

Biometric Security System For Voting Platform

Team ID : NM2023TMID00581

TEAM LEADER:

SANJAY STEPHENRAJA V (951320106035)

TEAM MEMBER 1:

ALAGARRAJA M (951320106001)

TEAM MEMBER 2:

MUTHU KRISHAN M (951320106026)

TEAM MEMBER 3:

JOHN GIFTSON P (951320106304)

TABLE OF CONTENT

S NO:	CONTENT	Pg No:
1	INTRODUCTION	<u>3</u>
2	LITERATURE SURVEY	<u>3</u>
3	IDEATION & PROPOSED SOLUTION	<u>5</u>
4	REQUIREMENT ANALYSIS	<u>8</u>
5	PROJECT DESIGN	<u>10</u>
6	PROJECT PLANNING & SCHEDULING	<u>11</u>
7	CODING & SOLUTIONING	<u>12</u>
8	PERFORMANCE TESTING	<u>15</u>
9	RESULTS	<u>16</u>
10	ADVANTAGES & DISADVANTAGES	<u>16</u>
11	CONCLUSION	<u>17</u>
12	FUTURE SCOPE	<u>18</u>
13	APPENDIX	<u>18</u>
14	GitHub & Project Demo Link	<u>20</u>

1. INTRODUCTION

1.1 Project Overview

Biometric security systems are technologies that use the unique physical or behavioral characteristics of individuals to verify their identity and grant them access to certain services or facilities. Biometric security systems can be applied to various domains, such as banking, border control, or health care. One of the most important and challenging applications of biometric security systems is voting. Voting is a fundamental right and a pillar of democracy, but it also faces many threats and challenges, such as fraud, coercion, impersonation, or exclusion. Biometric security systems can potentially enhance the integrity, efficiency, and inclusiveness of voting processes by ensuring that only eligible and registered voters can cast their ballots, and that each voter can only vote once. However, biometric security systems also pose some risks and challenges, such as data security, privacy, cost, accessibility, and public acceptance. In this paper, we will review the current state of the art of biometric security systems for voting, discuss their advantages and disadvantages, and propose some recommendations for their design and implementation.

1.2 Purpose

The purpose of a biometric security system for voting is to ensure that the voting process is fair, transparent, and accurate. A biometric security system uses the unique physical or behavioral characteristics of voters, such as fingerprints, iris, or face recognition, to identify and verify them. By doing so, a biometric security system can prevent voter fraud and tampering, such as multiple voting, impersonation, or exclusion. A biometric security system can also improve the efficiency and inclusiveness of voting by reducing the need for paper ballots, voter cards, or other documents. A biometric security system can enhance the integrity and credibility of elections by providing a unique list of voters with zero duplicate voters. However, a biometric security system also has some challenges and risks, such as data security, privacy, cost, accessibility, and public acceptance. Therefore, a biometric security system should be designed and implemented carefully, considering the legal, ethical, and social implications of using biometrics for voting.

2. LITERATURE SURVEY

2.1 Existing problem

Data security: Biometric data is sensitive and personal, and it can be vulnerable to hacking, theft, or misuse. If biometric data is compromised, it can affect the privacy and identity of the voter and also the integrity and credibility of the election.

Cost: Biometric security systems require expensive hardware and software, such as scanners, sensors, databases, and networks. They also need regular maintenance and updates. The cost of implementing and operating biometric security systems can be a burden for developing countries or region with resources.

Accessibility: Biometric security systems may not be accessible or convenient for all voters, especially those who have disabilities, injuries, or illnesses that affect their biometric features. For example, some voters may have poor eyesight, missing fingers, or skin diseases that prevent them from using iris, fingerprint, or face recognition systems. Biometric security

systems may also face technical issues such as low quality images, poor lighting or network failures.

Public acceptance: Biometric security systems may face resistance or distrust from the public, who may have concerns about their privacy, accuracy, or reliability. Some voters may not want to share their biometric data with the government or other entities, fearing that it may be used for surveillance or discrimination. Some voters may also doubt the effectiveness or fairness of biometric security systems and prefer to use traditional methods of voting .

2.2 References

Diponkar Paul and Suboj Kumar Ray, Member IACSIT, Vol. 3, No. 2, March 2013, “A preview n microcontroller Based electronic Voting machine”, International journal Of Information and Electronics Engineering.

D. Balzarotti, G. Banks, M. Cova, V. Felmetser, R. A. Kemmerer, W. Robertson, F. Valeur, and G. Vigna, vol.36, No. 4, 2010. “An Experience in Testing the Security of Real-World Electronic Voting Systems”, IEEE Transactions on Software Engineering.

A. Villafiorita and K. Weldemariam, and R. Tiella, vol. 4, No. 4, 2009. “Development Formal Verification and Evaluation of an E-Voting System with VVPAT”, IEEE Transactions on Information Forensics and Security. <http://www.bravenewballot.org/e-voting-in-india.html>.

Anil K. Jain, Arun Ross and Salil Prabhakar, Vol. 14, No.1, January 2004. “An Introduction to Biometric Recognition”, IEEE Transactions on Circuits and Systems For Video Technology, Special Issue on Image and Video Based Biometrics.

Anil K. Jain and Umut Uludag, Vol. 25, No. 11, pp.1094- 1098, Nov 2003. “Hiding Biometric Data”, IEEE Transactions on Pattern Analysis and Machine Intelligence. <http://uidai.gov.in/aadhaar.html>

S. Prabhakar, S. Pankanti, and A. K. Jain Vol. 1, No. 2, pp.33 -42, 2003 “Biometric Recognition: Security and Privacy Concerns”, IEEE Security and Privacy Magazine.

J. L. Wayman, Vol.1, No. 1, pp. 93-113, 2001, “Fundamentals of Biometric Authentication Technologies” International journal of Image and Graphics.

L. Hong, A. K. Jain, and S. Pankanti, ProcAutoID'99s, Pp.59-64, Oct 1999 “Can Multi Biometrics Improve Performance”? Summit (NJ), USA.

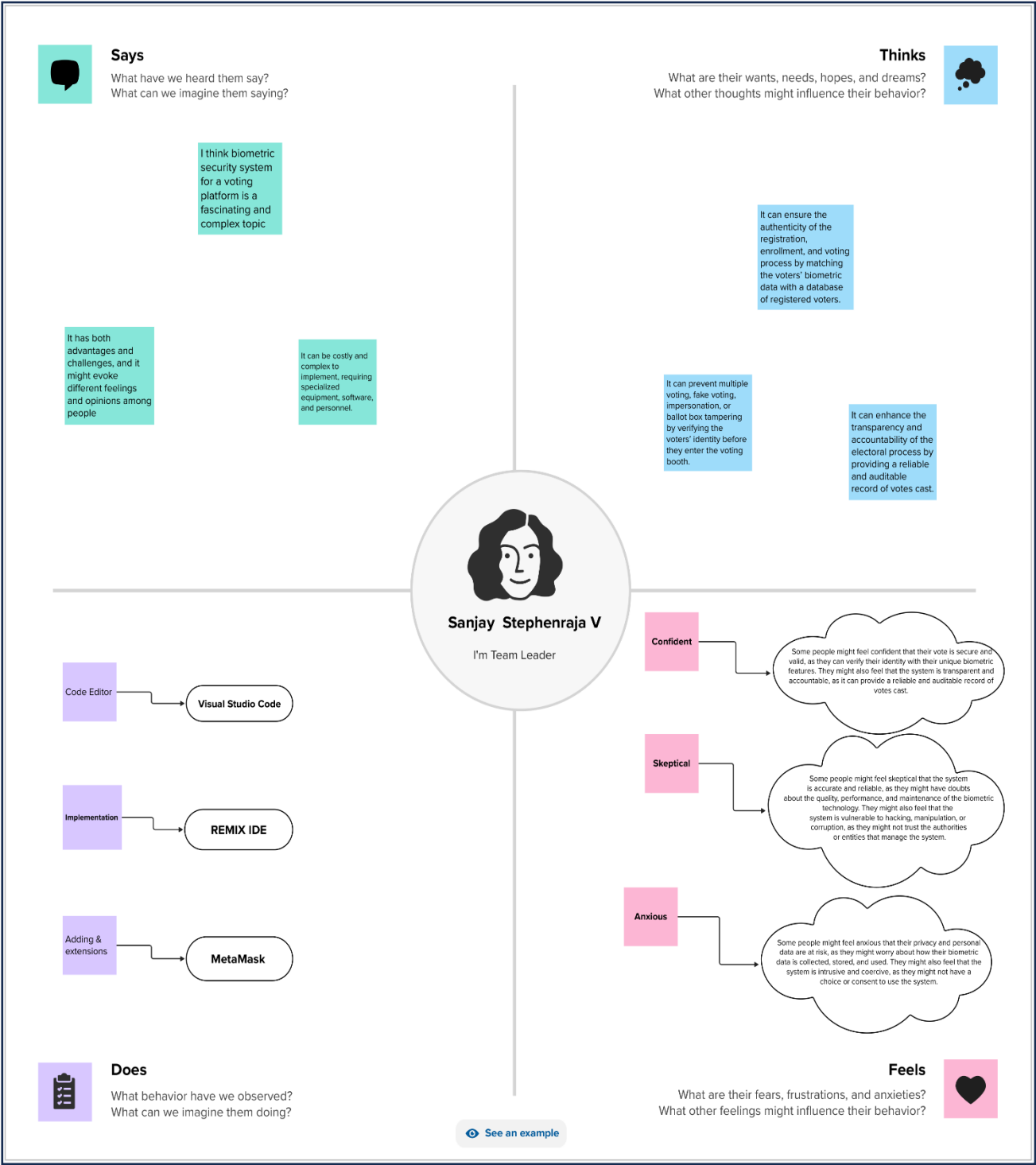
2.3 Problem Statement Definition

The current voting systems in many countries are vulnerable to various forms of fraud, tampering, and manipulation, which undermine the integrity and credibility of the elections. Moreover, the current voting systems are inefficient, costly, and inaccessible for many voters, especially those who live in remote areas, have disabilities, or lack proper identification documents. Therefore, there is a need for a more secure, efficient, and inclusive voting system that can use biometric technology to identify and verify the voters. The proposed system will use fingerprint recognition as the main biometric feature to authenticate the voters and allow them to cast their

ballots electronically. The proposed system will also use cloud-based databases to store and manage the voter information and the election results

3. IDEATION & PROPOSED SOLUTION


3.1 Empathy Map Canvas



3.2 Ideation & Brainstorming

Step-1: Team Gathering, Collaboration and Select the Problem Statement

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

Team gathering

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

Set the goal

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

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

How might we (biometric security system for a voting platform)?

Key rules of brainstorming

To run a smooth and productive session

Stay in topic.


Defer judgment.

Go for volume.

Encourage wild ideas.

Listen to others.

If possible, be visual.



Need some inspiration?

Here are featured projects of this template for brainstorming your next idea.

[Open examples](#)

Step-2: Brainstorm, Idea Listing and Grouping

2 Brainstorm

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

⌚ 10 minutes

Alagarraja

Download VS Code

Extract the project file and open in VS Code

Download the Zip file for the project

In this phase, voter authenticates himself at himself by showing his or her voting card. (We step is public, and verified by the presiding officer)

The voter takes place in a protected booth where voter cannot be seen by any person.

Muthu Krishan

At the end of authentication process, presiding officer give a ballot paper to voter to cast his or her vote

John Giftson

The voter cast their vote by writing it with a pen on the paper ballot, folds the ballot paper and put into the ballot box where all the votes are mixed

At the end of voting time, the presiding officer collect the ballot box containing all ballot papers and submit it to the counting centre

Sanjay Stephenraja

Various types of verification process are used, most procedure are public and verified by the representative of candidates of competing parties. Recount is also possible if there is any fraud or error

These systems are not efficient as they are conducted manually and therefore very often are not accurate. As a consequence, it is obligatory to carry the available voting through an electronic system

Conventional voting systems are not efficient due to long period of preparation, bogus voting, include papers, punch cards, mechanical levers, optical scan machines

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

Alagarraja

Download VS Code

Extract the project file and open in VS Code

Download the Zip file for the project

In this phase, voter authenticates himself or herself by showing his or her voting card. this step is public and verified by the presiding officer

Muthu Krishan

At the end of authentication process, presiding officer give a ballot paper to voter to cast his or her vote

John Giftson

The voter cast their vote by writing it with a pen on the paper ballot, folds the ballot paper and put into the ballot box where all the votes are mixed

At the end of voting time, the presiding officer collect the ballot box containing all ballot papers and submit it to the counting centre

Sanjay Stephenraja

Various types of verification process are used, most procedure are public and verified by the representative of candidates of competing parties. Recount is also possible if there is any fraud or error

These systems are not efficient as they are conducted manually and therefore very often are not accurate. As a consequence, it is obligatory to carry the available voting through an electronic system

Conventional voting systems are not efficient due to long period of preparation, bogus voting, include papers, punch cards, mechanical levers, optical scan machines

6

Step-3: Idea Prioritization

4

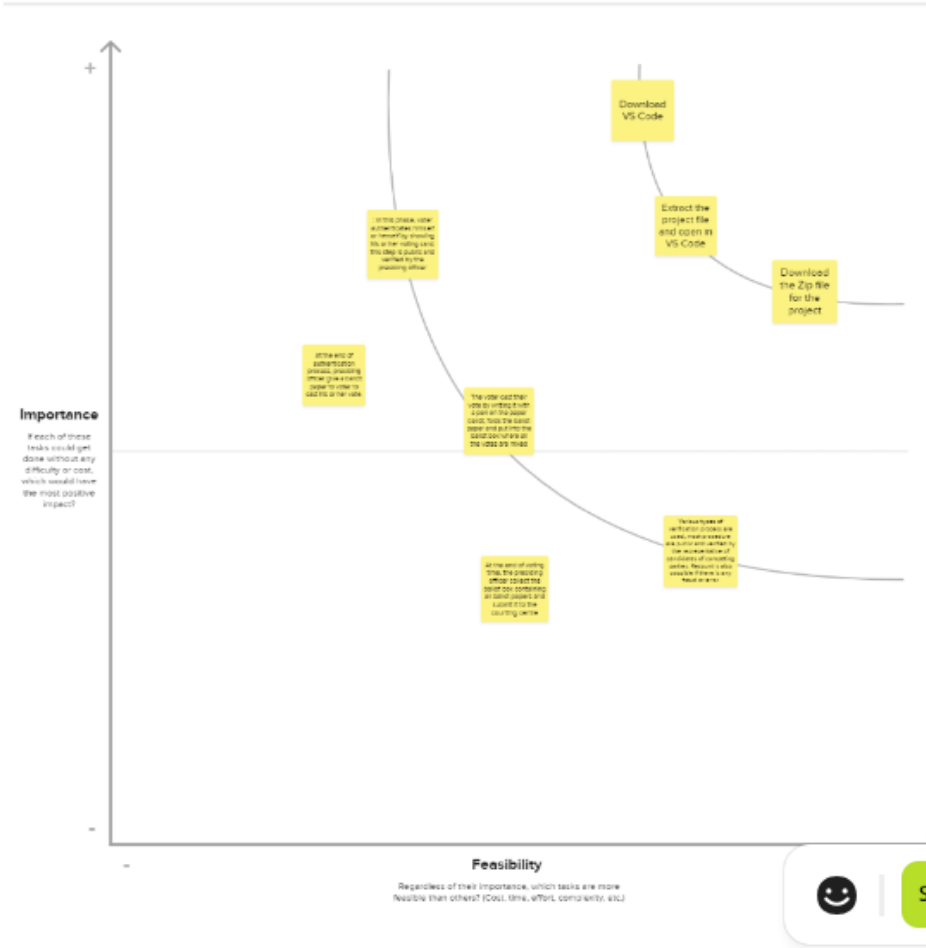
Prioritize

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 lower pointer holding the H key on the keyboard.



5

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 requirement

1.Vs Code

Visual Studio Code (VS Code) is a versatile and popular code editor that can be used for developing Biometric Security System For Voting Platform. Here are some steps and tips for using VS Code in such a project:

Installation:

Download and install Visual Studio Code from the official website (<https://code.visualstudio.com/>).

Extensions:

Install relevant extensions for blockchain and smart contract development. Some commonly used extensions for Ethereum and Solidity development include "Solidity" and "Truffle."

Smart Contract Development:

- ❖ Create a new folder for your project and open it in VS Code.
- ❖ Write, test, and deploy your smart contracts using the Solidity extension.
- ❖ Use Truffle, a development framework for Ethereum, to help you with smart contract development.

Front-End Development:

Develop the front-end of your Biometric Security System For Voting Platform using web development technologies like HTML, CSS, and JavaScript. Ensure that your front-end code interacts with your smart contracts on the blockchain.

Testing:

Write tests for your smart contracts and front-end code. VS Code can be used to run and manage test suites.

Deployment:

Deploy your smart contracts and front-end applications to the desired blockchain network and hosting platforms.

2. Nodejs

Node.js is a popular runtime environment for JavaScript, and it can be used in the development of a Biometric Security System For Voting Platform, particularly for building the backend server, handling API requests, and interacting with the blockchain. Here's how Node.js can be integrated into the development process:

Backend Development:

Node.js is commonly used to build the backend server of web applications. You can create an Express.js server to handle API requests from the front end and interact with the blockchain network (e.g., Ethereum).

Smart Contract Deployment:

You can use Node.js scripts to automate the deployment of smart contracts to the blockchain network during the development and testing phases.

Authentication and Authorization:

Implement user authentication and authorization using Node.js middleware, such as Passport.js, to secure **your application and control access to certain features**.

Real-Time Features:

For real-time features like property bidding, chat, or notifications, you can use Node.js with WebSocket libraries like Socket.io to provide instant updates to users.

4.2 Non-Functional requirements

1. Metamask

MetaMask is a popular cryptocurrency wallet and decentralized application (dApp) browser extension that allows users to interact with the Biometric Security System For Voting Platform blockchain. It serves as both a cryptocurrency wallet and a gateway to the world of decentralized applications.

MetaMask has gained popularity for its role in enabling users to access and participate in the decentralized finance (DeFi) ecosystem, interact with non-fungible tokens (NFTs), and securely manage their Ethereum-based assets

Integrating MetaMask into a Biometric Security System For Voting Platform can offer several benefits, especially when dealing with Ethereum-based tokens, smart contracts, and decentralized applications (dApps).

User Wallets: Each user, including property owners, buyers, and investors, can have their own MetaMask wallet. This wallet allows them to securely store and manage property tokens, Ether (ETH), or any other cryptocurrencies used within the system.

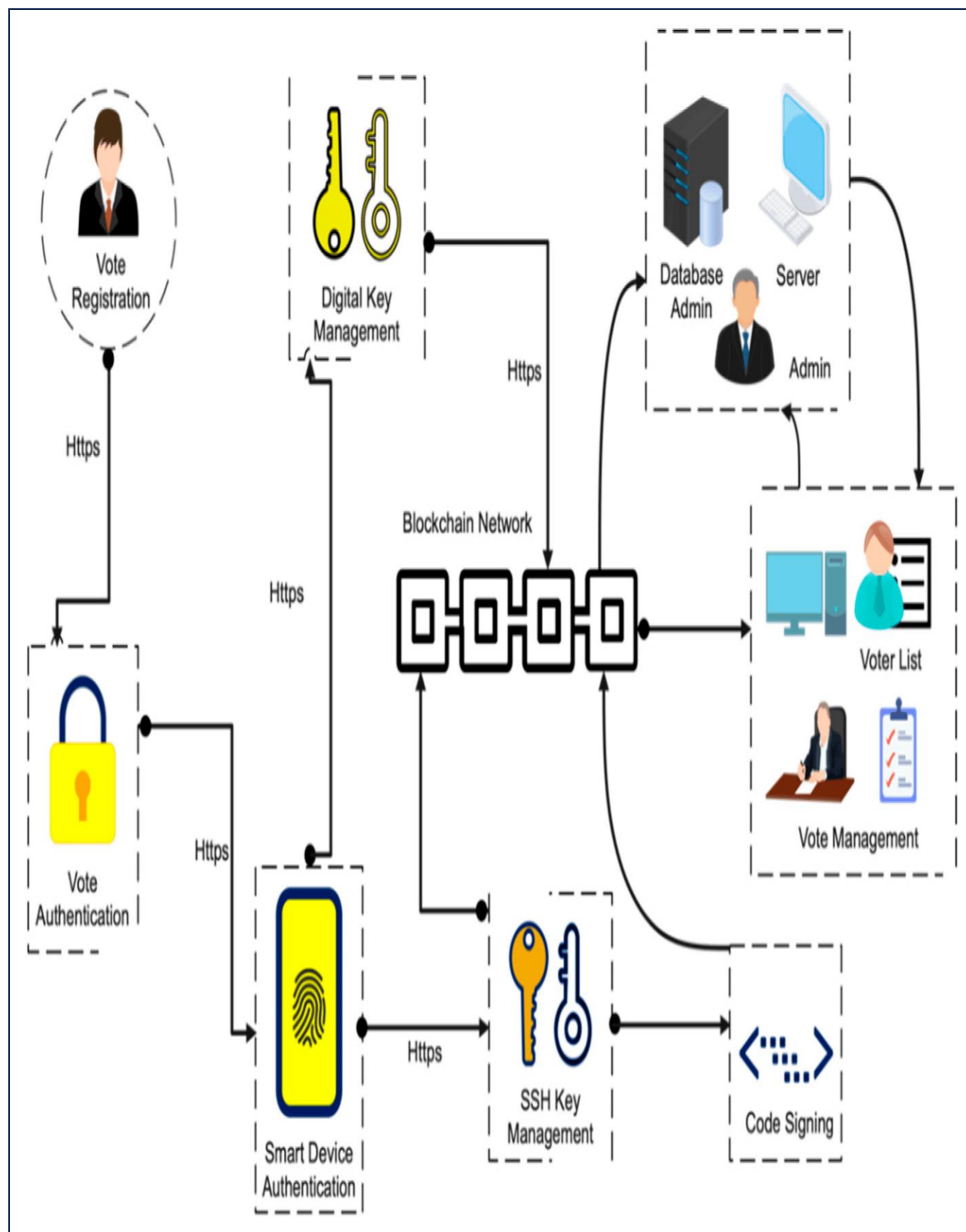
Access to dApps: MetaMask serves as a bridge to decentralized applications. Users can access the real estate management system's dApp through MetaMask, enabling them to initiate property transactions, view property details, and perform various management tasks.

Transaction Execution: Users can use MetaMask to interact with the smart contracts that govern property transactions. For example, they can initiate property purchases, sales, and transfers directly through their MetaMask wallets.

Token Management: Property tokens, which represent fractional ownership, can be managed within MetaMask. Users can view their token holdings, transfer tokens to other users, and participate in property investment activities.

5. PROJECT DESIGN

5.1 Data Flow Diagrams & User & Stories Solution Architecture



6. PROJECT PLANNING & SCHEDULING

6.1 Technical Architecture

The technical architecture of a Biometric Security System For Voting Platform is crucial for its functionality, security, and efficiency. The technical architecture of the Biometric Security System For Voting Platform is a complex, interconnected system that leverages blockchain technology, smart contracts, user interfaces, and security measures to provide a transparent, efficient, and secure real estate management solution. Careful design and continuous development and maintenance are essential to ensure its success and effectiveness.

6.2 Sprint Planning & Estimation

Sprint planning and estimation are crucial components of an agile development process for a Biometric Security System For Voting Platform. They help the development team organize work, set priorities, and estimate the effort required for each task within a sprint. Here's how sprint planning and estimation can be approached:

1. Product Backlog:

Begin with a well-defined product backlog. This backlog includes all the features, user stories, and tasks that need to be implemented in the system. It should be maintained and prioritized by the product owner.

2. Sprint Goal:

Determine the sprint goal for the upcoming sprint. This goal should be aligned with the overall project objectives and should specify what the team intends to achieve in the sprint.

3. Sprint Length:

Decide on the duration of the sprint. Common sprint lengths are 2 weeks, 3 weeks, or 4 weeks. Choose a sprint length that suits the team's work capacity and project complexity.

4. Sprint Planning Meeting:

Conduct a sprint planning meeting before the start of each sprint. This meeting involves the product owner, development team, and the scrum master. During the meeting, the team reviews the prioritized backlog items and selects the ones to be worked on in the upcoming sprint.

Sprint planning and estimation are iterative processes, and the team should continuously improve their estimation accuracy and sprint planning based on past performance and feedback. Effective sprint planning and estimation ensure that the Biometric Security System For Voting Platform is developed efficiently and with a focus on delivering value to users.

6.3 Sprint Delivery Schedule

The sprint delivery schedule for a Biometric Security System For Voting Platform depends on the sprint length and the scope of work for each sprint. Typically, agile development teams follow a sprint schedule with fixed sprint durations (e.g., 2 weeks, 3 weeks, or 4 weeks). Here's an example of a sprint delivery schedule for a 2-week sprint:

Sprint 1:

Sprint Duration: Week 1 and Week 2

Sprint Planning: Day 1 of Week 1

Daily Standup Meetings: Held daily throughout the sprint

Mid-Sprint Review: Day 5 of Week 1

Sprint Review and Demo: Day 5 of Week 2

Sprint Retrospective: Day 5 of Week 2

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

7.1 Feature 1

```
// 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.");
    _;
}

```

7.2 Feature 2

```

// 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

8.1 Performance Metrics

When evaluating the performance of a Biometric Security System For Voting Platform, it's important to consider various metrics to assess its effectiveness and efficiency. These metrics can help gauge the system's impact, user experience, and overall success. **Transaction Throughput:** Measure the number of real estate transactions processed per second or per minute. Higher throughput indicates better system performance.

Transaction Confirmation Time: Calculate the time it takes for a real estate transaction to be confirmed on the blockchain. Lower confirmation times are favorable, as they reduce delays in property transactions.

Scalability: Assess how well the system can handle an increasing number of users and transactions. Scalability metrics should indicate that the system can grow to accommodate a larger user base without a significant drop in performance.

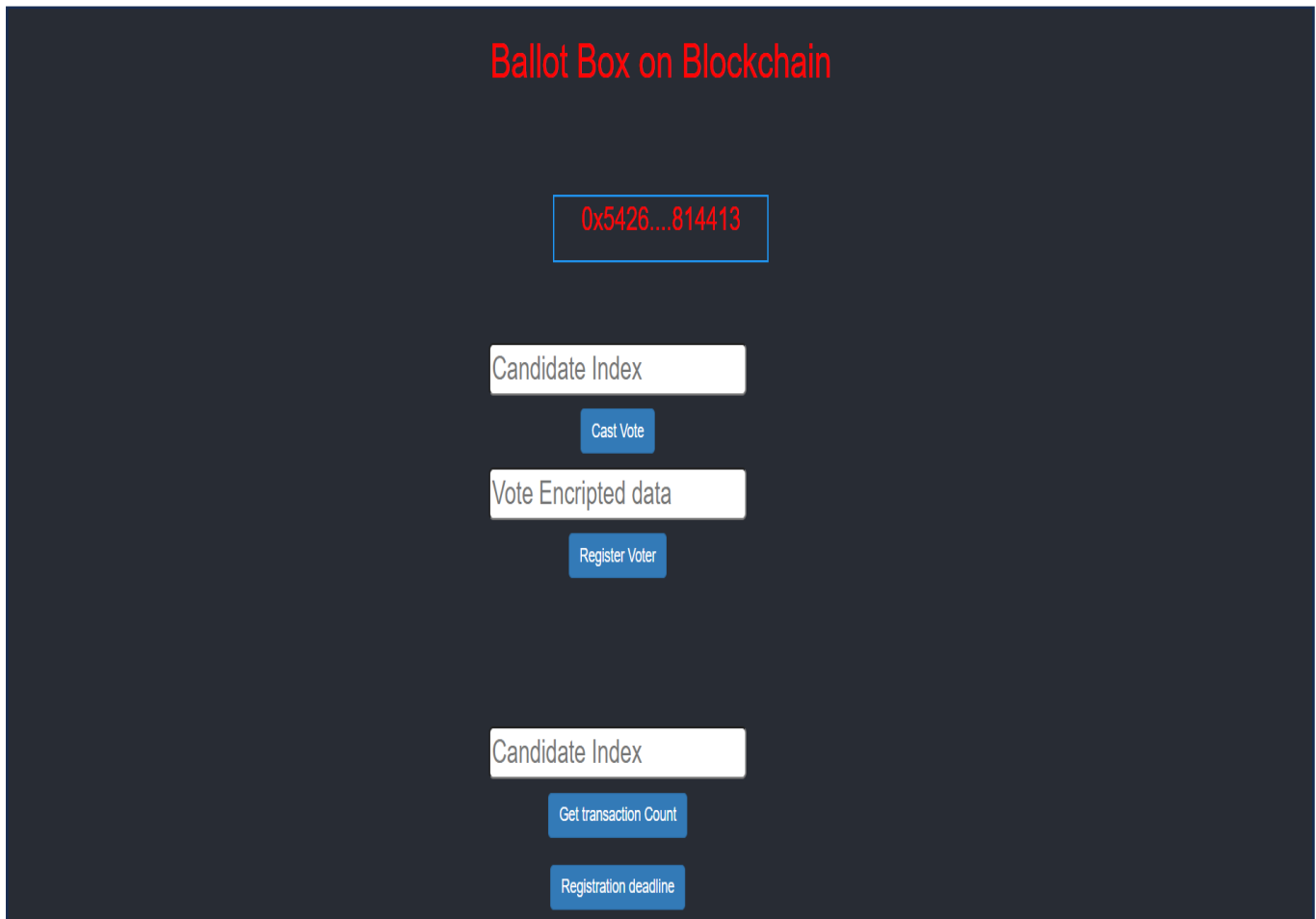
Cost Reduction: Evaluate the cost savings achieved through the elimination of intermediaries and streamlining of processes. Compare the cost of using the blockchain system to traditional real estate transaction costs.

Smart Contract Efficiency: Evaluate the efficiency of smart contracts in automating real estate transactions. Measure the number of successful, error-free smart contract executions.

Property Record Accuracy: Assess the accuracy of property records stored on the blockchain. Measure the number of disputes or discrepancies related to property ownership or history.

9. RESULTS

9.1 Output Screenshots



10. ADVANTAGES & DISADVANTAGES

Advantages:

AI biometric security system for voting is a system that uses the unique physical characteristics of voters, such as fingerprints, facial features, or iris patterns, to register and verify them before they cast their votes. This system aims to prevent fraud, identity theft, and multiple voting, and to ensure fair and transparent elections.

Some of the advantages of AI biometric security system for voting are:

Transparency: The biometric voting process is transparent from start to end, and the design of the system ensures that no one can tamper with the data or manipulate the results.

Participation: The biometric voting system can increase voter participation by making the registration and verification process easier and faster, and by reducing the barriers for people who do not have other forms of identification.

Scalability: The biometric voting system can handle large-scale elections with millions of voters, as it can quickly and accurately match the biometric data against a central database.

Fairness: The biometric voting system can ensure that every voter has an equal right to vote, and that every vote is counted fairly. The system can eliminate duplicate voters, fake voters, and other forms of electoral malpractice.

Security: The biometric voting system can protect the privacy and security of the voters and their data, as it uses encryption, authentication, and verification techniques to prevent unauthorized access or leakage

Disadvantages:

Privacy: Some people may feel uncomfortable or violated by having their biometric data collected and stored by the authorities. They may worry about the misuse or leakage of their personal information, or the potential violation of their human rights.

Errors: Biometric machines are not perfect and can make mistakes. Sometimes, they may accept an unauthorized person (False Acceptance Rate) or reject an authorized person (False Rejection Rate). These errors can affect the accuracy and fairness of the voting process.

Cost: Biometric voting systems can be expensive to implement and maintain. They require sophisticated hardware, software, and infrastructure, as well as trained staff and security measures. The cost may outweigh the benefits for some countries or regions.

Inclusivity: Biometric voting systems may not be inclusive for everyone. Some people may have physical or mental disabilities that prevent them from using biometric devices, such as missing fingers, eye problems, or cognitive impairments. Some people may also have religious or cultural objections to biometric identification

11. CONCLUSION

In conclusion, a Biometric Security System For Voting Platform holds great promise for revolutionizing the real estate industry by leveraging blockchain technology and smart contracts. This innovative approach offers transparency, efficiency, and security in property transactions and management.

In spite of these challenges, the advantages of a Biometric Security System For Voting Platform are significant. It offers transparency, trust, efficiency, and reduced costs. Property tokenization allows fractional ownership and liquidity, and smart contracts automate transactions, reducing paperwork and reliance on intermediaries.

To successfully implement a Biometric Security System For Voting Platform, collaboration between the technology and real estate sectors is essential. It's also crucial to work closely with legal experts to navigate regulatory complexities.

While the path to widespread adoption may be challenging, the potential for positive disruption and transformation in the real estate industry is undeniable. As blockchain technology matures and the industry becomes more accustomed to its benefits, it's likely that we will see more innovative and efficient approaches to real estate management and transactions.

12. FUTURE SCOPE

It can enhance the accuracy and reliability of the voting process by eliminating duplicate or fake votes.

It can reduce the cost and time of the voting process by eliminating the need for physical documents, ballot papers, or machines.

It can improve the convenience and accessibility of the voting process by allowing voters to vote from anywhere using their smartphones or other devices.

It can increase the voter turnout and participation by making the voting process more user-friendly and secure.

The privacy and security of the biometric data of the voters need to be ensured by using encryption, anonymization, or other techniques

The interoperability and compatibility of the biometric devices and systems need to be ensured by using common standards and protocols.

The legal and ethical issues related to the use of biometric data for voting need to be resolved by establishing clear policies and regulations.

The social and cultural acceptance of biometric security system for voting need to be increased by raising awareness and education among the voters.

13. APPENDIX

Source Code

// 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);  
}  
}
```

14.GitHub & Project Demo Link

Github link : <https://github.com/Aathianbu/NM2023TMID03676>

Project Demo link : <https://youtu.be/n2Uv1EpYjFQ>