Blockchain Based Voting Decentralized App

A Minor Project Synopsis Submitted to

Rajiv Gandhi Proudyogiki Vishwavidyalaya, Bhopal
Towards Partial Fulfillment for the Award of

Bachelor of Technology
(Computer Science and Engineering)

Under the Supervision of

Prof . Priyanka Jangde

Submitted By

Abhinav Nagar (0827CS201010)

Ajay Sonere (0827CS201019)

Ankit Buade (0827CS201033)

Department of Computer Science and Engineering
Acropolis Institute of Technology & Research, Indore
July-Dec 2022

1. Abstract

Even in the 21st century, online voting is still not possible in most countries. Voters need to leave their houses and cast votes. As a result, when the pandemic struck, elections were postponed since many people did not want to vote in person and could not vote online. This problem can solved by digital voting but the it is quite unsecure. The use of blockchain technology can address these issues by providing a secure and transparent voting system.

2. Introduction of the Project

This project focuses on the aspect of establishing digital voting by incorporating blockchain technology in it. The use of blockchain and smart contracts will ensure the security and integrity of the system. The project will be a Decentralized App with the owner or organization having control of registered voters and candidates. The voters can vote only once to their desirable candidates in easy and secure way.

3. Objective

The main objectives of this project are to:

- Develop a blockchain-based voting system that is secure, transparent, and tamper-proof.
- Implement a user-friendly interface for voters to cast their votes.
- Ensure the anonymity and privacy of voters.
- Implement a smart contract to automate the counting and reporting of votes.

4. Scope

The scope of this software is very large. This is a application which we can used by small organization as well as big organization mostly this will beneficial for private institutions and organizations. After few developments and like creating a own network and token for voting. It can also be implemented by the government for voting among the and can remove the traditional ineffective voting system.

5. Study of Existing System

The current voting systems in use in various countries around the world have several issues such as lack of transparency, vulnerability to fraud and manipulation, and inefficiency. A lot of effort is expended, and an entire day is devoted to the voting process in order to engage voters. Many countries hold general elections on Saturdays or Sundays to allow as many people as possible to vote. Election day is declared a public holiday in several countries or sections of countries that hold elections on a weekday.

Despite all of this effort and spending a whole day of the economy, the majority of areas had voting rates of 60-70%. This reveals far too much about the existing system's inefficiency. There are several flaws associated with current traditional voting, such as false/duplicate voting, counting mistakes, vote rejection, and human manipulation.

Furthermore, voting inside an organization cannot be done in this way, there are many issues. If the voting is scheduled on a weekday, all of the members will be unable to attend since they are at work. Their time is wasted if it is kept on the weekend. There will be complaints regarding time, place, and many more factors.

Digital voting solves many of these problems. Voters will be able to use their voting right in a few clicks from anywhere on the earth with an internet connection, whether using a mobile phone, a tablet, or a computer. It works efficiently for an organization and elections on a small level.

However, incorporating digital voting into traditional democratic systems is not as simple as it appears. The possibility of hacking and online identity verification are the two biggest obstacles. Although online identity verification is possible, this method is not safe from hackers. It is vulnerable to a variety of security concerns, such as DDoS attacks, vote alteration and manipulation, malware attacks, and so on.

There are several options, benefits, and drawbacks. Many nations are successfully carrying out this model of elections in local municipality elections. However, it has been shown that on a broad scale, this approach is not very dependable.

The voting process can be made more safe, transparent, immutable, and dependable by utilizing blockchain and many nations have also started implementing it. Let's examine it and its functioning in more detail.

6. Project Description

This project will mainly work on a user-friendly interface for voters to cast their votes, and will ensure the security of voters. The project will be developed using Solidity, JavaScript, React and Node.js. Solidity will be used to write smart contracts that will run on the Ethereum blockchain. JavaScript, React, and Node.js will be used to create the user interface, and to handle the front-end and back-end logic of the application.

The project will make use of Truffle, a development environment, testing framework, and asset pipeline for Ethereum, which will make it easier to develop, test, and deploy smart contracts. The project will also use Ganache, a personal blockchain for Ethereum development, that allows developers to create a private, virtual Ethereum blockchain for testing and development purposes.

The smart contracts will be responsible for counting the votes and reporting the results. The contract will be programmed to automatically count the votes and update the results in real-time, providing transparency and eliminating the need for manual counting. The system will also ensure the anonymity and privacy of voters by using zero-knowledge proofs, which allow for the verification of voter identity without revealing any personal information.

The system will also use Web3.js, a JavaScript library that allows for the interaction with the Ethereum blockchain, it provides a collection of APIs for interacting with the Ethereum network, such as sending transactions and querying.

7. Methodology/Planning of the Project work

The project will be developed using the Ethereum blockchain platform, which is well-suited for building decentralized applications. The voting system will be implemented as a smart contract on the Ethereum blockchain.

The user interface will be developed using a web-based application using react, which will allow voters to cast their votes using their smartphones or computers. The voting process will be secured using digital signatures and encryption.

The smart contract will be written in solidity which will be responsible for counting the votes and reporting the results. The contract will be programmed to automatically count the votes and update the results in real-time, providing transparency and eliminating the need for manual counting.

The system will also ensure the anonymity and privacy of voters by using zero-knowledge proofs, which allow for the verification of voter identity without revealing any personal information.

8. Expected Outcome

The blockchain-based voting system will provide a secure, transparent, and tamper-proof system for conducting elections. It will also increase voter turnout by making the voting process more convenient and accessible. The use of smart contracts will automate the counting and reporting of votes, providing real-time results and eliminating the need for manual counting. The system will also ensure the anonymity and privacy of voters.

9. Resources Required

The requirement of the resources for designing and developing are as follows

1. Solidity
2. Javascript
3. React
4. Node js
5. Truffle
6. Web3
7. Ganache

Solidity: Solidity is a programming language used for developing smart contracts on the Ethereum blockchain. It is a contract-oriented, high-level language that is similar to JavaScript and is used to write self-executing contracts with their own code and memory. Solidity can be used to create decentralized applications (DApps) that run on the Ethereum Virtual Machine (EVM).

JavaScript: JavaScript is a high-level, interpreted programming language that is commonly used to create interactive and responsive user interfaces for web applications. It is also used in the development of server-side applications using Node.js. JavaScript allows for the creation of dynamic and interactive elements on web pages and is an essential skill for building modern web applications.

React: React is a JavaScript library for building user interfaces. It is used to create reusable UI components that can be easily manipulated and rendered on a web page. React uses a virtual DOM (Document Object Model) to improve the performance of web applications by reducing the amount of direct manipulation of the actual DOM. React allows developers to create complex, interactive user interfaces with minimal code.

Node.js: Node.js is a JavaScript runtime built on Chrome's V8 JavaScript engine. It is used to run JavaScript on the server-side, allowing for the creation of fast, efficient, and scalable server-side applications. Node.js uses an event-driven, non-blocking I/O model, making it well-suited for real-time applications and data-intensive applications. It also has a large ecosystem of open-source packages, known as npm, which can be easily integrated into the project to add various functionalities.

Truffle: Truffle is a development environment, testing framework, and asset pipeline for Ethereum, aimed at making it easier to develop, test, and deploy smart contracts. It provides a suite of tools for compiling and deploying smart contracts, managing contract artifacts, and interacting with the Ethereum network. Truffle also includes a development console for testing and interacting with smart contracts, and a built-in development blockchain (Ganache) for local testing.

Web3: Web3.js is a JavaScript library that allows for the interaction with an Ethereum blockchain. It provides a collection of APIs for interacting with the Ethereum network, such as sending transactions, querying the blockchain, and interacting with smart contracts. Web3.js can be used in both client-side and server-side JavaScript applications and is a crucial component for building decentralized applications (DApps) on the Ethereum blockchain. It allows for the creation of a connection between the web application and the Ethereum network, providing the necessary functionality to interact with smart contracts and execute transactions on the blockchain.

Ganache: Ganache is a personal blockchain for Ethereum development. It is a tool that allows developers to create a private, virtual Ethereum blockchain for testing and development purposes. Ganache runs on the local machine and provides a simple, easy-to-use interface for creating and managing the blockchain. It allows developers to test and debug smart contracts, simulate different network conditions, and perform automated testing.

10. Conclusion

This blockchain-based voting system will provide a secure, transparent, and tamper-proof system for conducting elections. It will also increase voter turnout by making the voting process more convenient and accessible. The use of smart contracts will automate the counting and reporting of votes, providing real-time results and eliminating the need for manual counting. The system will also ensure the anonymity and privacy of voters, which is an important feature of a modern voting system.

11. References

1. Future of Blockchain Voting: Understanding the Blockchain Usecase. Coin Gabbar

.