“Online Voting System Using Blockchain”

PROJECT SYNOPSIS

OF MAJOR PROJECT

Bachelor of Technology

CSE

SUBMITTED BY

SHUBHAM VERMA
SHIVAM SHRIVAS
VIMAL KUMAR DUBEY

September 2022

KIET GROUP OF INSTITUTIONS,
Delhi-NCR, Ghaziabad (U.P.)

Department of Computer Science and Engineering

Submitted to: Dr. Sushil Kumar

Signature:

Table of Contents

INTRODUCTION

In any democratic country, Voting is a fundamental right of any citizen that enables them to choose the leaders of tomorrow. It gives individuals in a community the facility to voice their opinion. It helps them to realize the importance of citizenship. Online voting systems are software platforms used to securely conduct votes and elections. As a digital platform, they eliminate the need to cast your votes using paper or having to gather in person. They also protect the integrity of your vote by preventing voters from being able to vote multiple times. Electronic voting or e-voting has fundamental benefits over paper-based systems such as increased efficiency and reduced errors. The online voting system tends to maximize user participation, by allowing them to vote from anywhere and from any device that has an internet connection. The blockchain is an emerging, decentralized, and distributed technology with strong cryptographic foundations that promises to improve different aspects of many industries. Expanding e-voting into blockchain technology could be the solution to alleviate the present concerns in e-voting. Here we propose a blockchain-based voting system that will limit the voting fraud and make the voting process simple, secure and efficient.

Objective

The online voting system tends to maximize user participation, by allowing them to vote from anywhere and from any device that has an internet connection. The blockchain is an emerging, decentralized, and distributed technology with strong cryptographic foundations that promises to improve different aspects of many industries. Expanding e-voting into blockchain technology could be the solution to alleviate the present concerns in e-voting. Here we propose a blockchain-based voting system that will limit the voting fraud and make the voting process simple, secure, and efficient.

Literature Review

The authors of the paper [2] wrote an empirical review of e-voting applications using blockchain technology. They began by establishing the challenges that e-voting applications face: privacy, lack of evidence, fraud resistance, ease of use, scalability, speed, and cost. They compared a set of e-voting applications that they believed covered all aspects to create a robust system. Indeed, there were not enough criteria in this paper to compare all the specifics of e-voting applications.

The authors of the paper [6] described the use of blockchain-based e-voting applications in real-world scenarios. They then extracted a set of properties that a blockchain-based e-voting application should satisfy to be a fair, transparent, and democratic election system. These properties are public and individual verifiability, dependability and reliability, consistency, auditability, anonymity, transparency, scalability, eligibility, authentication, and fairness.

Our paper provides a more complete and richer comparison. It is designed as a guide for any research on the subject. Our paper not only compares the existing solutions but makes it easy to customize solutions based on specific criteria that are crucial for e-voting. We have determined a set of criteria that we have deemed very specific to include as many blockchain-based e-voting applications as possible in our comparison. In this way, our proposed work should help the community to focus on certain criteria and to compare the proposals of several applications as well.

Our paper used a similar methodology and objective to those published by the authors in the paper [24]. The analysis of the subject proposed in this paper is very interesting and our work is in line with it. Otherwise, our work differs from the latter and thus brings a new approach to the topic of blockchain-based e-voting. Indeed, our work complements the paper [24] since we propose about thirty additional references in the comparison of implementations. We have been able to use more recent papers and have used a more inclusive selection technique than the one proposed in [24]. In the latter, the authors identified several exclusion criteria that we did not apply in the preliminary search, allowing us to study other relevant papers. In addition, we relied on other published literature reviews that reference and compare voting applications. This allowed us to gather outside opinions, both positive and negative, on articles proposing voting applications, and thus gain credibility. While the overall approach to the systematic literature review is similar, the criteria for comparison differ for some. In our paper, section

IV-A details the voting process proposed by each voting application. Moreover, section IV-B compares the voter authentication methods, which is not specified in [24]. These differences make it possible to compare differently and more widely these e-voting implementations and thus bring novelty to the field of e-voting with the blockchain.

Proposed Approach

For our proposed plan of work we are considering two modules, first one is user-signup/login page and other is admin login page.

In the **user signup page** the use need four things to sign up and these are : name , email, password and confirm password after giving these details user can sign up by clicking on the signup button. If the email id already registered, then the user cannot signup twice and if the confirm password cannot match with the password then it shows an error. After the successfully login the message is shown that 'you are successfully registered'. After signup now user ready for login. In the login section user can need email and password by which user can sign up. If the any of the detail is wrongly filled by the user a message is shown that 'please enter valid id or password'. After successfully login user can enter in the landing page of the user where the first page is the user manual ,in this section there are basic guidelines for the user to understand the whole process .

There are two major processes from the voter side the first is the **voter registration** , in this there is a registration form and user needs to fill this , to eligible as a voter .After filling this registration form , voter will eligible to vote .Those who were not fill the voter registration form will not eligible for the vote .After the registration phase is over user cannot registered and will not able to vote .For the first phase of the registration, the user will have to enter the **Adhar card number** and the **account address** which the user will be using for the voting purpose , at this stage user age will be checked .If the user is 18 or above 18 years of age then only the he/she eligible to vote . The second stage of the registration is the **OTP verification**. This stage is required to validate the voter itself .After entering the correct OTP user will get the successfully registered.

The second important process is the **Voting Process,** whole process can carried out in the three phases, first phase is the registration phase where voter can register himself, then the second phase is voting phase where the user can able to vote , after the second phase the third phase is the result phase and it is the final stage of the voting process.

Now we moving towards the voting registration phase , in the registration phase the registration form is open and registration form is asking for the Aadhar number and the account address , account address must be those account address which we use to cast a vote to perform a transaction .We have created a **dummy database of the Adhar number** and there are few entries there , so we can validate weather the candidate is a valid candidate or not. If the Aadhar number enter by the user cannot match with the Adhar numbers in the database then, user can not able to registered .In the dummy Adhar database we also added a email-id in the database .If the Aadhar number is correct and user cannot previously registered then, user will get the OTP on the mail that is linked with the Aadhar. After entering the correct OTP user will be successfully registered. Now we move towards the voting area ,Voting area is a area where we will requiring to vote and perform a transaction.

In the **admin login page** the admin needs to enter the correct email and password after entering the correct details the can logged-in .There are the some functionalities which the admin can perform .The first functionality is the **add candidate information** . In the add candidate information admin must need to enter some details like name of the candidate , party , age and the qualification of the candidate after filling this details the admin can click on the add button , after clicking on the add button transaction has been generated and after confirming the transaction and message is shown that candidate will registered successfully and similarly admin can add the more candidates**.** The second functionality of the admin is the **voter registration section**. In the voter registration section all the users account address is shown and using this account address user will cast a vote or we simply say all the details of the user account address will be displayed in the voter registration section, and admin has needs to copy the users account address and then register it .If the admin completes the registration then only that users account will eligible to vote. The third functionality of the admin is the **change phase section, there** are only two phases one is registration and second is voting ,and only admin can able to change this phase. During the voting phase user cannot register.

In the voting area a user can choose a candidate after clicking on the vote button , and then a transaction takes place and after confirming the transaction a message is shown ‘voted successfully’ and a one mail is sent in the users registered email-id and this mail it is written the user can vote to whom. During the election the candidate will not able to go in the result section . Once after the election is over the user can see the results. Once the time of the election is over admin can change

the phase from the change phase section and after that no one will be able to vote. Once the election is over the results will start showing in the result section area.

For our proposed plan of work we are considering two modules that are to be completed in three phases. The first module is **Front-end** for the application and second module **Back end** using Solidity to implement Blockchain. Each of these modules will be considered as one phase and the remaining one phase will cover the connection and testing of these modules. In this **phase 1** we will cover the front-end module, in which we will build the interactive user-interface for the admin as well as the user. In parallel the research work related to the implementation of Blockchain in decentralized application will be done. In this phase2 we will cover the back-end module, we will implement the Blockchain using Ethereum framework and convert the system into a decentralized application. In the phase 3 the connection of two different module along with the testing of the platform will be completed in this phase.

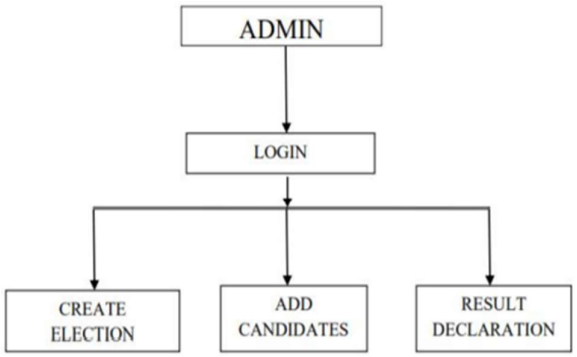

Fig-1

Division of Phase One: We have considered 2 main modules which are as follows. The first module of the phase1 is the **admin (**refer to fig 1**).** The admin module is divided into 5 components. The first component of the admin is the dashboard. The dashboard will contain various charts to display information such as number of parties, number of voters etc. The second component of the admin is the add candidate. In the add candidate component admin will add candidates who are standing in the election. After candidate is added it will be displayed on the user side. Third component of the admin is the Create Election. In the Create Election section admin will allow him to create election. A user can cast his vote only after

the election is created by admin. A user can cast vote between the start date and end date. The fourth component of the election is the Election Details and in the Election Details section the admin can update election details such as start date, end date etc. The fifth component of the admin is the Candidate details and in the candidate details section, the admin will Admin can update the candidate details if in case a wrong entry is done.

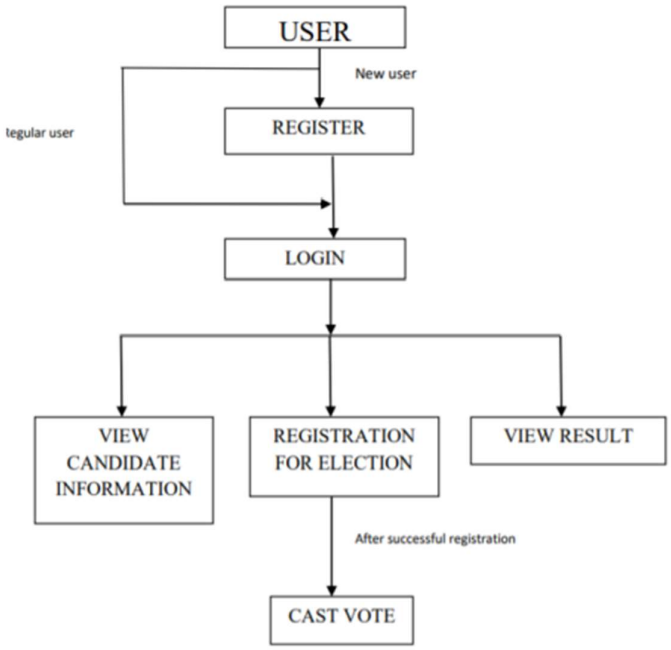

Fig-2

The second module of the phase 1 is the **User (**refer to fig 2**)**. The user module is divided into four components .The first component of the user module is the Dashboard. The user dashboard sections contain information about parties and their candidates. A user can see all the information about candidate. The second module of the user section is the Voter Registration. In the voter registration section firstly, user will have to register himself only then he will be able to cast his vote. The third component if the user section is the Voting Area**.** user is registered, then only he will be directed to this page and then he can cast his vote. The fourth component of the user section is the Results. In the result component the user will be able to see the results of the election.

Research Methodology of Phase Two:

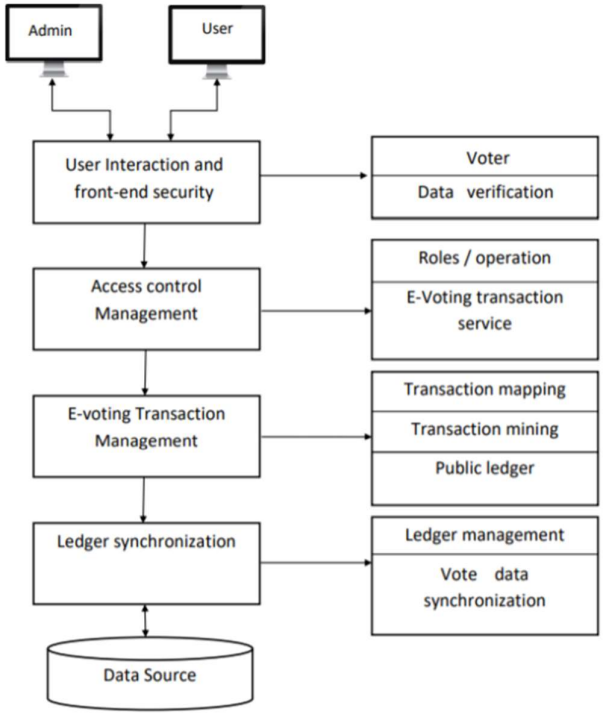

Fig-3

The blockchain is a digital ledger of transactions that is duplicated and distributed across the entire network of computer systems on the blockchain. It is a system of recording information in a way that makes it difficult or impossible to change, hack or cheat the system. Blockchain consist of very important concept called blocks. Every chain consist of multiple blocks and each block has three elements. First element is the data(transactions) the hash of the previous block and the block hash value .Hash value is a unique value, identity one block .It depends on the blocks content (data and previous block hash) , so each block has its unique hash

value, and its identifying this block only .Therefore each block can reference or point to the block , which means the four-block is taking a reference to the third one is taking a reference to the second , and so on and thus a chain of block is formed which we call as a blockchain. The key element that make blockchain immutable is cryptographic hashes, which is why blockchain is immutable. The data stored in blockchain is in the form of transactions. Blockchain transaction is the transfer of the crypto money. A transaction is a new record of exchange of some value or data between two public addresses of the blockchain. For developing E-voting using blockchain we used Ethereum -a popular platform for creating distributed Blockchain applications that support smart contracts. Ether (ETH) is the native cryptocurrency of the platform .Smart Contract are self – executing contracts which contains the terms and conditions of agreement between peers. They are simply programs stored on a blockchain that run when predetermined conditions are met. They typically are used to automate the execution of an agreement so that all participants can be immediately certain of the outcomes, without any intermediary involvement. Solidity language is used for writing the smart contract. It is the statically typed, support inheritance, libraries and complex user- defined types among other features . For performing any transaction on the blockchain we require an account address, this can be created by using the MetaMask chrome extension. MetaMask is a crypto wallet and gateway to blockchain apps. It generates passwords (in the form of mnemonic) and the keys on your device, so only you have access to your accounts and the data . It helps users in interacting with the blockchain. Since working with the main Ethereum network costs actual money for transactions, we are using a local RPC “Ganache”. Ganache is a local test network for rapid Ethereum and distributed applications development .It can be used across the entire development cycle ; enabling us to develop , deploy and test our decentralized apps in a safe and deterministic environment. Now to interact with our compiled smart contract in a hassle-free manner use Truffle suite. Once the user is identified, his vote must be added to the blockchain while preserving his anonymity. It is at this point that the vote encryption/hashing algorithms intervene to ensure security and integrity of transactions during the election there are different functions that can hash or encrypt the vote at different levels. One of the most used is **SHA-256** [5], [6]. SHA (Secure Hashing Algorithm) is a cryptographic hash function that produces a 256-bit hash value consisting of 64 hexadecimal characters. It was designed by the United States National Security Agency. SHA 256 is a new hash function that does not have collusion problems and seems to be reliable now. The advantage of this algorithm is that it accepts any input length and produces an arbitrary output length, whereas most other

algorithms produce a fixed output length. The **Homomorphism encryption** property allows to operate on ciphertexts without decrypting them [8]. In the case of a voting system, this property allows encrypted ballots to be counted by a third party without any information contained in the ballot being disclosed. The **Zero Knowledge Proof** is often used in a voting system [8] to prove that the statement is indeed what it claims without revealing any additional information about the statement itself. **Blind Signature and Ring Signature** are very useful to provide the user's anonymity and the signer's privacy [7]. Voting systems use blind signature to convince the tallying centre that the ballot is from a valid voter, without revealing the owner of the ballot [9]. For e-voting applications that are based on the Ethereum blockchain, they use an Ethereum-specific Hash function that can be added to other encryption algorithms [9], [10].

TECHNOLOGIES REQUIRED

1-BLOCKCHAIN

2-ETHEREUM

3-JAVASCRIPT

4-SOLIDITY

5-METAMASK

6-GANACHE

7-TRUFFLE

REFERENCES

[1] Umut Can Çabuk1, Eylül Adıgüzel2, Enis Karaarslan2 (2018); A Survey on Feasibility and Suitability of Blockchain Techniques for the E-Voting Systems; International Journal of Advanced Research in Computer and Communication Engineering. [Online].

[2] Aayushi Gupta1, Jyotirmay Patel2, Mansi Gupta1, Harshit Gupta1 (2017); Issues and Effectiveness of Blockchain Technology on Digital Voting; International Journal of Engineering and Manufacturing Science. ISSN 2249-3115 Vol. 7, No. 1 (2017). [Online].

[3] Pavel Tarasov and Hitesh Tewari (2017); the Future of E-Voting; IADIS International Journal on Computer Science and Information Systems Vol.12, No. 2, pp. 148- 165 I. [Online].

[4] Edureka (How Blockchain Works) - Simply Explained. [Online].

[5] R. Hanifatunnisa and B. Rahardjo, ‘‘Blockchain based E-voting recording system design,’’ in Proc. 11th Int. Conf. Telecommun. Syst. Services Appl. (TSSA), Oct. 2017, pp. 1–6.

[6] H. Yi, ‘‘Securing E-voting based on blockchain in P2P network,’’ EURASIP J. Wireless Commun. Netw., vol. 2019, no. 1, pp. 1–9, Dec. 2019.

[7] X. Yang, X. Yi, S. Nepal, A. Kelarev, and F. Han, ‘‘Blockchain voting: Publicly verifiable online voting protocol without trusted tallying authorities,’’ Future Gener. Comput. Syst., vol. 112, pp. 859–874, Nov. 2020.

[8] A. Trechsel. (2016). Potential and Challenges of E-Voting in the European Union. (Nov. 10, 2021). [Online]. Available: https://cadmus.eui.eu/bitstream/handle/1814/44926/EUDO_REPORT_2016_11.pdf?sequence=1

[9] F. Sheer Hardwick, A. Gioulis, R. Naeem Akram, and K. Markantonakis, ‘‘E-voting with blockchain: An E-voting protocol with decentralisation and voter privacy,’’ in Proc. IEEE Int. Conf. Internet Things (iThings) IEEE Green Comput. Commun. (GreenCom) IEEE Cyber, Phys. Social Comput. (CPSCom) IEEE Smart Data (SmartData), Jul. 2018, pp. 1561–1567.

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