Project Report

Date	26 October 2023
Team ID	NM2023TMIDO1678
Project Name	Electronic Voting System (Block Chain)

Team Members:

1. **Maharajan R-** 9BD0DE84195C82AA7374C808BA3E322D
2. **Mathankumar C-**BD0558CC1221FCBB71D4BFF5C0F9DAB8
3. **Balaji G-**633275AC7FE9BA6BC3C4FD9EF1ED48A9
4. **Vasanthakumar D-**F9606410090B960ACBF4978529AAA3E1

ELECTRONIC VOTING SYSTEM

12. FUTURE SCOPE 13. APPENDIX

Source Code

GitHub & Project Demo Link

1. INTRODUCTION

Electronic voting (also known as e-voting) is voting that uses electronic means to either aid or take care of casting and counting ballots. Depending on the particular implementation, e-voting may use standalone electronic voting machines (also called EVM) or computers connected to the Internet (online voting).

1.1 Project Overview

The electronic voting has emerged over time as a replacement to the paper -based voting to reduce the redundancies and inconsistencies. The historical perspective presented in the last two decades suggests that it has not been so successful due to the security and privacy flaws observed over time. This paper suggests a framework by using effective hashing techniques to ensure the security of the data. The concept of block creation and block sealing is introduced in this paper. The introduction of a block sealing concept helps in making the blockchain adjustable to meet the need of the polling process. The use of consortium blockchain is suggested, which ensures that the blockchain is owned by a governing body (e.g., election commission), and no unauthorized access can be made from outside. The framework proposed in this paper discusses the effectiveness of the polling process, hashing algorithms' utility, block creation and sealing, data accumulation, and result declaration by using the adjustable blockchain method. This paper claims to apprehend the security and data management challenges in blockchain and provides an improved manifestation of the electronic voting process.

1.2 Purpose

I.MODELING OF ENTIRE E-VOTING PROCESS :

The system modeling helps in drawing the entire system on paper to develop a deep understanding of the system and to identify errors and flaws that can be observed before the system can be implemented.

II. DETERMINATION OF THE SUITABLE TECHNOLOGY PLATFORM TO ENSURE ANONYMITY, PRIVACY, AND SECURITY:

The e-voting process requires the features like privacy, security, anonymity, and verifiability as the core function of this solution, it is important that the choice of the underlying technology is consistent to meet these challenges. It has been identified that the Blockchain technology sufficiently deals with all such challenges.

III. DEVELOPMENT & TECHNOLOGY INTEGRATION WITH THE PERCEIVED E-III. VOTING MODEL:

Based on the system model, the system will be developed and will be integrated with the baseline technology.

2. LITERATURE SURVEY

2.1 Existing problem

- Political instability
- Compromised writ of the government
- Mistrust over the electoral process
- Compromised governance
- Disorder in the state institution
- Chain of command to run state affairs
- Economic instability

2.2 References

When discussing electronic voting systems, it is important to refer to reliable and authoritative sources to ensure accurate and up-to-date information. Here are a few reputable references that can provide valuable insights into electronic voting systems:

1. International Foundation for Electoral Systems (IFES): IFES is a global organization that promotes democratic elections. They provide resources and publications on various aspects of electoral systems, including electronic voting. Their website (www.ifes.org) offers reports, guidelines, and case studies related to electronic voting.
2. Election Assistance Commission (EAC): The EAC is an independent federal agency in the United States that provides information and assistance on election administration. Their website (www.eac.gov) offers resources on electronic voting systems, including guidelines, research reports, and best practices.
3. Verified Voting: Verified Voting is a non-profit organization that advocates for accuracy, transparency, and verifiability in elections. Their website (www.verifiedvoting.org) provides information on electronic voting systems, including news, research papers, and policy recommendations.
4. National Institute of Standards and Technology (NIST): NIST is a federal agency that develops and promotes measurement standards. They have published guidelines and technical reports on electronic voting systems, which can be accessed through their website (www.nist.gov).
5. Academic Journals and Research Papers: Academic journals in the fields of computer science, political science, and public administration often publish research papers on electronic voting systems. Examples of relevant journals include the Journal of Electronic Voting, Election Law Journal, and Government Information Quarterly.

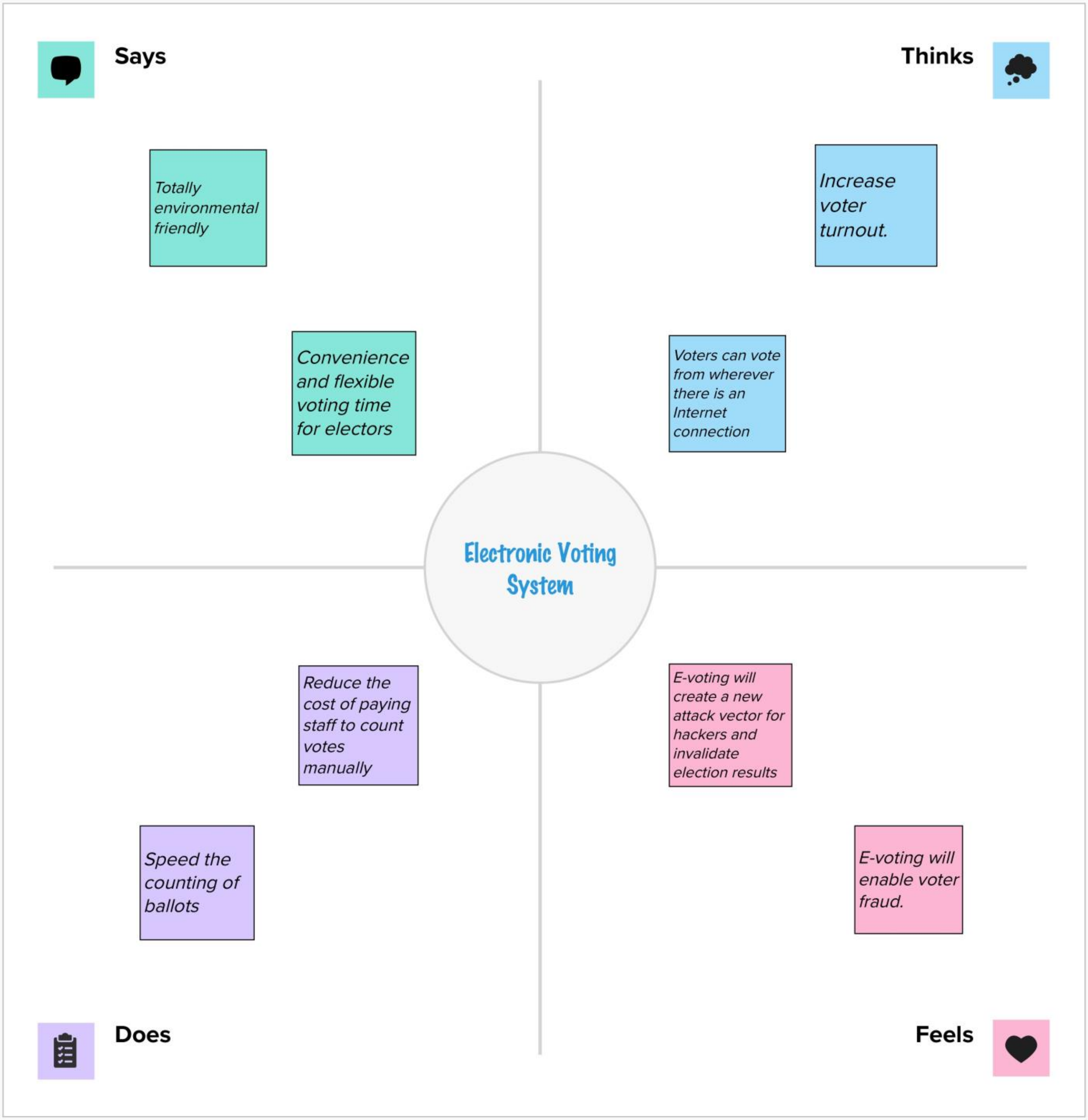
When using these references, it is important to critically evaluate the information and consider multiple perspectives to form a well-rounded understanding of electronic voting systems

2.3 Problem Statement Definition

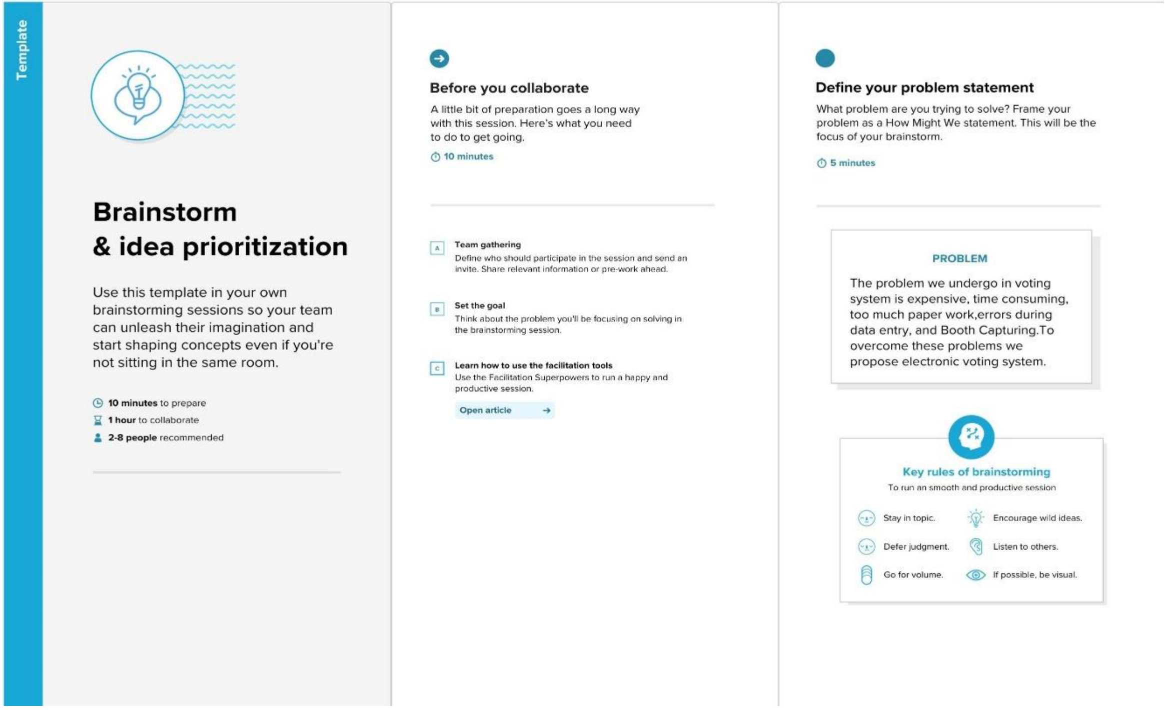
The problem we undergo in voting system is expensive, time consuming, too much paper work,errors during data entry, and Booth Capturing.To overcome these problems we propose electronic voting system**.**

3. IDEATION & PROPOSED SOLUTION

3.1 Empathy Map Canvas


Says
Thinks
Totally environmental friendly
Increase voter turnout.
Convenience and flexible voting time for electors
Voters can vote from wherever there is an Internet connection
Electronic Voting System
Reduce the cost of paying staff to count votes manually
E-voting will create a new attack vector for hackers and invalidate election results
Speed the counting of ballots
E-voting will enable voter fraud.
Does
Feels

3.2 Ideation & Brainstorming



Template

Brainstorm & idea prioritization

Use this template in your own brainstorming sessions so your team can unleash their imagination and start shaping concepts even if you're not sitting in the same room.

10 minutes to prepare

1 hour to collaborate

2-8 people recommended

Before you collaborate

A little bit of preparation goes a long way with this session. Here's what you need to do to get going.

10 minutes

A **Team gathering**
Define who should participate in the session and send an invite. Share relevant information or pre-work ahead.

B **Set the goal**
Think about the problem you'll be focusing on solving in the brainstorming session.

C **Learn how to use the facilitation tools**
Use the Facilitation Superpowers to run a happy and productive session.

Open article →

Define your problem statement

What problem are you trying to solve? Frame your problem as a How Might We statement. This will be the focus of your brainstorm.

5 minutes

PROBLEM

The problem we undergo in voting system is expensive, time consuming, too much paper work,errors during data entry, and Booth Capturing.To overcome these problems we propose electronic voting system.

Key rules of brainstorming

To run an smooth and productive session

Stay in topic.

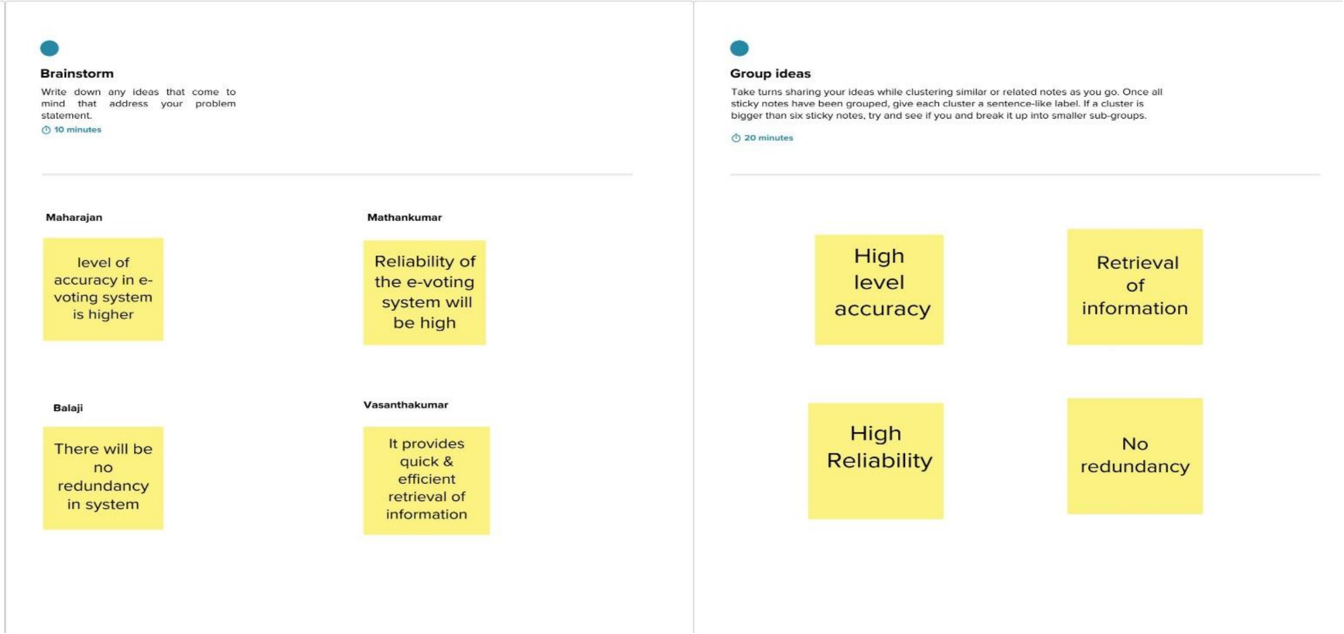
Encourage wild ideas.

Defer judgment.

Listen to others.

Go for volume.

If possible, be visual.



Brainstorm

Write down any ideas that come to mind that address your problem statement.

10 minutes

Maharajan

level of accuracy in e-voting system is higher

Mathankumar

Reliability of the e-voting system will be high

Balaji

There will be no redundancy in system

Vasanthakumar

It provides quick & efficient retrieval of information

Group ideas

Take turns sharing your ideas while clustering similar or related notes as you go. Once all sticky notes have been grouped, give each cluster a sentence-like label. If a cluster is bigger than six sticky notes, try and see if you and break it up into smaller sub-groups.

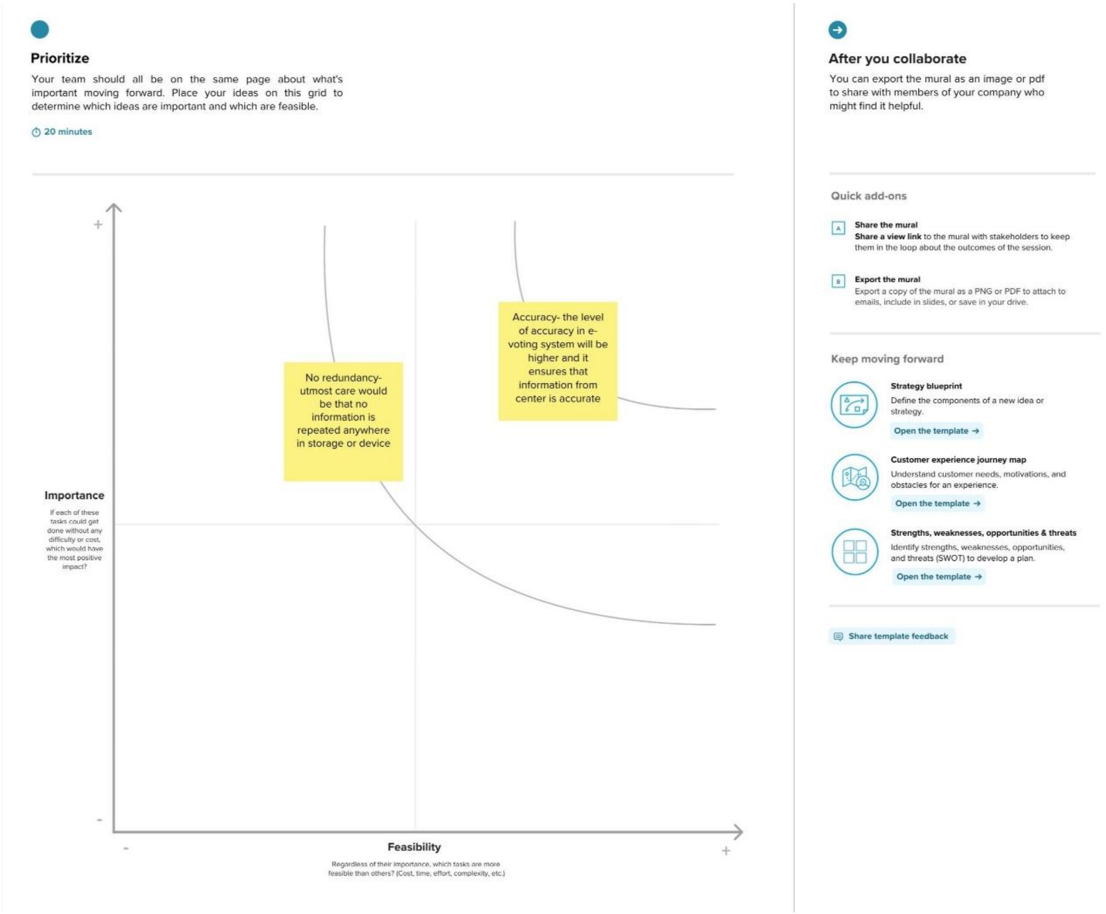
20 minutes

High level accuracy

Retrieval of information

High Reliability

No redundancy



Prioritize

Your team should all be on the same page about what's important moving forward. Place your ideas on this grid to determine which ideas are important and which are feasible.

20 minutes

No redundancy- utmost care would be that no information is repeated anywhere in storage or device

Accuracy- the level of accuracy in e-voting system will be higher and it ensures that information from center is accurate

Importance

If each of these tasks could get done without any difficulty or cost, which would have the most positive impact?

Feasibility

Regardless of their importance, which tasks are more feasible than others? (Cost, time, effort, complexity, etc.)

After you collaborate

You can export the mural as an image or pdf to share with members of your company who might find it helpful.

Quick add-ons

A **Share the mural**
Share a view link to the mural with stakeholders to keep them in the loop about the outcomes of the session.

B **Export the mural**
Export a copy of the mural as a PNG or PDF to attach to emails, include in slides, or save in your drive.

Keep moving forward

Strategy blueprint
Define the components of a new idea or strategy.

Customer experience journey map
Understand customer needs, motivations, and obstacles for an experience.

Strengths, weaknesses, opportunities & threats
Identify strengths, weaknesses, opportunities, and threats (SWOT) to develop a plan.

4. REQUIREMENT ANALYSIS

4.1Functional requirements

1. Mobility: The voter should not be restricted to cast his ballot at a single poll-site at his home precinct.

- Realistic: He shall be able to vote from any poll-site within the nation.

- Unrealistic/Expensive: He shall be able to vote from any county-controlled kiosk (situated at public places such as banks, shopping malls, etc.) within the nation. (Unrealistic because of logistical and cost issues).

- Infeasible: He shall be able to vote from virtually anywhere using an Internet connection. (Infeasible both for technical security issues as well as social science issues).

2. Convenience: The system shall allow the voters to cast their votes quickly, in one session, and should not require many special skills or intimidate the voter (to ensure Equality of Access to Voters).

3. User-Interface: The system shall provide an easy-to-use user-interface. Also, it shall not disadvantage any candidate while displaying the choices (e.g., by requiring the user to scroll down to see the last few choices).

4. Transparency: Voters should be able to possess a general knowledge and understanding of the voting process. 5. Flexibility: The system shall be flexible in that it allows a variety of ballot question formats including open-ended questions (e.g. Write-in candidates and survey questions).

4.2 Non-Functional requirements

1. Voter Authenticity: Ensure that the voter must identify himself (with respect to the registration database) to be entitled to vote. If voting other than at his home precinct, the voter may be asked to show some legal identification document.

2. Registration: The voter registration shall be done in person only. However, the computerized registration database shall be made available to polling-booths all around the nation.

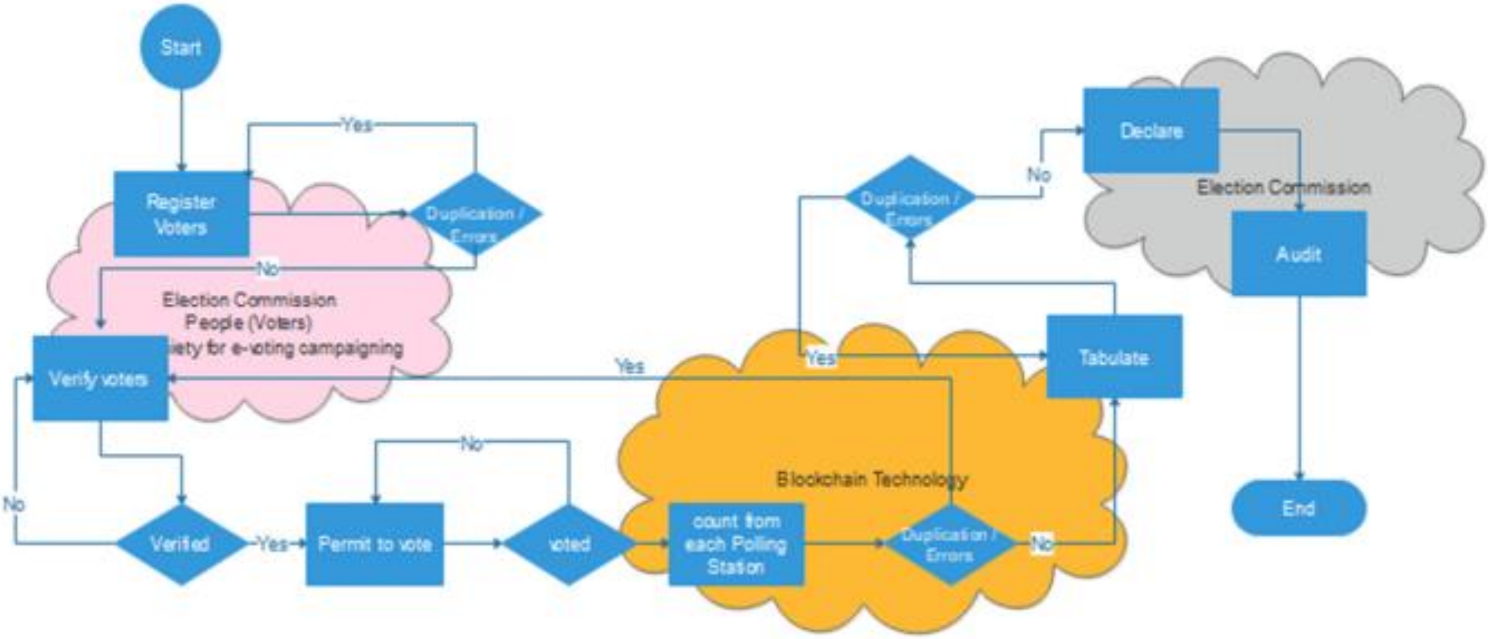
3. Voter Anonymity: Ensure that votes must not be associated with voter identity.

4. System Integrity: Ensure that the system cannot be re-configured during operation.

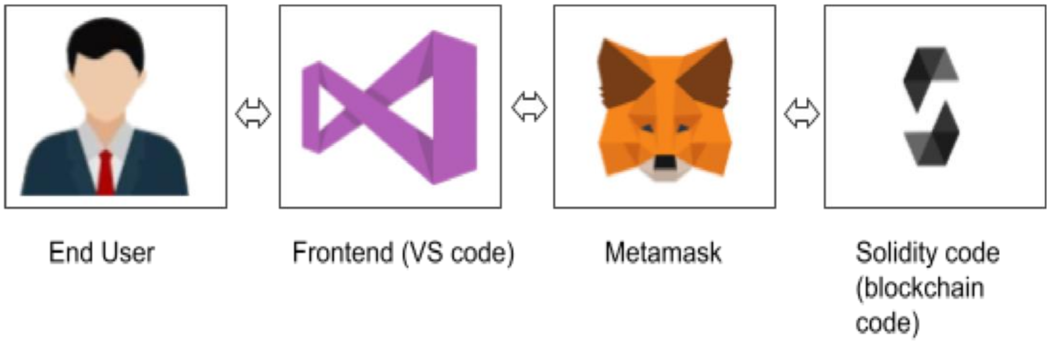
5. Data Integrity: Ensure that each vote is recorded as intended and cannot be tampered with in any manner, once recorded (i.e., votes should not be modified, forged or deleted without detection).

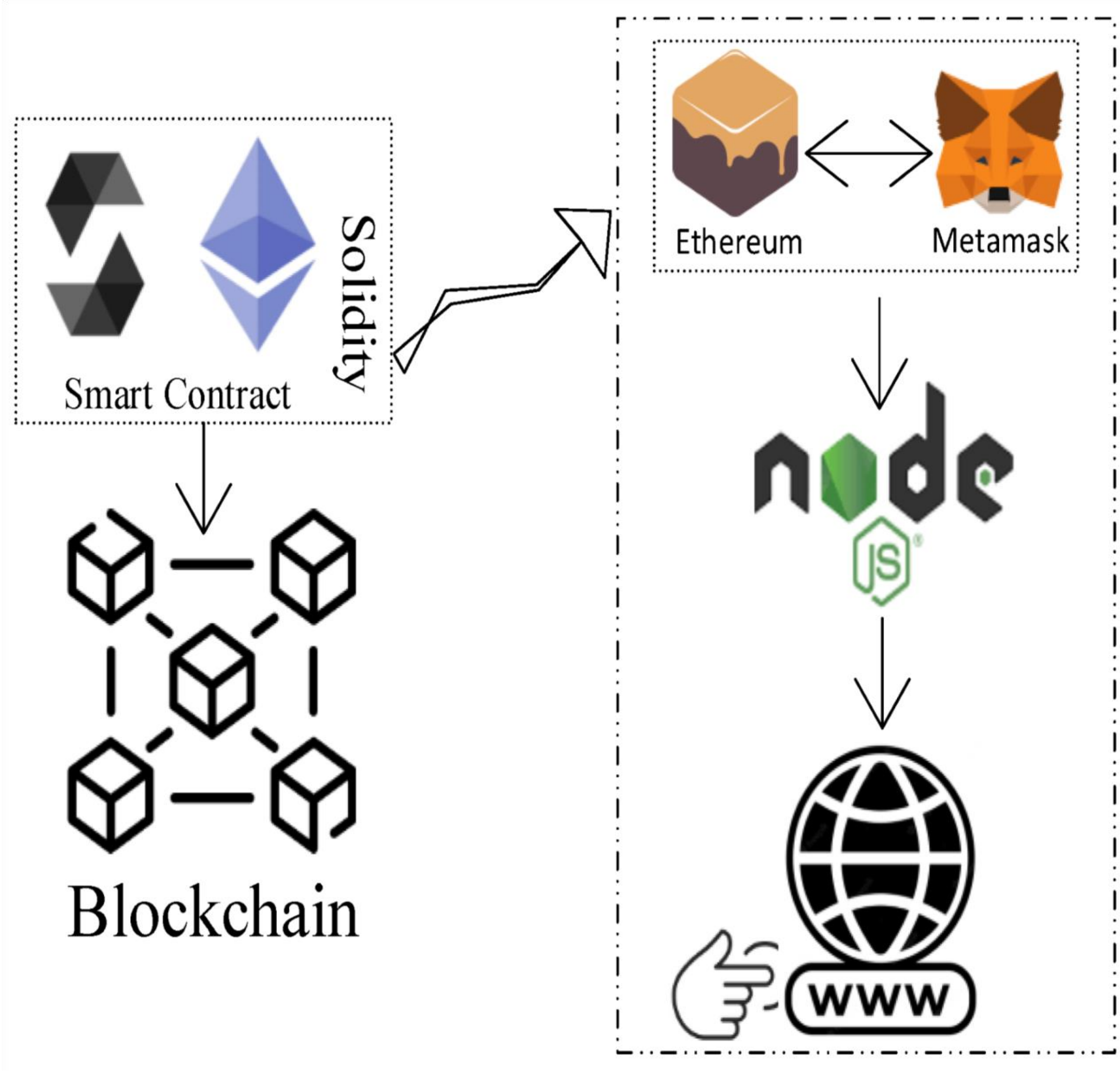
5. PROJECT DESIGN

5.1 Data Flow Diagrams & User Stories



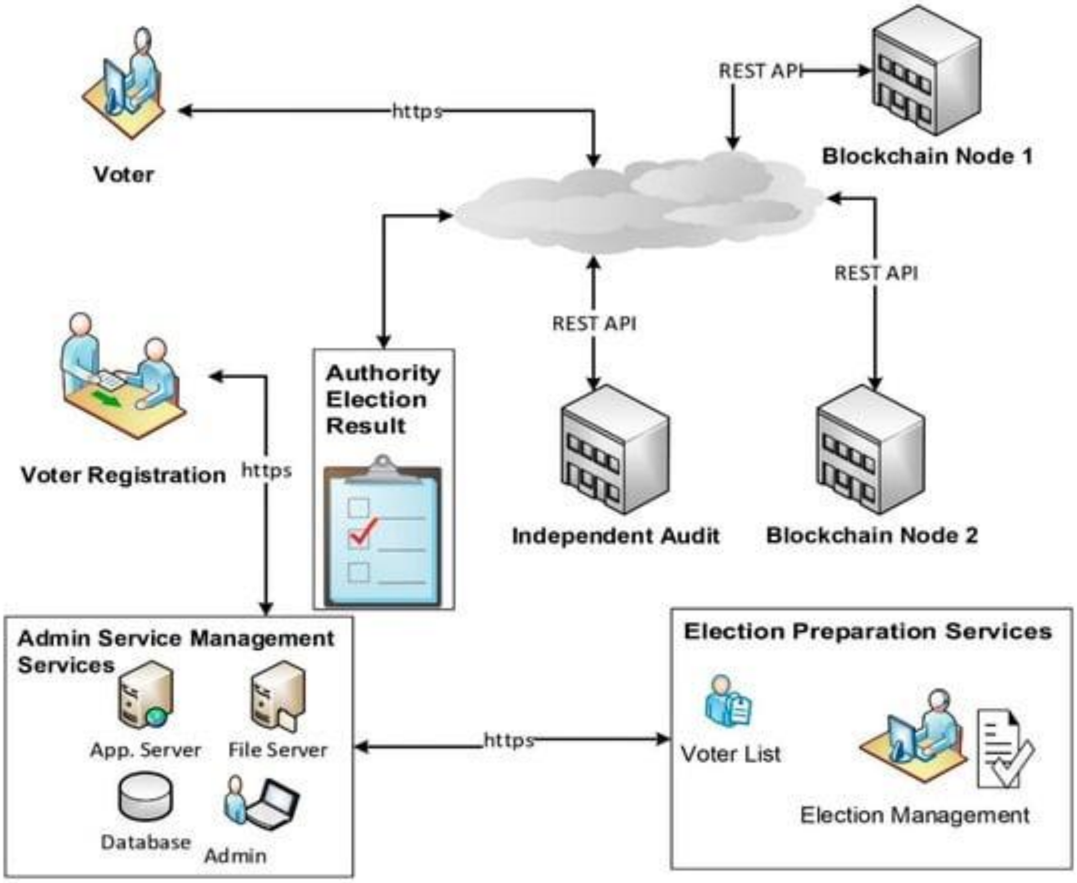
5.2 Solution Architecture





6. PROJECT PLANNING & SCHEDULING

6.1 Technical Architecture



6.2 Sprint Planning & Estimation

Sprint planning and estimation for an e-voting system would involve breaking down the project into smaller tasks and estimating the effort required for each task. This process typically includes the following steps:

1. Product Backlog: Create a list of all the features and functionalities required for the e-voting system. This serves as the product backlog.

2. User Stories: Convert the items in the product backlog into user stories, which describe the system's behavior from the perspective of the end user.

3. Prioritization: Prioritize the user stories based on their importance and value to the stakeholders. This helps in determining the order in which they will be implemented.

4. Sprint Planning: Select a set of user stories from the prioritized backlog to be implemented in the upcoming sprint. The sprint duration is typically 1-4 weeks.

5. Task Breakdown: Break down each user story into smaller tasks that can be completed within the sprint. These tasks should be specific, measurable, achievable, relevant, and time-bound (SMART).

6. Estimation: Estimate the effort required for each task. This can be done using techniques like story points, ideal days, or hours. The team can use historical data or expert judgment to make these estimates.

7. Capacity Planning: Determine the team's capacity for the sprint by considering factors like team size, availability, and any external dependencies.

8. Sprint Goal: Define a clear goal for the sprint, which aligns with the overall project objectives and the selected user stories.

9. Sprint Backlog: Create a sprint backlog by assigning the estimated tasks to team members based on their skills and availability.

10. Review and Refinement: Regularly review the progress during the sprint and refine the estimates and plans as needed.

It's important to note that the specific details of sprint planning and estimation may vary depending on the development methodology being used, such as Scrum or Kanban.

6.3 Sprint Delivery Schedule

Sprint delivery for an electronic voting system involves the iterative development and release of functionality in short, time-boxed sprints. Here is an overview of the sprint delivery process for an electronic voting system:

1. Sprint Planning: At the beginning of each sprint, the development team selects a set of user stories or features from the product backlog to be implemented and delivered within the sprint. These user stories should be small enough to be completed within the sprint duration, typically 1-4 weeks.

2. Development: During the sprint, the development team works on implementing the selected user stories. They collaborate, code, and test the functionality to ensure it meets the defined acceptance criteria.

3. Daily Stand-ups: The team holds daily stand-up meetings to discuss progress, challenges, and any adjustments needed to meet the sprint goal. This helps in maintaining transparency and addressing any issues promptly.

4. Continuous Integration and Testing: As the development progresses, continuous integration and testing practices are followed to ensure that the newly developed functionality integrates smoothly with the existing system and meets the required quality standards.

5. Sprint Review: At the end of the sprint, a sprint review meeting is conducted to demonstrate the completed functionality to stakeholders, such as product owners, users, and other relevant parties. Feedback is gathered, and any necessary adjustments or refinements are identified.

6. Sprint Retrospective: Following the sprint review, a sprint retrospective meeting takes place to reflect on the sprint process and identify areas for improvement. The team discusses what went well, what could be improved, and any action items to enhance future sprints.

7. Incremental Release: If the completed functionality meets the necessary criteria and is deemed ready for release, it can be deployed to a testing or production environment. This allows stakeholders to start using and evaluating the new features.

8. Next Sprint Planning: With the completion of one sprint, the cycle continues with the next sprint planning session, where the team selects new user stories or features to be implemented in the upcoming sprint.

By following this iterative approach, the electronic voting system can be developed and delivered incrementally, allowing for continuous feedback, improvement, and adaptation throughout the development process.

7. CODING & SOLUTIONING (Explain the features added in the project along with code)

Program:

Index:

```
<!DOCTYPE html>
<html lang="en">
  <head>
  <meta charset="utf-8" />
    <link rel="icon" href="%PUBLIC_URL%/favicon.ico" />
    <meta name="viewport" content="width=device-width,
initial-scale=1" />
    <meta name="theme-color" content="#000000" />
    <meta
      name="description"
      content="Web site created using create-react-app"
  />
    <link rel="apple-touch-icon"
href="%PUBLIC_URL%/logo192.png" />
    <!--
      manifest.json provides metadata used when your web app
is installed on a
      user's mobile device or desktop. See
https://developers.google.com/web/fundamentals/web-app-
manifest/
    -->
    <link rel="manifest" href="%PUBLIC_URL%/manifest.json"
/>
    <!--
      Notice the use of %PUBLIC_URL% in the tags above.
      It will be replaced with the URL of the `public`
folder during the build.
      Only files inside the `public` folder can be
referenced from the HTML.


   Unlike "/favicon.ico" or "favicon.ico",
"%PUBLIC_URL%/favicon.ico" will
      work correctly both with client-side routing and a
non-root public URL.
      Learn how to configure a non-root public URL by
running `npm run build`.
    -->
    <title>React App</title>
  </head>
  <body>
  <noscript>You need to enable JavaScript to run this
app.</noscript>
    <div id="root"></div>
    <!--
      This HTML file is a template.
      If you open it directly in the browser, you will see
an empty page.

      You can add webfonts, meta tags, or analytics to this
file.
      The build step will place the bundled scripts into the
<body> tag.

      To begin the development, run `npm start` or `yarn
start`.
      To create a production bundle, use `npm run build` or
`yarn build`.
    -->
  </body>
</html>
```

Voting:

```
// SPDX-License-Identifier: MIT
pragma solidity ^0.8.0;

contract VoteSystem{
    address public owner;
    constructor(){
        owner= msg.sender;
    }

 struct candidate {
     uint voterId;
     string name;
     uint age;
     uint voteCount;
 }

 mapping (uint => candidate) candidateMap;

 struct voters {
     uint voterId;
     string name;
     uint age;
     bool votingState;
 }

 mapping (uint => voters) votersMap;
 mapping (uint=>bool) registeredVoter;

 modifier checkVoterVoted(uint _votersVoterId){
     require (votersMap[_votersVoterId].votingState ==
false);
     _;
 }

modifier checkRegisteredVoter(uint _votersVoterId){
        require(registeredVoter[_votersVoterId]==true,
"Voter is not Registered");
        _;
}
uint[] voterIdlist;
uint[] candidateIdList;


 function enrollCandidate(uint _voterId,string memory
_name,uint  _age )  public {

 require (_age >= 25);
 require (candidateMap[_voterId].voterId != _voterId);

    candidateMap[_voterId].voterId = _voterId;
    candidateMap[_voterId].name = _name;
    candidateMap[_voterId].age = _age;

    candidateIdList.push(_voterId);
 }

 function enrollVoter(uint _voterId,string memory _name,uint
_age)  public  returns(bool){

require (_age >= 18);
require (votersMap[_voterId].voterId != _voterId);

     votersMap[_voterId].voterId = _voterId;
     votersMap[_voterId].name = _name;
     votersMap[_voterId].age = _age;

     voterIdlist.push(_voterId);
    return registeredVoter[_voterId]=true;

 }
 function getCandidateDetails(uint _voterId) view public
returns(uint,string memory,uint,uint) {

     return
(candidateMap[_voterId].voterId,candidateMap[_voterId].name,
candidateMap[_voterId].age,candidateMap[_voterId].voteCount)
;
 }

 function getVoterDetails(uint _voterId) view public returns
(uint,string memory,uint,bool){

     return
(votersMap[_voterId].voterId,votersMap[_voterId].name,voters
Map[_voterId].age,votersMap[_voterId].votingState);

 }

 function vote(uint _candidateVoterId,uint _votersVoterId)
public checkVoterVoted(_votersVoterId)
checkRegisteredVoter(_votersVoterId) {
     candidateMap[_candidateVoterId].voteCount += 1;
     votersMap[_votersVoterId].votingState = true;
 }

 function getVotecountOf(uint _voterId) view public
returns(uint){
        require(msg.sender== owner, "Only owner is allowed
to Check Results");
     return candidateMap[_voterId].voteCount;
 }

 function getVoterList() view public returns (uint[]
memory){
    return   voterIdlist;
    }

 function getCandidateList() view public returns(uint[]
memory){



 return candidateIdList;
 }



}
```

Performace Metrics

Performance metrics are measurements used to assess the effectiveness and efficiency of a system, process, or individual. In the context of an e-voting system, some relevant performance metrics could include:

1. Response Time: This metric measures the time taken for the system to respond to user actions, such as casting a vote or retrieving election results. Lower response times generally indicate better system performance.

2. Throughput: Throughput refers to the number of transactions or operations that the system can handle within a given time period. Higher throughput indicates better system capacity and efficiency.

3. Security: Security metrics evaluate the effectiveness of the system's security measures in protecting against unauthorized access, tampering, or manipulation of votes or voter information. This includes metrics such as the number of security incidents or the success rate of security controls.

4. User Satisfaction: User satisfaction metrics gauge the level of satisfaction and usability experienced by voters when interacting with the e-

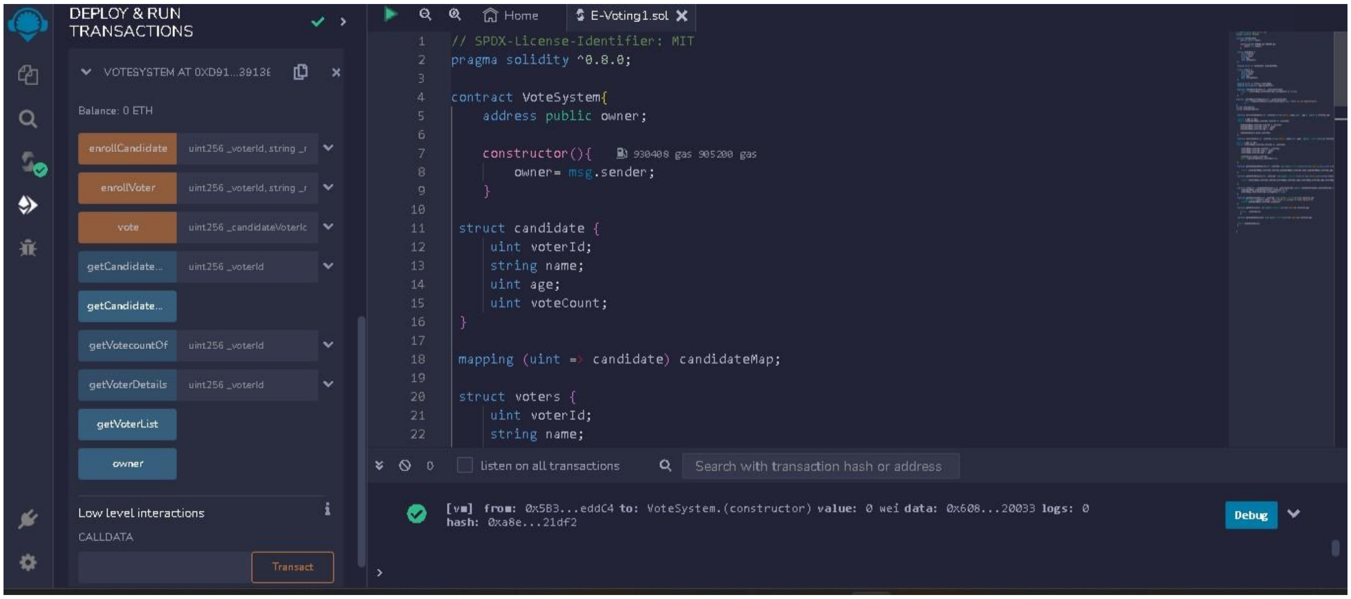
voting system. This can be measured through surveys, feedback, or user experience testing.

5. Error Rate: Error rate measures the frequency of errors or failures encountered during the operation of the e-voting system. Lower error rates indicate better system reliability and stability.

It is important to define specific targets or thresholds for these performance metrics to track the system's performance over time and identify areas for improvement. Regular monitoring and analysis of these metrics can help ensure the e-voting system's optimal performance and user satisfaction.

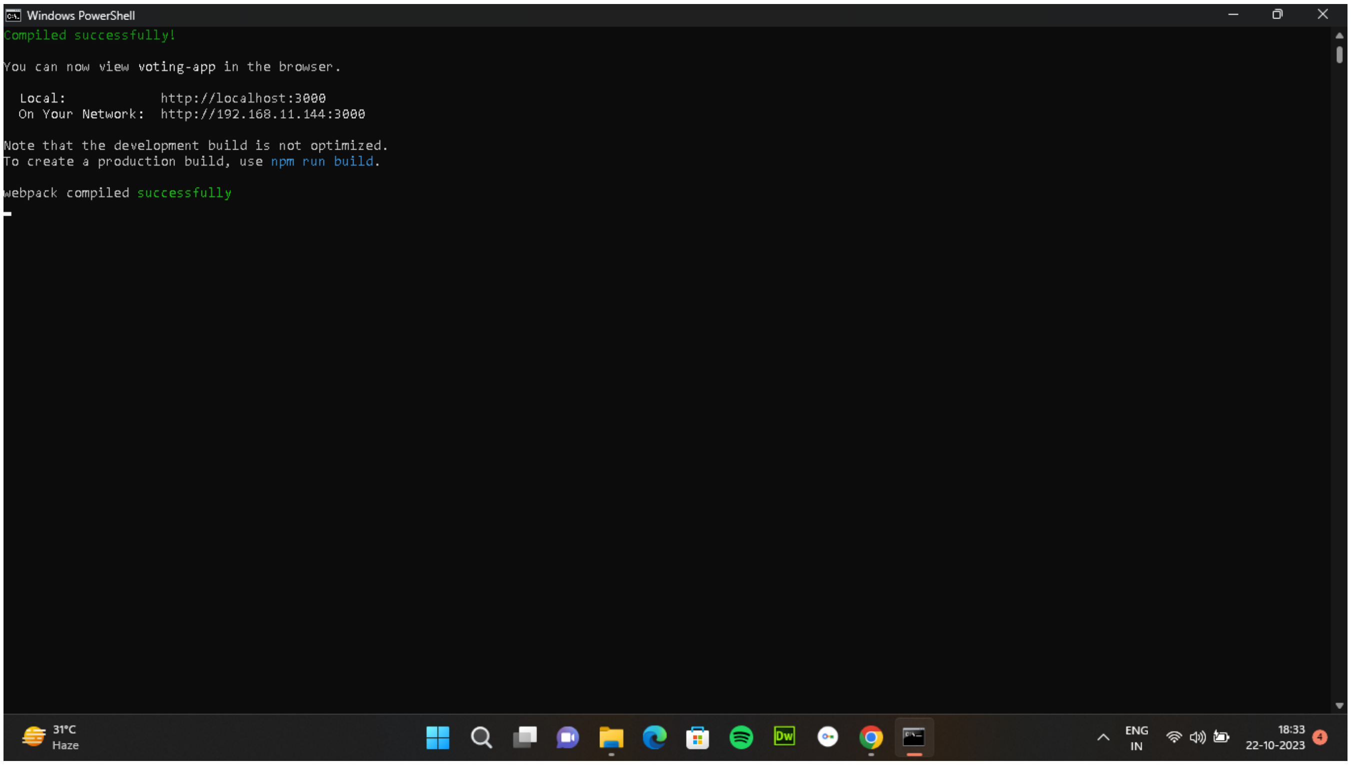
9. RESULTS

Output Screenshots

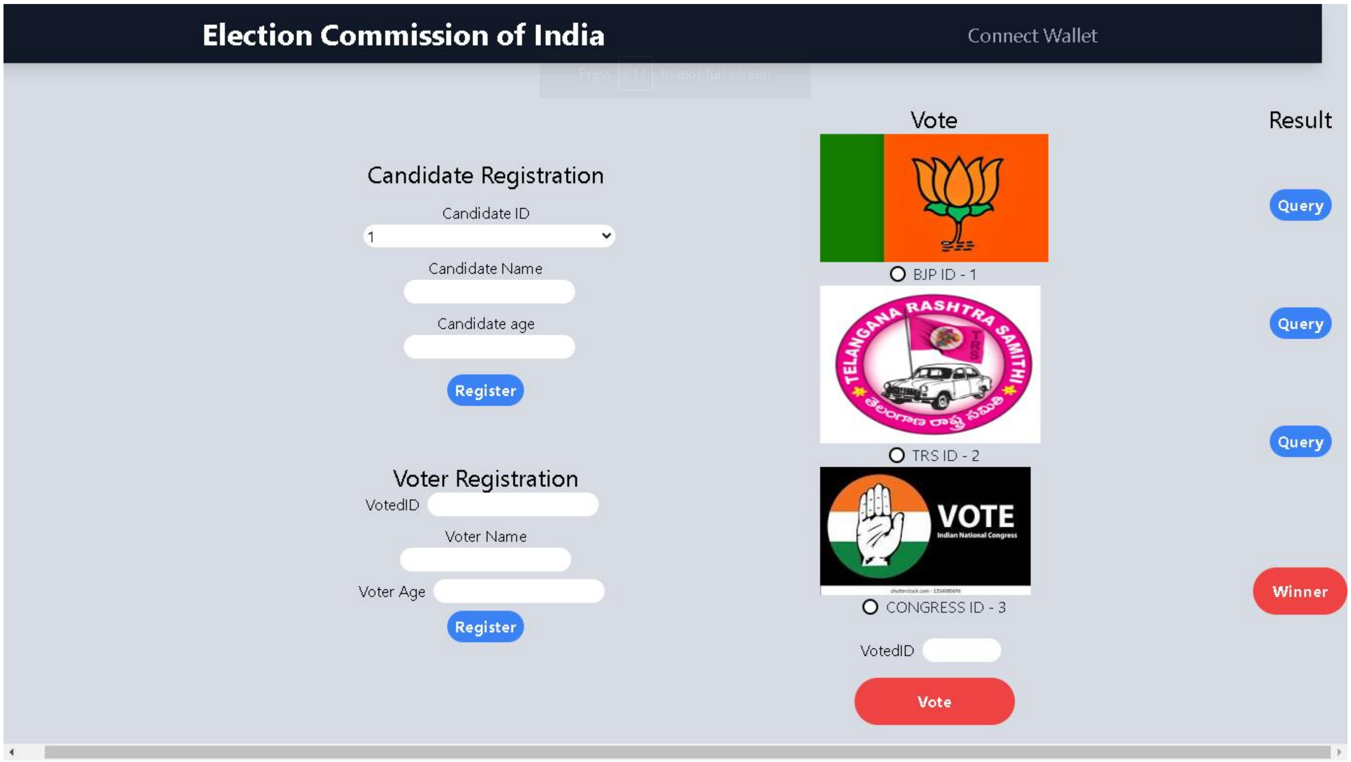


```
// SPDX-License-Identifier: MIT
pragma solidity ^0.8.0;

contract VoteSystem{
    address public owner;

    constructor(){
        owner= msg.sender;
    }

 struct candidate {
     uint voterId;
     string name;
     uint age;
     uint voteCount;
 }

 mapping (uint => candidate) candidateMap;

 struct voters {
     uint voterId;
     string name;
```

```
[vm] from: 0x5B3...eddC4 to: VoteSystem.(constructor) value: 0 wei data: 0x608...20033 logs: 0
hash: 0xa8e...21df2
```



```
Compiled successfully!

You can now view voting-app in the browser.

  Local:            http://localhost:3000
  On Your Network:  http://192.168.11.144:3000

Note that the development build is not optimized.
To create a production build, use npm run build.

webpack compiled successfully
```



10. ADVANTAGES & DISADVANTAGES

ADVANTAGES

The advantages of an electronic voting system include:

1. Efficiency: Electronic voting systems can streamline the voting process, reducing the time and effort required for both voters and election officials. It eliminates the need for manual counting of paper ballots, leading to faster and more accurate results.

2. Accessibility: Electronic voting systems can be designed to accommodate various accessibility needs, such as providing options for visually impaired individuals or those with physical disabilities. This ensures that voting is accessible to a wider range of voters.

3. Convenience: Electronic voting allows voters to cast their ballots from anywhere, reducing the need for physical presence at polling stations. This convenience can encourage higher voter turnout and participation.

4. Accuracy: Electronic voting systems can minimize errors and reduce the likelihood of invalid or spoiled ballots. The use of technology can help in validating voter choices and ensuring accurate vote counting.

5. Transparency: Electronic voting systems can provide a transparent and auditable process. They can include features like digital signatures, encryption, and secure databases, which enhance the integrity of the voting process and make it easier to detect and prevent fraud.

6. Flexibility: Electronic voting systems offer flexibility in terms of ballot design and customization. They can accommodate different types of elections, such as national, regional, or local, and allow for easy modifications to the voting process if needed.

7. Environmental Impact: By reducing the use of paper and other physical resources, electronic voting systems can have a positive environmental impact. They contribute to sustainability efforts by minimizing waste and energy consumption associated with traditional paper-based voting methods.

It's important to note that the successful implementation of an electronic voting system requires robust security measures, user-friendly interfaces, and public trust in the technology. Proper planning, testing, and ongoing maintenance are crucial to ensure the integrity and reliability of the system.

DISADVANTAGES

While e-voting offers several advantages, it also has some disadvantages that need to be considered. Here are a few:

1. Security Concerns: E-voting systems are vulnerable to hacking, tampering, and other cyber threats. Ensuring the integrity and confidentiality of votes is a significant challenge. Malicious actors could potentially manipulate or alter the results, undermining the trust in the electoral process.

2. Lack of Transparency: E-voting systems often rely on complex algorithms and proprietary software, making it difficult for voters and even experts to fully understand and verify the accuracy of the system. This lack of transparency can raise doubts about the legitimacy of the results.

3. Accessibility and Inclusivity: E-voting may exclude certain segments of the population, such as elderly or disabled individuals who may have difficulty using electronic devices or lack access to the necessary technology. This can lead to unequal participation and potential disenfranchisement.

4. Technological Dependencies: E-voting systems require reliable and robust technology infrastructure, including internet connectivity and power supply. In areas with limited infrastructure or during technical failures, the voting process may be disrupted, leading to delays or even complete system failures.

5. Cost and Complexity: Implementing and maintaining e-voting systems can be expensive, requiring significant investments in hardware, software, training, and security measures. Additionally, the complexity of these systems may pose challenges for election officials and voters alike, potentially leading to confusion and errors.

6. Lack of Paper Trail: Some e-voting systems do not provide a verifiable paper trail or audit trail, making it difficult to conduct post-election audits or recounts. This can undermine the ability to verify the accuracy of the results and address any potential disputes.

It is important to carefully consider these disadvantages and address them through robust security measures, transparency, accessibility provisions, and thorough testing before implementing e-voting systems.

11. CONCLUSION

The electronic voting, however, has emerged as an alternative but still not being practiced at a large scale. The electronic voting is anticipated to have a great future yet the past is not that glorious. In some countries e-voting is not an option while few are in a process to eliminate the security, verifiability, and anonymity concerns. There are issues that require immensely deep consideration by the legislatures, technologist, the people. This research has proposed a framework based on the adjustable blockchain that can apprehend the problems in the polling process, selection of the suitable hash algorithm, selection of adjustments in the blockchain, process of voting data management, and the security and authentication of the voting process. The power of blockchain has been used adjustably to fit into the dynamics of the electronic voting process.

12. FUTURE SCOPE

The future scope of electronic voting systems holds significant potential for enhancing the democratic process. Here are some potential areas of development and improvement:

1. Increased Accessibility: Electronic voting systems can be designed to be more accessible for individuals with disabilities or those who face physical barriers to traditional voting methods. This could include features such as assistive technologies, multiple language options, and user-friendly interfaces.

2. Enhanced Security: Continuous advancements in technology can lead to improved security measures in electronic voting systems. Implementing robust encryption, secure authentication methods, and tamper-proof infrastructure can help ensure the integrity and confidentiality of votes.

3. Blockchain Technology: The use of blockchain technology in electronic voting systems has gained attention due to its potential for transparency and immutability. Blockchain can provide a decentralized and verifiable ledger, reducing the risk of fraud or manipulation.

4. Remote Voting: The ability to vote remotely through electronic voting systems can increase voter participation, especially for individuals who are unable to physically visit polling stations. However, ensuring the security and authenticity of remote voting remains a challenge that needs to be addressed.

5. Improved Verification and Auditability: Future electronic voting systems can incorporate advanced verification mechanisms to allow voters to verify that their votes were accurately recorded and counted. Additionally, robust audit trails and mechanisms for post-election audits can enhance transparency and trust in the system.

6. Usability and User Experience: Focus on improving the usability and user experience of electronic voting systems can help increase adoption and reduce barriers to participation. Intuitive interfaces, clear instructions, and user testing can contribute to a smoother voting experience.

7. Integration with Identity Management Systems: Integration with reliable identity management systems can help ensure that only eligible voters can participate in the electronic voting process. This can involve leveraging technologies such as biometrics or digital identity verification.

8. Public Trust and Confidence: Building public trust and confidence in electronic voting systems is crucial. This can be achieved through transparent processes, independent audits, and open communication about the system's security measures and safeguards.

It's important to note that the implementation of electronic voting systems should be accompanied by thorough testing, stakeholder engagement, and legal frameworks to address concerns related to privacy, security, and inclusivity.

13.APPENDIX

Source Code

GitHub & Project Demo Link

Maharajan Project GitHub Link:
https://github.com/MaharajanR45/Electronic-Voting-System/tree/main/Project/Maharajan

Mathankumar Project GitHub Link:
https://github.com/MaharajanR45/Electronic-Voting-System/tree/main/Project/Mathankumar

Balaji Project GitHub Link:
https://github.com/MaharajanR45/Electronic-Voting-System/tree/main/Project/Balaji

Vasanthakumar Project GitHub Link:
https://github.com/MaharajanR45/Electronic-Voting-System/tree/main/Project/Vasanthakumar

Project Demo Link:
https://drive.google.com/file/d/1w2b21jg6DaP3T21LTs1xv0AGWEgiEOlt/view?usp=drive_link