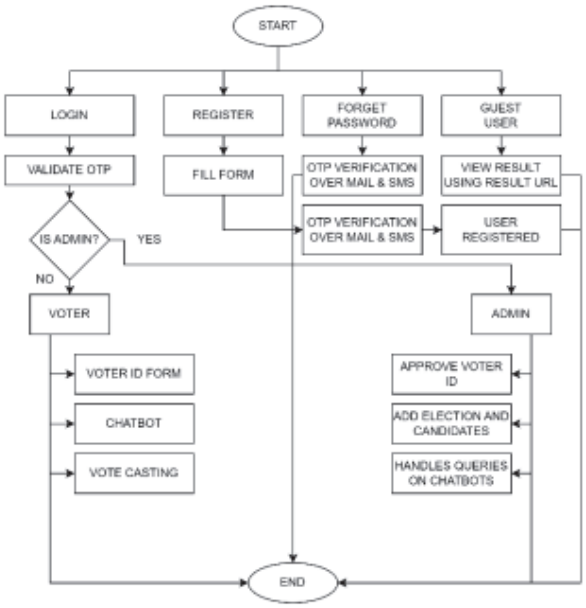

Fig 2: Process Flowchart with Functions

The admin can also register a voter and approve a voter. Admin can create an election by choosing the candidates and a proper duration after the successful hosting of the election voter will receive an email telling them about the election and the proper timing of voting then voters can log in to their account and choose election and then voter. The voting process is verified using OTP over Email and Face Authentication. After Successful voting, Voter will receive a thank you mail from the System, and at the proper time of the vote all the voter information is kept secure i.e. The Voter Chosen Candidates are not stored in the database. This Voting transaction is stored in the form of an immutable ledger known as Blockchain and with Smart Contracts. Fig 3 shows the MVC structure of the system and how the client interacts with the server and the database.

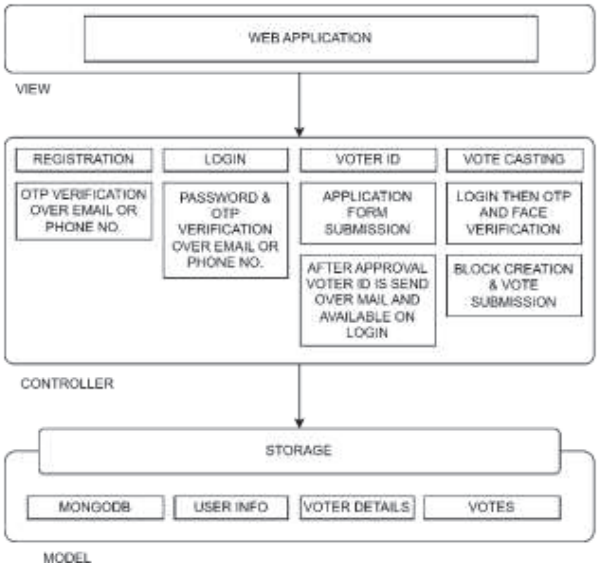

Fig 3: MVC Architecture of System

IV. Blockchain and Smart Contracts

Blockchain and Smart Contracts play a key role in setting up Blockchain-based networks with features of immutable records. In Blockchain, blocks of data are stored in the form of chains spread in a decentralized way and will be available to everyone on the network. Smart Contracts are programs written on Solidity Programming Language which run when predetermined conditions meet.

A. Blockchain

A Blockchain is a decentralized database that consists of a block that stores information on transactions that are chained together in a row such that each block includes information from previous nodes and these blocks are shared among all the nodes of the network. Blockchain technology become popular with cryptocurrencies like Bitcoin which was 1st cryptocurrency in the world. The First Block of a Blockchain is called a "Genesis Block" or "Block 0" and does not have the address of the previous Block. A blockchain structure is illustrated in Fig 4.

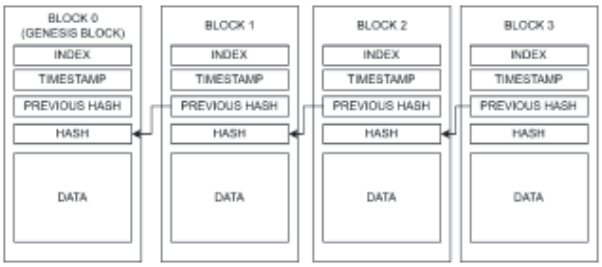

Fig 4: Structure of Blockchain Network [1]

By storing votes on a blockchain network, voters can rest assured that their data will be accurate and secure. The blockchain system is resistant to data tampering and hacking because it's decentralized and transparent. Plus, blockchain technology is also cost-efficient when it comes to using it for e-voting.

How does Blockchain work?

A blockchain is a network of blocks that are chained in a row each block of the blockchain consists of information about a Transaction and each block is identified by a Hash.

A Hash consists of 64 Hexadecimal characters of 4 Bits each and 256 Bits which is generated based on the information in the Block & previous Hash using the SHA 256 Algorithm when a Transaction occurs it creates a new block which is added in that chain. Adding chains requires additional processes like Proof of Work, and Competing chain protocol. The Block is immutable so they are not easily tempered [6]. The organization of votes is shown here in Fig 5.

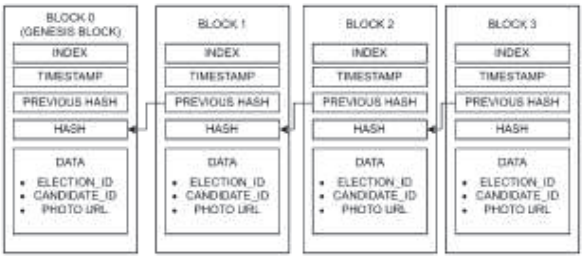

Fig 5: Blockchain Network of Votes

Hashing Algorithm: SHA 256 Algorithm

SHA 256 is one the popular algorithm used to generate a hash based on the information in the block and hash of the previous block for any minor changes in the information and previous hash the value of the new hash is complete changes (Avalanche effect) [10].

The main requirements of RSA 256 Algorithms are:

- One Way: Supports only Encryption
- Deterministic: Provides the same output each time
- Fast Computation
- Withstand Collision: Not easily hacked
- Avalanche Effect: for any minor changes data changes completely

Consensus Protocol

Consensus Protocol is the set of rules that must be followed for adding a block in the blockchain [11].

These Rules help in the prevention of attacks and help in the completion of the chain. There are two types:

- Proof of work (POW)
- Proof of Stake (POS) [11]

When any node discovers a Block then that node had to represent proof of work done by it in mining the block to each other node of the network [12]. These remaining nodes check that block by running an Algorithm if the algorithm says that the Block is genuine then it has been added to Blockchain.

B. Smart Contracts

Smart Contracts are generally programs that run on Blockchain Networks written in Solidity Programming Language which are executed once when specific conditions meet. They are mostly used to automate the execution of the agreement. Smart Contracts are also used to automate a workflow, triggering the next actions when particular conditions are met [13-14]. Smart Contracts are responsible for reading and writing data in Blockchain Networks and well as also responsible for applying logic [15,16,17]. In our

application, Smart Contracts logic is applied on the Blockchain Network which counts Votes throughout our network and the Candidates with the highest no of votes are declared the winner [18,19,20].

For this, we have to first create a Front End using (ReactJS) and a Server using NodeJS and install all dependencies after that we have to write smart contracts and deploy them on Blockchain. A smart Contract is declared with the "contract" keyword ending with the contract name. After that, we store the state variable which stores the data of votes. The state variable used to write data on the Blockchain constructor is called when a contract is deployed on the Blockchain network. The code is described here that shows the declaration of the contract, structure, and variables.

```
pragma solidity ^0.5.1;

contract Votes {
   struct Vote {
      string election_id;
      string candidate_id;
      string photo_url;
   }
   mapping (address => Vote[]) private Users;

   function addVote(string calldata _electionid,string calldata
_candidateid, string calldata _photourl) external{
     Users[msg.sender].push(Vote({
        election_id:_electionid,
        candidate_id:_candidateid,
        photo_url: _photourl
     }));} } }
```

Fig 6: Compilation Result

Here we have declared the struct Vote which has an Election_ID & Candidate_ID of unsigned integer type, Timestamp and Photo_URL of string type, and delegate of address type. To store these structs we use mapping in solidity which is a hash with key-value pairs. After creating the application and setup the smart contract we first login into Blockchain. To use Blockchain we need to use the account in ganache- A required dependency for Blockchain which provides free access to 10 accounts with account addresses and fake ethers on our Local system. Another dependency known as Meta Mask is also required. After completing all these we can interact with smart contracts in our system and we can cast votes easily. Fig 5 shows a sample compilation result, and Fig 6 a sample Vote Bank.

Fig 7: A Sample Vote Block

Each vote can generate a new data block, and after the election, the votes can be tallied to produce the Election result. In this System, only business elections, community elections, and college elections are permitted. There will be more challenges associated with a larger voter population. We utilize the Ethereum network, the scalability of which is currently unknown. It requires additional investigation; hence it is not suitable for large-scale elections.

V. Implementation & Results

In this section design, implementation, and functioning of the E-Voting application are discussed. This application can be accessed by both admin and user for their specific needs.

Both user [Fig 8] and admin [Fig 10] can access the application from where it is hosted and the user can cast their votes as well register him/her as a first-time voter and the admin can assign election/poll to the registered user and accept an application for first-time voters.

User-Driven Modules:

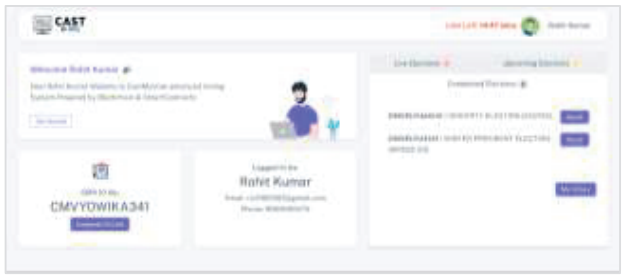

Fig 8: Snapshot of User Dashboard

- **Signup Module:** In this module, a voter can first register themselves through basic details like Name, Phone No, Email & Password after signup Voter can receive OTP over email once a verified Voter account is created into the System.
- **Voter Registration Module:** In this Module, voters can request their Voter ID by filling out a form with basic fields like DOB, Address, and Photo. After submitting of form Voter will receive an Application ID that shows the status of the application and the voter will receive mail for the same and the same is also displayed on the Voter Dashboard.
- **Login Module:** In this Module, Voters can sign in to their accounts to access all services, the voter will enter their email and password an OTP is sent to the registered Email, Once OTP verified Voter can enter into its dashboard and a login token is created with a limit of 15min after that voter will sign-out automatically.
- **Voter ID Module:** In this Module, voters can download their Voter ID Card. ID Card can be available when the Registration form is approved by the admin. This Voter Card is in the .pdf format with Name, Photo, and basic details mentioned on it and with a Voter ID Card Bar Code mentioned for further use.

- **Voting Module:** In this Module, voters can cast their vote just by choosing Election through Name and a list of Participants is available to him/her. Voter Just chooses Choice and submits voter with OTP verification and Face Verification with a Registration Photo and Voter Selfie is uploaded at the time of submission of the vote. After the successful submission of the Vote, a thank you mail was sent to the Voter. The voting process is here displayed in Fig 9
  User Login > Choose Election (with Live Status) > OTP Verification over Mail and by SMS > Face Verification > Choose Candidate > Submit > Success.

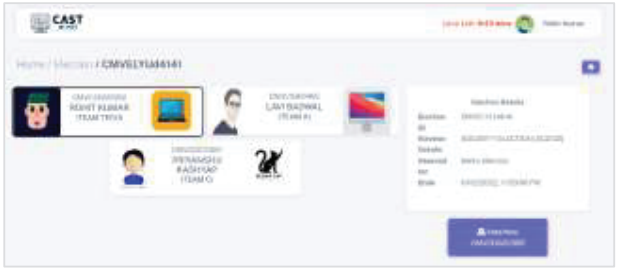

Fig 9:Voting Module

Admin-Driven Modules:
In the following section, some admin-driven modules are described.

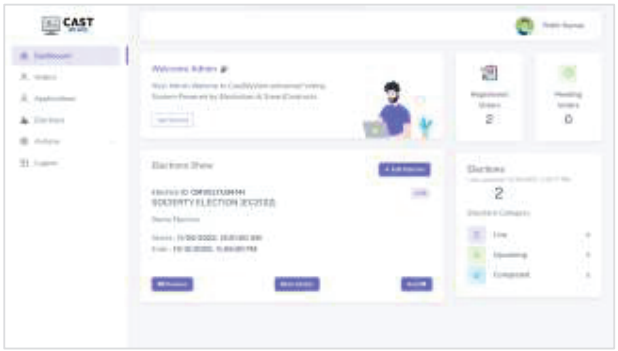

Fig 10: Snapshot of Admin Dashboard

- **Application Module:** In this module, The Applications submitted by Users will be displayed from where Admin can view the submitted form with supporting doc. From here user can Accept or discard the application. Once Discarded a discard mail is sent to the user with the discard reason and Once approved New Voter ID is issued to the application an SMS with details and an email with the Voter ID attached as an attachment is sent to the User and the User will then become a voter.
- **Election Module:** In this module, Admin can create a new Election with a name and a Unique Code after addition a Voting Invitation mail is sent to all the Voters with some basic details about the Election and a link to the tentative list of Candidate. After Election addition Admin can add, Update and delete candidates till Election starting time. Once Election started Anybody cannot change details like Election Info. Candidates etc. A complete description is available here in Fig 11.

Fig 11: Election Module

Common Modules:
- **Database:** All the records whether User Login Records, Voter IDs, or Election Details are stored in MongoDB Atlas (Non-SQL Database)
- **Blockchain Network:** All the Voting data are stored in Blockchains which are deployed in a decentralized Network.
- **Authentication:** The user, as well as Admin Login, can be verified through OTP via Email and SMS and this OTP creates a layer of security in the system OTP is required in various operations in our system. Multiple ways of authentication are also displayed in Fig 12.

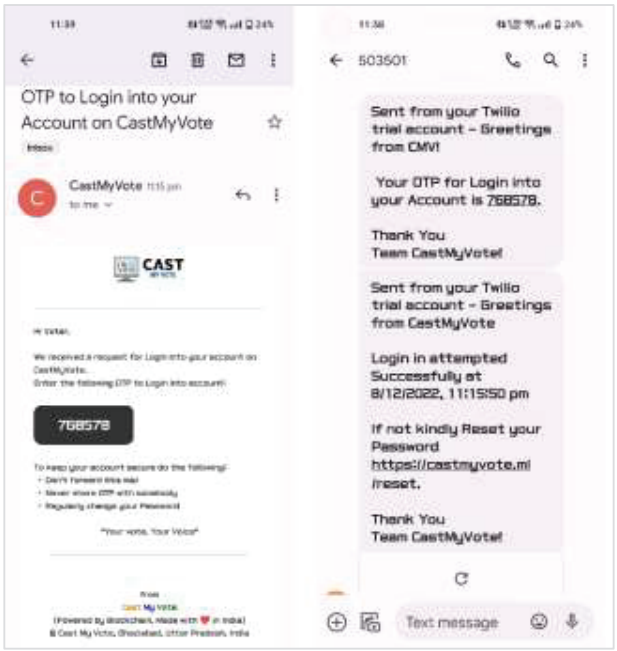

Fig 12: OTP Authentication

- **Result Module:** In this module, the result of the election is prepared Once Election got finished the System will check all the Blocks on the Blockchain Network and count the Votes. After the Counting of all Votes, the Result is available on Admin and Voter Login, and a Result mail is sent to Everyone and Result is also displayed on Results URL [see Fig 13] send on the Election thank you mail.

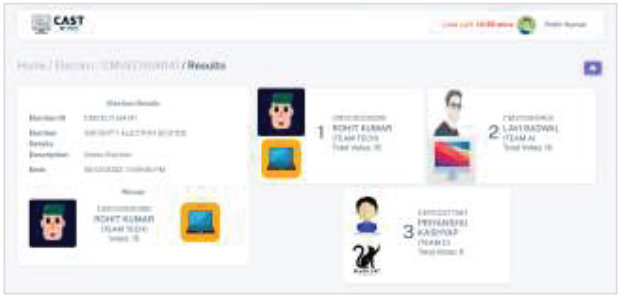

Fig 13: Results Module

VI. Discussion

In this project, the scope is limited to small-scale elections like elections at the college level, polls, society elections, etc. Larger voting with millions of votes requires a lot of infrastructure and Ethereum network stability is still unknown as it requires further research. Here this System is ideal for small Elections or polls like College Position Election or Organization Poll Here SMS Sending API has limited access for trail accounts Emails Sending API has also some limitations and our MongoDB Database have a limited amount of space. So, implementing this will requires funds as well as a dedicated server and Infrastructure.

VII. Conclusion

In this research paper, a Blockchain-powered E-Voting system is proposed that users/voters can use to cast their votes without going to the polling booth. An innovative electronic voting system based on blockchain technology is described in detail that ensures voter privacy while facilitating secure and economical elections. After going through a detailed study on recent Blockchain Technology, it is concluded that Blockchain offers democratic countries a new opportunity to replace the old means like EVM and Ballot system with this E-Voting system that is cost- and time-effective, while also enhancing the security features of the existing system and presenting new opportunities for transparency. This research paper presents a blockchain-based online voting framework based on MERN that uses smart contracts to keep voter privacy intact while keeping the process of elections safe and affordable. In addition to current blockchain-based voting systems, the blockchain offers the ability to significantly improve electronic voting as well as prospective future research avenues. The Blockchain is an excellent fit for a decentralized electronic voting system.

References

[1] Yadav, Abhishek. (2020). E-Voting using Blockchain Technology. International Journal of Engineering Research and. V9. 10.17577/IJERTV9IS070183.

[2] Jafar, U.; Aziz, M.J.A.; Shukur, Z. Blockchain for Electronic Voting System—Review and Open Research Challenges. Sensors 2021, 21, 5874. https://doi.org/10.3390/ s21175874.

[3] Benny, Albin, Blockchain-based E-voting System (July 11, 2020). Available at SRN: https://ssrn.com/abstract=3648870 or http://dx.doi.org/10.2139/ssrn.3648870.

[4] S. Nakamoto, "Bitcoin: a peer-to-peer electronic cash system", [Online]. Available: https://bitcoin.org/bitcoin.pdf

[5] Nir Kshetri, Jeffrey Voas, "Blockchain-Enabled E-Voting".

[6] Shahzad, B.; Crowcroft, J. Trustworthy Electronic Voting Using Adjusted Blockchain Technology. *IEEE Access* **2019**, *7*, 24477–24488.

[7] Gao, S.; Zheng, D.; Guo, R.; Jing, C.; Hu, C. An Anti-Quantum E-Voting Protocol in Blockchain with Audit Function.

[8] Kim, T.; Ochoa, J.; Faika, T.; Mantooth, A.; Di, J.; Li, Q.; Lee, Y. An overview of cyber-physical security of battery management systems and adoption of blockchain technology.

[9] Ali Kaan Koç, Emre Yavuz, Umut Can Çabuk, Gökhan Dalkılıç "Towards Secure E-Voting Using Ethereum Blockchain".

[10] E. Maaten, "Towards remote e-voting: Estonian case", Electronic Voting in Europe-Technology, Law, Politics, and Society, vol. 47, pp. 83-100, 2004.

[11] Avasthi, S., Chauhan, R., & Acharjya, D. P. (2021). Techniques, applications, and issues in mining large-scale text databases. In *Advances in Information Communication Technology and Computing* (pp. 385-396). Springer, Singapore.

[12] Avasthi, S., Chauhan, R., & Acharjya, D. P. (2022). Topic Modeling Techniques for Text Mining Over a Large-Scale Scientific and Biomedical Text Corpus. *International Journal of Ambient Computing and Intelligence (IJACI)*, *13*(1), 1-18.

[13] Pham, H. L., Tran, T. H., Phan, T. D., Le, V. T. D., Lam, D. K., & Nakashima, Y. (2020). Double SHA-256 hardware architecture with compact message expander for bitcoin mining. *IEEE Access*, *8*, 139634-139646.

[14] Zhang, S., & Lee, J. H. (2020). Analysis of the main consensus protocols of blockchain. ICT Express, 6(2), 93-97.

[15] Avasthi, S., Sanwal, T., Sharma, S., & Roy, S. (2023). VANETs and the Use of IoT: Approaches, Applications, and Challenges. *Revolutionizing Industrial Automation Through the Convergence of Artificial Intelligence and the Internet of Things*, 1-23.

[16] Idelberger, F., Governatori, G., Riveret, R., & Sartor, G. (2016, July). Evaluation of logic-based smart contracts for blockchain systems. In *International symposium on rules and rule markup languages for the semantic web* (pp. 167-183). Springer, Cham.

[17] Wang, S., Ouyang, L., Yuan, Y., Ni, X., Han, X., & Wang, F. Y. (2019). Blockchain-enabled smart contracts: architecture, applications, and future trends. *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, *49*(11), 2266-2277.

[18] Chauhan, R., Avasthi, S., Alankar, B., & Kaur, H. (2021). Smart IoT Systems: Data Analytics, Secure Smart Home, and Challenges. In *Transforming the Internet of Things for Next-Generation Smart Systems* (pp. 100-119). IGI Global.

[19] Khan, S. N., Loukil, F., Ghedira-Guegan, C., Benkhelifa, E., & Bani-Hani, A. (2021). Blockchain smart contracts: Applications, challenges, and future trends. *Peer-to-peer Networking and Applications*, *14*(5), 2901-2925.

[20] Trustworthy Electronic Voting Using Adjusted Blockchain Technology - Basit Shahzad Raju, Jon Crowcroft in the year 2019.