A Project Report

On

Online Voting System

submitted for partial fulfillment of the requirements

for the award of the degree of

Bachelor of

Technology in

Computer Science

Submitted by

Ashish Kumar Gupta (2000290120043)

Aditya Aggarwal (2000290120013)

Saurabh Pundir (2000290110149)

Under supervision of

Prof. Akash Goel

KIET Group of Institutions, Ghaziabad

Dr. A.P.J. Abdul Kalam Technical University, Lucknow

2023 - 2024

DECLARATION

I/We hereby declare that this submission is our own work and that, to the best of our knowledge and belief, it contains no material previously published or written by another person nor material which to a substantial extent has been accepted for the award of any other degree or diploma of the university or other institute of higher learning, except where due acknowledgment has been made in the text.

Signature

Name:- Ashish Kumar Gupta

Roll No.:- 2000290120043

Date:-

Signature

Name:- Aditya Aggarwal

Roll No.:- 2000290120013

Date:-

Signature

Name:- Saurabh Pundir

Roll No.:- 2000290110149

Date:-

CERTIFICATE

This is to certify that Project Report entitled “Online Voting System” which is submitted by in partial fulfillment of the requirement for the award of degree B. Tech. in Department of Computer Science of Dr. A.P.J. Abdul Kalam Technical University, Lucknow is a record of the candidates own work carried out by them under my supervision. The matter embodied in this report is original and has not been submitted for the award of any other degree.

.

Date: **Supervisor**

Akash

Goel (Assistant

Professor)

ACKNOWLEDGEMENT

It gives us a great sense of pleasure to present the report of the B. Tech Project undertaken during B. Tech. Final Year. We owe special debt of gratitude to Professor Akash Geol Department of Computer Science, KIET, Ghaziabad, for his constant support and guidance throughout the course of our work. His sincerity, thoroughness and perseverance have been a constant source of inspiration for us. It is only his cognizant efforts that our endeavors have seen light of the day.

We also take the opportunity to acknowledge the contribution of Dr. Ajay Kumar Shrivastava, Head of the Department of Computer Science, KIET, Ghaziabad, for his full support and assistance during the development of the project. We also do not like to miss the opportunity to acknowledge the contribution of all the faculty members of the department for their kind assistance and cooperation during the development of our project.

Last but not the least, we acknowledge our friends for their contribution in the completion of the project.

Signature

Name:- Ashish Kumar Gupta

Roll No.:- 2000290120043

Date:-

Signature

Name:- Aditya Aggarwal

Roll No.:- 2000290120013

Date:-

Signature

Name:- Saurabh Pundir

Roll No.:- 2000290110149

Date:-

ABSTRACT

Building a secure electronic voting system that offers the fairness and privacy of current voting schemes, while providing the transparency and flexibility offered by electronic systems has been a challenge for a long time. In this work-in-progress paper, we evaluate an application of blockchain as a service to implement distributed electronic voting systems. The paper proposes a novel electronic voting system based on blockchain that addresses some of the limitations in existing systems and evaluates some of the popular blockchain frameworks for the purpose of constructing a blockchain-based e-voting system In particular, we evaluate the potential of distributed ledger technologies through the description of a case study; namely, the process of an election, and the implementation of a blockchain-based application, which improves the security and decreases the cost of hosting a nationwide election. This technology will improve the trust of voters that there action is secure.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

LIST OF ABBREVIATIONS

DFD	Data Flow Diagram
ER	Entity Relationship
IPFS	InterPlanetary File System
ROI	Return on investment
IEEE	Institute of Electrical and Electronics Engineers
SWOT	Strengths, Weaknesses, Opportunities, and Threats
PEST	Political, Economic, Social and Technological factors
MVP	Minimum Viable Product
UAT	User Acceptance Testing
API	Application Programming Interface
IDE	Integrated Development Environment
UI	User Interface
ECC	Error correction code memory

CHAPTER 1
INTRODUCTIO
N

1.1 INTRODUCTION

In this project, our aim is to create a cutting-edge electronic voting system leveraging blockchain technology. This system will revolutionize the voting process by allowing individuals to cast their votes conveniently from anywhere using electronic devices like mobile phones or computers. Despite the potential benefits, widespread adoption of such systems has been hindered by legitimate security concerns. Issues such as hacking and manipulation pose significant threats to the integrity of elections, prompting caution in their implementation on a larger scale. However, by incorporating blockchain's decentralized and secure nature, we endeavor to address these concerns and pave the way for a more transparent and trustworthy voting process.

Benefits of blockchain based e-voting system to customers.

Fairness and Privacy: With blockchain technology, votes are encrypted and stored securely, ensuring that each vote remains anonymous and tamper-proof. This ensures the integrity of the voting process and maintains the privacy of voters, fostering trust in the system.

Speed and Efficiency: Electronic voting via blockchain eliminates the need for manual counting and reduces the time required to tally votes. This results in a faster and more efficient voting process, enabling quicker declaration of results and reducing the likelihood of errors.

Transparency: The decentralized nature of blockchain ensures that every transaction (in this case, votes) is recorded transparently and cannot be altered retroactively. This transparency builds trust among voters, as they can independently verify the integrity of the voting process.

Immutability: Once a vote is recorded on the blockchain, it becomes immutable, meaning it cannot be changed or deleted. This feature ensures the integrity of the voting data and prevents any unauthorized alterations, thus enhancing the reliability of the electoral outcome.

1.2 PROJECT CATEGORY

The “Online Voting System” is an Internet-Based Application or System Development:

The development of an online voting system using blockchain technology represents a pivotal endeavor in modernizing the electoral process, aiming to establish a trustworthy, inclusive, and efficient platform for democratic participation. This ambitious project entails crafting a user-friendly web application or system that empowers voters to cast their ballots remotely via electronic devices like computers, smartphones, or tablets. By harnessing the power of blockchain technology, the integrity and immutability of the voting process are upheld, with each vote securely encrypted, recorded, and stored on a decentralized ledger. Moreover, the system may incorporate advanced features such as identity verification, fraud prevention, and real-time result monitoring to further enhance transparency and security. Across various stages, including requirements analysis, system design, software development, testing, and deployment, meticulous attention is dedicated to ensuring reliability, security, and usability, thereby advancing the integrity and accessibility of democratic elections in the digital era.

1.3 OBJECTIVES

Our objective is to develop a Digital Voting System with several key functionalities to ensure the integrity, accessibility, and trustworthiness of the electoral process. Firstly, the system will authenticate voters' identities to ensure that only registered individuals are eligible to cast their ballots, thereby preventing unauthorized access and maintaining the integrity of the voting process. Additionally, stringent measures will be implemented to ensure that each registered voter can only vote once, preventing instances of double voting or voter fraud. Furthermore, the system will securely store each voter's individual vote, safeguarding the confidentiality and privacy of their choices while also enabling accurate auditing and verification of the voting results.

Secondly, the user interface of the Digital Voting System will be intuitively designed to facilitate easy and seamless voting for all eligible individuals. This includes providing accessibility features to accommodate voters with diverse needs and ensuring compatibility with a wide range of electronic devices such as smartphones, tablets, and computers. Moreover, efforts will be made to promote widespread participation by eliminating barriers to entry and ensuring that every eligible voter, regardless of location or circumstance, can exercise their democratic right to vote.

Lastly, paramount importance will be given to the transparency and trustworthiness of the vote tallying process. Through the utilization of robust cryptographic techniques and blockchain technology, the Digital Voting System will ensure the verifiability and immutability of the voting results, thereby instilling confidence in the accuracy and fairness of the electoral outcomes. By providing a secure, accessible, and transparent voting platform, our objective is to enhance democratic governance and foster public trust in the electoral process.

1.4 PROBLEM FORMULATION

The traditional methods of conducting elections are often plagued by inefficiencies, vulnerabilities, and lack of transparency, leading to challenges such as voter fraud, tampering of results, and logistical constraints. In light of these issues, there is a pressing need for the development of a secure, transparent, and accessible voting system that leverages blockchain technology to ensure the integrity and fairness of the electoral process. The absence of a robust online voting solution leaves a void in democratic governance, hindering the ability to conduct elections efficiently, accurately, and inclusively. Therefore, the primary objective of this project is to design and implement an Online Voting System using blockchain technology, addressing the aforementioned challenges and providing a reliable platform for citizens to exercise their democratic rights with confidence and trust.

1.5 PROPOSED SYSTEM

1.5.1 ARCHITECTURE

The envisioned architecture for the e-voting system has been structured into multiple layers, aiming for a modular design approach. These distinct layers are elaborated upon as follows:

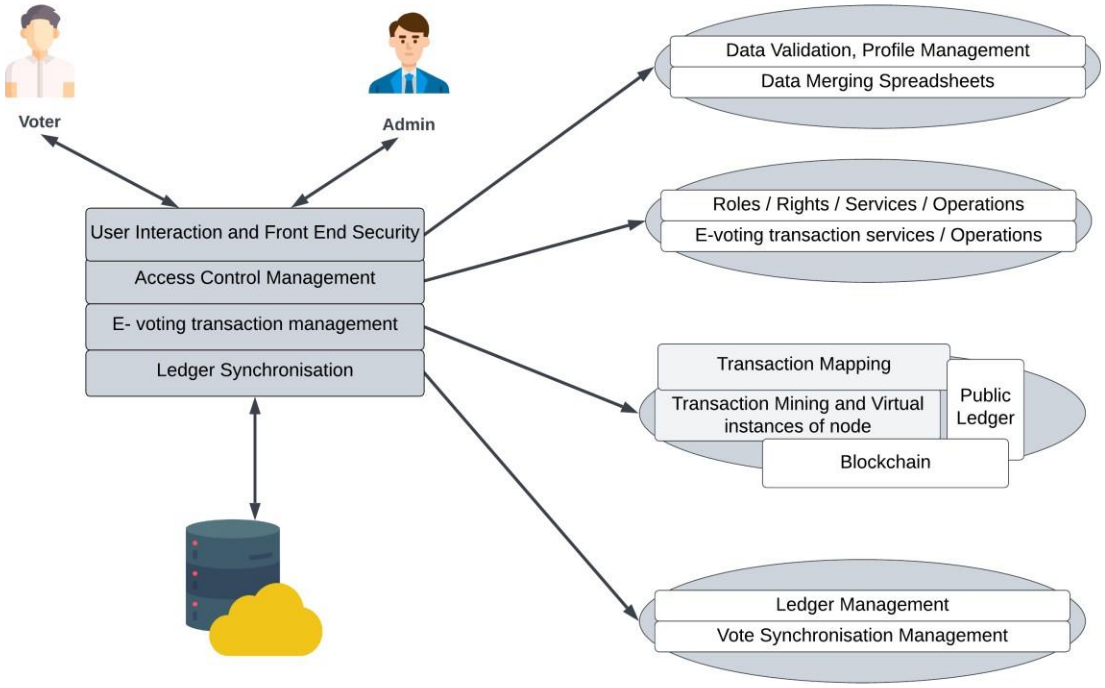

Voter
Admin
User Interaction and Front End Security
Access Control Management
E- voting transaction management
Ledger Synchronisation
Data Validation, Profile Management
Data Merging Spreadsheets
Roles / Rights / Services / Operations
E-voting transaction services / Operations
Transaction Mapping
Transaction Mining and Virtual instances of node
Public Ledger
Blockchain
Ledger Management
Vote Synchronisation Management

Fig 1.1 Architecture

1.5.1.1 User Interaction and Front-End Security

The User Interaction and Front-end Security layer manages interactions with both voters (enabling vote casting) and administrators (overseeing the election process). Its core functions revolve around authenticating and authorizing users—ensuring that system access remains exclusive to authorized individuals as per predefined access control policies. This layer employs diverse authentication methods, ranging from conventional username/password setups to sophisticated techniques like fingerprinting or iris recognition, tailored to suit the architecture's specifics. Essentially, it serves as the primary interface for users, responsible for validating user credentials based on systemspecific policies.

1.5.1.2 Access Control Management Layer

The Access Control Management layer serves as a foundational support system for both Layer 1 and Layer 3, providing essential services crucial for these layers to fulfill their respective objectives. Its services encompass role definition, establishing associated access control policies, and outlining specifications for voting transactions. Role definition and management form the core of access control operations in Layer 1, while voting transaction definitions aid in mapping blockchain-based transactions for processing in Layer 3. Essentially, this layer synchronizes the functionalities of the proposed system by providing fundamental elements necessary for each individual layer's operations.

1.5.1.3 Transaction Management Layer

The E-Voting Transaction Management layer serves as the central core within the architecture. Here, the e-voting transaction constructed at the Role Management/Transactions layer is linked to the blockchain transaction for mining purposes. This linked transaction incorporates the voter's provided credentials from Layer 1, such as the voter's fingerprint, for authentication. This data is utilized to generate a cryptographic hash that contributes to forming the transaction ID. The verification of these credentials is intended to occur at the User Interaction and Front-end Security layer (Layer 1). The process involves several virtual node instances participating in mining to finalize the transaction's entry into the blockchain.

1.5.1.4 Ledger Synchronization Layer

The Ledger Synchronization layer is responsible for harmonizing the Multichain ledger with a dedicated local application database, utilizing an established database technology. Votes cast are logged in the backend data tables of this database. Voters are furnished with a unique identifier upon the immediate addition of their vote into the blockchain ledger, enabling them to track their votes. The security of votes relies on blockchain technology, employing cryptographic hashes to ensure secure end-to-end communication. Additionally, voting results are stored within the application's database, intending to ease auditing processes and enable further operations at subsequent stages.

1.5.1.5 ALGORITHM USED

Blockchains function based on consensus algorithms, which enable agreement among distributed nodes. These mechanisms are pivotal in fostering reliability, trust, and security within the network. Consensus methods like proof of work (PoW) or proof of stake (PoS) act as vital safeguards, preventing unauthorized validation of inaccurate transactions and bolstering network security. They are crucial for maintaining the confidentiality and integrity of shared information across the blockchain. Additionally, Elliptic Curve Cryptography (ECC) is utilized. ECC, as a form of asymmetric cryptography, relies on the mathematical principles of elliptic curves for encryption.

$y^2 = x^3 + ax + b$…………..

Working of ECC

1. **Key Pair Generation**: - Within Elliptic Curve Cryptography (ECC), a user's private key is randomly generated, while the corresponding public key is generated using elliptic mathematics. Both the private and public keys are maintained in secrecy to ensure the security of the cryptographic system.

2. **Public Key Distribution**: - The public key is shared with the server or other person who wants to send encrypted data. The public key is safe to distribute widely.

3. **Encrypting the data**: - When the server or other person wants to send the data, the data is encrypted with the help of recipient’s public key.

4. **Decryption**: - The recipient possesses a corresponding private key, utilized specifically for decrypting the data. The mathematical properties of elliptic curves guarantee the efficiency of the private key in executing this decryption process.

Consider a scenario where two individuals, referred to as Person 1 and Person 2, engage in communication and data exchange. Both parties share a mutual elliptic curve equation along with a generator point denoted as G.

Let private keys of ***Person 1*** and ***Person 2*** are $\boldsymbol{nA}$ and $\boldsymbol{nB}$ respectively. Now, public keys of both are given as,

$$K_1 = nAG$$

and $$K_2 = nBG$$

If Person 1 intends to send a message M to Person 2, they utilize the public key of Person 2 to encrypt the message. The resulting ciphertext is computed as follows:

$$C = \{\lambda G, M + \lambda K2 \}$$

where $\boldsymbol{\lambda}$ is any random number, which makes sure that for same message each time a different cipher text is generated. This will make it hard for someone who is trying to decrypt the message illegally.

Now for decryption process of message, person 2 can decrypt the message by subtracting the coordinate of λG multiplied by nB from $M + \lambda K2$. The decrypted message is given by,

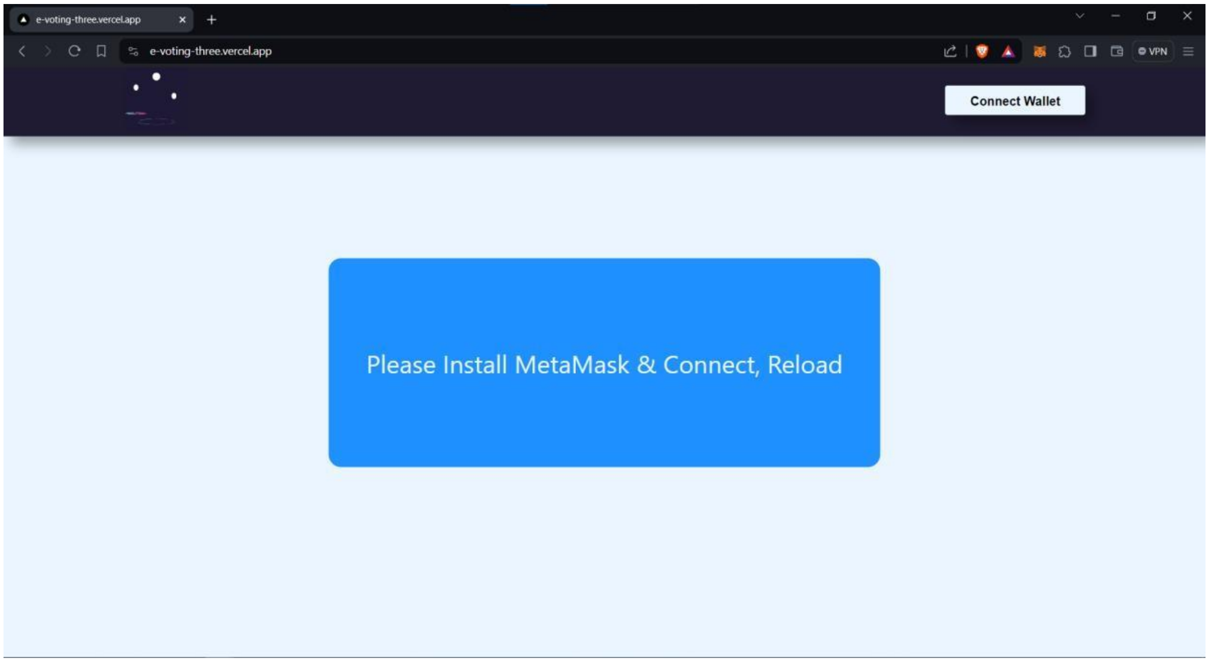
$$M = \{M + \lambda K2 - nB\lambda G\}$$

(This multiplication is not simple algebraic multiplication, but it is multiple addition of points - geometrical).

1.5.1.4 IMPLEMENTATION

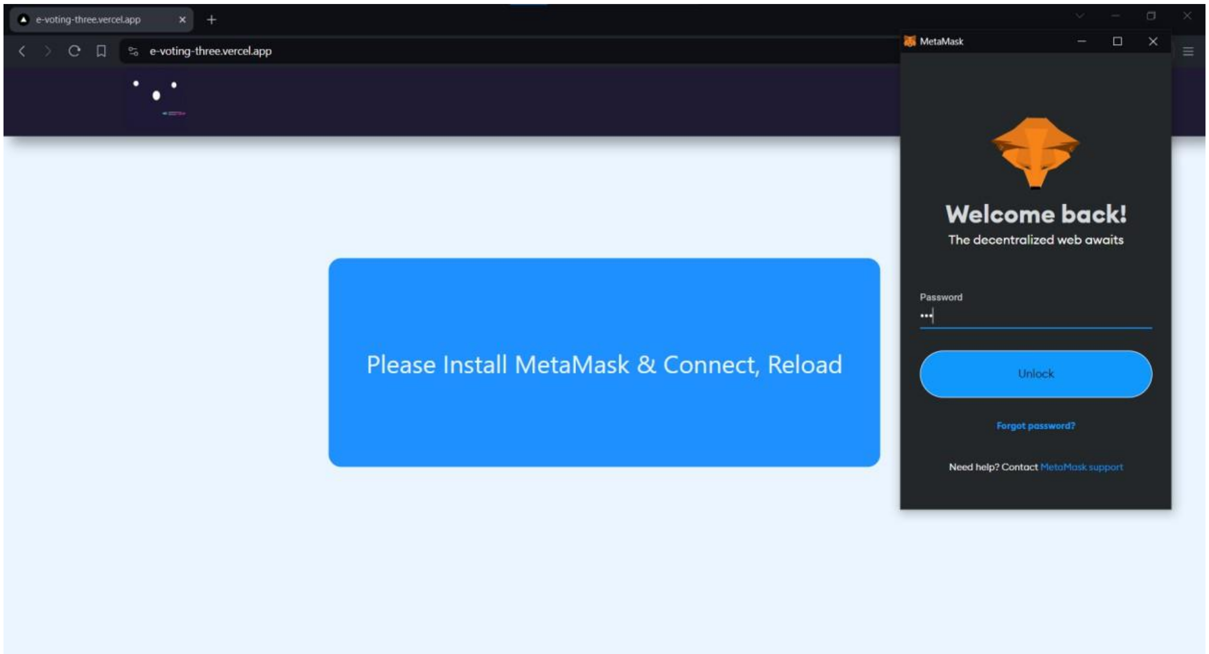
Description of the implementation

1. Logging in to the web application with the help of unique address on which Smart contract is deployed. Only the deployed address has administration privileges.



Please Install MetaMask & Connect, Reload

Fig 1.2.Home page of e-voting system



Please Install MetaMask & Connect, Reload

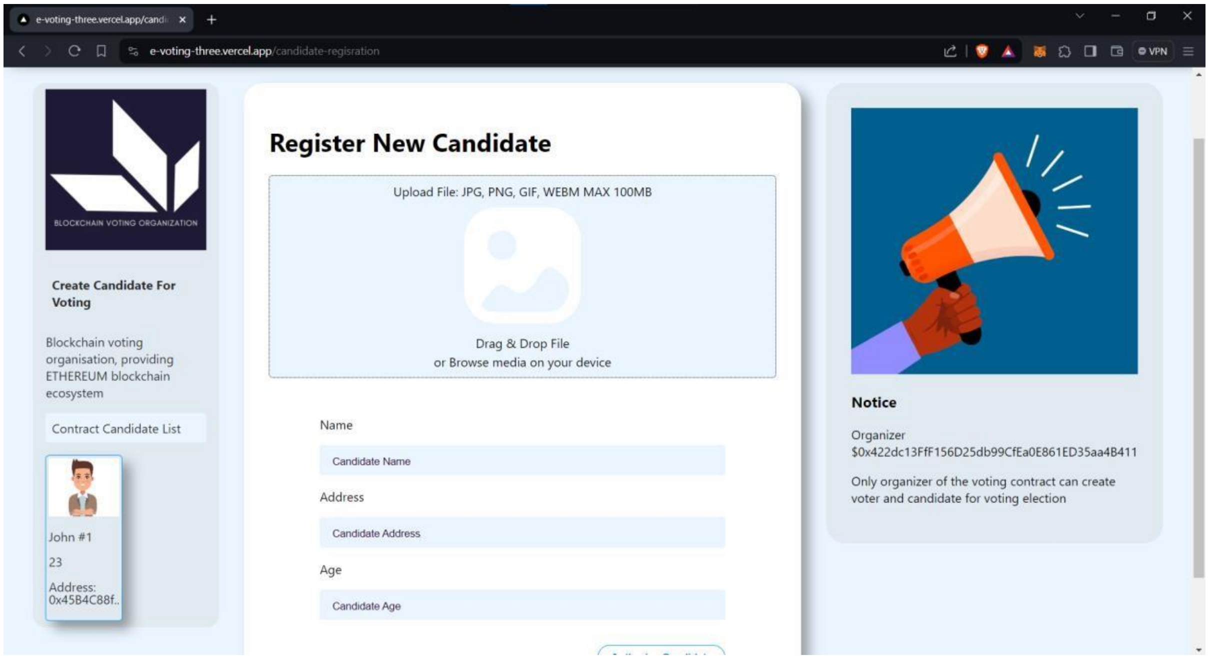
Welcome back!
The decentralized web awaits

Need help? Contact MetaMask support

Fig 1.3 Logging in to MetaMask

Registering the candidate

Navigating to the candidate registration page and registering the candidate. Entering the candidate details with the unique address. Uploading the image to the IPFS using the upload image button. Entering name address and other details.


e-voting-three.vercel.app/candi
e-voting-three.vercel.app/candidate-regisration
VPN

BLOCKCHAIN VOTING ORGANIZATION

Create Candidate For Voting

Blockchain voting organisation, providing ETHEREUM blockchain ecosystem

Contract Candidate List

John #1

23

Address: 0x45B4C88f..

Register New Candidate

Upload File: JPG, PNG, GIF, WEBM MAX 100MB

Drag & Drop File
or Browse media on your device

Name

Candidate Name

Address

Candidate Address

Age

Candidate Age

Notice

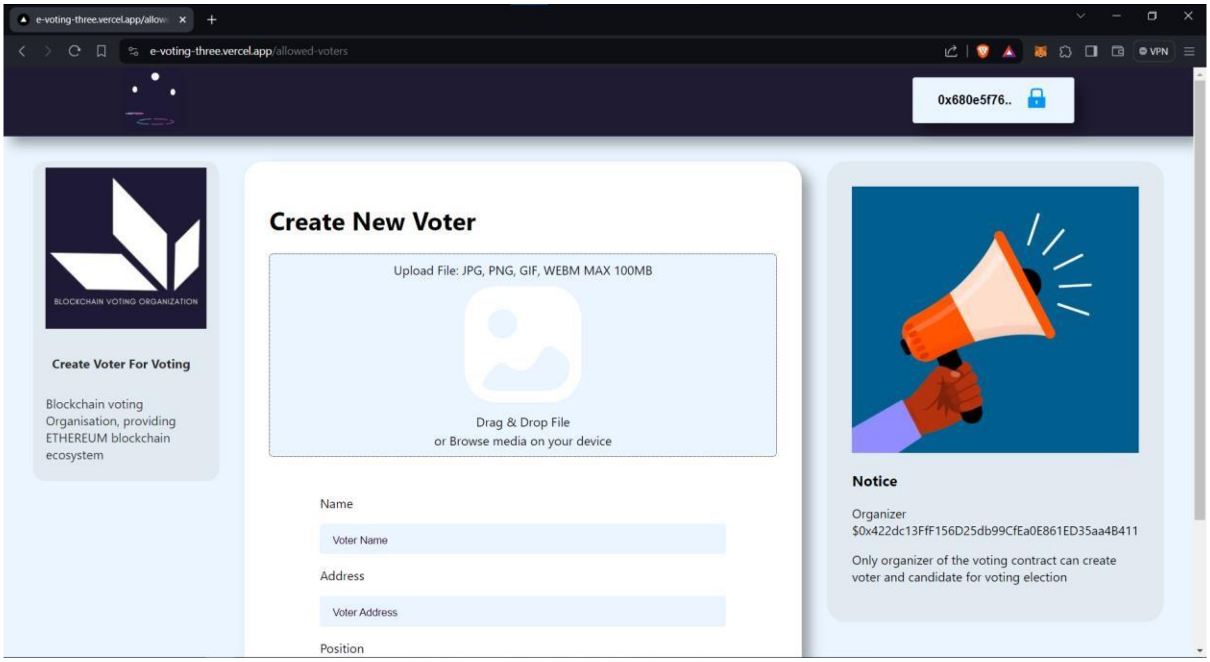
Organizer
$0x422dc13FfF156D25db99CfEa0E861ED35aa4B411

Only organizer of the voting contract can create voter and candidate for voting election

Fig 1.4 Registering the Candidate

Registering the Voter

Registering the voter same as the candidate registration.


e-voting-three.vercel.app/allow
e-voting-three.vercel.app/allowed-voters
VPN
0x680e5f76..

BLOCKCHAIN VOTING ORGANIZATION

Create Voter For Voting

Blockchain voting Organisation, providing ETHEREUM blockchain ecosystem

Create New Voter

Upload File: JPG, PNG, GIF, WEBM MAX 100MB

Drag & Drop File
or Browse media on your device

Name

Voter Name

Address

Voter Address

Position

Notice

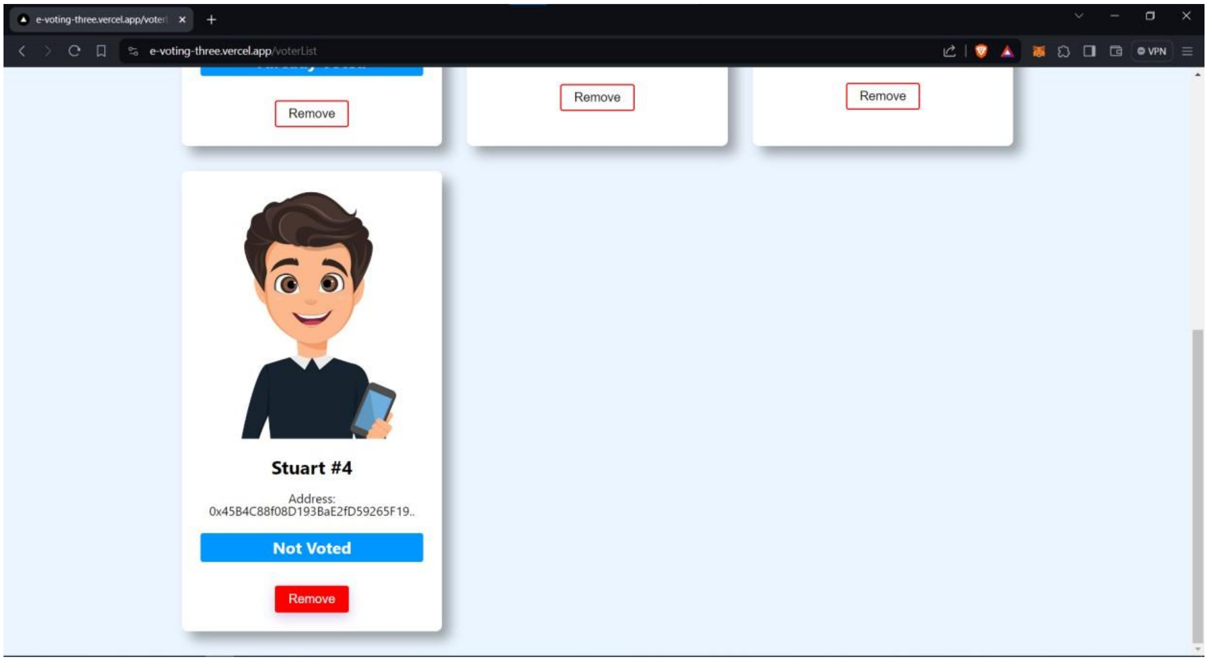
Organizer
$0x422dc13FfF156D25db99CfEa0E861ED35aa4B411

Only organizer of the voting contract can create voter and candidate for voting election

Fig 1.5 Registering the Voter

Voting process

Voter’s logging to the application using registered account through MetaMask. After successful login the candidates are shown at the homepage itself. Voters vote by clicking the ‘vote’ button.


e-voting-three.vercel.app/voterl
e-voting-three.vercel.app/voterList
VPN
Remove
Remove
Remove
Stuart #4
Not Voted
Remove

Address:
0x45B4C88f08D193BaE2fD59265F19..

Fig 1.6 Registered Candidates

This is the whole mechanism of the voting process done by the proposed E-voting system where administrators can register and remove candidates and voters. Voters can vote to their choice of Candidates. As the whole process is done in the form of transaction on Ethereum blockchain it makes it more reliable.

1.6 UNIQUE FEATURES OF THE SYSTEM

Blockchain Integration: The system will leverage blockchain technology to provide an immutable and transparent ledger for recording and storing voting transactions. This ensures that each vote is securely encrypted and timestamped, preventing tampering or manipulation of election results.

Decentralized Verification: Through the decentralized nature of blockchain, the voting process will be verified by a distributed network of nodes, eliminating the reliance on a central authority and enhancing the system's resilience against cyber attacks and fraud.

Voter Anonymity: While ensuring authentication and security, the system will maintain the anonymity of voters, protecting their privacy and confidentiality throughout the voting process.

Voter Education and Engagement: The project will include initiatives to educate voters about the benefits and procedures of online voting, fostering greater participation and trust in the electoral process among citizens of all demographics.

Immutable Voter Records: Utilizing blockchain technology, the system will maintain immutable records of voter registration, ensuring that once registered, a voter's information cannot be altered or tampered with, thereby enhancing the integrity of the voter database.

CHAPTER 2

REQUIREMENT ANALYSIS AND SYSTEM SPECIFICATION

2.1 FEASIBILITY STUDY

1. Introduction:

Online voting systems have the potential to revolutionize democratic processes by making voting more accessible, secure, and transparent. Leveraging blockchain technology can address many of the challenges associated with traditional voting systems, such as fraud, tampering, and logistical issues. This feasibility study examines the technical, economical, and operational aspects of implementing an online voting system using blockchain technology.

2. Technical Feasibility:

a. Blockchain Infrastructure:

Assess the scalability of blockchain technology to handle large-scale voting operations. Evaluate different blockchain platforms (e.g., Ethereum, Hyperledger) for suitability in terms of transaction throughput, consensus mechanisms, and smart contract capabilities. Determine the level of decentralization required to ensure security and integrity of the voting process.

b. Security Measures:

Implement cryptographic techniques to ensure the anonymity and confidentiality of votes. Design robust authentication mechanisms to prevent unauthorized access and ensure voter eligibility. Evaluate methods for preventing double voting and ensuring the integrity of the voting process.

c. User Experience:

Develop user-friendly interfaces for both voters and election administrators. Test the system for usability and accessibility across different devices and platforms. Incorporate features such as voter verification, ballot preview, and receipt generation to enhance transparency and trust in the system.

3. Economical Feasibility:

a. Cost Analysis:

Estimate the initial investment required for developing the online voting system, including software development, infrastructure setup, and security measures. Assess ongoing operational costs, including maintenance, hosting, and support services. Compare the costs of the proposed online voting system with traditional voting methods to determine cost-effectiveness.

b. Return on Investment (ROI):

Evaluate the potential savings achieved through increased efficiency, reduced administrative overhead, and lower printing and distribution costs. Consider the long-term benefits of improved voter participation, enhanced transparency, and trust in the electoral process.

4. Operational Feasibility:

a. Legal and Regulatory Compliance:

Ensure compliance with relevant laws and regulations governing elections, data protection, and cybersecurity. Collaborate with legal experts to address legal challenges associated with online voting, such as jurisdictional issues and dispute resolution mechanisms.

b. Stakeholder Engagement:

Engage with key stakeholders, including government agencies, election authorities, political parties, and civil society organizations, to garner support for the adoption of online voting. Conduct public awareness campaigns to educate voters about the benefits and security measures of the online voting system.

c. Risk Management:

Identify potential risks and vulnerabilities associated with the online voting system, such as cyber attacks, technological failures, and voter coercion. Develop contingency plans and mitigation strategies to address these risks and ensure the integrity and reliability of the voting process.

5. Conclusion:

The feasibility study demonstrates that an online voting system using blockchain technology holds significant promise in enhancing the democratic process by providing a secure, transparent, and accessible platform for conducting elections. However, successful implementation requires careful consideration of technical, economical, and operational factors, as well as proactive measures to address potential challenges and risks. With proper planning, collaboration, and stakeholder engagement, the adoption of online voting systems can contribute to a more inclusive and trustworthy electoral system.

2.2 SOFTWARE REQUIREMENT SPECIFICATION DOCUMENT

1. Introduction

1.1 Purpose

The purpose of this document is to provide a detailed software requirements specification for an electronic voting system using blockchain technology. The system will enable voters to cast their votes securely and transparently in a way that prevents tampering, fraud, and other malicious activities. This document will adhere to the standards set by the IEEE for software requirements specification.

1.2 Document Conventions

This document follows the IEEE standard for software requirements specification conventions.

1.3 Intended Audience and Reading Suggestions

This document is intended to provide suggestions to developers, project managers etc. but it is readable for every person.

1.4 Product Scope

The e-voting system using blockchain technology will be a web-based platform that allows eligible voters to cast their votes electronically. The system will use blockchain technology to ensure that each vote is recorded securely, anonymously, and immutably. The system will be designed to meet the requirements of various electoral systems, including single or multiple candidate elections, approval voting, and preferential voting.

2. Overall Description

2.1 Product Perspective

The e-voting system using blockchain technology is a standalone software product that allows voters to cast their votes electronically in a secure and transparent manner. The system will be built using blockchain technology, which provides a decentralized, immutable, and tamperproof record of the voting process.

2.2 Product Functions

The e-voting system will provide the following functions:

Registration of eligible voters.

Authentication of voters.

Ballot creation and distribution, Vote casting and recording.

Vote counting and tallying.

Result declaration.

2.3 User Classes and Characteristics

The e-voting system will be used by the following user classes:

Voters: eligible individuals who will use the system to cast their votes. Administrators: system administrators who will manage and oversee the system's operation and maintenance.

Election officials: officials who will use the system to create and, monitor the voting process, and declare the results.

2.4 Operating Environment

The e-voting system will be a web-based application accessible via a standard web browser. The system will require a stable internet connection, and it will be hosted on a secure server.

2.5 Design and Implementation Constraint

The e-voting system will be built using blockchain technology, which requires specialized programming knowledge and expertise. The system will also need to comply with local and national regulations and guidelines regarding the conduct of online voting.

2.6 Assumption and Dependencies

The e-voting system assumes that users have access to a stable internet connection, and that they are familiar with basic computer skills such as using a web browser and entering data into online forms.

3. External Interface Requirements

3.1 User Interfaces

The e-voting system will provide a user-friendly interface that is easy to navigate and understand. The interface will be accessible via a standard web browser and will be designed to be compatible with a wide range of devices and platforms.

3.2 Hardware Interfaces

The e-voting system will require a stable internet connection and access to a secure server for hosting the application.

3.3 Software Interfaces

The e-voting system will be built using blockchain technology and will require specialized programming knowledge and expertise. The system will also need to integrate with other software and technologies as needed.

3.4 Usability

The system should be user-friendly and easy to use. The system should provide clear instructions and guidance to the users during the voting process. The system should also be accessible to users with disabilities.

4. System Features

4.1 System Feature 1: User Registration

The e-voting system will allow eligible voters to register for the system by providing their personal details, including their name, address, and identification documents. The system will verify the eligibility of the user and assign them a unique digital identity that will be used to authenticate the user during the voting process.

4.2 System Feature 2: Ballot Creation and Distribution

The e-voting system will allow election officials to create and distribute ballots for the election. The system will provide a user-friendly interface that will allow officials to customize the ballot based on the specific election and the candidates or issues being voted on. The system will also provide a mechanism for distributing the ballot to eligible voters securely.

4.3 System Feature 3: Vote Casting and Recording

The e-voting system will allow eligible voters to cast their votes electronically using their unique digital identity. The system will record each vote securely in the blockchain ledger, ensuring that the vote is anonymous and tamper-proof.

4.4 System Feature 4: Vote Counting and Tallying

The e-voting system will use blockchain technology to count and tally the votes cast during the election. The system will provide an accurate and transparent tally of the votes, and officials will be able to monitor the process in real-time.

4.5 System Feature 5: Result Declaration

The e-voting system will automatically declare the election results based on the votes cast and the tally calculated using the blockchain ledger. The system will provide an accurate and transparent record of the election results, ensuring the integrity of the process.

5. Other Non-Functional Requirements

5.1 Performance Requirements

The e-voting system must be able to handle a large volume of traffic during the election period, and the system must be able to process votes quickly and accurately. The system must also be able to withstand cyber-attacks and other security threats.

5.2 Safety Requirements

The e-voting system must be designed to prevent voter fraud and other forms of election manipulation. The system must also be designed to ensure the privacy and confidentiality of the voter's identity and the votes cast.

5.3 Security Requirements

The e-voting system must be designed with strong security features to prevent unauthorized access and tampering. The system must also provide a secure mechanism for user authentication and data encryption.

5.4 Software Quality Attributes

The e-voting system must be reliable, maintainable, and scalable. The system must be designed with a modular architecture that allows for easy updates and modifications as needed.

6. Other Requirements

The e-voting system must be tested thoroughly before deployment to ensure that it meets all of the functional and non-functional requirements. The system must also be monitored regularly to ensure that it continues to operate effectively and efficiently

Appendix A: Analysis Models

"PEST Analysis: Understanding the External Environment" by MindTools. This model can be used to analyze the political, economic, social, and technological factors that can impact the implementation and adoption of e-voting systems. For example, political factors may include government regulations and policies around voting, while technological factors may include the availability of secure and reliable hardware and software. "SWOT Analysis: Discover New Opportunities, Manage and Eliminate Threats" by MindTools. This model can be used to identify the strengths, weaknesses, opportunities, and threats facing e-voting systems. For example, strengths may include the potential for increased voter turnout and accessibility, while weaknesses may include concerns around security and privacy. "Porter's Five Forces: Analyzing the Competition" by Harvard Business Review. This model can be used to evaluate the competitive forces that can impact the e-voting industry, including the bargaining power of suppliers (such as technology providers), the threat of substitutes (such as traditional paper ballots), and the intensity of competitive rivalry between e-voting providers. "Value Chain Analysis: Identify Activities That Create Value" by MindTools. This model can be used to identify the key activities involved in the implementation and operation of e-voting systems, from the design and development of the software and hardware components to the training of election officials and the maintenance of the system. It can help identify areas where value can be added, or costs can be reduced.

Appendix B: To Be Determined List

Security Measures:

The type of encryption to be used for the blockchain.

The level of encryption for voter data and their votes.

The security measures for voter identification and verification.

The protocol for handling errors, malfunctions, or attacks.

Governance and Administration:

The governance structure for the e-voting system.

The roles and responsibilities of the parties involved, including the election commission, blockchain administrators, and third-party auditors.

The process for selecting blockchain administrators and auditors.

The process for resolving disputes or challenges to the election results.

Transparency and Accountability:

The level of transparency for the blockchain, including the visibility of the voting process and results.

The auditing and reporting procedures for the e-voting system.

The process for verifying the authenticity of votes and ensuring the accuracy of the results.

The availability of the data for public access and analysis.

Accessibility and Inclusivity:

The accessibility of the e-voting system for voters with disabilities or limited internet access.

The language options available for the e-voting system.

The usability of the system for non-technical users, including the elderly or those with limited computer skills.

The measures in place to ensure equal access and opportunity for all voters.

Sustainability:

The environmental impact of the e-voting system, including energy consumption and carbon footprint.

The cost-effectiveness of the system and its scalability for large-scale elections.

The ability to upgrade or maintain the system over time.

The potential for integrating the e-voting system with other government services and systems.

2.3 SDLC MODEL TO BE USED

Agile Software Development Model with Iterative Prototyping:

1. Planning Phase:

Project Initiation: Define project objectives, scope, and requirements for the online voting system.

Stakeholder Analysis: Identify stakeholders, including students, faculty, election authorities, and IT team members.

Initial Requirements Gathering: Conduct interviews, surveys, and workshops to gather initial requirements and prioritize features.

2. Development Phase:

Iteration 1: Minimum Viable Product (MVP) Development:

Develop a basic version of the online voting system with essential features. Implement blockchain integration for vote recording and verification.

Focus on building a secure and user-friendly interface for casting votes.

Iteration 2: Feedback and Enhancement:

Gather feedback from stakeholders and users on the MVP.

Identify areas for improvement and additional features based on feedback.

Incorporate enhancements and iterate on the system design and functionality.

3. Testing Phase:

Unit Testing: Conduct unit tests to ensure the correctness of individual components and modules.

Integration Testing: Verify the integration of different system components, including blockchain integration, authentication mechanisms, and user interfaces.

Security Testing: Perform security assessments to identify vulnerabilities and ensure robustness against attacks, including penetration testing and code reviews.

4. Deployment Phase:

Beta Testing: Release the online voting system to a limited group of users for beta testing.

User Acceptance Testing (UAT): Allow stakeholders and end-users to test the system in a simulated environment and provide feedback.

Gradual Rollout: Deploy the system gradually, starting with a small-scale pilot and scaling up based on feedback and performance.

5. Maintenance Phase:

Ongoing Support: Provide ongoing support and maintenance for the deployed system, including bug fixes, security updates, and performance optimizations.

Continuous Improvement: Continuously gather feedback from users and stakeholders to identify opportunities for further enhancement and refinement.

6. Risk Management:

Risk Identification: Identify potential risks and uncertainties associated with the development and deployment of the online voting system.

Risk Mitigation: Develop risk mitigation strategies to address identified risks, such as security vulnerabilities, usability issues, and regulatory compliance.

Contingency Planning: Prepare contingency plans to mitigate the impact of unforeseen events or failures during the development and deployment process.

By adopting an Agile Software Development Model with Iterative Prototyping, the project team can effectively manage the complexity of developing an online voting system using blockchain technology. This approach allows for continuous feedback, collaboration, and adaptation throughout the project lifecycle, leading to the successful delivery of a secure, transparent, and user-friendly voting system.

CHAPTER 3

REQUIREMENT ANALYSIS AND SYSTEM SPECIFICATION

3.1 DETAIL DESIGN

1. System Architecture:

The system architecture will consist of multiple layers:

Presentation Layer: The user interface through which voters interact with the system.

Application Layer: The logic and business rules governing the voting process.

Blockchain Layer: The underlying blockchain network for recording and verifying votes.

Integration Layer: Interfaces for integrating with external systems (e.g., authentication, voter registration databases).

2. Components:

User Interface: Develop a user-friendly interface accessible via web or mobile devices. This interface should allow voters to authenticate securely, view available elections, cast votes, and verify their vote has been recorded accurately.

Authentication Module: Implement robust authentication mechanisms to verify the identity of voters securely. This may include methods such as two-factor authentication, biometrics, or digital signatures.

Voter Registration: Develop a module for voter registration, allowing eligible voters to register and verify their identity before participating in elections. This information will be stored on the blockchain to ensure transparency and integrity.

Voting Module: Design a module for casting votes securely. Each vote will be encrypted and stored as a transaction on the blockchain, ensuring immutability and auditability. Smart contracts will enforce rules such as voter eligibility and prevent double voting.

Blockchain Integration: Integrate the online voting system with a suitable blockchain platform (e.g., Ethereum, Hyperledger). Use smart contracts to implement voting rules, manage voting tokens, and record votes on the blockchain.

Security Layer: Implement robust security measures to protect against unauthorized access, tampering, and fraud. This includes encryption of sensitive data, secure communication protocols, and regular security audits.

3. Technical Considerations:

Scalability: Ensure the system can handle a large number of concurrent users and transactions, especially during peak voting periods. Consider techniques such as sharding or off-chain processing to improve scalability.

Privacy: Implement privacy-enhancing technologies to protect the anonymity of voters while still ensuring transparency and auditability. Techniques such as zero-knowledge proofs or ring signatures can be used to achieve this.

Resilience: Design the system to be resilient to various types of attacks, including DDoS attacks, network failures, and malicious actors. Implement redundancy, failover mechanisms, and disaster recovery plans to ensure continuous operation.

4. Data Management:

Data Storage: Store voter registration information, voting records, and other sensitive data securely on the blockchain. Use encryption and access control mechanisms to protect data confidentiality.

Data Retention: Define policies for data retention and archiving to comply with legal and regulatory requirements. Consider techniques such as data pruning or off-chain storage for managing blockchain data.

5. Compliance and Governance:

Regulatory Compliance: Ensure the online voting system complies with relevant laws and regulations governing elections, data protection, and cybersecurity. Collaborate with legal experts to address compliance requirements.

Auditability: Design the system to provide transparency and auditability throughout the voting process. Allow independent observers to verify the integrity of the system and audit voting records stored on the blockchain.

6. Testing and Deployment:

Testing: Conduct thorough testing of the online voting system, including functional testing, security testing, and performance testing. Use techniques such as penetration testing and code reviews to identify and address vulnerabilities.

Deployment: Deploy the system in a controlled environment initially, such as a pilot election or a limited group of users. Gradually scale up deployment based on feedback and performance metrics.

3.2 SYSTEM DESIGN USING DFD LEVEL 0 AND LEVEL 1

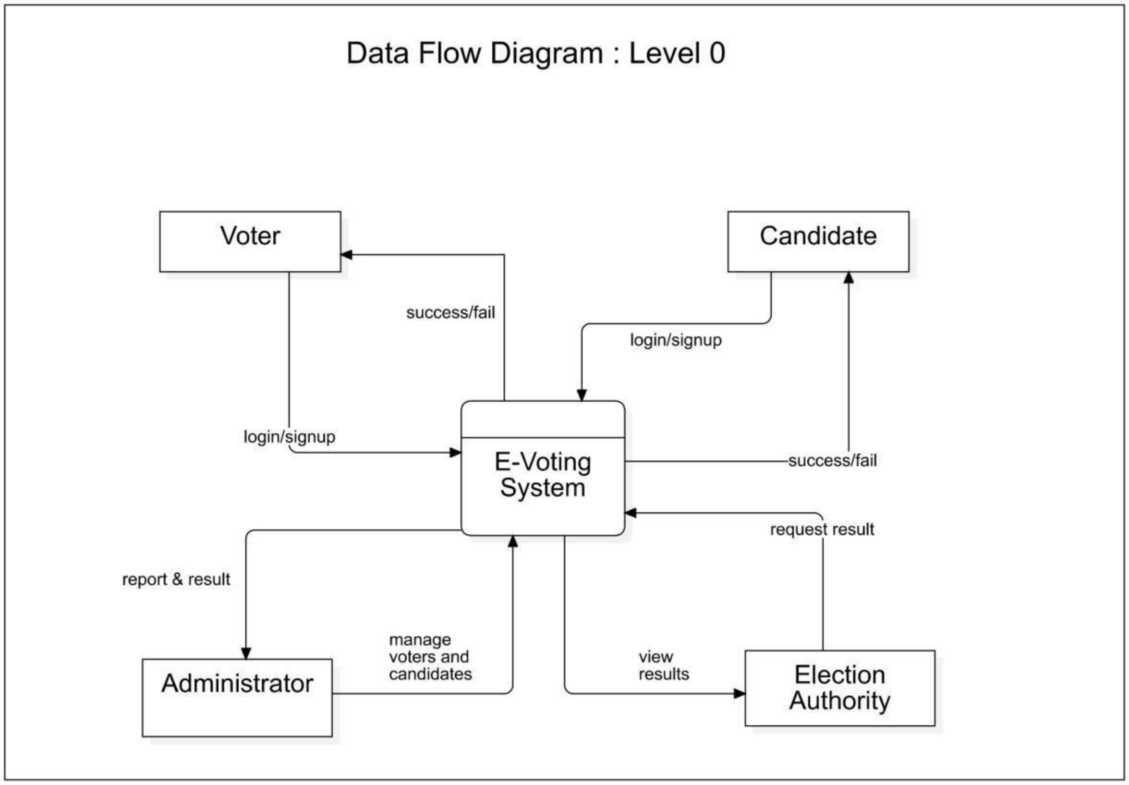


Fig 3.1 DFD Level 0

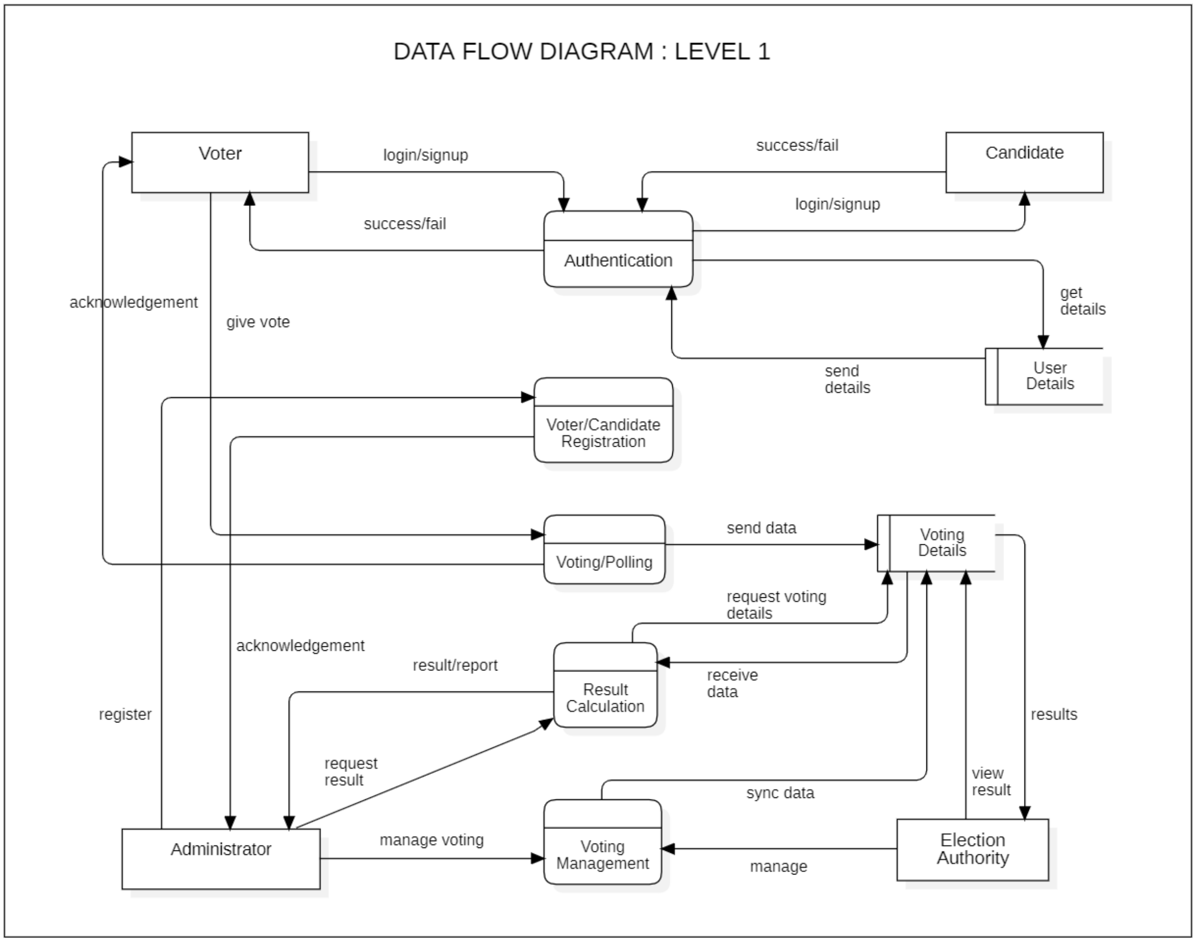


Fig 3.2 DFD Level 1

3.3 USE CASE DIAGRAM

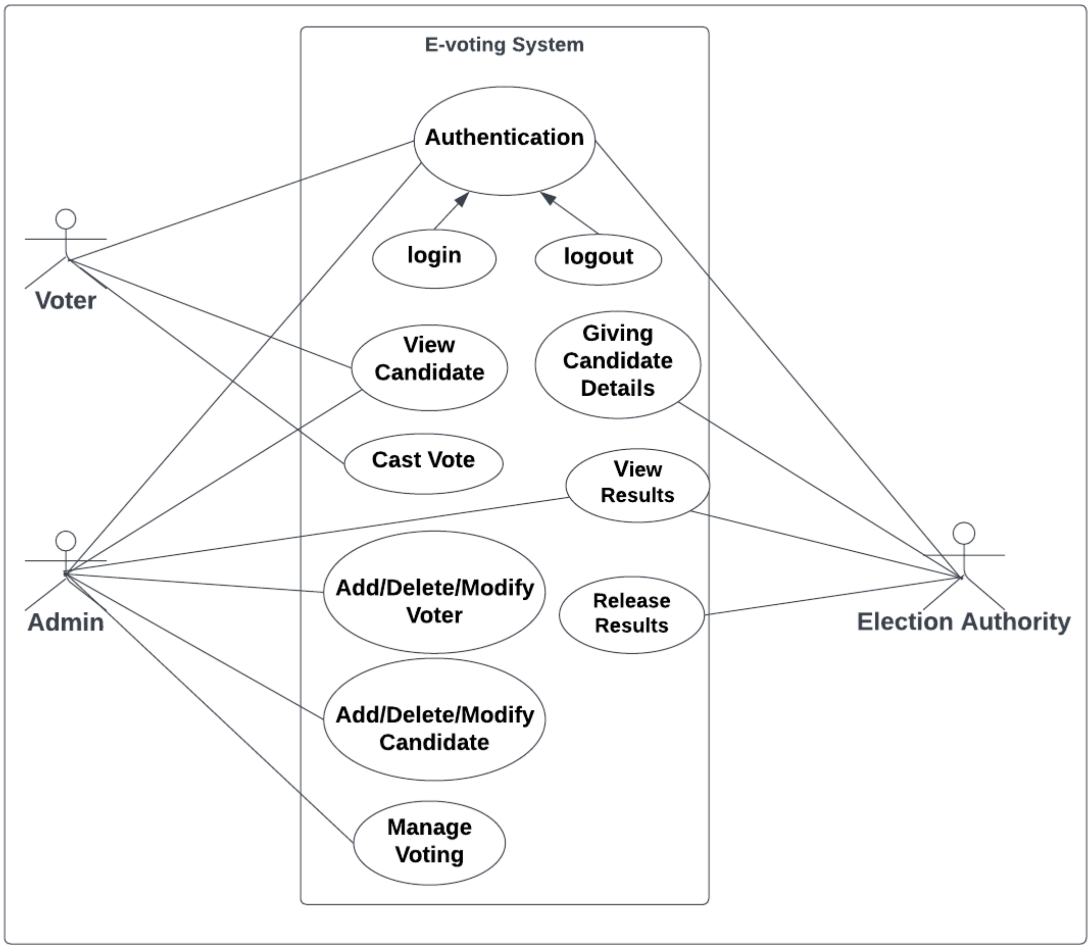
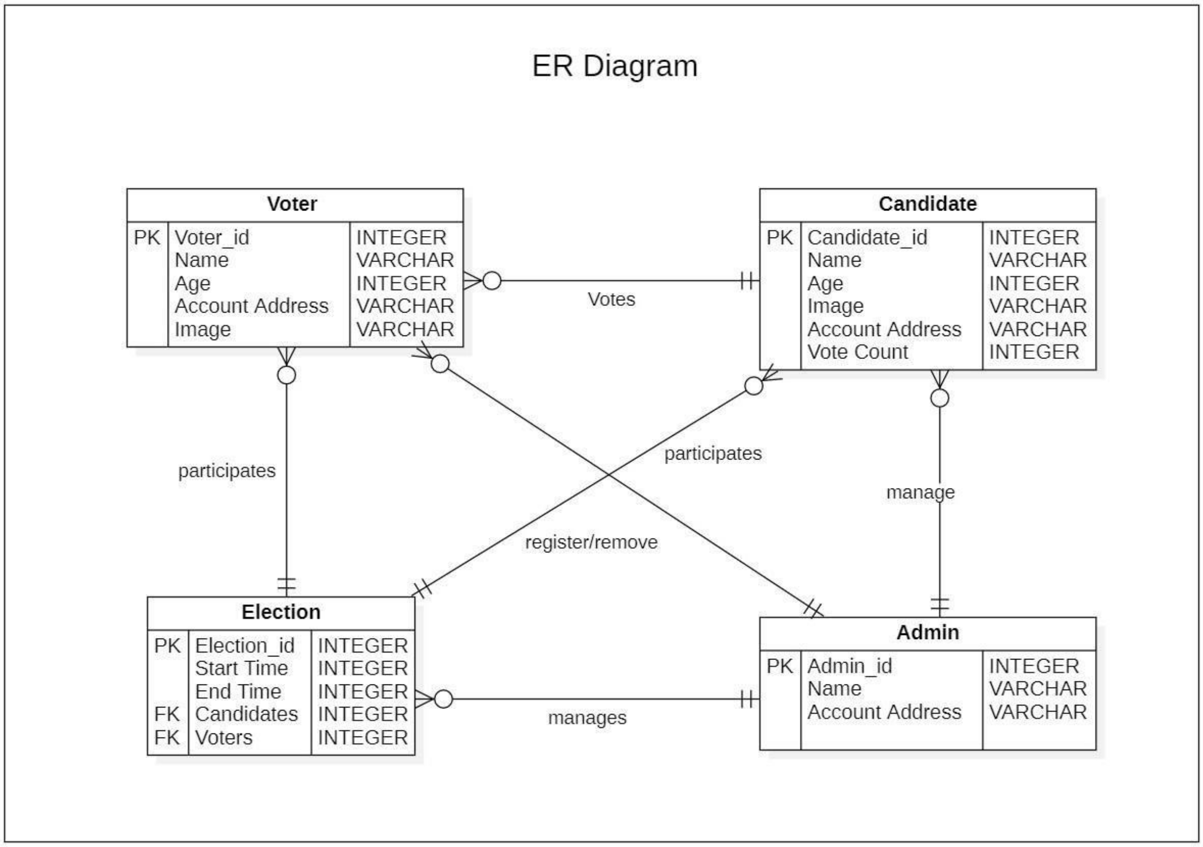


Fig 3.3 Use Case Diagram

3.4 DATABASE DESIGN



Voter		
PK	Voter_id	INTEGER
	Name	VARCHAR
	Age	INTEGER
	Account Address	VARCHAR
	Image	VARCHAR

Candidate		
PK	Candidate_id	INTEGER
	Name	VARCHAR
	Age	INTEGER
	Image	VARCHAR
	Account Address	VARCHAR
	Vote Count	INTEGER

Election		
PK	Election_id	INTEGER
	Start Time	INTEGER
	End Time	INTEGER
FK	Candidates	INTEGER
FK	Voters	INTEGER

Admin		
PK	Admin_id	INTEGER
	Name	VARCHAR
	Account Address	VARCHAR

Fig 3.4 ER Diagram

CHAPTER 4

IMPLEMENTATION, TESTING, AND MAINTENANCE

Introduction

Building a secure electronic voting system that offers the fairness and privacy of current voting schemes, while providing the transparency and flexibility offered by electronic systems has been a challenge for a long time. In this work-in-progress paper, we evaluate an application of blockchain as a service to implement distributed electronic voting systems. The paper proposes a novel electronic voting system based on blockchain that addresses some of the limitations in existing systems and evaluates some of the popular blockchain frameworks for the purpose of constructing a blockchain-based e-voting system In particular, we evaluate the potential of distributed ledger technologies through the description of a case study; namely, the process of an election, and the implementation of a blockchain-based application, which improves the security and decreases the cost of hosting a nationwide election. This technology will improve the trust of voters that there action is secure.

Scope

In

Scope

This shows the aspects of the E-Voting System that are within the scope of our testing efforts. It includes the following:

- Voting Interface: Testing the user interface to ensure that it is user-friendly, accessible, and accurately captures voter choices.
- Vote Encryption and Decryption: Ensuring the cryptographic processes used to secure votes are functioning correctly.
- Vote Recording: Validating that each vote is accurately recorded on the blockchain ledger.
- Security Measures: Verifying the effectiveness of security measures to protect against unauthorized access.
- Performance: Assessing the system's responsiveness and scalability to handle a significant volume of votes.
- Usability: Evaluating the overall user experience and accessibility of the system.

Out of Scope

For our E-Voting System, the following are out of scope:

- **No. of voters:** Only limited number of entities can be tested.

- **Blockchain Technology**: The core blockchain technology itself, as it should have undergone extensive testing during its development.
- **Hardware Infrastructure:** The physical hardware infrastructure supporting the system is out of scope.
- **Network Infrastructure:** The broader network infrastructure is not within our testing purview.

Quality Objective

Our quality objectives are:

- **Accuracy:** Ensuring that each vote is recorded and counted accurately.
- **Security:** Guaranteeing the integrity, confidentiality, and availability of the voting data.
- **Usability:** Creating a user-friendly system that is accessible to a broad range of voters.
- **Performance:** Ensuring the system can handle a significant load without degradation.

Roles and Responsibilities

Detail description of the Roles and responsibilities of team members

- Test Manager (Ashish Kumar Gupta)
- Tester (Ashish Kumar Gupta) – Tested the test cases.
- Developers (Ashish Kumar Gupta, Saurabh Pundir, Aditya Aggarwal) - Addressed and resolved issues identified during testing.
- Project Manager (Mr. Akash Goel) - Oversee the project's progress and ensure alignment with testing efforts.

Test Methodology

Overview

For our E-Voting System project, we have adopted the Waterfall test methodology. The choice of this methodology is driven by several key factors specific to our project's requirements and characteristics:

- As our requirements are clearly defined and stable we are using Waterfall methodology.
- The voting process is subject to strict regulations, and a Waterfall approach allows for thorough planning and documentation to ensure compliance with legal and security requirements.
- Waterfall provides a structured approach.

Test Levels

Following are the testing levels that are defined based on the case of the E-Voting System:

- Unit Testing: Ensuring that the all the modules are working correctly.
- Integration Testing: Testing that all the integrated modules are working as expected.
- System Testing: Evaluating the entire system to verify its correctness and compliance with election regulations.
- User Acceptance Testing (UAT): Allowing end-users, including election officials and voters, to validate the system for usability and suitability.

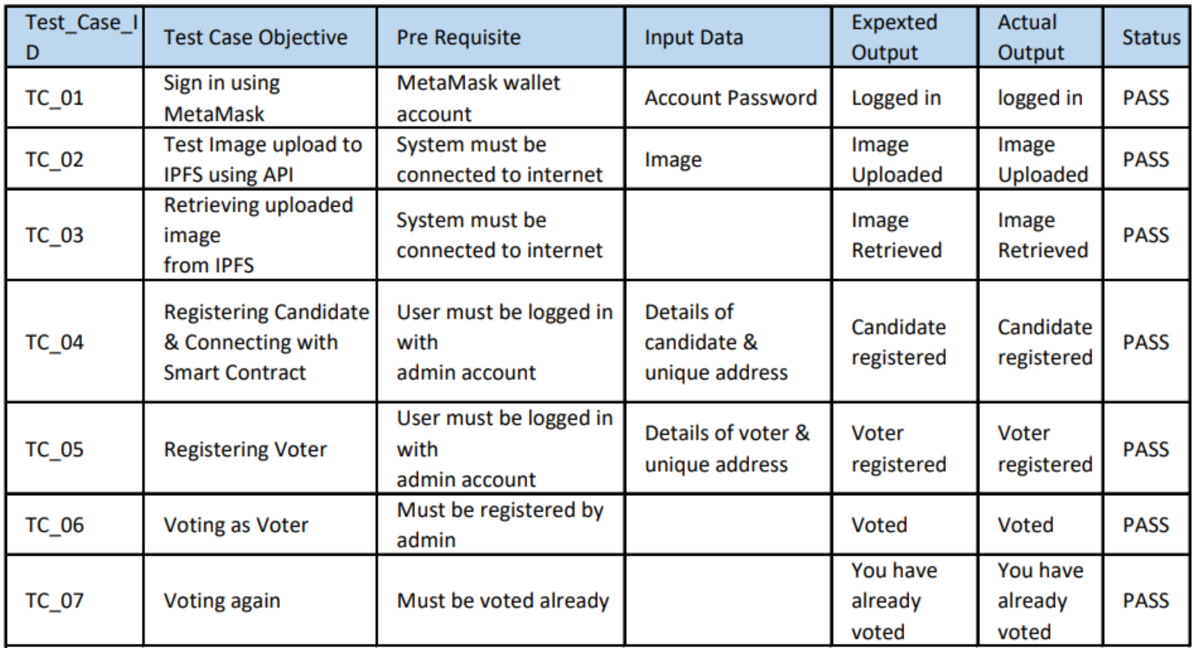
Suspension Criteria and Resumption Requirements

- Suspension Criteria: If a critical security vulnerability is discovered during requirements testing, further testing may be suspended until the issue is resolved.
- Resumption Criteria: In the case of the security vulnerability, testing can resume when the issue is fixed, and the system is confirmed secure.

Test Completeness

- All requirements are met.
- Compatibility: Software is compatible with other platforms, browsers, devices, OS.
- 100% test coverage
- All Test cases executed
- All open bugs are fixed or will be fixed in next release.

Manual Testing



Test_Case_ID	Test Case Objective	Pre Requisite	Input Data	Expexted Output	Actual Output	Status
TC_01	Sign in using MetaMask	MetaMask wallet account	Account Password	Logged in	logged in	PASS
TC_02	Test Image upload to IPFS using API	System must be connected to internet	Image	Image Uploaded	Image Uploaded	PASS
TC_03	Retrieving uploaded image from IPFS	System must be connected to internet		Image Retrieved	Image Retrieved	PASS
TC_04	Registering Candidate & Connecting with Smart Contract	User must be logged in with admin account	Details of candidate & unique address	Candidate registered	Candidate registered	PASS
TC_05	Registering Voter	User must be logged in with admin account	Details of voter & unique address	Voter registered	Voter registered	PASS
TC_06	Voting as Voter	Must be registered by admin		Voted	Voted	PASS
TC_07	Voting again	Must be voted already		You have already voted	You have already voted	PASS

Fig 4.1 Mannual Testing

Automation Testing

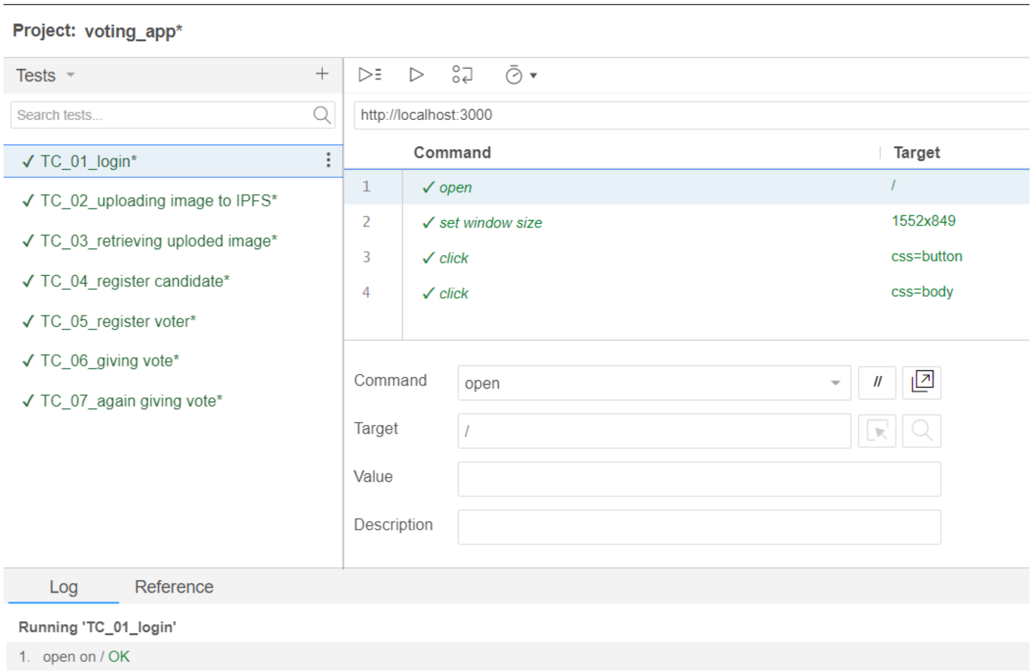
Test Case 1: Sign In Using Metamask

Test Description: This scenario evaluates the functionality of signing into the E-Voting System using MetaMask, a popular Ethereum wallet and gateway to blockchain applications.

Steps:

1. Open the E-Voting System application.
2. Click on the "Sign In" option.
3. Select "Sign in using MetaMask."
4. Connect the MetaMask extension.
5. Verify successful sign-in and access to the user's account.

Expected Result: The user can sign in using MetaMask, and their account is accessible within system.


Project: voting_app*

Tests

- ✓ TC_01_login*
- ✓ TC_02_uploading image to IPFS*
- ✓ TC_03_retrieving uploded image*
- ✓ TC_04_register candidate*
- ✓ TC_05_register voter*
- ✓ TC_06_giving vote*
- ✓ TC_07_again giving vote*

http://localhost:3000

	Command	Target
1	✓ *open*	/
2	✓ *set window size*	1552x849
3	✓ *click*	css=button
4	✓ *click*	css=body

Command	open
Target	/
Value	
Description	

Log Reference

```
Running 'TC_01_login'
1.  open on / OK
```

Fig 4.2 AutomationTesting

Logs:

- Running 'TC_01_login'20:47:36
- 1.open on / OK20:47:36
- 2.setWindowSize on 1552x849 OK20:47:37
- 3.click on css=button OK20:47:37
- 4.click on css=body OK20:47:39
- 'TC_01_login' completed successfully20:47:39

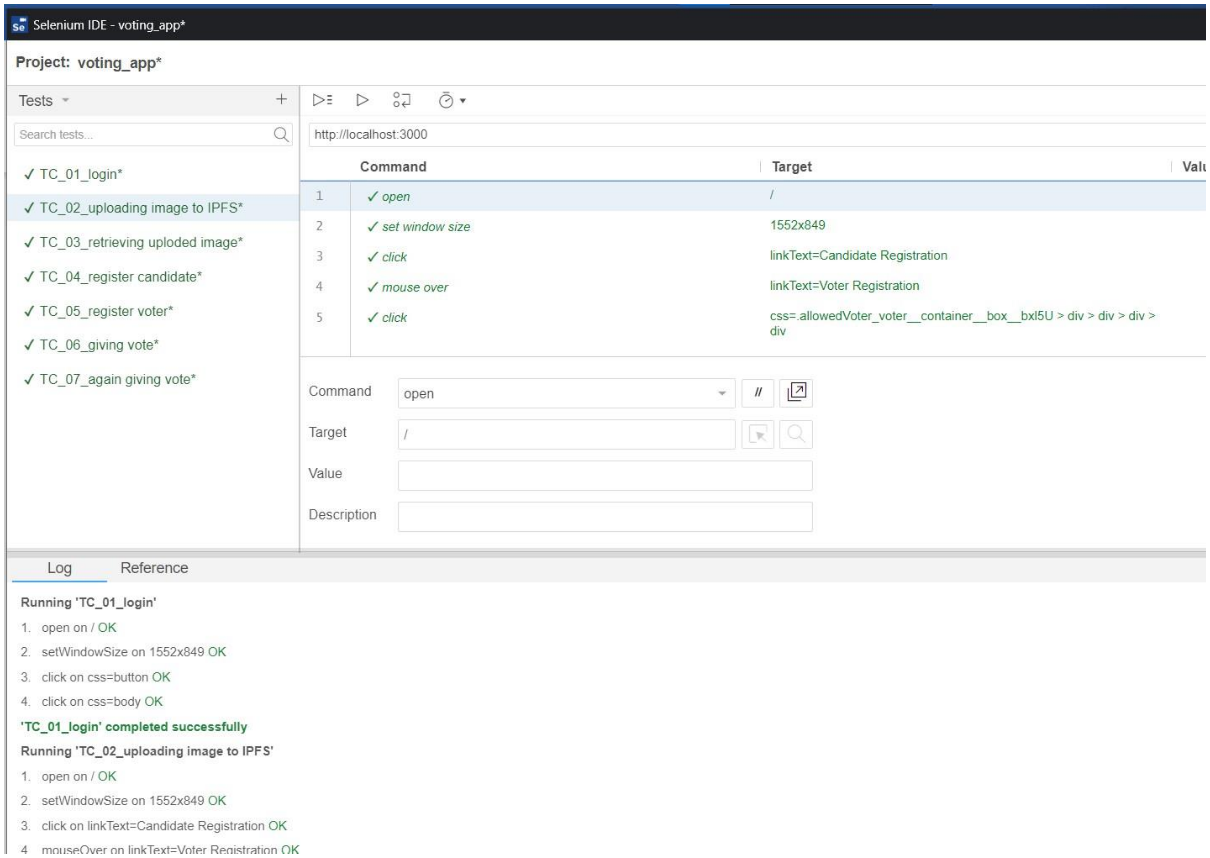
Test Case 2: Test Image Upload to IPFS Using API:

Test Description: This scenario examines the capability of the system to upload an image to IPFS via an API.

Test Steps:

1. Log in to the E-Voting System.
2. Navigate to the "Candidate Registration" section.
3. Upload an image using the upload button.
4. Confirm successful image upload.

Expected Result: The system allows users to upload images to IPFS through the IPFS API, and the image is successfully uploaded.


Selenium IDE - voting_app*
Project: voting_app*
Tests
Search tests...
✓ TC_01_login*
✓ TC_02_uploading image to IPFS*
✓ TC_03_retrieving uploded image*
✓ TC_04_register candidate*
✓ TC_05_register voter*
✓ TC_06_giving vote*
✓ TC_07_again giving vote*
Log
Reference

http://localhost:3000

	Command	Target	Valu
1	✓ open	/	
2	✓ set window size	1552x849	
3	✓ click	linkText=Candidate Registration	
4	✓ mouse over	linkText=Voter Registration	
5	✓ click	css=.allowedVoter_voter__container__box__bxl5U > div > div > div > div	

Command: open

Target: /

Value:

Description:

```
Running 'TC_01_login'
1.  open on / OK
2.  setWindowSize on 1552x849 OK
3.  click on css=button OK
4.  click on css=body OK
'TC_01_login' completed successfully
Running 'TC_02_uploading image to IPFS'
1.  open on / OK
2.  setWindowSize on 1552x849 OK
3.  click on linkText=Candidate Registration OK
4.  mouseOver on linkText=Voter Registration OK
```

Fig 4.3 Test Case 2

Test Case 3: Retrieving Uploaded Image from IPFS (after upload automatically)

Test Description: This scenario tests the system's ability to retrieve an image previously uploaded to IPFS.

Test Steps:

1. Upload the image to the IPFS.
2. It will automatically retrieve and show the image.
3. Verify that the correct image is displayed.

Expected Result: The system successfully retrieves and displays the uploaded image from IPFS.

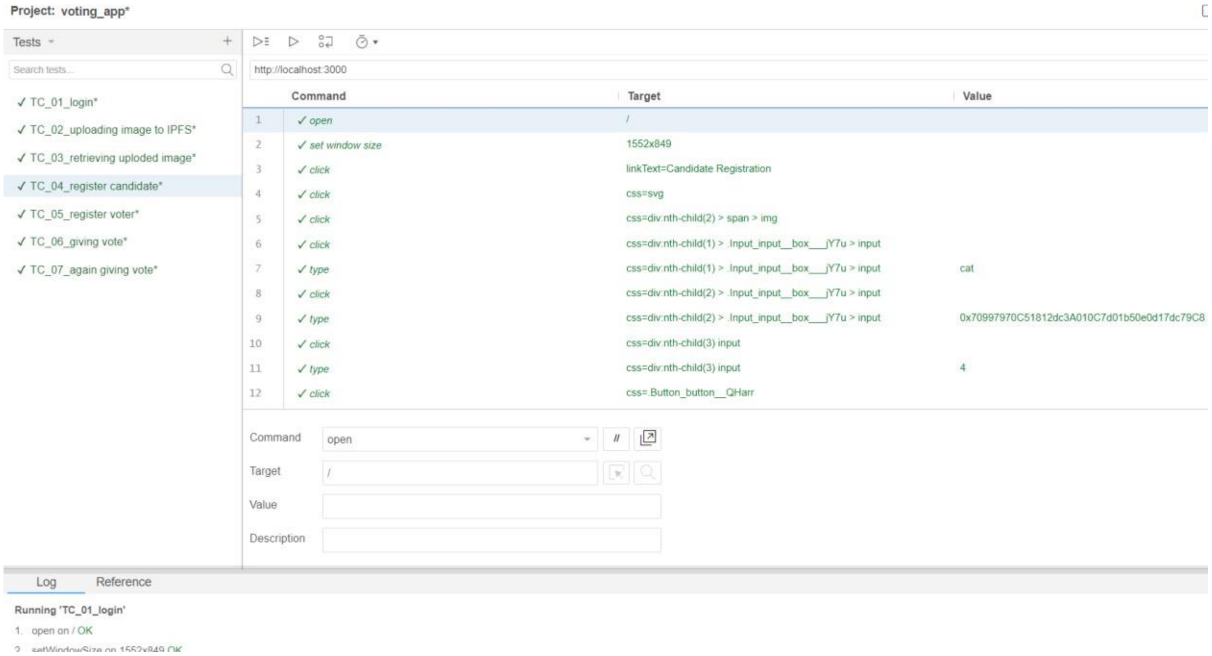
Test Case 4: Registering Candidate & Connecting with Smart Contract

Test Description: This scenario assesses the system's functionality to register a candidate and establish a connection with the underlying Smart Contract.

Test Steps:

1. Log in to the E-Voting System as an administrator.
2. Access the "Candidate Registration" section.
3. Register a candidate, providing relevant details and uploading image.
4. Confirm the transaction process.

Expected Result: The system successfully registers the candidate and establishes a connection with the Smart Contract, allowing the candidate to participate in the election.



Project: voting_app*

Tests

- ✓ TC_01_login*
- ✓ TC_02_uploading image to IPFS*
- ✓ TC_03_retrieving uploded image*
- ✓ TC_04_register candidate*
- ✓ TC_05_register voter*
- ✓ TC_06_giving vote*
- ✓ TC_07_again giving vote*

http://localhost:3000

	Command	Target	Value
1	✓ open	/	
2	✓ set window size	1552x849	
3	✓ click	linkText=Candidate Registration	
4	✓ click	css=svg	
5	✓ click	css=div:nth-child(2) > span > img	
6	✓ click	css=div:nth-child(1) > .Input_input__box___jY7u > input	
7	✓ type	css=div:nth-child(1) > .Input_input__box___jY7u > input	cat
8	✓ click	css=div:nth-child(2) > .Input_input__box___jY7u > input	
9	✓ type	css=div:nth-child(2) > .Input_input__box___jY7u > input	0x70997970C51812dc3A010C7d01b50e0d17dc79C8
10	✓ click	css=div:nth-child(3) input	
11	✓ type	css=div:nth-child(3) input	4
12	✓ click	css=.Button_button__QHarr	

Command: open

Target: /

Value:

Description:

Log Reference

```
Running 'TC_01_login'
1.  open on / OK
2.  setWindowSize on 1552x849 OK
```

Fig 4.4 Test Case 4

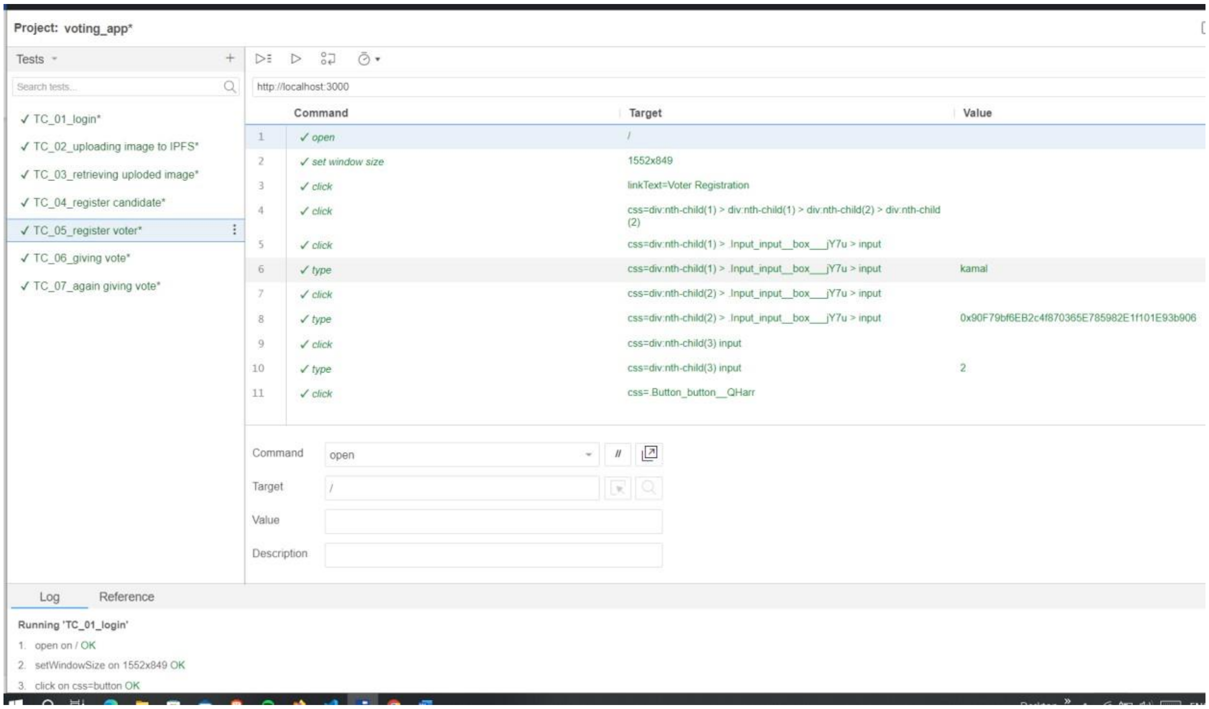
Test Case 5: Registering Voter

Test Description: This scenario evaluates the system's capability to register voters for the election.

Test Steps:

1. Log in to the E-Voting System as an administrator.
2. Navigate to the "Voter Registration" section.
3. Register a voter by entering their details.
4. Verify that the voter's registration is recorded in the system.

Expected Result: The system registers the voter, making them eligible to participate in the election.



✓ TC_01_login*
✓ TC_02_uploading image to IPFS*
✓ TC_03_retrieving uploded image*
✓ TC_04_register candidate*
✓ TC_05_register voter*
✓ TC_06_giving vote*
✓ TC_07_again giving vote*

http://localhost:3000

	Command	Target	Value
1	✓ open	/	
2	✓ set window size	1552x849	
3	✓ click	linkText=Voter Registration	
4	✓ click	css=div:nth-child(1) > div:nth-child(1) > div:nth-child(2) > div:nth-child (2)	
5	✓ click	css=div:nth-child(1) > .Input_input__box___jY7u > input	
6	✓ type	css=div:nth-child(1) > .Input_input__box___jY7u > input	kamal
7	✓ click	css=div:nth-child(2) > .Input_input__box___jY7u > input	
8	✓ type	css=div:nth-child(2) > .Input_input__box___jY7u > input	0x90F79bf6EB2c4f870365E785982E1f101E93b906
9	✓ click	css=div:nth-child(3) input	
10	✓ type	css=div:nth-child(3) input	2
11	✓ click	css=.Button_button__QHarr	

Command: open
Target: /
Value:
Description:

```
Running 'TC_01_login'
1. open on / OK
2. setWindowSize on 1552x849 OK
3. click on css=button OK
```

Fig 4.5 Test Case 5

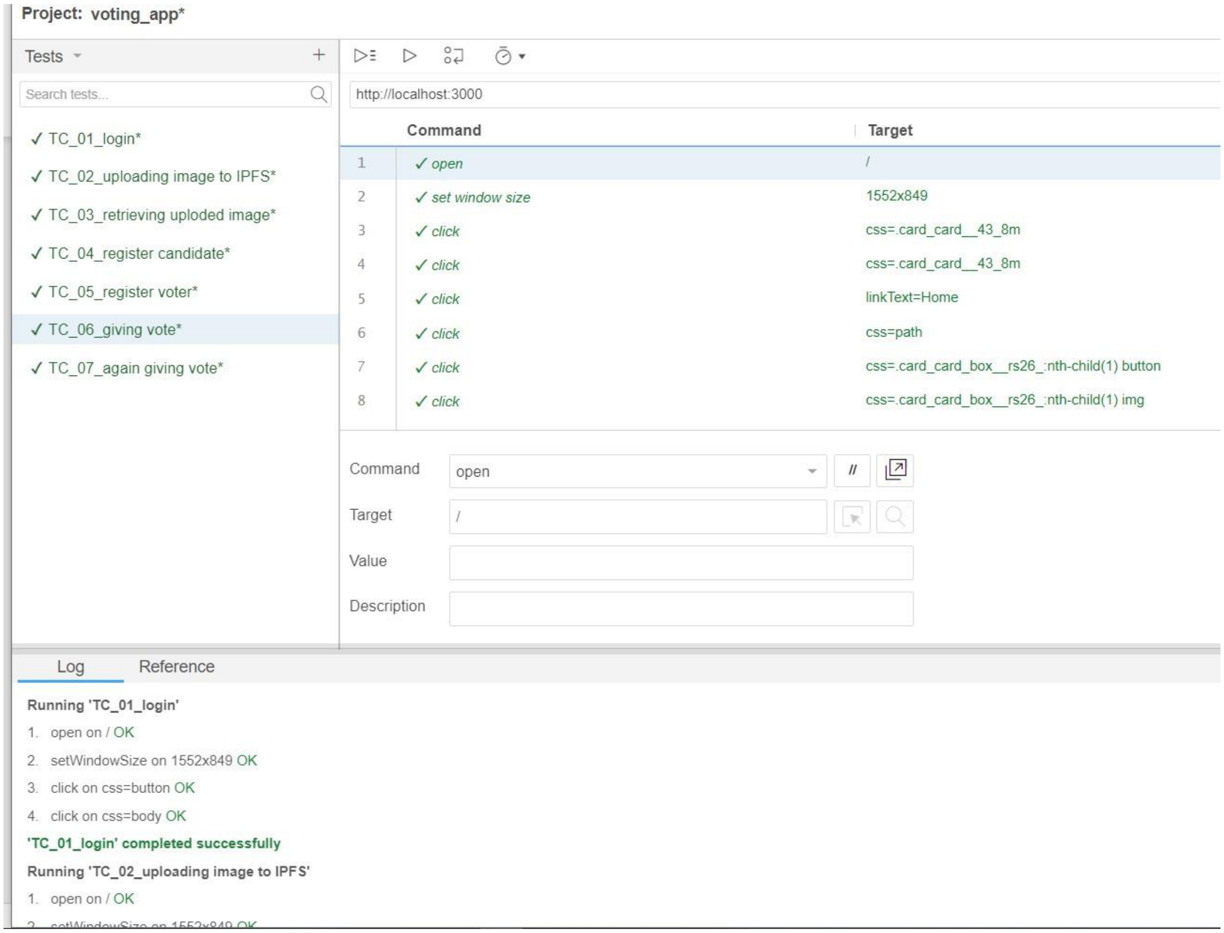
Test Case 6: Voting as Voter

Test Description: This scenario evaluates the voting process for a registered voter.

Test Steps:

1. Log in as a registered voter.
2. Go to homepage of web application.
3. Cast a vote for a candidate by clicking vote button.
4. Verify that the vote is recorded in the system.

Expected Result: Registered voters can cast their votes successfully, and the system records their choices.



Project: voting_app*

Tests

- ✓ TC_01_login*
- ✓ TC_02_uploading image to IPFS*
- ✓ TC_03_retrieving uploded image*
- ✓ TC_04_register candidate*
- ✓ TC_05_register voter*
- ✓ TC_06_giving vote*
- ✓ TC_07_again giving vote*

http://localhost:3000

	Command	Target
1	✓ open	/
2	✓ set window size	1552x849
3	✓ click	css=.card_card__43_8m
4	✓ click	css=.card_card__43_8m
5	✓ click	linkText=Home
6	✓ click	css=path
7	✓ click	css=.card_card_box__rs26_:nth-child(1) button
8	✓ click	css=.card_card_box__rs26_:nth-child(1) img

Command: open

Target: /

Value:

Description:

```
Running 'TC_01_login'
1.  open on / OK
2.  setWindowSize on 1552x849 OK
3.  click on css=button OK
4.  click on css=body OK
'TC_01_login' completed successfully
Running 'TC_02_uploading image to IPFS'
1.  open on / OK
2.  setWindowSize on 1552x849 OK
```

Fig 4.6 Test Case 6

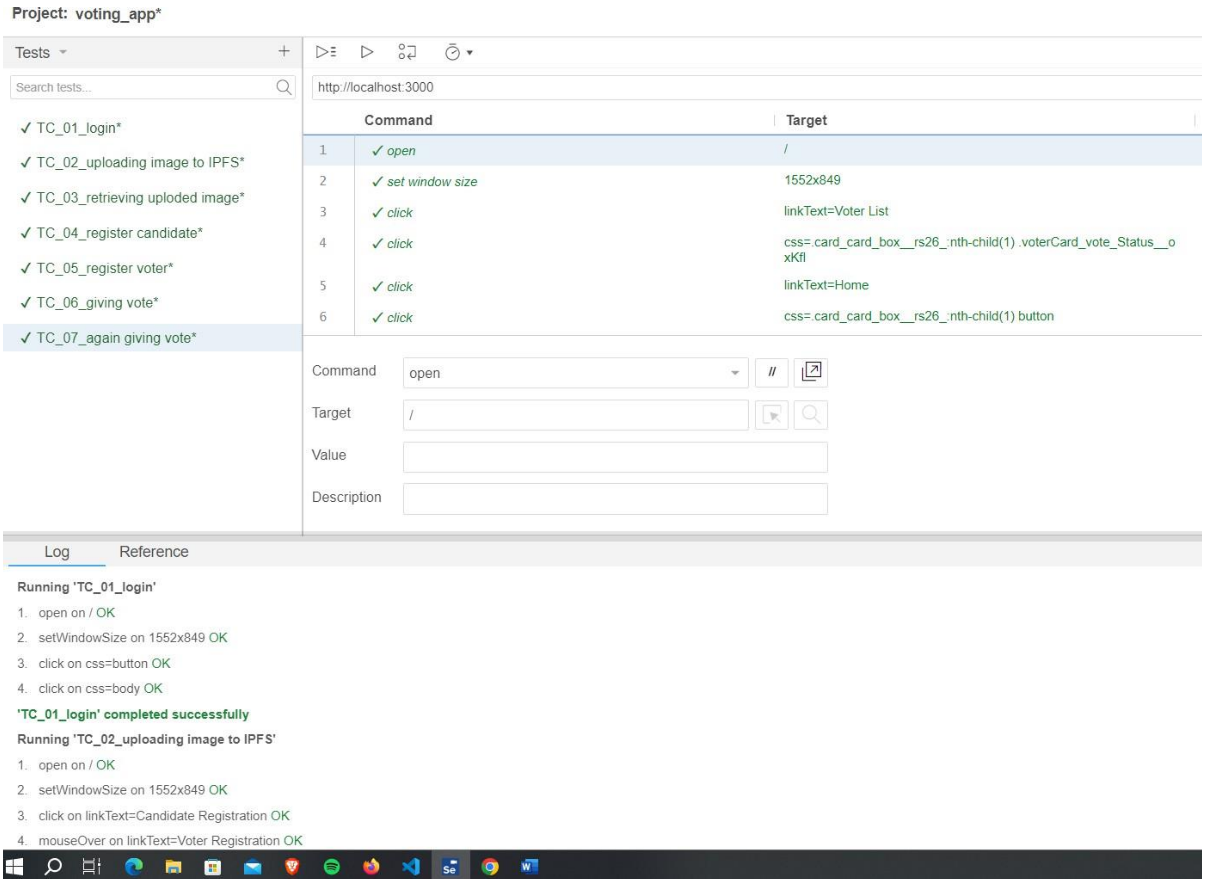
Test Case 7: Voting Again

Test Description: This scenario verifies the system's ability to prevent a voter from casting multiple votes.

Test Steps:

1. Log in as a registered voter.
2. Cast a vote for a candidate.
3. Attempt to vote again using the same voter account.
4. Confirm that the system prevents the voter from casting multiple votes.

Expected Result: The system should restrict voters from casting multiple votes, ensuring the integrity of the election process.



Project: voting_app*

- ✓ TC_01_login*
- ✓ TC_02_uploading image to IPFS*
- ✓ TC_03_retrieving uploded image*
- ✓ TC_04_register candidate*
- ✓ TC_05_register voter*
- ✓ TC_06_giving vote*
- ✓ TC_07_again giving vote*

http://localhost:3000

	Command	Target
1	✓ open	/
2	✓ set window size	1552x849
3	✓ click	linkText=Voter List
4	✓ click	css=.card_card_box__rs26_:nth-child(1) .voterCard_vote_Status__oxKfl
5	✓ click	linkText=Home
6	✓ click	css=.card_card_box__rs26_:nth-child(1) button

Command: open

Target: /

Value:

Description:

```
Running 'TC_01_login'
1. open on / OK
2. setWindowSize on 1552x849 OK
3. click on css=button OK
4. click on css=body OK
'TC_01_login' completed successfully
Running 'TC_02_uploading image to IPFS'
1. open on / OK
2. setWindowSize on 1552x849 OK
3. click on linkText=Candidate Registration OK
4. mouseOver on linkText=Voter Registration OK
```

Fig 4.7 Test Case 7

Testing Tools

Following Tools like

- Selenium IDE
- Browser (Firefox or chrome)

Test Environment

Following software’s are required in addition to client-specific software.

- Windows 10 and above
- Browser
- Microsoft Visual Studio

Terms/Acronyms

Make a mention of any terms or acronyms used in the project

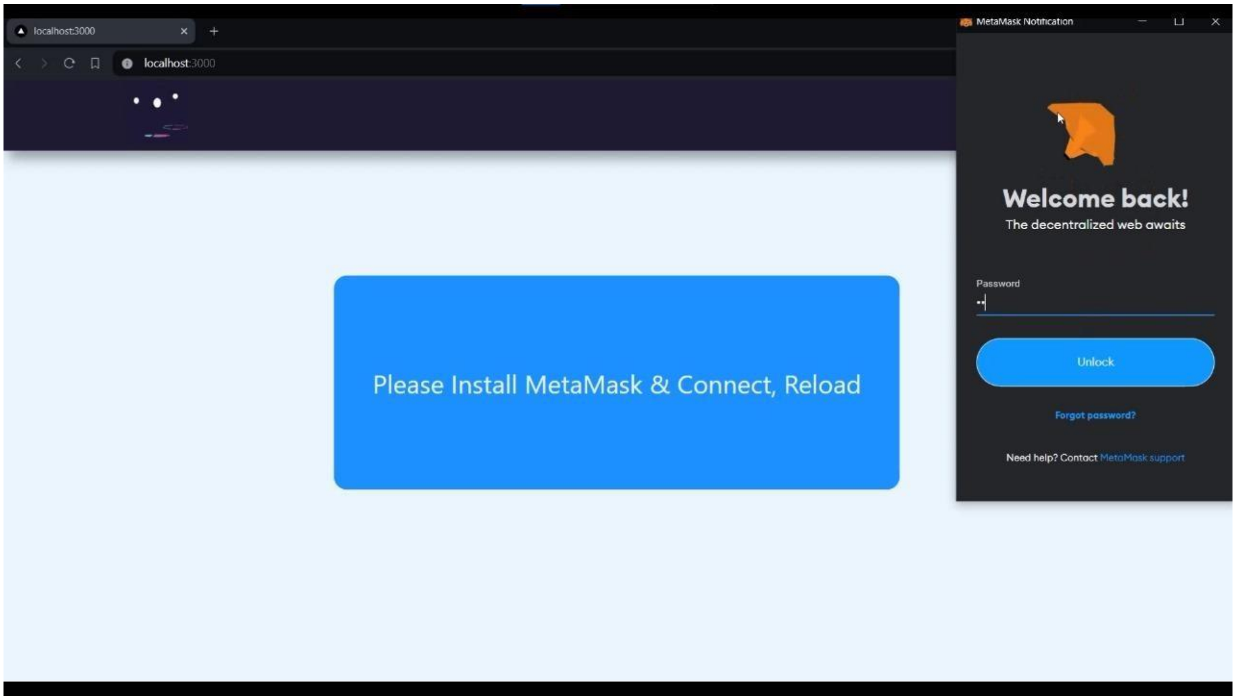
TERM/ACRONYM	DEFINITION
API	Application Program Interface
AUT	Application Under Test
IPFS	Inter Planetary File System
IDE	Integrated Development Environment

Fig 4.8 Acronyms

CHAPTER 5

RESULTS AND DISCUSSIONS

5.1 USER INTERFACE REPRESENTATION

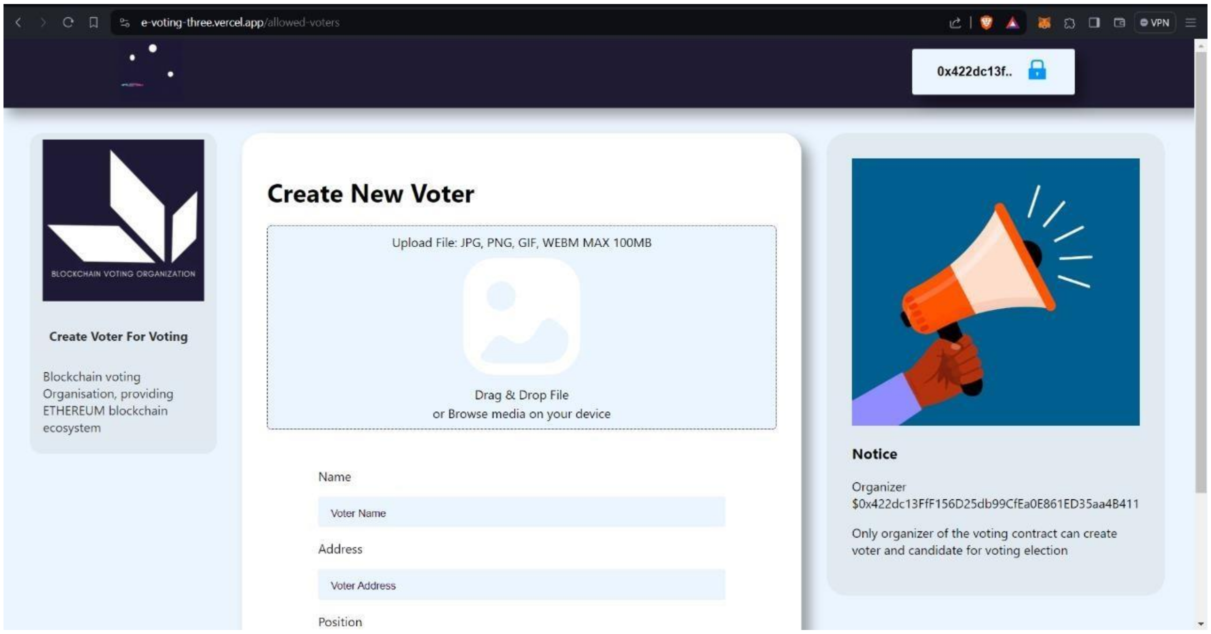

Please Install MetaMask & Connect, Reload

Welcome back!

The decentralized web awaits

Need help? Contact MetaMask support

Fig 5.1 Homepage



e-voting-three.vercel.app/allowed-voters
0x422dc13f..
BLOCKCHAIN VOTING ORGANIZATION

Create Voter For Voting

Blockchain voting Organisation, providing ETHEREUM blockchain ecosystem

Create New Voter

Upload File: JPG, PNG, GIF, WEBM MAX 100MB

Drag & Drop File
or Browse media on your device

Name: Voter Name

Address: Voter Address

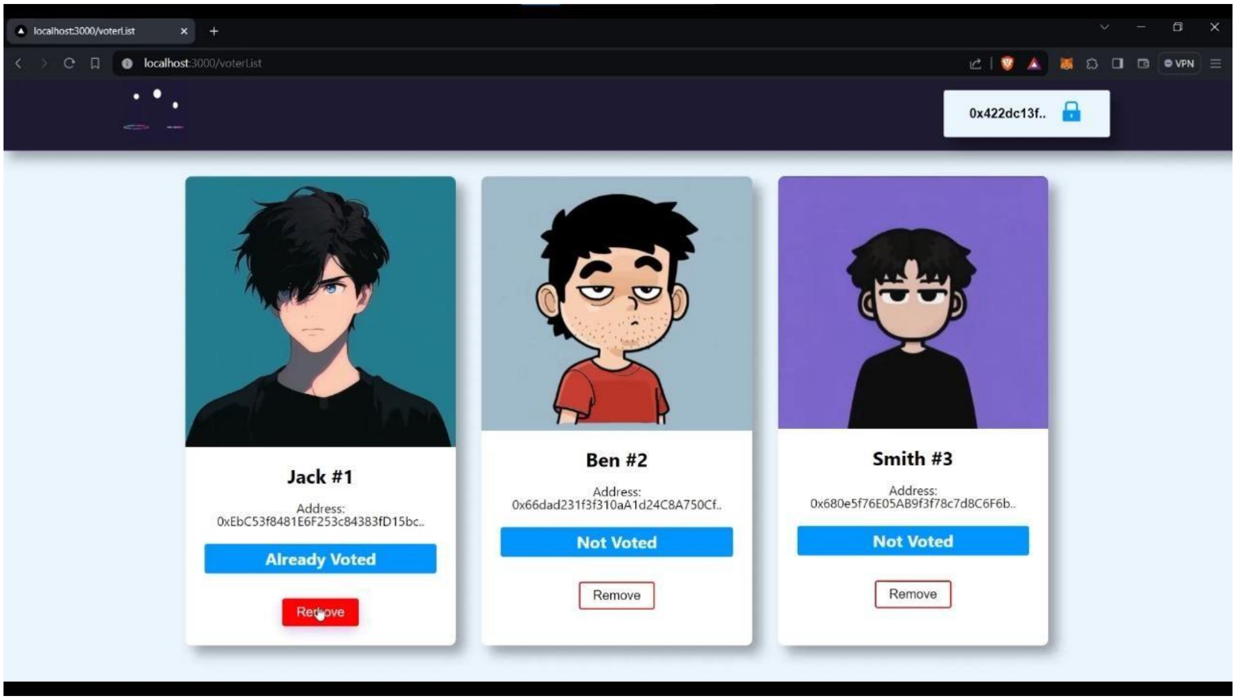
Position

Notice

Organizer
$0x422dc13FfF156D25db99CfEa0E861ED35aa4B411

Only organizer of the voting contract can create voter and candidate for voting election

Fig 5.2 Voter Registration



Jack #1
Address:
0xEbC53f8481E6F253c84383fD15bc..

Already Voted

Remove

Ben #2
Address:
0x66dad231f3f310aA1d24C8A750Cf..

Not Voted

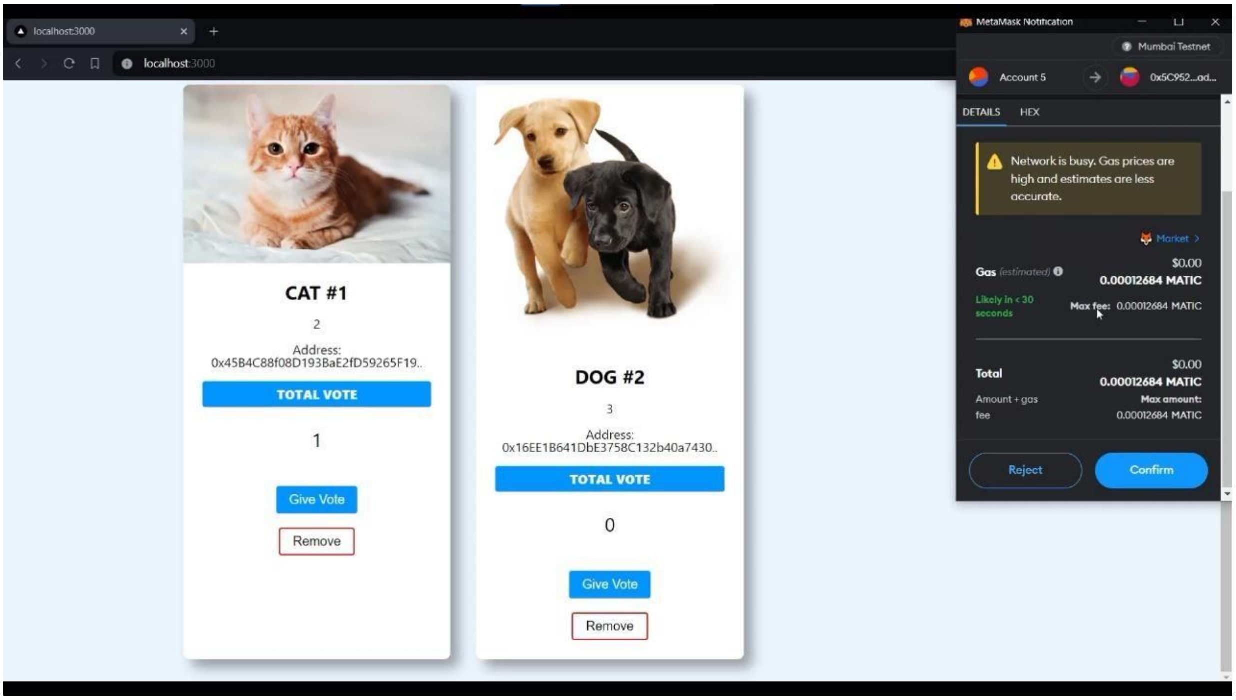
Remove

Smith #3
Address:
0x680e5f76E05AB9f3f78c7d8C6F6b..

Not Voted

Remove

Fig 5.3 Voter List



CAT #1

2

Address:
0x45B4C88f08D193BaE2fD59265F19..

TOTAL VOTE

1

DOG #2

3

Address:
0x16EE1B641DbE3758C132b40a7430..

TOTAL VOTE

0

Mumbai Testnet

Account 5 → 0x5C952...ad...

Network is busy. Gas prices are high and estimates are less accurate.

Gas (estimated) $0.00 0.00012684 MATIC

Likely in < 30 seconds Max fee: 0.00012684 MATIC

Total $0.00 0.00012684 MATIC

Amount + gas fee Max amount: 0.00012684 MATIC

Fig 5.4 Casting Vote

5.1.1 BRIEF DESCRIPTION OF VARIOUS MODULES OF THE SYSTEM

User Authentication Module: This module handles the authentication and authorization of voters, ensuring that only eligible voters can participate in the voting process. It verifies the identity of voters using secure authentication methods such as biometrics, digital signatures, or multi-factor authentication.

Voter Registration Module: This module manages the registration of voters, collecting necessary information such as identification details and voter eligibility criteria. It ensures that each voter is registered only once and maintains a secure database of registered voters.

Voting Interface Module: This module provides an intuitive and user-friendly interface for voters to cast their votes securely. It may include web or mobile applications that allow voters to access their ballots, select their choices, and submit their votes using cryptographic techniques for security.

Blockchain Integration Module: This module integrates blockchain technology into the voting system to record and store voting transactions securely and transparently. It utilizes decentralized consensus mechanisms to ensure immutability, transparency, and tamper-resistance of the voting data.

Vote Recording Module: This module records and timestamps each vote as a transaction on the blockchain, associating it with the corresponding voter's identity while maintaining voter anonymity. It ensures that votes are securely and accurately recorded without the possibility of manipulation or fraud.

Vote Counting Module: This module aggregates and counts the votes recorded on the blockchain to determine the outcome of the election. It may implement cryptographic algorithms or smart contracts to automate the counting process while preserving the anonymity and integrity of individual votes.

5.2 SNAPSHOTS OF SYSTEM WITH BRIEF DETAIL OF EACH

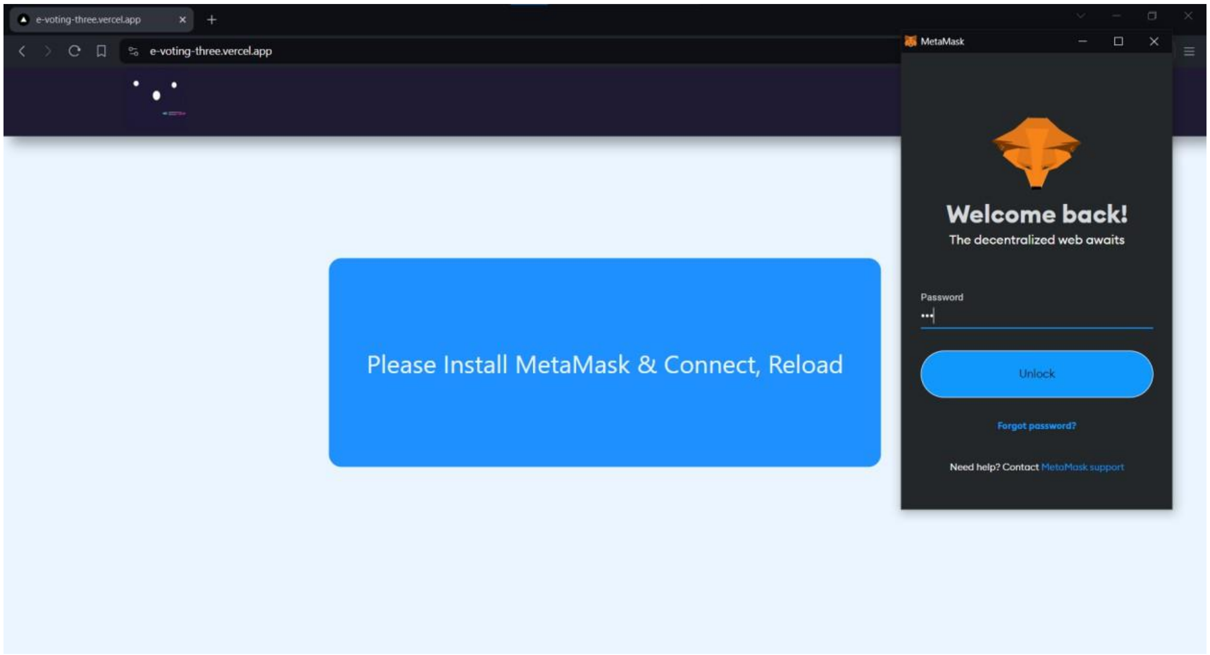
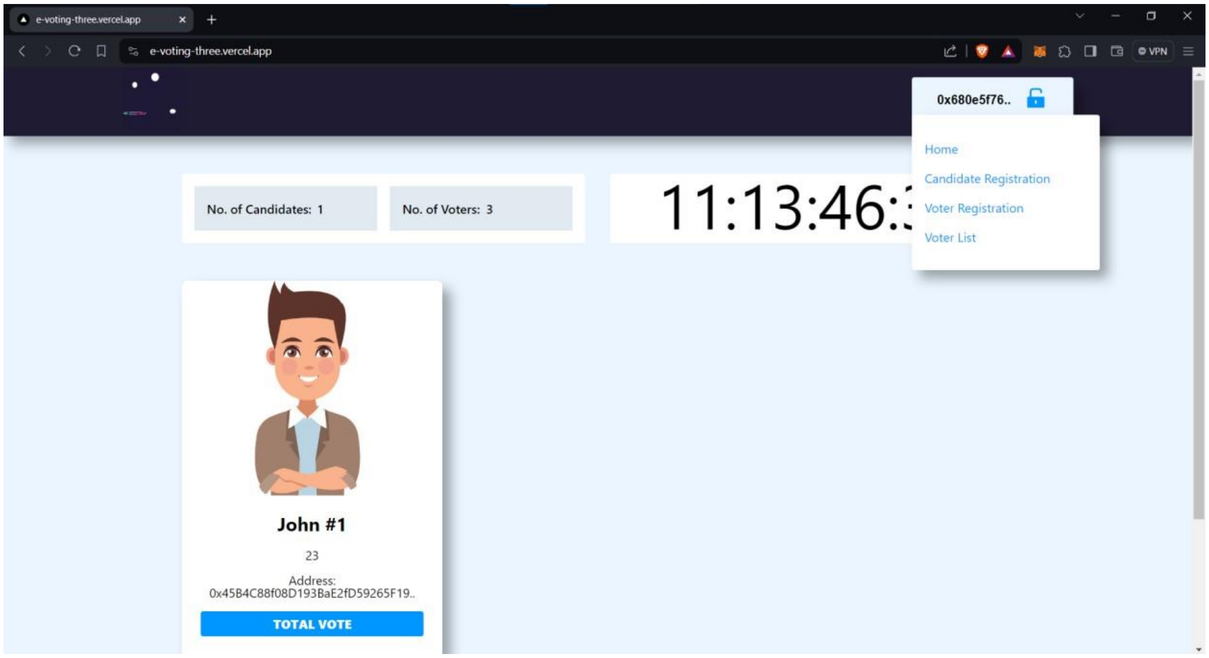

e-voting-three.vercel.app
e-voting-three.vercel.app
MetaMask
Please Install MetaMask & Connect, Reload
Welcome back!
The decentralized web awaits
Password
Unlock
Forgot password?
Need help? Contact MetaMask support

Fig 5.5 Homepage and Metamask Login

The homepage displays the login screen, which requires the user to log in to the voting system with a MetaMask Account. By logging in with the MetaMask account, the user will be able to access the UI.



No. of Candidates: 1

No. of Voters: 3

11:13:46:

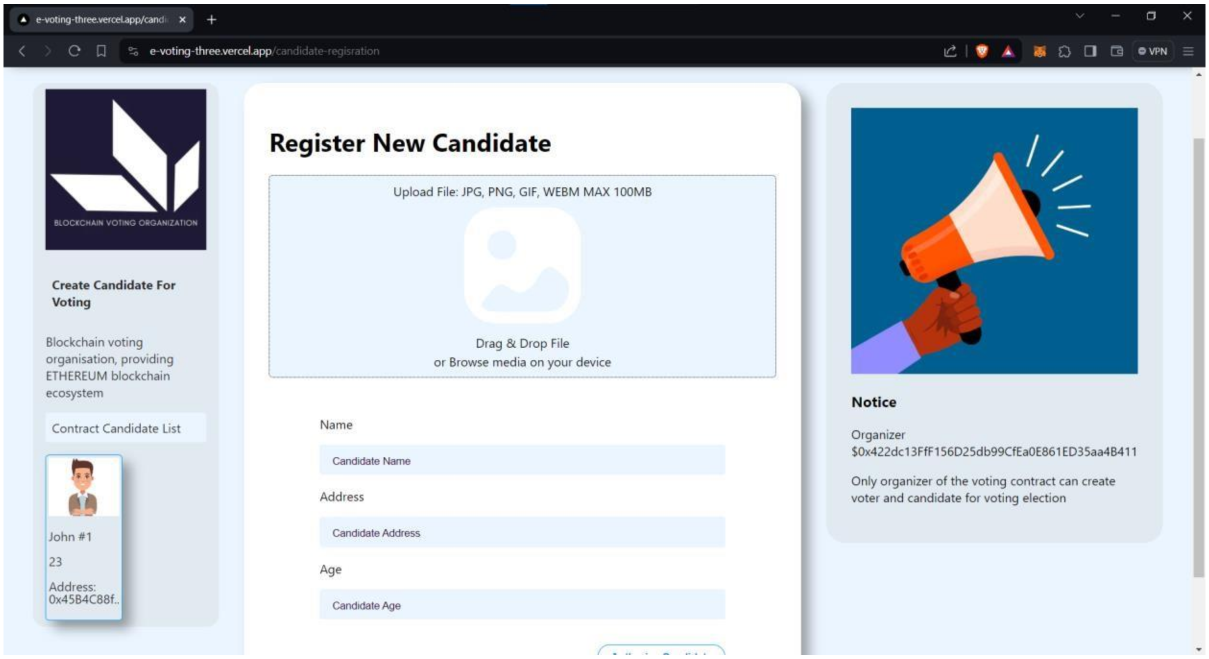
John #1

23

Address:
0x45B4C88f08D193BaE2fD59265F19..

Fig 5.6 User Interface

This is the main UI of the app, which displays the candidates, the number of voters, and the candidate information.


e-voting-three.vercel.app/cand
e-voting-three.vercel.app/candidate-regisration
BLOCKCHAIN VOTING ORGANIZATION
Contract Candidate List

Create Candidate For Voting

Blockchain voting organisation, providing ETHEREUM blockchain ecosystem

John #1

23

Address:
0x45B4C88f..

Register New Candidate

Upload File: JPG, PNG, GIF, WEBM MAX 100MB

Drag & Drop File
or Browse media on your device

Name

Candidate Name

Address

Candidate Address

Age

Candidate Age

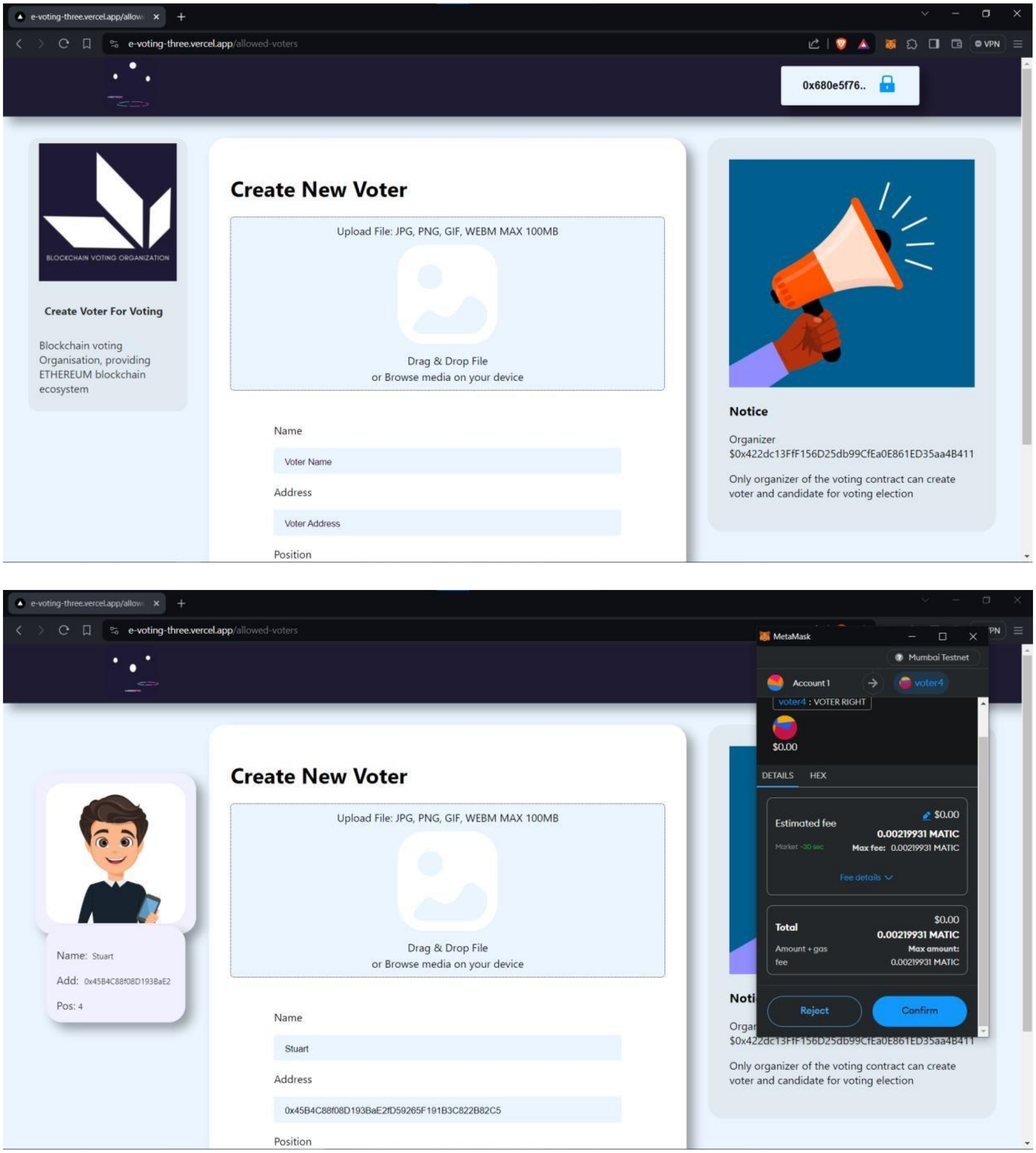
Notice

Organizer
$0x422dc13FfF156D25db99CfEa0E861ED35aa4B411

Only organizer of the voting contract can create voter and candidate for voting election

Fig 5.7 Candidate Registration

This is the candidate registration page, which includes a form for entering name, account address, age, and image upload.


e-voting-three.vercel.app/allow
e-voting-three.vercel.app/allowed-voters
VPN
0x680e5f76..
BLOCKCHAIN VOTING ORGANIZATION

Create Voter For Voting

Blockchain voting Organisation, providing ETHEREUM blockchain ecosystem

Create New Voter

Upload File: JPG, PNG, GIF, WEBM MAX 100MB

Drag & Drop File
or Browse media on your device

Name

Voter Name

Address

Voter Address

Position

Notice

Organizer
$0x422dc13FfF156D25db99CfEa0E861ED35aa4B411

Only organizer of the voting contract can create voter and candidate for voting election

e-voting-three.vercel.app/allow
e-voting-three.vercel.app/allowed-voters
MetaMask
Mumbai Testnet
Account 1
voter4
DETAILS
HEX
Fee details
Reject
Confirm

voter4 : VOTER RIGHT

$0.00

Estimated fee $0.00
0.00219931 MATIC
Market -30 sec Max fee: 0.00219931 MATIC

Total $0.00
0.00219931 MATIC
Amount + gas fee Max amount: 0.00219931 MATIC

Name: Stuart

Add: 0x45B4C88f08D193BaE2

Pos: 4

Create New Voter

Upload File: JPG, PNG, GIF, WEBM MAX 100MB

Drag & Drop File
or Browse media on your device

Name

Stuart

Address

0x45B4C88f08D193BaE2fD59265F191B3C822B82C5

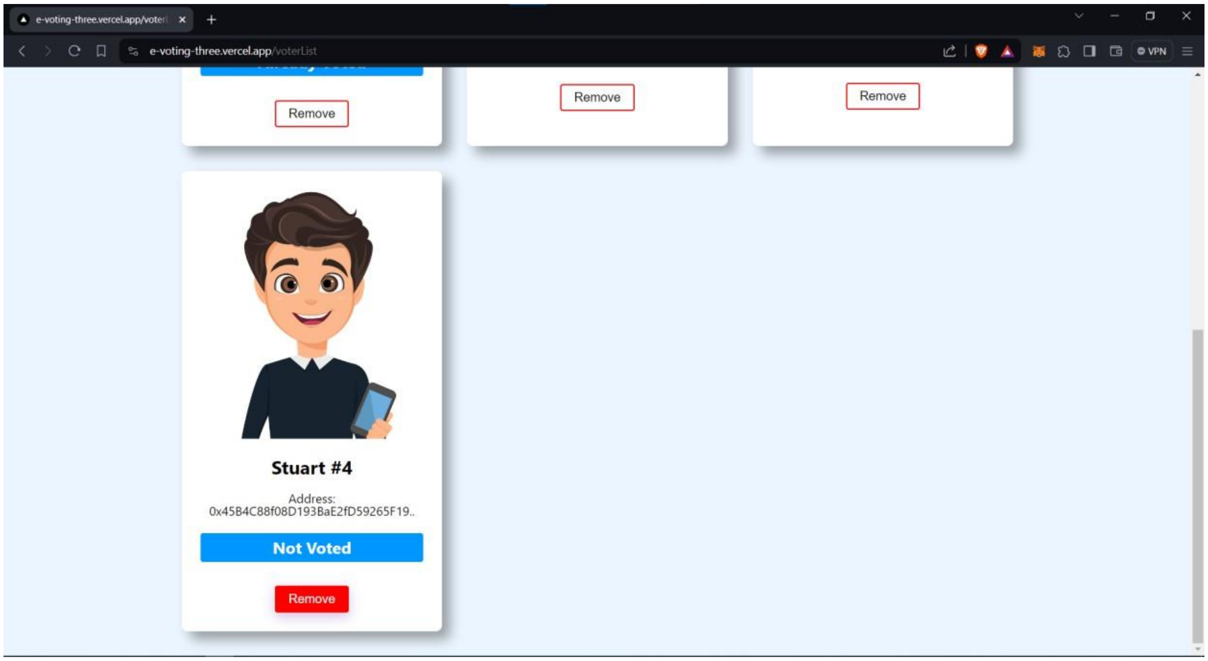
Position

Organizer
$0x422dc13FfF156D25db99CfEa0E861ED35aa4B411

Only organizer of the voting contract can create voter and candidate for voting election

Fig 5.8 Voter Registration

Voter registration is the same as candidate registration.

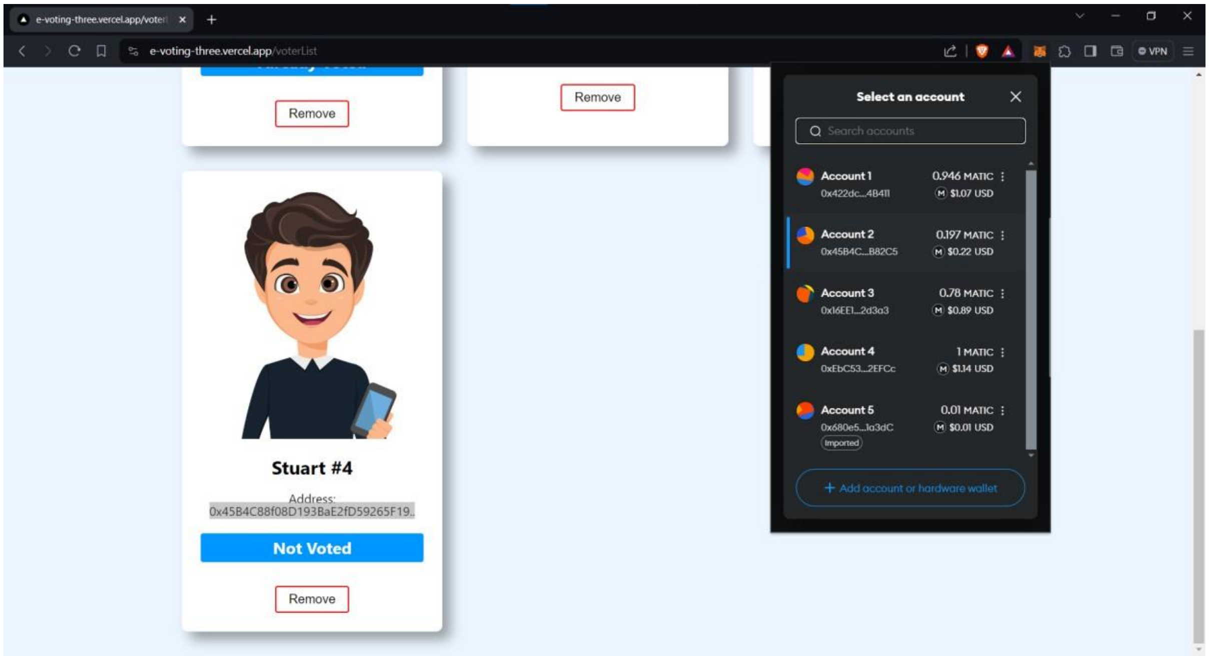

e-voting-three.vercel.app/voterList

Stuart #4

Address:
0x45B4C88f08D193BaE2fD59265F19..

Fig 5.9 Voter List

This shows the voter list and their status, indicating whether they have voted or not.

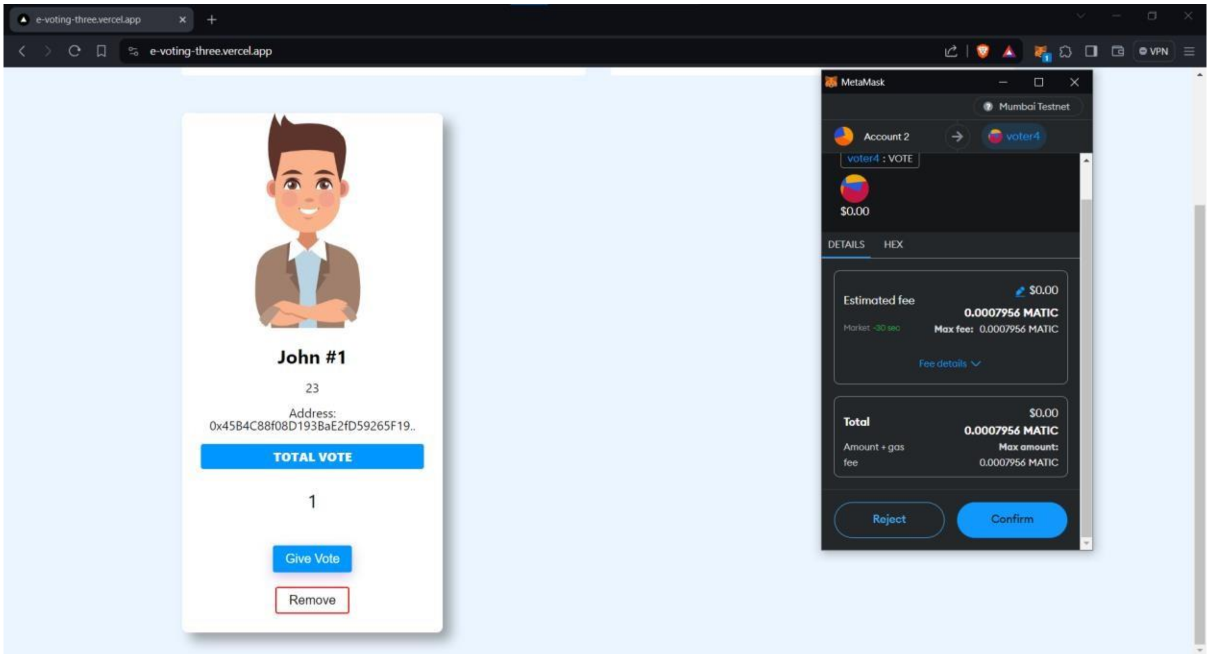

e-voting-three.vercel.app/voterList

Stuart #4

Address:
0x45B4C88f08D193BaE2fD59265F19..

Select an account

Account 1 0x422dc...4B411	0.946 MATIC $1.07 USD
Account 2 0x45B4C...B82C5	0.197 MATIC $0.22 USD
Account 3 0x16EE1...2d3a3	0.78 MATIC $0.89 USD
Account 4 0xEbC53...2EFCc	1 MATIC $1.14 USD
Account 5 0x680e5...1a3dC Imported	0.01 MATIC $0.01 USD



John #1

23

Address:
0x45B4C88f08D193BaE2fD59265F19..

1

voter4 : VOTE

$0.00

Estimated fee $0.00
0.0007956 MATIC
Market ~30 sec
Max fee: 0.0007956 MATIC

Total $0.00
0.0007956 MATIC
Amount + gas fee
Max amount: 0.0007956 MATIC

Fig 5.10 Vote Casting

In this image, voting is being conducted, and transactions are completed with MetaMask.

5.3 BACK ENDS REPRESENTATION

Blockchain Network: At the core of the backend representation is the blockchain network, which serves as a decentralized and immutable ledger for recording all voting transactions. The blockchain network is typically implemented using a distributed ledger technology such as Ethereum, which utilizes smart contracts to automate and enforce voting rules and procedures.

Smart Contracts: Smart contracts are self-executing contracts with the terms of the agreement directly written into code. In the context of online voting, smart contracts are deployed on the blockchain to facilitate various functions, including voter registration, ballot generation, vote casting, and tallying. These smart contracts ensure transparency, security, and trust in the voting process by automating tasks and eliminating the need for intermediaries.

Voter Authentication and Authorization: The backend includes mechanisms for authenticating and authorizing voters before they can participate in the voting process. This may involve integrating identity verification systems such as biometrics, digital signatures, or multi-factor authentication to verify the identity of voters securely.

Voter Registration Module: A voter registration module manages the registration of voters, collecting and storing relevant voter information securely on the blockchain. This module ensures that each voter is registered only once and maintains a tamper-proof record of eligible voters.

5.3.1 SNAPSHOTS OF DATABASE TABLES WITH BRIEF DESCRIPTION

IPFS is a decentralized protocol designed to create a peer-to-peer network for storing and sharing hypermedia content. Unlike traditional centralized storage systems, where data is stored on a single server or a group of servers controlled by a central authority, IPFS distributes data across a network of nodes. Each piece of data is assigned a unique cryptographic hash, which serves as its address on the IPFS network.

In the context of an application, IPFS can be used to store various types of data, including text, images, videos, and documents. When data is uploaded to IPFS, it is broken down into smaller chunks, encrypted, and distributed across multiple nodes in the network. This decentralized approach to storage offers several advantages:

Redundancy and Fault Tolerance: Since data is replicated across multiple nodes in the IPFS network, there is no single point of failure. Even if some nodes go offline or become inaccessible, the data remains available and accessible from other nodes in the network.

Content Addressing: Each piece of data stored on IPFS is assigned a unique cryptographic hash based on its content. This hash serves as a content-based identifier, allowing users to retrieve data by its hash rather than its location. This ensures that data integrity is maintained, as any change to the content would result in a different hash.

Caching and Performance: IPFS incorporates a caching mechanism that allows nodes to store and retrieve frequently accessed data locally. This improves performance and reduces latency, as data can be retrieved from nearby nodes rather than from the original source.

Decentralization and Data Sovereignty: By storing data on IPFS, users retain full control over their data and are not dependent on centralized servers or third-party providers. This promotes data sovereignty and ensures that users can access and share their data freely without restrictions.

CHAPTER 6

CONCLUSION AND FUTURE SCOPE

Electronic voting, dating back to the 1970s, presents considerable advantages over traditional paper-based systems, offering increased efficiency and reduced errors. The rise of blockchain technology has spurred investigations into its potential application in enhancing electronic voting systems. This paper highlights one such effort, leveraging blockchain's cryptographic underpinnings and transparency to create a more efficient e-voting solution. Using Multichain, the proposed methodology has been practically implemented and rigorously evaluated, demonstrating its effectiveness in meeting essential e-voting system requirements. To achieve an end-to-end verifiable e-voting scheme, we emphasize the critical need for a dependable provenance model in e-voting systems. Ongoing efforts focus on developing an additional provenance layer to complement the existing blockchain-based infrastructure, aiming to bolster the overall integrity and credibility of the system.

REFERENCES

[1] Fakhar ul Hassan, Anwaar Ali, Siddique Latif, Junaid Qadir, Salil Kanhere, Jatinder Singh, and Jon Crowcroft.(2019) Blockchain And The Future of the Internet: A Comprehensive Review
https://www.researchgate.net/publication/331730251_Blockchain_And_The_Future_of_the_Internet_A_Comprehensive_Review

[2] Simin Ghesmati, Walid Fdhila, Edgar Weippl (2023) User-Perceived Privacy in Blockchain
https://www.researchgate.net/publication/372564944_UserPerceived_Privacy_in_Blockchain

[3] Burcu Sakız, Aysen Hic Gencer (2019) Blockchain Technology and its Impact on the Global Economy
https://www.researchgate.net/publication/345386992_Blockchain_Technology_and_its_Impact_on_the_Global_Economy

[4] B Laurie Hughes, Yogesh Kumar Dwivedi, Santosh K. Misra, Nripendra Rana (2019) Blockchain research, practice and policy: Applications, benefits, limitations, emerging research themes and research agenda https://www.researchgate.net/publication/344959048_Blockchain_research_practice_and_policy_Applications_benefits_limitations_emerging_research_themes_and_research_agenda

[5] Satoshi Nakamoto (2009) Bitcoin: A Peer-to-Peer Electronic Cash System https://www.researchgate.net/publication/228640975_Bitcoin_A_Peer - to- Peer_Electronic_Cash_System.

[6] Merlinda Andoni, Valentin Robu, D. Flynn, Simone Abram (2018) Blockchain technology in the energy sector: A systematic review of challenges and opportunities https://www.researchgate.net/publication/328760651_Blockchain_technology_in_the_energy_sector_A_systematic_review_of_challenges_ and_opportunities

[7] Diego Moussallem, Matthias, Axel-Cyrille Ngonga Ngomo (2017) Machine Translation Using Semantic Web Technologies: A Survey https://www.researchgate.net/publication/321325421_Machine_Translation_Using_Semantic_Web_Technologies_A_Survey

[8] Zibin Zheng, Shaoan Xie, Hong-Ning Dai, Xiangping Chen (2017) An Overview of Blockchain Technology: Architecture, Consensus, and Future Trends https://www.researchgate.net/publication/318131748_An_Overview_of_Blockchain_Technology_Architecture_Consensus_and_Future_Tr ends

[9] Gustavo Ansaldi Oliva, Ahmed E. Hassan, Zhen Ming (Jack) Jiang (2020) An Exploratory Study of Smart Contracts in the Ethereum Blockchain Platform https://www.researchgate.net/publication/337603517_An_Explorator y_Study_of_Smart_Contracts_in_the_Ethereum_Blockchain_Platfor m

[10] Peter D. DeVries (2016) An Analysis of Cryptocurrency, Bitcoin, and the Future https://www.researchgate.net/publication/316656878_An_Analysis_o f_Cryptocurrency_Bitcoin_and_the_Future

[11] Van Giang Phan Mai, Lã Minh Vũ, Đỗ Hoàng Sơn, and Nguyễn Tuấn Khải (2023) A Blockchain-based User Authentication Model Using MetaMask https://www.researchgate.net/publication/370339058_A_Blockchainbased_User_Authenticati on_Model_Using_MetaMask

[12] Laiphrakpam Dolendro Singh, Khumanthem Manglem Singh (2015) Implementation of Text Encryption using Elliptic Curve Cryptography https://www.researchgate.net/publication/283186188_Implementation _of_Text_Encryption_using_Elliptic_Curve_Cryptography

[13] Nwosu Anthony, S B Goyal, Anand Singh Rajawat, Sardar M. N. Islam (2022) An Innovative Blockchain-Based Secured Logistics Management Architecture: Utilizing an RSA Asymmetric Encryption Method https://www.researchgate.net/publication/366156239_An_Innovative _BlockchainBased_Secured_Logistics_Management_Architecture_Utilizing_an_ RSA_Asymmetric_Encryption_Method

[14] Blockchain Technology Based Healthcare Supply Chain Management, Dr. Pushpa, International, Taylor & Francis Group, 978-1-003-20196- 0 (Book Chapter).

[15] Vikarnt Shokeen, Sachin Goel, Nidhi Gupta, Chhaya Sharma, Parita Jain, “Blockchain Technology and Its Industrial Utilization: A Review”, DE, pp. 14570-14579, Sep. 2021