

# **BIOMETRIC SECURITY SYSTEM FOR VOTING PLATFORM**

## **PROJECT REPORT**

### **SUBMITTED BY**

**TEAM ID: NM2023TMID04472**

ASHIKHA A -963520106013

MENTOR: Mrs. BABY SOLA

ABINAYA S -963520106003

SPOC: Dr. BHARADWAJ

BALA DEVIKA B -963520106015

JEMI J A BENISHA -963520106022

In the partial fulfillment of the requirements for the award of a degree of

**BACHELOR OF ENGINEERING**

**IN**

**ELECTRONICS AND COMMUNICATION ENGINEERING**

**STELLA MARY'S COLLEGE OF ENGINEERING**

**ARUTHENGANVILAI, KALLUKATTI JUNCTION**

**AZHIKKAL (PO), KANYAKUMARI-629202.**

**2023 – 2024 (ODD)**

# CONTENT

1. **INTRODUCTION**
  - 1.1 Project Overview
  - 1.2 Purpose
2. **LITERATURE SURVEY**
  - 2.1 Existing Problem
  - 2.2 References
  - 2.3 Problem Statement Definition
3. **IDEATION & PROPOSED SOLUTION**
  - 3.1 Empathy Map Canvas
  - 3.2 Ideation & Brainstorming
4. **REQUIREMENT ANALYSIS**
  - 4.1 Functional Requirements
  - 4.2 Non-Functional Requirements
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
  - 6.3 Sprint Delivery Schedule
7. **CODING & SOLUTION**
8. **PERFORMANCE TESTING**
9. **RESULT**
- 10 **ADVANTAGES & DISADVANTAGES**  
.
- 11 **CONCLUSION**  
.
- 12 **FUTURE SCOPE**  
.
- 13 **APPENDIX**  
.

# 1. Introduction

Block chain is a revolutionary technology that has fundamentally transformed the way we think about data, transactions, and trust in the digital age. It emerged as the underlying technology for crypto currencies like Bit coin, but its applications extend far beyond digital currencies. At its core, block chain is a decentralized, distributed ledger technology that offers a secure and transparent way to record and verify transactions and data.

In a traditional centralized system, a single entity, such as a bank or a government, maintains a central ledger to record and verify transactions. In contrast, block chain operates as a decentralized ledger shared across a network of computers, known as nodes. Each node stores a copy of the block chain, and the system uses a consensus mechanism to ensure that all copies of the ledger remain in sync and accurate.

There is no central authority or intermediary in control of the block chain. This decentralized nature makes it resistant to censorship and tampering. All transactions and data recorded on the block chain are visible to all participants in the network. This transparency enhances trust and accountability. Block chain employs cryptographic techniques to secure data and transactions. Once a block of data is added to the chain, it is virtually impossible to alter, ensuring the integrity of the ledger.

Block chain technology continues to evolve, and its potential applications are continually expanding. Its decentralized, secure, and transparent nature makes it a powerful tool for industries and sectors seeking to enhance trust and efficiency in a digital world. In this paper we introduce about the Biometric Service of Voting platform in block chain.

## 1.1 Project Overview

The "Biometric Security System for Voting Platform" project is a comprehensive initiative aimed at enhancing the security, accuracy, and transparency of the voting process through the integration of biometric technology. This project seeks to address the critical issues of voter authentication, fraud prevention, and the overall integrity of the electoral system. By leveraging biometric data, the project aims to establish a reliable and tamper-proof voting platform that ensures legitimate voters can participate while thwarting unauthorized or fraudulent attempts.

The project will commence with the enrollment of eligible voters, during which biometric data, such as fingerprints, iris scans, or facial recognition, will be collected and securely stored in a central database. Biometric data gathered during enrollment will serve as the primary method for voter authentication during the actual voting process, thus reinforcing the system's security.

An intuitive, user-friendly voting application or kiosk will be developed to streamline the voting process. Voters will be required to authenticate themselves through their biometric data before casting their ballots. The project will diligently ensure compliance with all relevant laws and regulations, including data protection and election legislation, to uphold the highest standards of legality and ethics.

The "Biometric Security System for Voting Platform" project represents a significant stride in modernizing and securing the electoral process, ensuring that every eligible voter can participate with confidence in the integrity and security of the system.

## **1.2 Purpose**

The purpose of implementing a Biometric Security System for a Voting Platform is to enhance the integrity, security, and reliability of the voting process. Ensure that each voter is accurately and securely authenticated, preventing unauthorized or fraudulent voting. Mitigate the risk of voter fraud, identity theft, and tampering with the electoral process. Create an immutable and transparent ledger of all votes, making it extremely difficult to alter or manipulate the voting data. Build trust among voters by providing a secure and transparent voting platform, ultimately increasing voter confidence in the electoral process. Ensure that all eligible voters, including those with disabilities, can participate in the voting process through user-friendly biometric authentication methods. Safeguard biometric data and voting records, complying with data protection regulations to protect voter privacy. Reduce the risk of individuals voting multiple times at different polling stations or locations. Improve the accuracy of the voting process by reducing errors in voter identification and authentication.

In summary, the primary purpose of implementing a Biometric Security System for a Voting Platform is to create a secure, transparent, and accessible voting environment that safeguards the rights of eligible voters, prevents fraud, and upholds the principles of democracy. This system is designed to ensure that each vote is accurately counted and that the electoral process is conducted with the highest level of integrity and trustworthiness.

## **2. Literature Survey**

### **2.1 Existing Problem**

While the concept of implementing a Biometric Security System for a Voting Platform holds significant promise for enhancing the integrity of elections, there are several existing problems and challenges associated with its implementation. These challenges need to be carefully addressed to ensure the system's effectiveness and fairness.

Collecting and storing biometric data, such as fingerprints or facial recognition, raises significant privacy concerns. Voters may worry about how their sensitive personal data will be used, protected, and potentially exposed in the event of a data breach. Biometric systems are not infallible. They can produce false positives and false negatives, potentially leading to legitimate voters being denied their right to vote or allowing unauthorized individuals to cast ballots.

While biometrics can enhance security, they are not immune to attacks. Biometric data can be spoofed, stolen, or manipulated. Ensuring the security of the biometric data and the system itself is a complex challenge. Implementing biometric systems can be expensive, requiring investments in hardware, software, and personnel training. Smaller or financially constrained electoral authorities may struggle to afford such systems. There may be concerns about the accuracy, fairness, and transparency of biometric voting systems. Addressing these existing problems involves a combination of robust technological solutions, legal and regulatory frameworks, public education, and ongoing vigilance. It is essential to strike a balance between improving election security and preserving the accessibility, privacy, and trustworthiness of the voting process.

## 2.2 References

Academic databases like IEEE Xplore, ACM Digital Library, or Google Scholar for peer-reviewed articles and research papers on biometric security in voting systems.

Government election commissions or relevant agencies often publish reports and guidelines on election security, including biometric authentication.

Alaguvel R., Gnanavel G., Jagadhambal K. – "Biometrics using Electronic Voting System with Embedded Security", pp. 1065, 2013.

O.M. Olaniyan, T. Mapayi & S.A. Adejumo – "A Proposed Multiple Scan Biometric-Based System for Electronic Voting", African Journal Comp. & ICT Volume 4. No. 2. Issue 1pp. 12, 2011.

Kashif H.M., Dileep Kumar and Syed Muhammad Usman, "Next Generation A Secure E-Voting System Based On Biometric Fingerprint Method" 2011 International Conference on Information and Intelligent Computing IPCSIT vol.18 (2011) pp .26-27.

OASIS Election & Voter Services Technical Committee – "Requirements for common data formats and standards for e-Voting", NIST Paper. 18 August 2009 (Retrieved October 10, 2014).

Government election commissions or relevant agencies often publish reports and guidelines on election security, including biometric authentication.

Look for news articles and reports from reputable news outlets that discuss the implementation and challenges of biometric security in voting systems.

Organizations specializing in election technology and security often publish whitepapers and case studies on the subject.

De Giusti A., Feierherd G., Pesado P., Depetris B. "Una aproximación a los requerimientos del software de voto electrónico de Argentina". Congreso Argentino de Ciencias de la Computación. 2004.

Tula M. "Voto Electrónico". Ariel Ciencias Políticas. 2005.

Cantijoch Cunill M. "El voto electrónico ¿Un temor justificado?". Revista TEXTOS de la CiberSociedad, 7. <http://www.cibersociedad.net>. 2005.

Arsaute G. A., Tutores: Nasisi Óscar Herminio M. M. "Reconocimiento de características en huellas dactilares para la identificación humana". Universidad Nacional de San Juan. Facultad de Ingeniería. Instituto de Automática. 1997.

Beavan Colin. "Huellas dactilares. Los orígenes de la dactiloscopía". Ed. Alba. 1990.

Arrieta A., Marín J., Sánchez L. G., Romero L., Sánchez L. A., Batista V. "Gestión y Reconocimiento Óptico de los Puntos Característicos de Imágenes de Huellas Dactilares". Universidad de Salamanca.

Reid P. "Biometrics for Network Security". Prentice Hall. 2004.

Chirillo J. y otros. "Implementing Biometric Security". Wiley Publishing. 2003.

## 2.3 Problem Statement Definition

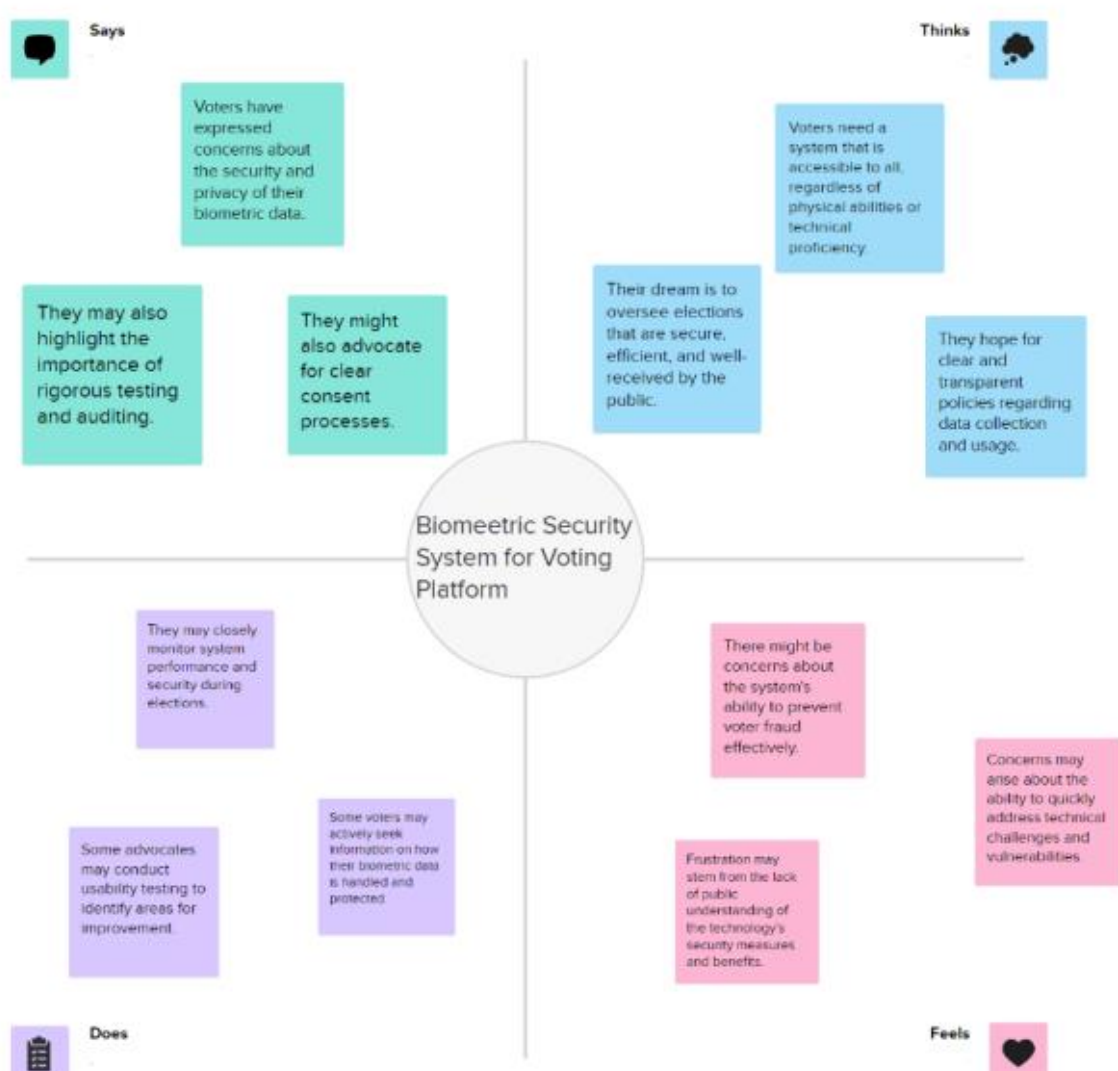
In modernizing and securing the voting process, the deployment of a Biometric Security System for a Voting Platform presents several complex challenges. The current electoral landscape is fraught with issues related to voter authentication, fraud prevention, privacy concerns, and technology implementation. This problem statement seeks to address these challenges and find effective solutions for the successful integration of biometric security into voting systems, ensuring the integrity, accessibility, and privacy of the electoral process.

Existing voting systems are vulnerable to fraud, such as voter impersonation and multiple voting. This can undermine the integrity of democratic elections and lead to a loss of public trust in the voting process.

All people have their voting rights, but in some cases their rights are being taken from them. So, the people get frustrated on losing their rights.


## 3. Ideation & Proposed Solution

### 3.1 Empathy Map Canvas



## 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

1

Team gathering

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

2

Set the goal

Think about the problem you'll be focusing on solving in the brainstorming session.

3

Learn how to use the facilitation tools

Use the Facilitation Superpowers to run a happy and productive session.

[Open article](#) ➔

1

#### 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

Existing voting systems are vulnerable to fraud, such as voter impersonation and multiple voting. This can undermine the integrity of democratic elections and lead to a loss of public trust in the voting process.

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.

2

#### Brainstorm

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

⌚ 10 minutes

Asishka

Use biometric data to register voters.

Use blockchain-based platform to create a secure and tamper-proof record of voter eligibility.

Adanya

Use biometric data to register voters.

Use blockchain-based platform to create a secure and tamper-proof record of voter eligibility.

Beto Devika

Use biometric data to register voters.

Use blockchain-based platform to create a secure and tamper-proof record of voter eligibility.

Jenil Beriksha

Use biometric data to register voters.

Use blockchain-based platform to create a secure and tamper-proof record of voter eligibility.

TP

You can select a sticky note and use the panel toolbar to easily (don't start drawing)

3

#### 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 can break it up into smaller sub-groups.

⌚ 20 minutes

Voters could register their biometric data, such as their fingerprint or facial scan, on a blockchain-based platform. This would create a secure and tamper-proof record of voter eligibility.

Send OTP to confirm the person

Scan the eye to check whether it match or not

Use their own finger print

On election day, voters could use their biometric data to authenticate themselves at the polls. This would prevent voter impersonation and multiple voting.

voters could use their biometric data to authenticate themselves at the polls

It can process a large number of transactions quickly and securely.

TP

Add customizable tags to sticky notes to make it easier to find, browse, organize, and categorize important ideas as themes within your cluster.

4

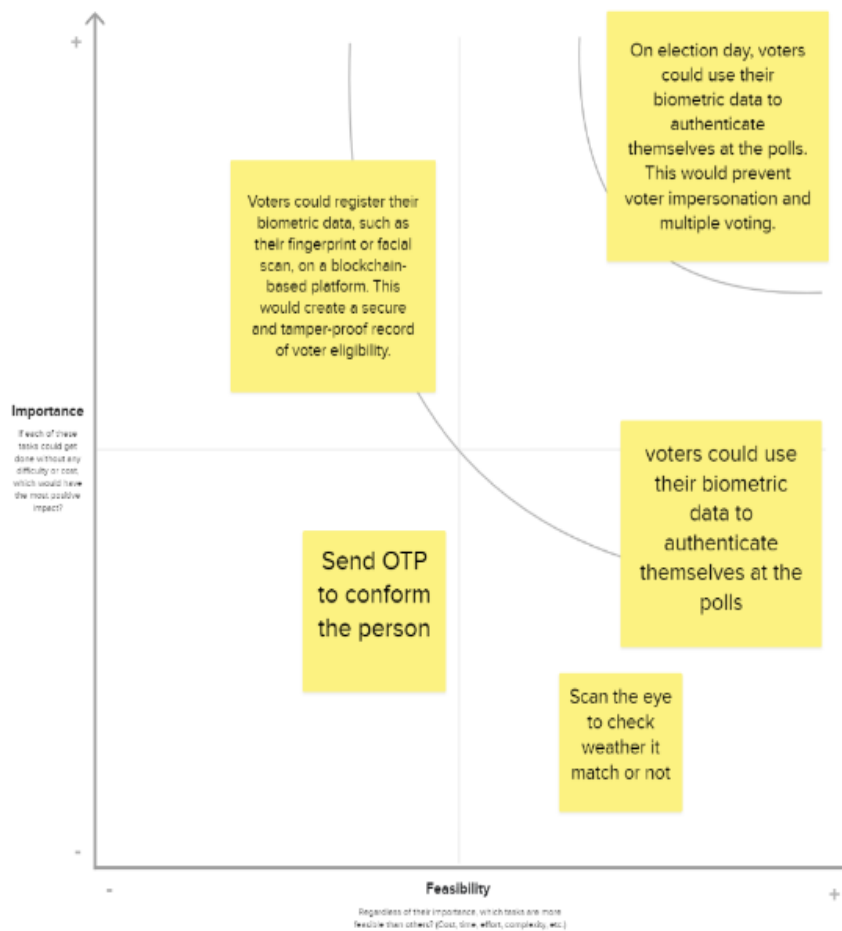
## Prioritize

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

### TIP

Participants can use their cursor to point at where sticky notes should go on the grid. The facilitator can confirm the spot by using the laser pointer holding the H key on the keyboard.



→

## 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



#### Share the mural

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



#### 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.

[Open the template →](#)



#### Customer experience journey map

Understand customer needs, motivations, and obstacles for an experience.

[Open the template →](#)



#### Strengths, weaknesses, opportunities & threats

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

[Open the template →](#)

[Share template feedback](#)



## **4. Requirement Analysis**

### **4.1 Functional Requirements**

Functional requirements are a critical aspect of developing a Biometric Security System for a Voting Platform based on block chain technology. These requirements define what the system should do and outline its specific features and capabilities.

#### **1. Voter Registration and Enrollment:**

The system should allow eligible voters to register and enroll their biometric data securely.

Voters' biometric data should be stored in an encrypted and tamper-proof manner on the blockchain.

#### **2. Biometric Data Capture:**

The system should support multiple biometric data types, such as fingerprints, facial recognition, or iris scans. It should capture and authenticate biometric data during the voter registration process.

#### **3. Identity Verification:**

Use block chain to provide decentralized identity verification to ensure voters are eligible to participate.

#### **4. Voter Authentication:**

The system should authenticate voters using their stored biometric data via a secure and user-friendly interface. Biometric authentication should take place in real-time to ensure accuracy.

#### **5. Voter Anonymity:**

Implement measures to ensure voter anonymity while still allowing for secure authentication. Use encryption techniques to protect voter identity and choices.

#### **6. Secure Voting Process:**

Develop a secure and user-friendly voting application or kiosk for casting ballots. Implement smart contracts to automate the voting process and prevent double voting.

#### **7. Vote Recording:**

Record each vote as a transaction on the block chain in an immutable and transparent manner. Use cryptographic techniques to pseudonymize votes while preserving their authenticity.

#### **8. Voting Accessibility:**

Ensure that the voting system is accessible to voters with disabilities, with features like voice commands, screen readers, and tactile interfaces.

## 9. Consensus Mechanism:

Choose and implement a robust consensus mechanism, such as Proof of Work (PoW) or Proof of Stake (PoS), to validate and secure transactions on the block chain.

## 10. Public and Private Block chain Options:

Allow for both public and private block chain configurations, depending on the specific needs of the voting system. Public block chains provide transparency, while private block chains offer increased control and privacy.

## 4.2 Non-Functional Requirements

Non-functional requirements describe the characteristics or qualities of a system that are not directly related to its specific functions but are crucial for its overall performance and effectiveness.

### 1. Security:

**Data Security:** The system must provide a high level of security for biometric data, voting records, and authentication processes. It should use robust encryption and access control measures to protect sensitive information.

**Resilience to Attacks:** The system should be resilient to various cyberattacks, including Distributed Denial of Service (DDoS), man-in-the-middle, and intrusion attempts.

**Smart Contract Security:** Ensure the security of smart contracts to prevent vulnerabilities and unauthorized access.

### 2. Scalability:

The system should be able to handle a large number of voters and transactions, especially during national elections, without a significant drop in performance or responsiveness.

### 3. Performance:

**Real-Time Processing:** Provide real-time processing for biometric authentication and vote recording to ensure a smooth and efficient voting experience.

**Low Latency:** Minimize system response times to prevent voter frustration and bottlenecks in the voting process.

### 4. Reliability and Availability:

Ensure that the system is highly reliable, with minimal downtime. It should be available during election hours and capable of recovering quickly from any interruptions.

### 5. Compliance and Legal Regulations:

Adhere to local and international data protection and privacy regulations, ensuring compliance with relevant laws. Facilitate audits and reporting to demonstrate legal and regulatory compliance.

## 6. Usability:

The system should be user-friendly, with intuitive interfaces for both voters and election officials. It should be accessible to users with varying technical proficiencies.

## 7. Interoperability:

Ensure compatibility with existing voting systems and infrastructure, allowing for smooth integration with the broader electoral process.

## 8. Auditability:

Provide a comprehensive audit trail for all transactions on the blockchain, allowing for transparency and accountability. Support independent audits of the system's security, privacy, and compliance.

## 9. Disaster Recovery:

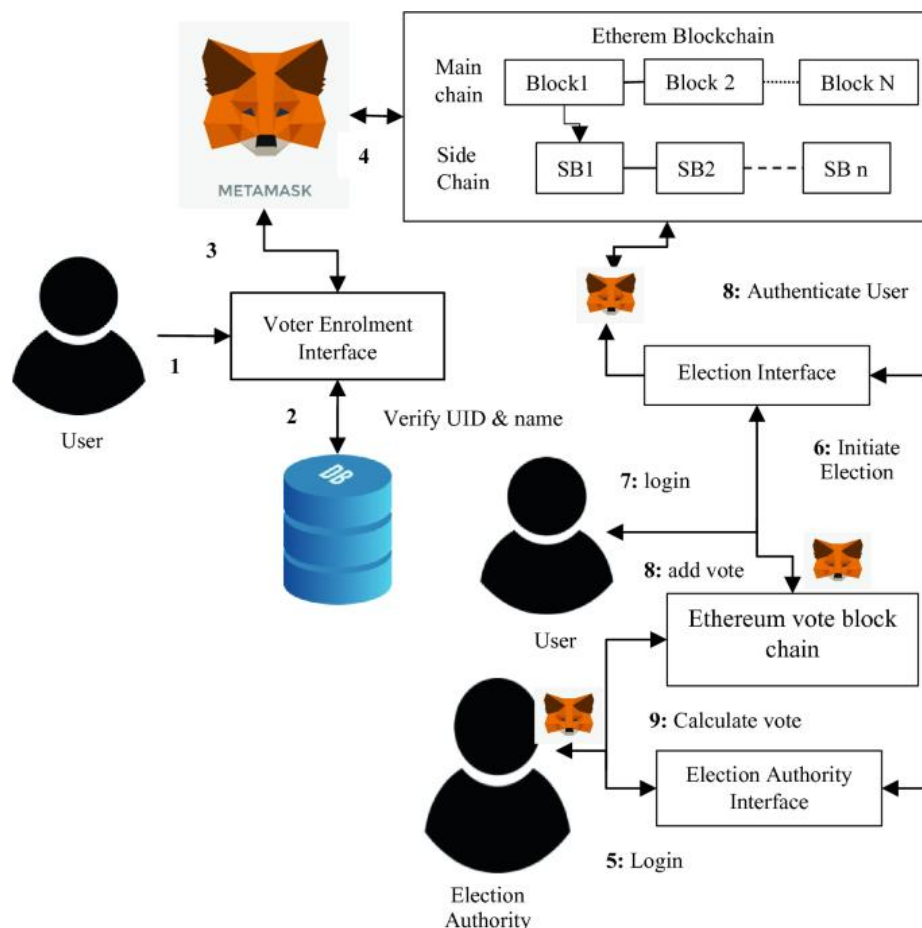
Implement robust disaster recovery measures to maintain system operations during unforeseen events, such as natural disasters or technical failures.

## 10. Resource Efficiency:

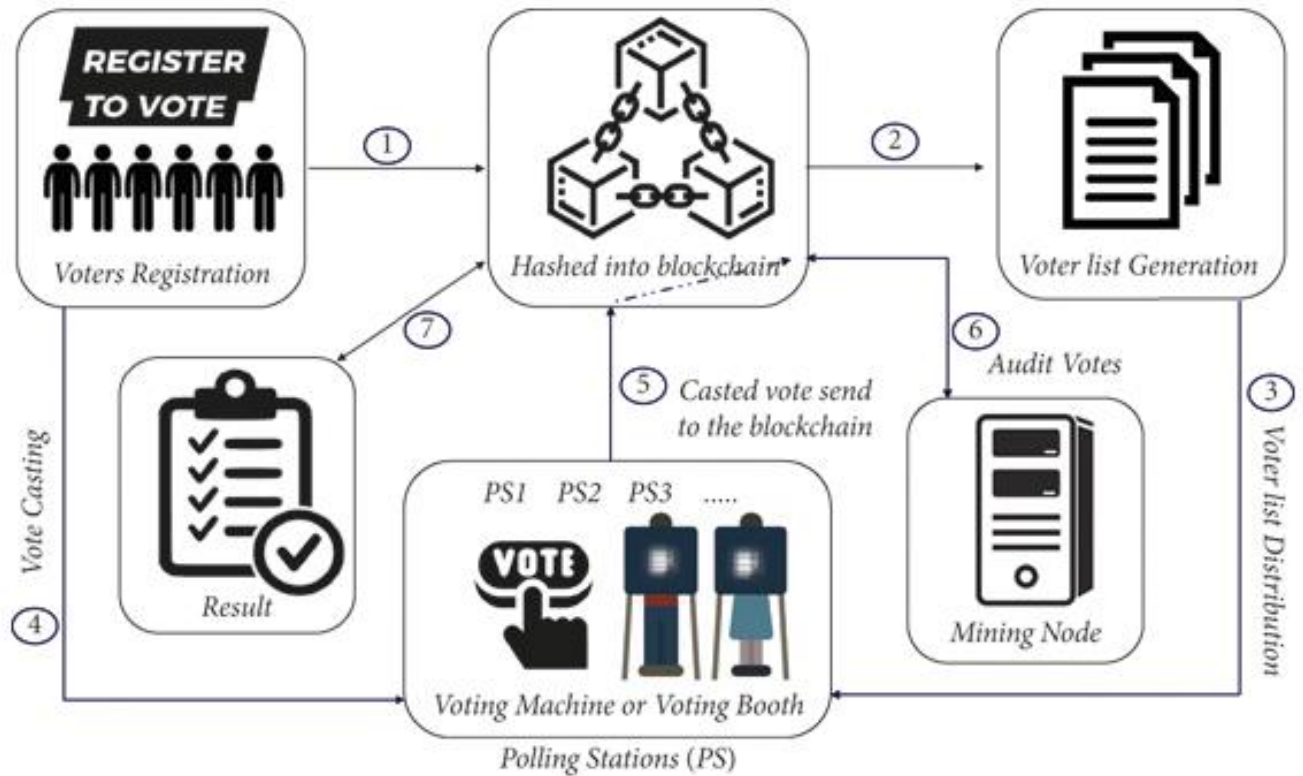
Optimize resource usage, including computational power, storage, and bandwidth, to make the system cost-effective and sustainable.

# 5. Project Design

## 5.1 Data Flow Diagram & User Stories



## 5.2 Solution Architecture



In Step 1, the voters must register by providing their identity number and other credentials; the block will be created against the voter record, and the private and public keys will be assigned to that particular voter.

In Step 2, the system adds the voter to the list based on the specific district (from which the voter hails).

Step 3, the voter's list is distributed (allocated) to the various polling stations

. In Step 4, the voters cast their vote against a certain candidate

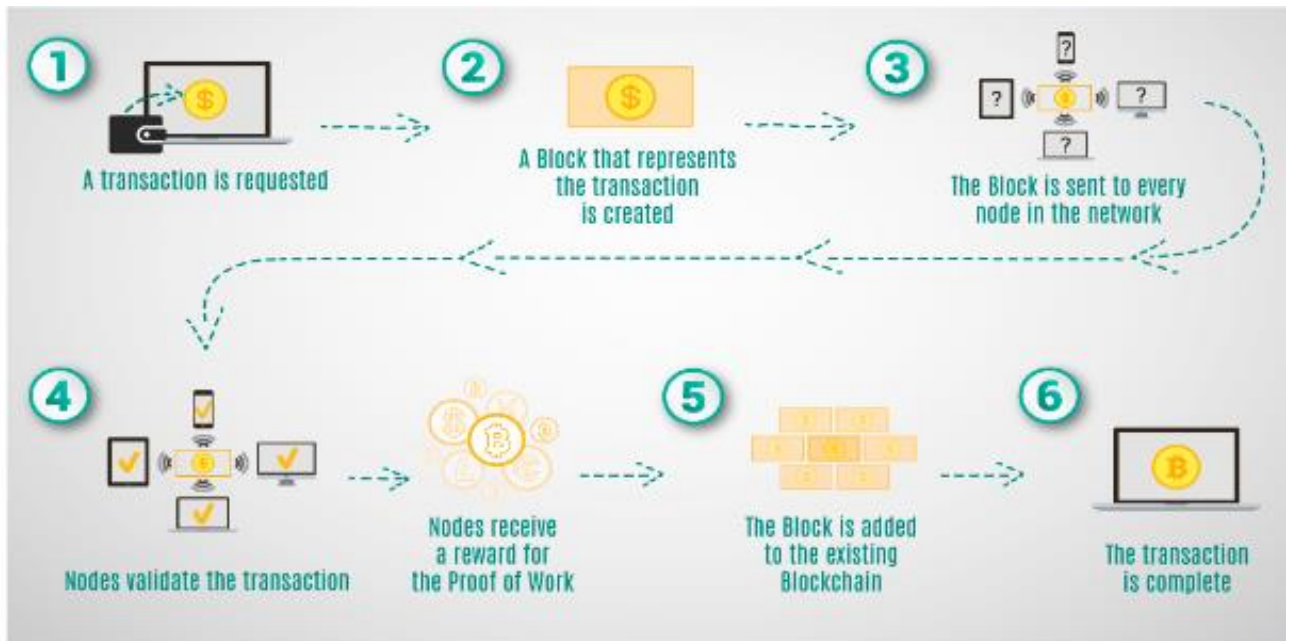
. In Step 5, polling stations update the block chain with the results while

In Step 6, the system (mining node) verifies the record for tampering and sends it back to the block chain.

In Step 7, at the end, the final result will be displayed when the election is over, as the proposed architecture discussed above is illustrated.

## 6. Project Planning & Scheduling

### 6.1 Technical Architecture



### 6.2 Sprint Planning & Estimation

#### Sprint Planning

##### 1. Product Backlog Refinement:

Before sprint planning, the product backlog should be continually refined to ensure that all tasks, features, and requirements are up-to-date, well-defined, and prioritized.

##### 2. Sprint Goals:

Define clear sprint goals and objectives. What specific features or tasks do you want to complete in this sprint? Sprint goals should align with the broader project objectives.

##### 3. Backlog Prioritization:

The product owner should prioritize items in the backlog, placing the highest-priority items at the top. These items should contribute to achieving the sprint goals.

##### 4. Task Selection:

During the sprint planning meeting, the development team selects a set of tasks from the top of the backlog that they believe can be completed within the sprint timeframe.

## **5. Estimation:**

The development team estimates the effort required for each selected task. Common estimation techniques include story points, ideal days, or t-shirt sizing.

## **6. Capacity Planning:**

Consider the team's capacity for the sprint. How many working days are available? Deduct time for meetings, administrative tasks, and potential unforeseen issues.

## **7. Task Breakdown:**

Break down selected tasks into smaller, actionable sub-tasks. This provides a more granular view of the work involved.

## **8. Dependency Analysis:**

Identify and resolve any dependencies between tasks. Dependencies can affect the order in which tasks are completed.

## **9. Definition of Done:**

Establish clear criteria for when a task or feature is considered complete. This should include testing, documentation, and quality assurance.

## **10. Sprint Backlog:**

The sprint backlog is a list of tasks selected for the sprint, along with their estimates. It's a commitment from the development team to complete these tasks within the sprint.

# **Estimation**

Estimating tasks accurately is crucial for successful sprint planning. Consider the following points when estimating tasks for a Biometric Security System for a Voting Platform in block chain technology:

## **1. Complexity:**

Assess the complexity of each task, taking into account factors like the integration of biometrics, block chain protocols, and security measures.

## **2. Historical Data:**

Utilize historical data from previous sprints or similar projects to inform your estimations. How long did similar tasks take in the past?

## **3. Expertise:**

Consider the expertise and experience of your development team. Tasks may be estimated differently based on the team's skill set.

#### **4. Risks:**

Identify potential risks and uncertainties associated with each task. Tasks with higher risks may require more conservative estimates.

#### **5. Dependencies:**

Account for task dependencies. If a task relies on the completion of another task, ensure that these dependencies are factored into the estimates.

#### **6. Buffer:**

Include a reasonable buffer for unforeseen issues or changes in requirements. This helps manage uncertainty and risk.

#### **7. Regular Refinement:**

Continually refine and update your estimates as the project progresses and more information become available.

#### **8. Consensus:**

Encourage team members to reach a consensus on estimates, as collective knowledge often leads to more accurate estimations.

#### **9. Documentation:**

Keep detailed records of estimates and actual time spent on tasks to improve future estimation accuracy.

### **6.3 Sprint Delivery Schedule**

Creating a sprint delivery schedule for a Biometric Security System for a Voting Platform in block chain technology involves breaking down the project into manageable sprints, each with specific goals and deliverables.

#### **Sprint 1 - Project Initiation:**

**Sprint Goal:** Set up the project infrastructure and establish the development environment.

##### **Tasks:**

- Define the project scope and objectives.
- Create the project plan, including sprint schedules and milestones.
- Set up version control, collaboration tools, and communication channels.
- Establish the initial block chain network and develop a prototype of the voting platform.

##### **Deliverables:**

- Project initiation documentation.

- Initial block chain network.
- Voting platform prototype.

## **Sprint 2 - Voter Registration and Biometric Data Capture:**

**Sprint Goal:** Develop the voter registration and biometric data capture features.

### **Tasks:**

- Design the voter registration process.
- Implement biometric data capture and encryption.
- Develop the backend for storing biometric data on the block chain.

### **Deliverables:**

- Voter registration system.
- Biometric data capture functionality.
- Initial block chain integration for data storage.

## **Sprint 3 - Voter Authentication and Secure Voting:**

**Sprint Goal:** Build voter authentication and secure voting features.

### **Tasks:**

- Implement voter authentication using biometric data.
- Create a secure and user-friendly voting application.
- Develop smart contracts for secure voting.

### **Deliverables:**

- Voter authentication system.
- Secure voting application.
- Smart contracts for secure voting.

## **Sprint 4 - Privacy and Compliance:**

**Sprint Goal:** Focus on privacy, data protection, and compliance.

### **Tasks:**

- Enhance data privacy features, including encryption and pseudonymization.
- Ensure compliance with data protection regulations.

### **Deliverables:**

- Enhanced data privacy measures.
- Compliance documentation.

## **Sprint 5 - Testing and Quality Assurance:**

**Sprint Goal:** Test the entire system rigorously, identify and fix issues, and ensure high-quality performance.



**Tasks:**

- Conduct extensive testing, including functional, security, and usability testing.
- Address any identified vulnerabilities and bugs.
- Optimize system performance and resilience.

**Deliverables:**

- Testing reports and documentation.
- A more robust and optimized voting platform.

**Sprint 6 - Accessibility and Usability:**

**Sprint Goal:** Improve accessibility and usability, ensuring the system accommodates users with disabilities and is user-friendly.

**Tasks:**

- Enhance accessibility features, such as voice commands and screen reader compatibility.
- Gather user feedback and make usability improvements.

**Deliverables:**

- Accessibility enhancements.
- Usability improvements.

**Sprint 7 - Final Testing and Security Audit:**

**Sprint Goal:** Conduct a final round of testing and a security audit.

**Tasks:**

- Perform additional testing, including penetration testing and vulnerability assessments.
- Engage external security experts for a comprehensive security audit.

**Deliverables:**

- Final testing reports.
- Security audit findings and recommendations.

**Sprint 8 - Deployment and Documentation:**

**Sprint Goal:** Prepare for deployment and create comprehensive documentation.

**Tasks:**

- Prepare for the deployment of the voting platform.
- Create user and administrator documentation.

**Deliverables:**

- Deployed voting platform.
- User and administrator documentation.

## **Sprint 9 - Post-Deployment Review:**

**Sprint Goal:** Evaluate the system's performance post-deployment and address any immediate issues.

### **Tasks:**

- Monitor the system's performance in a real-world environment.
- Address any post-deployment issues or concerns.

### **Deliverables:**

- Post-deployment review report.

## **7. Coding & Solution**

// SPDX-License-Identifier: MIT

pragma solidity ^0.8.0;

contract BallotBox {

// Define the owner of the contract (election authority).

address public owner;

// Define the structure of a voter.

struct Voter {

bytes32 biometricData; // Encrypted biometric data

bool hasVoted; // Indicates if the voter has cast a vote

}

// Define the structure of a candidate.

struct Candidate {

string name;

uint256 voteCount;

}

```

// Define the election parameters.

string public electionName;

uint256 public registrationDeadline;

uint256 public votingDeadline;


// Store the list of candidates.

Candidate[] public candidates;


// Store the mapping of voters.

mapping(address => Voter) public voters;


// Event to announce when a vote is cast.

event VoteCast(address indexed voter, uint256 candidateIndex);


// Modifiers for access control.

modifier onlyOwner() {

    require(msg.sender == owner, "Only the owner can call this function.");

    _;

}

modifier canVote() {

    require(block.timestamp < votingDeadline, "Voting has ended.");

    require(block.timestamp < registrationDeadline, "Registration has ended.");

    require(!voters[msg.sender].hasVoted, "You have already voted.");

    _;

}

```

```

// Constructor to initialize the contract.
constructor(
    string memory _electionName,
    uint256 _registrationDeadline,
    uint256 _votingDeadline,
    string[] memory _candidateNames
) {
    owner = msg.sender;
    electionName = _electionName;
    registrationDeadline = _registrationDeadline;
    votingDeadline = _votingDeadline;

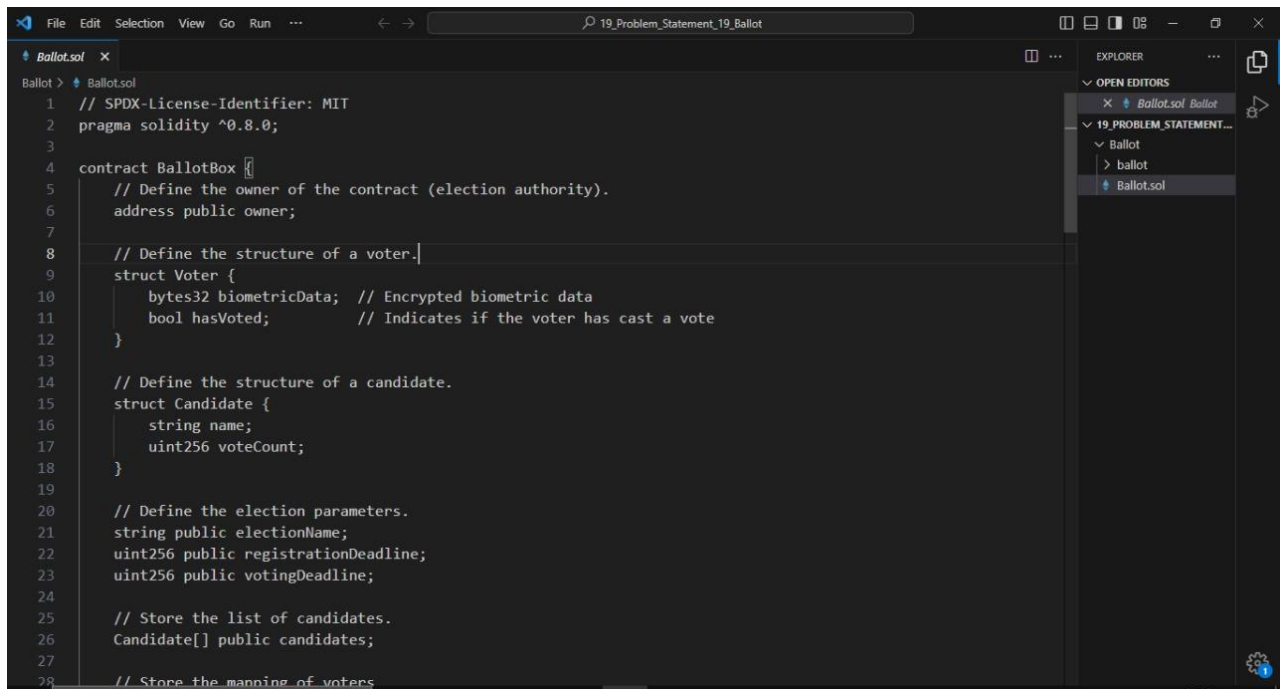
    // Initialize the list of candidates.
    for (uint256 i = 0; i < _candidateNames.length; i++) {
        candidates.push(Candidate({
            name: _candidateNames[i],
            voteCount: 0
        }));
    }
}

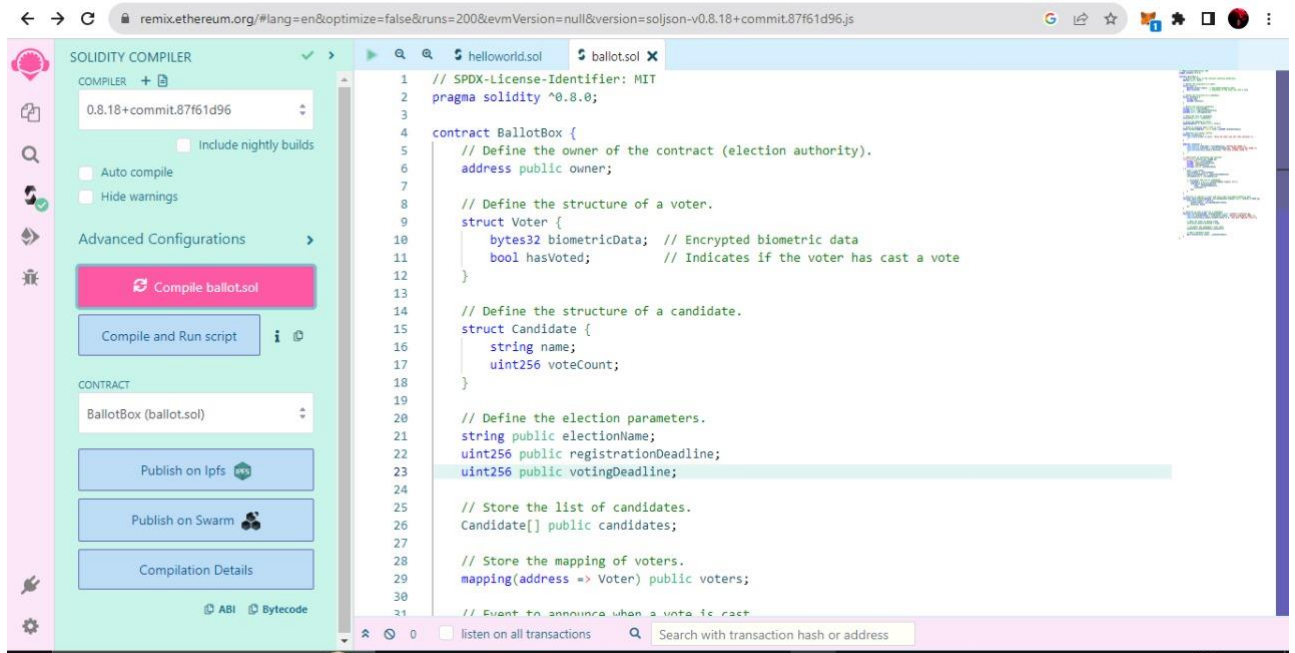
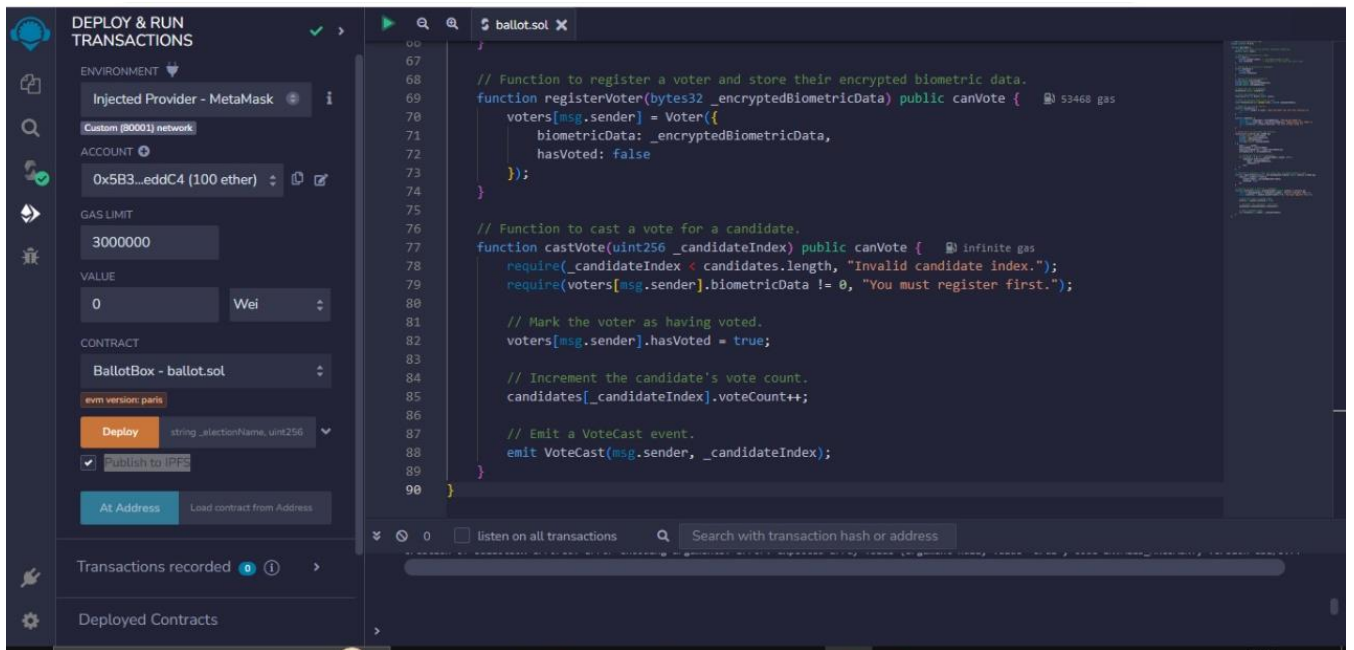
// Function to register a voter and store their encrypted biometric data.
function registerVoter(bytes32 _encryptedBiometricData) public canVote {
    voters[msg.sender] = Voter({
        biometricData: _encryptedBiometricData,
        hasVoted: false
    });
}

```

// Function to cast a vote for a candidate.

```
function castVote(uint256 _candidateIndex) public canVote {  
  
    require(_candidateIndex < candidates.length, "Invalid candidate index.");  
  
    require(voters[msg.sender].biometricData != 0, "You must register first.");  
  
    // Mark the voter as having voted.  
  
    voters[msg.sender].hasVoted = true;  
  
    // Increment the candidate's vote count.  
  
    candidates[_candidateIndex].voteCount++;  
  
    // Emit a VoteCast event.  
  
    emit VoteCast(msg.sender, _candidateIndex);  
}  
  
}
```





## 8. Performance Testing

### 1. Test Environment Setup:

- Create a dedicated testing environment that replicates the target blockchain network (e.g., Ethereum's Rinkeby or a private blockchain).
- Use development tools like Truffle or Hardhat for local testing and deployment.

### 2. Load Testing:

- Simulate a high volume of voter registrations and voting transactions to determine how the contract performs under normal load conditions.
- Measure the contract's response time, gas consumption, and transaction throughput.

### 3. Stress Testing:

- Push the system to its limits by significantly increasing the load beyond normal expectations. This helps identify potential bottlenecks and assess the system's resilience.

### 4. Concurrency Testing:

- Test how well the contract handles a large number of concurrent transactions and voter registrations.
- Assess whether the contract can scale horizontally to accommodate the expected number of voters.

### 5. Transaction Throughput:

- Measure the number of transactions the contract can process per second (TPS) while maintaining acceptable response times.

### 6. Response Time Analysis:

- Monitor and analyze the response times of key contract functions, such as voter registration and voting, under different loads.
- Ensure that responses are within acceptable time frames.

### 7. Gas Cost Evaluation:

- Analyze the gas consumption of contract functions. High gas costs can lead to transaction delays and increased costs for voters.

### 8. Resource Utilization:

- Monitor and analyze the utilization of system resources, such as CPU, memory, and storage. Identify any resource bottlenecks that could impact performance.

### 9. Scalability Testing:

- Evaluate how the contract behaves as the number of registered voters, candidates, and transactions increases.
- Determine the system's capacity to scale with growing demand, both in terms of voters and candidates.

### 10. Security and Privacy Assessment:

- Assess whether the biometric data and voting records remain secure and private during high-load scenarios. Ensure that sensitive information is protected.

### 11. Documentation and Reporting:

- Thoroughly document the results of each performance test, including bottlenecks, issues, and areas for improvement.
- Provide detailed reports to the development team.

#### 12. Optimization and Iteration:

- Based on the test results, work on optimizing the contract's code and architecture to enhance its capacity and efficiency.
- Continue testing and iterating to verify performance improvements.

#### 13. Load Balancing (if applicable):

- Consider load balancing strategies if the contract interacts with multiple nodes or external services to ensure even distribution of traffic.

#### 14. Resource Planning:

- Based on test results, develop resource planning strategies to allocate the necessary computing resources (CPU, RAM, etc.) for the production environment.

#### 15. Validation of Regulatory Compliance:

- Ensure that the system meets any regulatory and legal requirements, especially regarding voter privacy and data protection.

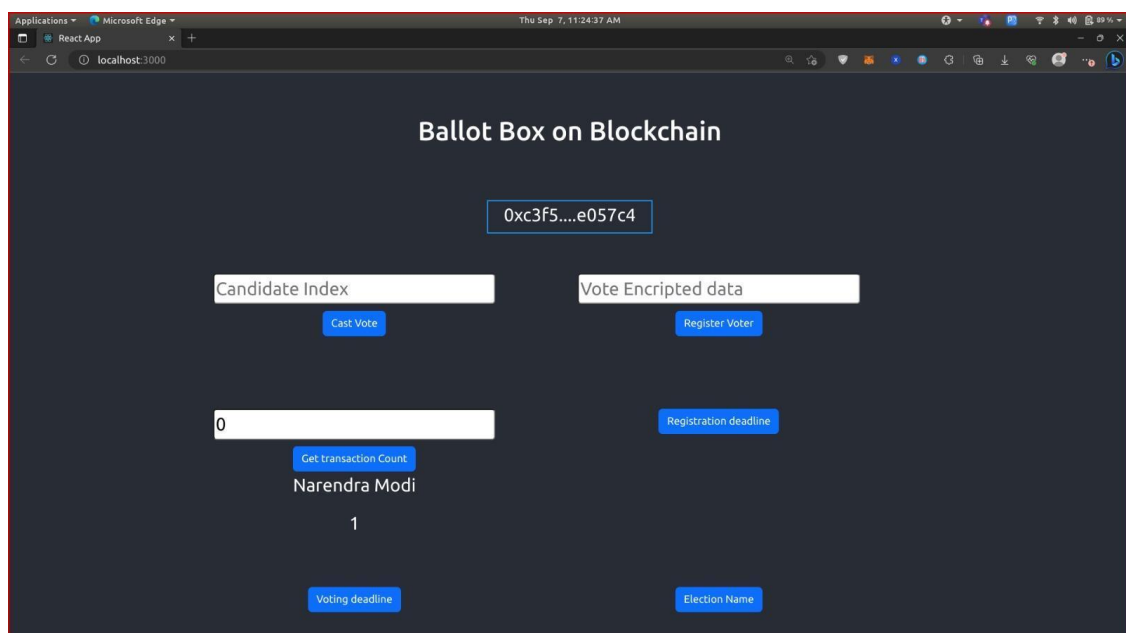
#### 16. Accessibility Testing:

- Verify that the system remains accessible to voters with disabilities, even under high load.

#### 17. Continuous Monitoring:

- Implement continuous monitoring and profiling tools to keep track of the contract's performance in the production environment and address any issues promptly.

## 9. Result





## 10. Advantages & Disadvantages

### Advantages

- Block chain technology is very secure and tamper-proof.
- All of the transactions on a block chain are publicly visible, making it easy to audit and verify the election results.
- Biometric authentication is very accurate and reliable.
- Biometric authentication is also very convenient for voters.
- Voters can simply scan their fingerprint or face to authenticate them and cast their ballots.

### Disadvantages

- Developing and implementing a biometric security system for voting platform in block chain technology would be very expensive.
- Block chain technology is complex and can be difficult to implement and manage.
- Biometric data is sensitive personal information.
- Biometric authentication is not always accurate.
- Block chain technology is still a relatively new technology, and there are some security risks associated with it.

## 11. Conclusion

A biometric security system for voting platform in block chain technology has the potential to revolutionize the way we vote. By using block chain technology to ensure the security and transparency of the election process, and by using biometric authentication to verify voter identities, a biometric security system can help to reduce voter fraud, increase voter turnout, and improve public trust in the electoral process. It is also important to consider the potential social and ethical concerns associated with the use of biometric security systems for voting. For example, some people argue that biometric authentication is discriminatory and could disproportionately impact marginalized groups. Others argue that biometric authentication is a form of surveillance and could erode individual privacy.

Overall, a biometric security system for voting platform in block chain technology has the potential to make voting more secure, reliable, and efficient. However, it is important to carefully consider the potential advantages and disadvantages of such a system before implementing it.

## 12. Future Scope

- **Improved biometric authentication:** New biometric authentication technologies are being developed all the time, such as iris scans and voice recognition. These technologies can provide even more accurate and reliable voter authentication than fingerprint scans and facial recognition.
- **More secure and scalable block chain platforms:** New block chain platforms are also being developed that are specifically designed for voting applications. These platforms can provide even greater security and scalability than existing block chain platforms.
- **Integration with other technologies:** Biometric security systems for voting platforms can be integrated with other technologies, such as artificial intelligence and machine learning, to

further improve security and efficiency. For example, AI can be used to detect and prevent fraud, and machine learning can be used to improve the accuracy of biometric authentication.

- **Remote voting:** Voters could cast their ballots from anywhere in the world using their smartphones or other devices. This would make it easier for people to vote, especially those who have difficulty traveling to the polls.
- **Electronic poll books:** Electronic poll books could be used to verify voter identities and prevent voter fraud.
- **Post-election audits:** Biometric security systems could be used to conduct post-election audits to ensure that the results are accurate.

## 13. Appendix

**Github Link:** <https://github.com/Jemibenisha/Biometric-Security-System-for-Voting-Platform.git>