International Journal of
INTELLIGENT SYSTEMS AND APPLICATIONS IN ENGINEERING
ISSN:2147-6799 www.ijisae.org Original Research Paper

Transparent E-Voting: Paving the Way for the Future of Democracy using Blockchain

Bhoomi Shah [1*], Anil Vasoya[2], Kamal Shah[3], Sanjay Sange[4], Parshvi Shah [5] , Baban Rindhe[6], Mohini Reddy[7]

Submitted: 22/08/2023 **Revised:** 10/10/2023 **Accepted:** 23/10/2023

Abstract: In this work, an online E-voting system has been proposed that provides security, transparency, non-repudiation, etc. as the current voting systems which use Electronic Voting Machines (EVMs) have many limitations. These limitations are owing to the fact that EVMs are susceptible to tampering. This may cause a crucial impact on the judgement on any voting system and hence the entire system can be manipulated. Owing to widespread employment of voting system, it becomes necessary to provide a transparent and tamperproof voting system. For the same, a blockchain based system has been proposed that can be implemented for reality shows. This block chain based decentralized system is capable of breaking through the conventional centralized system and thus has a huge scope for widespread deployment. The proposed system uses a decentralized ledger for storing information which is maintained by the people in the network opposite to the centralized authority. The authors have utilized Ethereum as the foundational technology for constructing and operating the proposed system. Based on the implementation of the proposed system, it is evident that the proposed system can be widely used in real life nullifying the chances of manipulations in voting.

Keywords: *Reality Show Voting, Process Optimization, Smart Contracts, Web3.0, Solidity, Blockchain, Decentralization, Voting*

1. Introduction

Voting is one of the most important activity for the smooth and proper functioning of any democracy. Over the period of time, voting has become an unavoidable facet of an egalitarian society. The perpetuation of voting systems since the 17th century demonstrates societal faith in the system. However, there have been many cases on records where the integrity of the voting system has been compromised. Considering the widespread deployment of voting system, it is imperative to devise a system that maintains sanctity and correctness of the system. Further over the period of time, the horizon of voting has grown beyond politics. Now, voting has observed its widespread deployment in various sectors viz. corporate elections and reality TV shows etc. Hence, the requirement to have a tamper proof voting system is further strengthened.

The work in this paper mainly deals with the application of voting in reality TV shows.

The present-day voting system in these shows involves the usage of a centralized system where the producers are in complete control of the voting process.

It can either be a SMS-based system or an online portal where viewers can place their votes. However, the fact remains that the process is not transparent and the public has no idea of what happens behind the scenes. There is no possible way to determine whether their votes are actually taken into consideration. Producers of TV shows can be accused of overturning the votes in case it might harm the TRP of the show. To avoid such hassles and increase trust in such systems this paper proposes a decentralized approach to voting. It provides transparency and security to the entire process. This is where Blockchain [1] comes in. Blockchain technology has been at the forefront of ground-breaking technology ever since the introduction of bitcoin [2].

Blockchain, initially considered only as a solution for secure monetary transactions has started gaining ground in several other fields. Blockchain provides a very secure system of storing information that cannot be altered once stored. It is a Peer to Peer (P2P) network consisting of blocks [3]. Each block contains several transactions which are encrypted before adding to the ledger [4]. A series of these blocks form a chain thus the name, Blockchain. There are 3 main types of Blockchain: Public, Permissioned, and Federated Blockchain. A public

[1]*The Maharaja Sayajirao University of Baroda, Vadodara, Gujarat, India*
ORCID ID: 0000-0001-6881-2919
[2] *Thakur College of Engineering and Technology, Mumbai, India*
ORCID ID: 0009-0009-6495-9142
[3] *Thakur College of Engineering and Technology, Mumbai India*
ORCID ID: 0000-0002-6369-6960
[4] *Shah And Anchor Kutchhi Engineering College, Mumbai, India*
ORCID ID: 0000-0035-2938-5900
[5]*Fr. Conceicao Rodrigues College of Engineering, Mumbai, India*
ORCID ID: 0000-0002-2291-8993
[6] *K. C. College of Engineering & Mgt. Studies & Research, Mumbai, India*
ORCID ID: 0000-0001-8462-757X
[7] *NMIMS Mukesh Patel School of Technology Management and Engineering, Mumbai, India*
ORCID ID: 0009-0004-0962-7368

Blockchain is permissionless wherein anybody can participate without any restrictions. A permissioned Blockchain allows only certain users to access specific parts of the data, whereas a federated Blockchain is controlled by selected nodes of stakeholders. The implementation proposed in this paper makes use of the permissioned Blockchain such that only the election organizer has access to create or make changes to an election. The implementation also utilizes smart contracts [5].

A smart contract is an unchangeable piece of code present on the blockchain. It also consists of data associated with the code stored at a specific address on the chain. An address is associated with an account that contains some balance like a certain cryptocurrency. The account is used for storing the currency and after every transaction the cost of the transaction is deducted from this account. Smart contracts are not governed by a user but are deployed to the blockchain and will always run according to how they are programmed. Smart contracts are secure as they are automatically enforced and transactions made by them are irreversible. Adding a candidate, voting for a candidate, and updating the vote count are all done through smart contracts. The report discusses all implementation details and challenges faced in detail in the method proposed in this paper in the following sections.

The current work is organized into various sections. The need of tamperproof voting system is discussed in section 1 and related work is presented in section 2. Section 3 focuses on the proposed architecture and implementation is given in section 4. Results are illustrated in section 5. conclusion and further direction of research is given in section 6 and 7 respectively.

2. Literature Review

For reality show voting the most common method used is the missed call-based and SMS-based voting system. The problem with the following voting systems is that they use phone numbers as unique identification numbers for the viewer and fail to take into consideration that a viewer can have more than one mobile number [6]. Due to this, the results are skewed.

According to articles mentioned by famous news companies, there has been outrage amongst people about the integrity of voting occurring on reality shows [7]. Due to a lack of transparency people have questioned the integrity of voting systems. Questions have been raised about famous shows like “Dancing with the Stars” which conduct voting using a web2.0-based website or SMSs [8]. Thus, a blockchain-based system of electronic voting has been proposed in this paper. This system, in lieu of its public immutable ledger, provides security, transparency, and integrity. It reinforces the fact that every vote counts and the votes cast by every user are counted and the proof for this is given by transaction hashes of the vote transaction.

In the following paragraphs, a comparison of various blockchain-based e-voting systems is done with the system proposed in this paper. Now comparing the e-voting blockchain system proposed in this paper with other e-voting blockchain-based systems. The system proposed in this paper [9] is based on the core concepts of blockchain. It considers scalability and verifiability for the voters. It is based on the bitcoin blockchain and thus does not allow the usage of user-written smart contracts to execute core functionality. Hence, the system proposed in this paper proposes the usage of the Ethereum blockchain which enable code execution alongside transactions.

According to Estonia’s system of Voting, each block would contain block size, transaction counter, block header, and transaction. This would require a lot of storage and processing overdue due to the use of a separate blockchain for each candidate [10]. The storage and processing required are reduced by making use of a single blockchain for both candidates as well as voters.

These papers [3, 11] compare different E-voting protocols and show the implementation by dividing them into different phases which gives a clear understanding of the process. This approach is not scalable. As the size will increase the time required for each phase will increase drastically. The system proposed in this paper is very scalable since it uses self-scaling Ethereum smart contracts thus providing high throughput, low latency, and low cost of transactions.

In a general blockchain-based e-voting system, SHA encryption of the voter information is performed before adding it to the selected candidate’s chain. A disparate/separate chain is used for every candidate. This system does not discuss the implementation technique and no framework is also suggested. In the proposed system of this paper, a single chain is used for the voting system which makes use of the solidity framework for implementing smart contracts [12].

There are many existing applications for E-voting using blockchain. The major drawbacks were increased storage and processing overhead due to the usage of multiple blockchain networks for multiple candidates. Also, the usage of smart contracts increases the flexibility of the system allowing us to add programmable content to the blockchain.

In this paper, a blockchain-based voting system coupled with a Mongo-DB database built on the Ethereum blockchain is proposed which is better than the above-mentioned systems as explained below. Unlike the above-mentioned literature, this paper describes the

implementation in detail.

3. Proposed Architecture

The proposed architecture of the system requires a backend in web 2, web 3, and a corresponding frontend for the same. The backend in web 2 and web 3 should work with each other for the smooth working of the system.

In the web 2 backend Node.js is used to authenticate the user. The Node.js backend enables us to maintain two types of users and store the data of each user. It helps ensure that only the admin can create an election. Node.js is required because it is not ideal to retrieve data from the blockchain.

For the blockchain solidity was used to code the Ethereum smart contracts. Smart contracts are basically programs which add or retrieve data from a block present in a blockchain. Each blockchain consists of a number of smart contracts. Once a smart contract is executed the changes made by it cannot be reversed.

For the frontend React js is being used. React js provides an interface to the backend to enable the user to interact and enter their information. It provides a means for the user to transact his/her vote on the blockchain. It is a way for the user to interact with both the web 2.0 backend and the blockchain.

Applications such as ganache, truffle, and meta mask are used to help check the execution of the system on the Ethereum blockchain. Ganache provides 10 accounts having 100 Ethereum each. Truffle enables the execution of smart contracts on the Ethereum blockchain. MetaMask [13] provides a wallet for accessing the accounts provided by ganache. It allows the user access to 100 Ethereum in his/her wallet. The architecture of the proposed method is shown in Fig. 1.

A blockchain consists of simple programs stored on it that run when predetermined conditions are met. These simple programs are called smart contracts. It is a piece of code and data associated with the code that resides at a specific address on the Ethereum blockchain. They can be synonymized with a type of Ethereum account. This implies that they have a balance and can send transactions on the network. They are deployed on the network and cannot be controlled by the user and run as they were programmed to. User accounts can interact with it by calling functions for which they have to submit transactions. Like a regular contract, even in a smart contract rules are defined and here the rules are automatically enforced by a piece of code, unlike a regular contract. For E-voting using blockchain the smart contracts which have to be executed are as follows:

The first is election creation, an election is created by the admin user. This system consists of two types of users, one being a normal user who takes part in elections and another being an admin who selects the candidates standing for the elections and creates elections.

Candidate and voter registration are other functions to be executed by smart contracts. The admin selects the candidates which will stand for the elections. These candidates need to be registered on the blockchain network and data about the candidates standing for the elections is mandatory for creating an election. Candidate registration on the blockchain ensures that the admin doesn't accidentally assign the same candidate for an election twice. Voter registration on the blockchain ensures that the same voter is not allowed to participate in an election more than once.

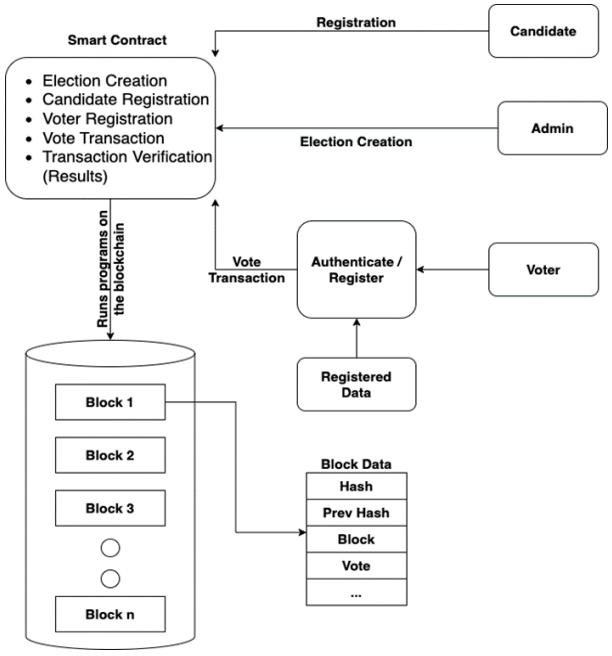

Fig. 1. System architecture of decentralized E-voting

Vote transaction is another important function to be executed by the smart contract. The core idea behind e-voting using blockchain is that the votes made are immutable which increases the security of the system and the best way to implement that is by the transaction of votes on a blockchain. Since the blockchain's ledger is visible to the public, forging votes is an impossible task.

Transaction verification is another function to be executed by the smart contract. The votes which have been transacted need to be counted in order to display the result of the elections. The function of this smart contract is mainly to calculate the result.

The first step in the flow of the election is the admin login. Once the admin enters his credentials and successfully logs in, he is redirected to the election creation page. The admin can then enter the details about the elections and click submit. Upon submitting a smart contract is triggered and the election is created. The registered candidates can now be added by the admin to a specific election. The addition

of candidates to the election is also a transaction on the blockchain and requires some transaction fee. It is also executed using a smart contract. The voters aren't required to log in to the system as once they have registered as a voter they are given a unique wallet account that can only be accessed by them. In order to vote they need to authorize a transaction through their account. While voting the users receive a MetaMask prompt asking to confirm the vote/transaction. Every time a user votes, the voteCount attribute of the candidate he/she voted for is incremented by 1 which makes tallying of votes easier. Once the election ends, the admin will be able to see the total votes received by each candidate on the View Vote count page. The workflow of election used in the system proposed by this paper is shown in Fig. 2.

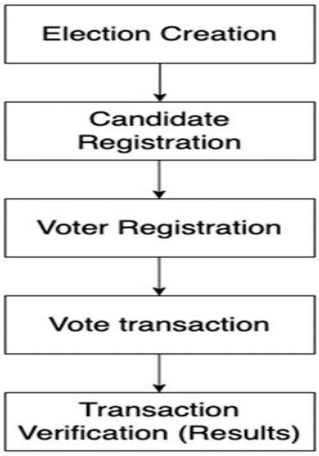

Fig. 2. Workflow which shows the steps in this system.

4. Implementation

4.1. Length of Papers

A form was created for the purpose of finding out information from the public. The form consisted of the following questions: 1. Do you trust the voting system being used in various reality shows? 2. Are you happy with the voting system currently in use in various reality shows? 3. Would you like the voting system used by these reality shows to be more transparent about the votes being placed? The form was then circulated among eligible voters. The form got a total of 273 responses and 84 percent of them did not trust the voting system used in reality shows, and 76 percent of them were not happy with the voting process in use. 82 percent of them wanted the voting process to be more transparent with the votes being cast. The responses to the survey conducted were from a diverse group of people of different sexes and genders. The graphs below show the split in responses when it comes to age and gender.

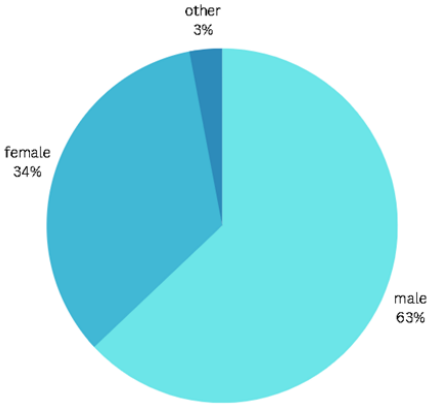

Fig. 3 Form response analysis according to gender

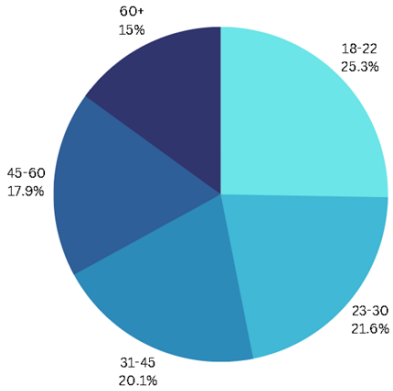

Fig. 4 Form response analysis according to age

The objective of the survey was conducted to find out what people thought about the already existing voting system. The survey's main goal was to highlight the drawbacks of the already existing voting system. This survey brought to light that the voting system currently in use wasn't trusted by many viewers of shows since they did not know the number of people voting the number of votes each participant got and did not have any guarantee about their vote-making a difference to the shows which they loved watching. Not only do the missed call and SMS-based voting system lack transparency but it also takes a couple of days for the result. The survey helped conclude that if the system in use was more transparent viewers would be a lot more willing to take part in voting and such a transparent process might help increase viewer retention of a show.

4.2. System description

Based on the results of the survey we built an implementation that not only suited the people's needs but also incorporated an improved architecture as compared to the existing systems. The implementation of the proposed system has been shown below.

4.2.1 User Type

An end user of the website will either be a user or an admin. The difference is that a user takes part in voting and

cannot create an election whereas an admin can create an election, select the election candidates, and declare the result of the election. The screen where the end user chooses whether he or she wants to be either a user or an admin is shown below in Fig. 3.

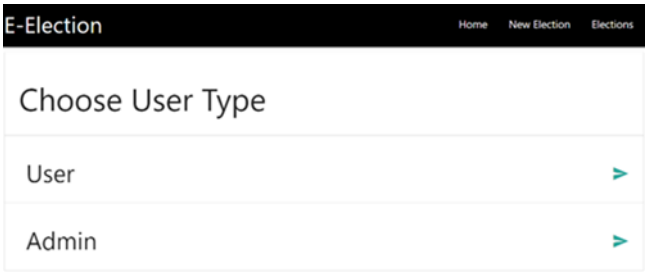

Fig 5. User Type Screen

4.2.2 Admin Login

An admin must login before he/she creates an election. The registration screen is shown below in Fig. 6.

Fig 6. Admin Login Screen

4.2.3 Election Creation

For election creation, an admin must enter the election name, organizer, and password. The election screen is shown below in Fig. 7.

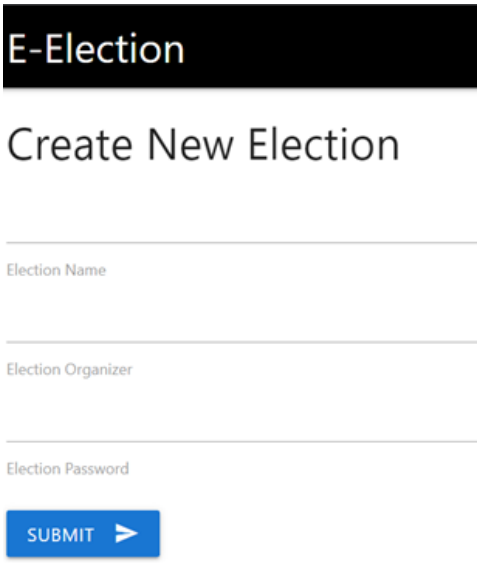

Fig 7. Election Creation Screen

4.2.4 Adding Candidates to an election

After creating an election an admin needs to add the candidates standing for the elections. The admin needs to enter the name and other details of the candidate. The election candidate adding screen is shown below in Fig. 8.

Fig. 8 Adding Candidates Screen

4.2.5 Election List

Once an election is created by an admin, a user can see a list of elections and can select which election to vote for. A user can only vote once for an election. The election list screen is shown below in Fig. 9.

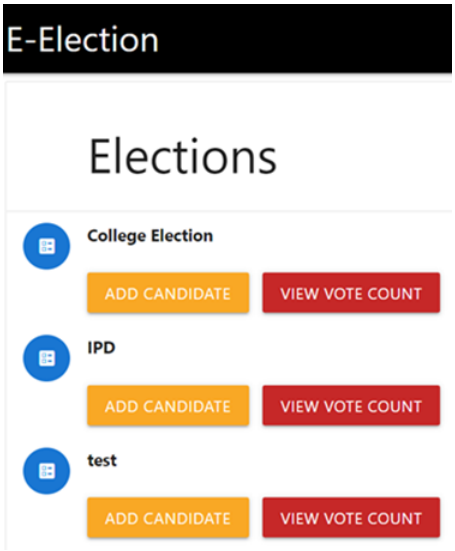

Fig. 9 Election List Screen

4.2.6 Vote

The user can now vote from the available candidates by clicking on the vote button. They can only vote once in a particular election.

5. Result and Discussion

The following section discusses the testing performed and discussion of results.

5.1 Testing

The proposed application was tested on two levels. Before starting with end-to-end functional testing unit tests were

written using the mocha library and all functions of the smart contract were tested in a development environment provided by truffle. It was ensured after every change all unit tests were passed.

The first level of testing was done by running Ganache on localhost:7545 which provides 10 Ethereum accounts with 100 eths each. Using one account as the admin, deployment of the smart contract, creation of election, adding of candidates, and calculating the result(ending the election) is done and then the registering of the other 9 accounts occurs where all accounts are added to the preferred wallet. In the proposed system MetaMask was used and then they are authenticated as registered voters post which the same accounts were used to vote for candidates. This was level 1 of testing where everything was done on a single system. transaction fees were very low and since only 1 node was running and the transaction speed was also very high. The transaction cost was 0.004 eth which at the time this paper was written amounts to 3.17 INR.

In level 2 of testing, the application was deployed using Netlify and the smart contract was deployed on the Goreli [13] test network. The rest of the testing conditions were the same except for the fact that there was no limit on the number of accounts that could be used. Ether obtained from faucet using the Goreli test network which provides one Ether at a time which is more than sufficient for the required used case. The transaction fees were the same as the previous system mimicking the main net only increasing at times of congestion or when a few nodes of the test network are down. The transaction speed was slightly slower as the number of people on the network increased. Overall, after these two phases of testing, the system was ready to be deployed on the Ethereum main net.

In Fig. 8. the transaction hashes, receiver addresses, sender addresses, and type of transaction can be seen. Below is an image of the public immutable ledger where one can go and view all transactions with respect to our application, check all transactions of a particular address by searching the address up on etherscan, and get metadata of every transaction by searching the transaction. In this way, the security and transparency of our application is tested.

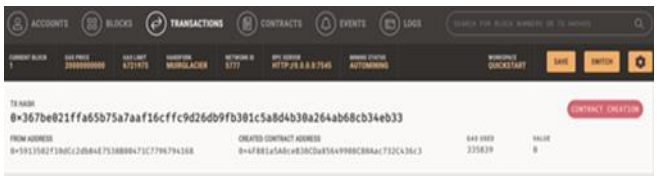

Fig. 10 Decentralized Immutable Ledger

5.2 Results

After subsequent tests using the above testing methodology, it can be concluded that this system has 100% accuracy since it always shows the correct vote count of individual candidates on the result declaration. The time taken for result calculation is very less since votes are counted as they are cast and no manual counting of votes is involved. The system is entirely transparent because of the decentralized nature of blockchain. The use of blockchain in e-voting makes it highly secure and immune to tampering because blockchain has an immutable ledger.

Analyzing the results of SMS, missed calls, and website-based voting systems we concluded that the proposed system is holistically better. In many ways like security, integrity, transparency, etc. The table below shows the comparison of the proposed system with currently used reality show voting systems.

TABLE 1 : COMPARATIVE STUDY OF DIFFERENT VOTING METHODS

	SMS	**Missed Call**	**Website**	**Proposed System**
Integrity	Low	Low	Medium	High
Transparency	Cannot be ensured	Cannot be ensured	Difficult to ensure	Completely Transparent
Security	Low	Low	Medium	High
System maintenance	Low to establish, difficult to maintain	Low to establish, difficult to maintain	Costly but easy to maintain	Costly but easy to maintain
Cost to user	Moderate	Moderate	Little or no cost	Moderate (less than SMS and Missed Call)

6. Conclusion

Voting in reality shows is one of the most crucial aspects since the elimination of contestants is regulated by this. The system needs to be secure, reliable, dependable, and transparent to keep the public happy, satisfied and involved. The older system that uses centralized technologies like websites, SMSs, and calls can be accused of being tampered with. Apart from this, a lack of transparency causes the viewers to term the show as

scripted and at times discontinue watching the show.

To solve this challenge of secure online E-voting, Blockchain is a promising technology. Blockchain technology ensures that the voting is carried out in a secure manner and provides transparency as the ledger will be publicly accessible by everyone. Doing so will increase trust in the shows. Furthermore, it also provides mathematically accurate results as there is no chance of miscalculation or chances of tampering with votes due to the immutable nature of blockchains. Apart from this it also provides other sources of revenue that help in making the show more profitable too. The proposed system not only helps in publicity but also in the future a fungible token can be made into an asset that people hold just like other cryptocurrencies.

7. Future Scope

The product built has great scope and can be easily implemented on a larger scale. However, with the number of users increasing, Polygon can be used as a scaling solution in the future to increase transaction speed and further lower costs. In the future to further lower costs a distinct blockchain and its cryptocurrency called the VoteCoin can be built on top of Ethereum where every user before every election on registration could buy VoteCoins used for voting.

These VoteCoin can be given custom names based on the name of the show, for example, the X show can have the name of the crypto as X coin. The highest holder of these coins can be given special privileges like the opportunity to call or meet the contestants or live show passes etc. More utility can be built around the coin by selling merchandise having exclusive fan events passes be sold for these coins and much more.

The scope of blockchain is still relatively new and its application to domains outside of currencies is being explored. It is safe to assume that as the move towards web3 is being made, the technology supporting it will mature and its scalability will increase.

Author contributions

Bhoomi Shah: *Conceptualization, Methodology, Software, Field* study ***Anil Vasoya, Kamal Shah*****:** Data curation, Writing-Original draft preparation ***Sanjay Sange, Parshvi shah:*** Software, Validation., Field study ***Baban Rindhe, Mohini Reddy:*** Visualization, Investigation, Writing-Reviewing and Editing.

Conflicts of interest

The authors declare no conflicts of interest.

References

[1] Hanifatunnisa, Rifa, and Budi Rahardjo. "Blockchain based e- voting recording system design." 2017 11th International Conference on Telecommunication Systems Services and Applications (TSSA). IEEE, 2017.

[2] Lu, Ning, et al. "BEvote: Bitcoin-Enabled E-Voting Scheme with Anonymity and Robustness." Security and Communication Networks 2021 (2021).

[3] Ayed, Ahmed Ben. "A conceptual secure blockchain-based electronic voting system." International Journal of Network Security & Its Applications 9.3 (2017): 01-09.

[4] Di Pierro, Massimo. "What is the blockchain?." Computing in Science & Engineering 19.5 (2017): 92-95.

[5] Soni, Sumit, and Bharat Bhushan. "A comprehensive survey on blockchain: working, security analysis, privacy threats and potential applications." 2019 2nd international conference on intelligent computing, instrumentation and control technologies (ICICICT). Vol. 1. IEEE, 2019.

[6] S. Sen, “Missed calls are most preferred voting system for realityshows”, Sep. 28, 2021.

[7] R.Dawn, “Is reality-show voting rigged? Well, sort of ...” today,Sep 28, 2021.

[8] W. Entriken, “Introduction to smart contracts,” ethereum.org, 06-Jun-2022. [Online]. Available: https://ethereum.org/en/developers/docs/smart-contracts/. [Accessed: 25-Jun-2022].

[9] Oo, Htet Ne, and A. M. Aung. "A survey of different electronic voting systems." International Journal of Scientific Engineering and Technology Research 3.16 (2014): 3460-3464.

[10] Hardwick, Freya Sheer, et al. "E-voting with blockchain: An e-voting protocol with decentralisation and voter privacy." 2018 IEEE International Conference on Internet of Things (iThings) and IEEE Green Computing and Communications (GreenCom) and IEEE Cyber, Physical and Social Computing (CPSCom) and IEEE Smart Data (SmartData). IEEE, 2018.

[11] Patil, Harsha V., Kanchan G. Rathi, and Malati V. Tribhuwan. "A study on decentralized e-voting system using blockchain technology." Int. Res. J. Eng. Technol 5.11 (2018): 48-53.

[12] Jokić, Stevo, et al. "Comparative analysis of cryptocurrency wallets vs traditional wallets." Ekonomika 65.3 (2019): 65-75.

[13] Bogner, Andreas, Mathieu Chanson, and Arne Meeuw. "A decentralised sharing app running a smart

contract on the ethereum blockchain." Proceedings of the 6th International Conference on the Internet of Things. 2016.

[14] Hiroshi Yamamoto, An Ensemble Learning Approach for Credit Risk Assessment in Banking , Machine Learning Applications Conference Proceedings, Vol 1 2021.

[15] Waheeb , M. Q. ., SANGEETHA, D., & Raj , R. . (2021). Detection of Various Plant Disease Stages and Its Prevention Method Based on Deep Learning Technique. Research Journal of Computer Systems and Engineering, 2(2), 33:37. Retrieved from https://technicaljournals.org/RJCSE/index.php/journal/article/view/30