Voting Platform using Blockchain and the Concept of Demiblocks

Hardik Aggarwal[1], Dr Anu Rathee[2]

[1]Student, Dept of Information Technology, Maharaja Agrasen Institute of Technology, Delhi, India
[2]Assistant Professor, Dept of Information Technology, Maharaja Agrasen Institute of Technology, Delhi, India

---

Abstract – *The traditional methods of elections like ballots or electronic voting machine have several concerns like vote tampering, low turnout, delays etc. Majority of these problems can be solved by using the emerging technology of Blockchain. Blockchain helps in creating a secure, reliable, transparent platform using which any discrepancy can be caught in a flick of second. The focus of the study article is on creating a voting platform using Ethereum blockchain with the help of other technologies of web development. It will result in increasing voter turnout as well as public trust and confidence in elected officials or leaders. Also, a mental construct of demiblocks is also discussed which can prove to be a catalyst in the security of the system if it is further experimented.*

***Key Words*: Blockchain, Voting, Ethereum, Smart Contract, Real-time, Elections**

1.INTRODUCTION

Elections are an integral part of democracy. For electing people to power, there must be a strong reliable voting system which must have minimal flaws and is easy to use. In a populous country like India, where the count of voters is more than the total population of majority countries, there is a serious need of a better voting system.

There are many cases of booth capturing and allegations of EVM tampering but many of these cases remain unattended.

Blockchain is an emerging technology which can prove to be a concrete solution for the voting platform problem. In a blockchain, a data once entered, can't be changed. Therefore, there will be minimal chances of discrepancy. We can catch the culprit in a split of second, because once, a block is changed, the whole blockchain gets disrupted.

So, a transparent and secure voting system can be build using various tools and technologies.

There will be a special number which will be given to each voter and after confirming the identity of that person, he/she will be able to cast the vote.

After that nobody will be able to change it which will result in solution of a major problem in the election system. This system can also be used in various other election purposes like in clubs, corporations, societies, associations, internal election in a political party etc.

There are still some loopholes in blockchain technology like the incessant use of electricity in mining blocks, and the 51% attack which we think will be solved in due course of time.

We can start testing this system on smaller scale and gradually upscale it. We can also include biometrics to solve the problem of verification of the identity of the voter.

2. THEORETICAL FRAMEWORK

2.1 Blockchain

Blockchain is a new technology which is proving to be a game-changer in the field of data security. It uses cryptography as its underlying principle, and its various algorithms like SHA-256(Secure Hash Algorithm) etc. Blockchain is basically a chain of linked blocks which contain encrypted data. As they are linked using hashing, if there is a change in even a single alphabet, whole chain gets disrupted and we can check which block has been tampered. Therefore, if developed properly, it will be a milestone.

2.2 Ethereum

Ethereum is one of the cryptocurrencies based on blockchain technology. But not only the coin, Ethereum has developed into a bunch of services and has created a smart contract language known as Solidity. It is seriously helping the blockchain technology to bloom.

2.3 Solidity

Solidity is a smart contract language developed by Ethereum company for creation of smart contracts so that developers can take use of functions for various purposes. A smart contract is nothing but a piece of code which is stored in a blockchain and it uses some wei(small unit of ether) to implement the functions mentioned in that smart contract. Solidity is an object-oriented language which contains many features like structures, maps etc. which helps us in understanding it better.

2.4 Web Development

Node.js is used for back-end development of the website. There are few packages downloaded from Node Package Manager which are used for various functions like compiling, connecting etc. Top three are mentioned here.

SolC compiler is a compiler which is used to compile the smart contracts written in solidity language. Truffle is used to create a personalized blockchain before uploading it to the Ethereum. Web3.js is used as a connection between the website and the blockchain.

2.5 Ganache and Metamask

Ganache is used to test the blockchain. It is a software in which we can maintain our own truffle blockchain or Ethereum test network.

Metamask is an Ethereum wallet which actually contains your Ethereum coins or Ethereum test coins. It is used debiting and crediting the ethers for deployment or transactions in our blockchain. There are many wallets available but Metamask is the one which is most used.

2.6 Demiblocks

A demi-block is a cloned block with false information which is used to increase the length of blockchain. It can help in lowering the probability of data alteration and an assistance to the SHA-256 algorithm which is used to encrypt the data. The demi-block having similar but unoriginal data can help in increasing the time, a hacker tries to hack a blockchain, therefore preventing frequent damages.

2.7 Challenges

Emerging Technology: Blockchain is still emerging technology which is in exploratory stage and there are many technical and legal issues.

Room for Improvement : Blockchain business and market influencers to work together and change the business, deploy blockchain technology in market, and introduce innovative ideas.

Lack of Understanding of blockchain innovation : Businesses have to develop their understanding of blockchain innovation, it’s worth, its chances, and its dangers.

3. METHODOLOGY

3.1 Starting a project

The project started by initializing the Node Package Manager and installing all the necessary packages. Then a directory was created for all the necessary JavaScript files and other solidity files for the smart contracts.

The directory structure included contracts, migrations, src folders and truffle.js file.

3.2 Working on the blockchain packages

Working on the blockchain packages and smart contract was the main work of the project.

The contract included:

State Variables: They are used to store values which are then used for functioning of smart contract.

Functions: They are the main parts of the smart contract which helped us to perform the basic functions of voting and counting of votes.

Events: They are used for logging important values to the client terminal.

Struct: These are similar to the structures present in various programming languages which act as a storage for various types of information.

Mappings: We used two mappings of voter and candidate.

Basically, a Voter and a Candidate are described by two structures. We were able to assign many attributes to them using Structs, like emails, addresses, and so on.

We separated Voters and Candidates into different mappings that were integer indexed to keep track of them.

We also kept track of how many voters and candidates there were, which helped us organise them. A new Voter struct is created and added to the mapping when a user votes. All Candidates and Voters' histories will be stored in the mappings.

3.3 Creating a local blockchain

After the Smart Contract was completed, it was tested on the test blockchain before being uploaded to the blockchain.

Before the test blockchain can begin, a file named 2 deploy contracts.js must be generated in the /contracts folder that instructs it to move the Voting Smart Contract.

3.4 Testing with some functionality

We may test the code with our own testing procedures or with the mocha testing package, which allows us to test all of the functionality.

All of the functionalities have been thoroughly tested to ensure that there will be no issues when the application is connected to the Rinkeby Ethereum test network. The contract was then relocated. Migrations is a Truffle script that allows us to change the status of our app's contract as we create it. We were able to shift data around thanks to migrations. We set up a web3.0 instance with JavaScript on the browser whenever the program launched after we published the smart contract to the Blockchain.

3.5 Connecting the Ethereum Test Network: Rinkeby

As mentioned before, Rinkeby Test network is used to experience blockchain practically because it is a large blockchain therefore it helps the users in developing. We also used Metamask here which is the wallet for storing the Ethereum test coins. We got free Ethereum test coins from *rinkeby.faucet.io*.

Connecting to this gave us the practical experience of how this system really works. Rinkeby is a test network, which means it has nothing to do with money, but it is a large network which was enough to give us the feel like we are working on real Ethereum blockchain.

3.6 Full stack work for the website to function

The final step was to create the application's interface. This included HTML, CSS, and JavaScript, which are all required for every online application.

The HTML is a simple page with a user ID input form and buttons for voting and counting votes. When certain buttons are pressed, appropriate voting functions are invoked, and the number of votes for each candidate is calculated.

However, there are three key div components with ids: candidate-box, msg, and vote-box, which will include checkboxes for each candidate, a message, and the total number of votes, respectively. Also included are jQuery, Bootstrap, and app.js.

It was necessary to communicate with the Contract in order to build voting and counting functions for each candidate. We utilized jQuery to modify the DOM, and Promises to make transactions and requests to the Blockchain.

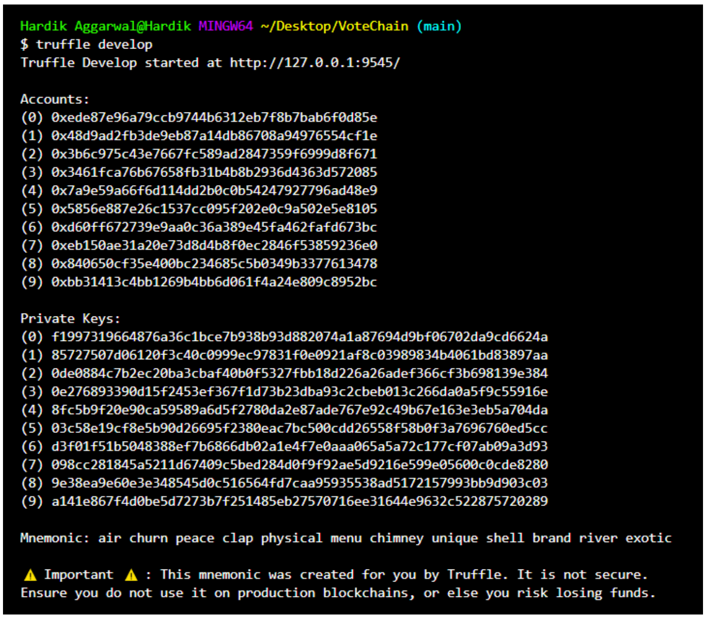
3.7 Final Testing and Deployment

For the final testing, we made some of our friends try the website so that we could find the real time errors that the users experienced while working on the website. It is the part of software engineering. It could be deployed to Heroku and other hosting servers. The final application was run on the localhost and it used Rinkeby Test Network and Ganache for the Ethereum Blockchain services.

4. IMPLEMENTATION AND RESULTS

First, we created a local blockchain using truffle. 9 accounts are created by default.

Command: truffle develop



```
Hardik Aggarwal@Hardik MINGW64 ~/Desktop/VoteChain (main)
$ truffle develop
Truffle Develop started at http://127.0.0.1:9545/

Accounts:
(0) 0xede87e96a79ccb9744b6312eb7f8b7bab6f0d85e
(1) 0x48d9ad2fb3de9eb87a14db86708a94976554cf1e
(2) 0x3b6c975c43e7667fc589ad2847359f6999d8f671
(3) 0x3461fca76b67658fb31b4b8b2936d4363d572085
(4) 0x7a9e59a66f6d114dd2b0c0b54247927796ad48e9
(5) 0x5856e887e26c1537cc095f202e0c9a502e5e8105
(6) 0xd60ff672739e9aa0c36a389e45fa462fafd673bc
(7) 0xeb150ae31a20e73d8d4b8f0ec2846f53859236e0
(8) 0x840650cf35e400bc234685c5b0349b3377613478
(9) 0xbb31413c4bb1269b4bb6d061f4a24e809c8952bc

Private Keys:
(0) f1997319664876a36c1bce7b938b93d882074a1a87694d9bf06702da9cd6624a
(1) 85727507d06120f3c40c0999ec97831f0e0921af8c03989834b4061bd83897aa
(2) 0de0884c7b2ec20ba3cbaf40b0f5327fbb18d226a26adef366cf3b698139e384
(3) 0e276893390d15f2453ef367f1d73b23dba93c2cbeb013c266da0a5f9c55916e
(4) 8fc5b9f20e90ca59589a6d5f2780da2e87ade767e92c49b67e163e3eb5a704da
(5) 03c58e19cf8e5b90d26695f2380eac7bc500cdd26558f58b0f3a7696760ed5cc
(6) d3f01f51b5048388ef7b6866db02a1e4f7e0aaa065a5a72c177cf07ab09a3d93
(7) 098cc281845a5211d67409c5bed284d0f9f92ae5d9216e599e05600c0cde8280
(8) 9e38ea9e60e3e348545d0c516564fd7caa95935538ad5172157993bb9d903c03
(9) a141e867f4d0be5d7273b7f251485eb27570716ee31644e9632c522875720289

Mnemonic: air churn peace clap physical menu chimney unique shell brand river exotic

⚠ Important ⚠ : This mnemonic was created for you by Truffle. It is not secure.
Ensure you do not use it on production blockchains, or else you risk losing funds.
```

Fig -1: Creating local blockchain using truffle

Here is the front page of the project.

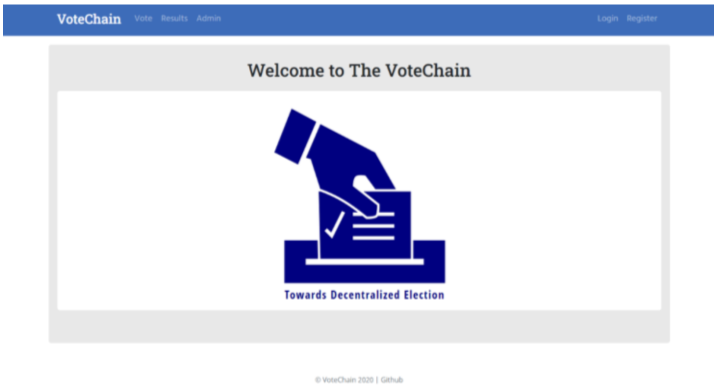


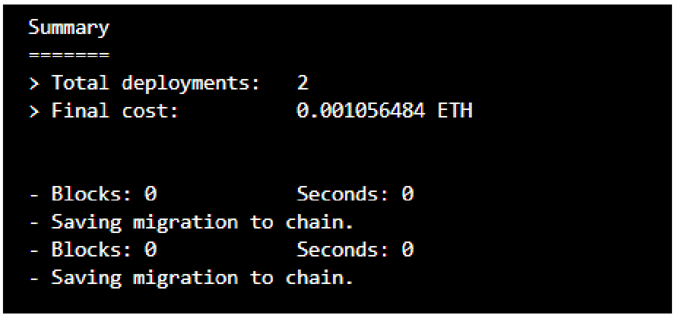
Fig -2: The home page

Below are the main terminal commands used to deploy and migrate the smart contracts

>truffle develop

>migrate

Next is the figure showing the summary after compilation and migration of smart contracts.



```
Summary
=======
> Total deployments:   2
> Final cost:          0.001056484 ETH


- Blocks: 0            Seconds: 0
- Saving migration to chain.
- Blocks: 0            Seconds: 0
- Saving migration to chain.
```

Fig -3: The summary after migration and deployment.

The above shows that the cost of deployments of two smart contracts name Voting and Migration is *0.001056484 ETH* which signifies those smart contracts were successfully deployed to the blockchain of Rinkeby test network.

After deployment, we were able to use the vote function and count vote function through the website. Hence, it fulfilled the purpose of this project.

5. FUTURE SCOPE

There are few more features which can be added to this project. One of them is AADHAR based authentication which will make this project more secure as the voter will be authenticated very easily using biometrics.

Because of the safe and unchangeable nature of blockchain, voters can cast their ballots via computer or mobile device rather than going to a local polling station or sending in a mail-in ballot to be manually processed by election officials. Votes tracked on a blockchain can be counted faster and more securely, perhaps leading to increased voter engagement, improved ballot security, and lower costs.

The agencies associated with various types of elections must organize hackathons and brainstorming sessions to induct blockchain in the voting processes.

This study was presented to address the issues afflicting voting systems, notably the absence of real-time results and transparency. We urge that all small and medium sized firms use this blockchain-based Electronic Voting System to assure the impartiality, secrecy, and timeliness of election results, hence boosting openness.

Following that, it might be deployed on a wide scale, such as in national elections inside countries, to assure electoral system transparency.

6. CONCLUSIONS

In this paper, we presented a Blockchain-based secure E-voting system that allows a decentralized database to vote in a modern way. We've demonstrated how blockchain technology can address concerns such as security, transparency, fairness, and trust, as well as lowering the hurdles to E-voting systems. Nobody will be able to tamper

Blockchain technology because it will be openly verifiable and disseminated. We also mentioned the disadvantages of our electronic voting technology, which can be used for further researches. The mental construct of demiblocks can also be used by someone who wants to use personal blockchain and not Ethereum. This project aimed to decrease election irregularities such as non-transparency, unfairness, duplicate votes, and delayed election results. The above was made possible thanks to Ethereum blockchain technology, which is a distributed ledger that is shared among all nodes connected to the blockchain system or network. Blockchain technology maintains uptime since storage and processing are not centralized. Because blockchain technology has proven to be effective in distributing immutable information across all nodes within a network, it was used to ensure data integrity and transparency in this study's voting procedure.

ACKNOWLEDGEMENT

First and foremost, I'd like to express my sincere appreciation to my mentor Dr. Anu Rathee, for her unwavering support throughout the project and research paper, as well as her patience, encouragement, excitement, and vast knowledge. Her advice was invaluable during the research and drafting of this report. For my project, I could not have asked for a greater counsellor and mentor.

REFERENCES

[1] Díaz-Santiso J, Fraga-Lamas P. E-Voting System Using Hyperledger. Engineering Proceedings. 2021; 7(1):11. https://doi.org/10.3390/engproc2021007011

[2] M. Hellman, Yavuz Emre, Ali Kaan Koç, Umut Can Çabuk and Gökhan Dalkihç, "Towards secure e-voting using Ethereum blockchain", 2018 6th International Symposium on Digital Forensic and Security (ISDFS), pp. 1-7, 2018.

[3] Freya Sheer Hardwick, Raja Naeem Akram and Konstantinos Markantonakis, E-Voting with Blockchain: An E-Voting Protocol with Decentralization and Voter Privacy, 2018.

[4] Faour Nazim, Transparent Voting Platform Based on Permissioned Blockchain, 2018.

[5] Mark, L., Ponnusamy, V., Wicaksana, A., Christyono, B.B. and Widjaja, M. (2021). A Secured Online Voting System by Using Ethereum Blockchain as the Medium.