Smart Election Voting System Using Blockchain

Submitted in partial fulfilment of the requirement of

University of Mumbai for the degree of

Bachelor of Engineering (Information Technology)

By

Student Name: Sahil Ahire **Roll No. A-64**

Student Name: Abhishek Badole **Roll No. A-48**

Student Name: Prathamesh Yadav **Roll No. A-51**

Student Name: Abhishek Lakhane **Roll No. A-65**

Under the Guidance of

Prof. Smita Pai

Department Of Information Technology

Terna Engineering College

University Of Mumbai

(2022-2023)

CERTIFICATE

This is to certify that the project entitled "**Smart Election Voting System Using Blockchain**" is a Bonafide work of **"Sahil Ahire (A-64)"**, **"Abhishek Badole (A-48)"**, **"Prathamesh Yadav (A-51)"**, **"Abhishek Lakhane (A-65)"**, submitted to the University of Mumbai in partial fulfillment of the requirement for the award of the degree of Bachelor of Engineering in Information Technology, Terna Engineering College, Nerul, Navi Mumbai.

Project Guide	**Project Coordinator**
(Prof. Smita Pai)	**(Prof. Smita Pai)**

Project Convener	**Head of Department**	**Principal**
(Dr. Vijayalakshmi Kadroli)	**(Dr. Vaishali Khairnar)**	**(Dr. L. K. Ragha)**

Project Report Approval for B.E. (Semester VII)

The Project report entitled **"Smart Election Voting System Using Blockchain"** by **"Sahil Ahire" (A-64), "Abhishek Badole"(A-48), "Prathamesh Yadav"(A-51), "Abhishek Lakhane"(A-65)** is approved for the degree of ***Bachelor of Engineering*** in Information Technology.

Examiners

1. _____

2. _____

Date: _____

Place: _____

Declaration

We declare that following written submission is in our own words and where we have taken the others idea/fact/submission we have mentioned that. We understand that any violation or misuse of the above will cause disciplinary action by the institute.

Signature of Student:

Sahil Ahire
[TU4F1920066]

Abhishek Badole
[TU4F1920051]

Prathamesh Yadav
[TU4F1920055]

Abhishek Lakhane
[TU4F1819068]

Contents of Report

List of Figures

Sr No. Name of Figure Page No.

List of Tables

Chapter 1

Introduction

1.1 Scope

In the digital era where hacking and bypassing a system is very easy, tampering of data is can always leads towards the bad situations. Blockchain is used to store data which is near impossible to change or tamper with as it is very secure in nature. Voting is an very important process in any respective field like political election, Local community leader, college president, school president any many more which are the essential event and if votes get miscalculated by any external source it will be harmful.

1.2 Motivation

Our project proposes a decentralized Smart election voting system using blockchain technology. It includes an admin panel to schedule the voting, manage candidates and declare the results. The web application will provide the users with an interface to enter their Name, Mobile No. and a Live photo of themselves at the time of voting. The eligibility of the voter will be checked at the time they enter their User Id. Eligible voter's phone numbers will be verified via One Time Password (OTP). After voter verification, individual voters will be considered eligible for voting. During voting, voters will be monitored through a webcam/front camera. The votes will be stored in a blockchain and any tampering would be detected easily.

1.3 Objective

- Voters will be able to vote securely.
- The registration will be done by user entering their data and by giving their facial authentication.
- Each voter will get login with unique id to give vote.
- After voting is done, the results will be declared on the same day in the given period of time as set by admin.

Chapter 2

Literature Survey

2.1 Existing System:

There are lot of practices are made to introduce the variations in electronic and online voting systems where different techniques and methodologies are used. Some of them guarantees the confidentiality and security to the system at some extent, still the voting information and process need to be control and manage with advanced systems that will ensures and guarantees the security and privacy of voter's and voter's information.

Basic E-voting approach /architecture

The systems that are developed to caste the vote by means of digital approach using online portals and electronic devices use various encryption and decryption techniques to guarantee the secure data transaction.

Homomorphic Encryption Technique:

Homomorphic encryption is a well-known powerful technique with many useful applications. Recently, it has been applied to the design of online voting system . The voting system based on this encryption uses the exponential ElGamal cryptosystem. Before submission, the contents of each cast ballot are encrypted using the exponential ElGamal encryption. The additive homomorphism property of this crypto system makes it possible to tally encrypted ballots directly without decrypting them.

Centralized architecture:

However, numbers of techniques are present to convert the data in coded format to prevent from manipulation while transferring to the network. One drawback can be discussed here that after the correct data have been stored in the database trust and security is required at substantial level. Centralized storage is inconvenient if the data is esteemed because unauthorized access and attack by hackers will challenge the system in terms of reliability.

2.2 Summary of Literature Survey

Table 2.1 Summary of literature survey

Publishing Year	Title of the Paper	Highlighted Proposed System	Features	Limitation
14th November 2018	Decentralized Voting Platform Based on Ethereum Blockchain [IEEE]	1. Web application 2. Event Management Server 3. Smart Contracts, registration contract and voting contract	SMS Gateway, Mobile Application, Ethereum Virtual Machine and decentralized application	Third Party person can interfere with the data. Strong Data integrity but low availability.
2nd July 2018	Blockchain-Based E-Voting System [IEEE]	1. Verifiability: The ledger is decentralized, replicated and distributed over multiple locations 2. District Node and Bootnode 3. Geth	Go Ethereum: Proof of authority (POA), Tallying Results is done on smart contracts, District Node runs on static IP, which peers faster.	Go Ethereum: POA is used which is optionally decentralized. The required secured authentication is not guaranteed by default.
1st September 2020	E-Voting Systems using Blockchain	Smart contract admin, quorum framework, Exonium, Trusted Third Party (TTP), Multichain framework	Hyperledger sawtooth, RSA Encryption, Distributed Ladger, three tier architecture (National, constituency and local)	Exonium is a paid service provider and it is costly. Due to multichain framework and third party inference, it creates a public address and store it against the voters.

May 2017	**A Conceptual Secure Blockchain-Based Electronic voting system**	**1. Authentication 2. Anonymity 3. Accuracy 4. Verifiability**	**The voting system will have a node in each district to ensure the system is decentralized.**	**System's inability to reverse a vote in the event of a user error is one of its shortcomings. The user will only be allowed to vote once.**
December 2017	**The future of E-voting**	**They present technology used in the voting system is a payment scheme, which offers anonymity of transactions, a trait not seen in blockchain protocols to date.**	**Basic Function of the SHA-256 Hash, SMS gateway**	**The verification of the security protocol is poor.**
November 2018	**General explanation of blockchain based voting systems**	**Generalized e-voting system using blockchain is proposed with SHA encryption of voter information. The vote block is added to the selected candidate's blockchain**	**Vote Privacy Robustness and Integrity Fingerprint authentication, SMS Gateway**	**A different chain for each candidate introduces greater overhead. The system does not discuss implementation using any specific framework.**
May 2019	**Securing e-voting based on blockchain in P2P network**	**The blockchain-based e-voting scheme is public, distributed, and decentralized. It can record votes from voters across many mobile devices and computers. The blockchain-based e-voting scheme allows the voters to audit and verify the votes inexpensively**	**It is designed synchronized model based on DLT. There is a design made for user credential model based on ECC.**	**No security authentication while registration. The disadvantages to distributed ledger technology are inefficiency, the inherent risk of non-reversible transactions, and the possibility of a 51% attack. For distributed ledgers, and blockchains in**

				particular, inefficiency can be an issue.
January 6, 2017	**Using blockchain for enabling internet voting**	**Using blockchains to provide secure and reliable internet voting protocols.**	**Given the Estonian internet voting system as a basis, we look at some aspects of the system and propose uses for the blockchain technology.**	**Coercion-resistance, Estonian system allows for revoting.**
July 2018	**Election as a smart contract**	**In they propose a potential new e-voting protocol that utilizes the blockchain as a transparent ballot box.**	**Agora, District node which manages the smart contract of the boot node. Frameworks recommended are Exonium, Quorum and Geth.**	**Proposed system consists of a district node which manages the smart contract of the boot node. Frameworks recommended are Exonium, Quorum and Geth.**

Chapter 3

Problem Definition

3.1 Problem Statement:

- The Local Voting System has a lot of history of fake voting and double voting where people use to still the voting machine system and vote as much as they want.
- In Local Voting System Booth Capturing can take place by political pressure, which affects on actual voting result.

Chapter 4

Proposed Methodology

4.1. System Architecture

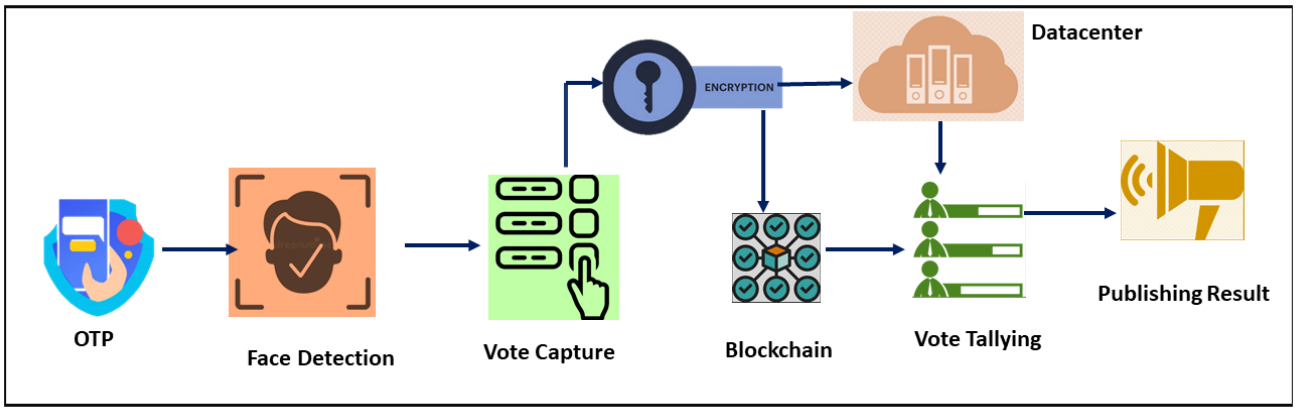


Fig 4.1: System Architecture of Smart Voting System using Blockchain

Once the voter and candidate is registered, the voter will continue the further process of voting by login. He/she will get the OTP while trying to login. Web cam will turn on to check the face of the voter, and start monitoring the face till the voting process is complete. Once the vote is captured, it will be saved in blockchain, so no one can interfere with the vote data. After the voting period is over,Vote tallying process will take place to count the votes, And then the result will be displayed to both voters and candidates.

4.2 Block Diagram:

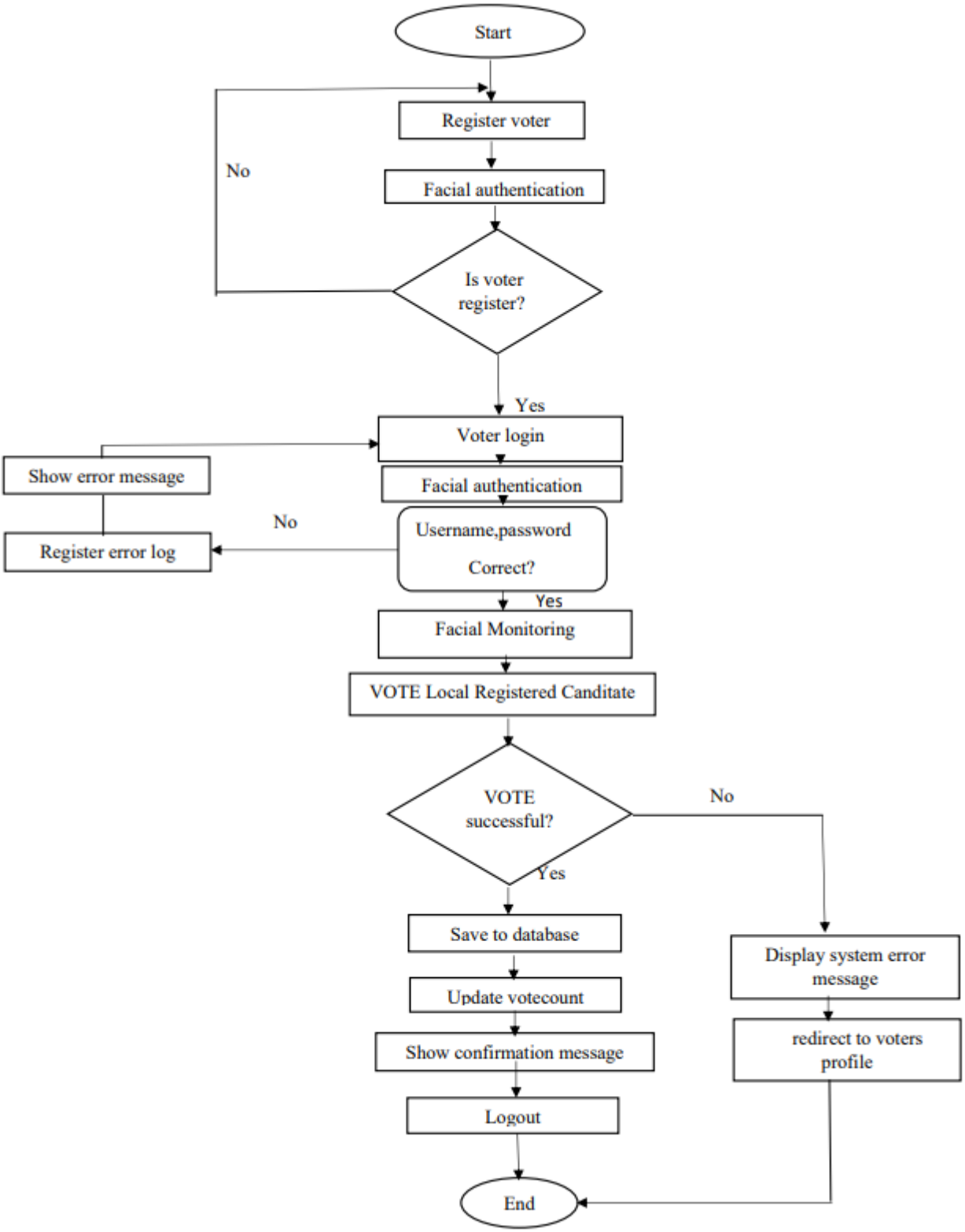


Fig 4.2: Block Diagram of Smart Voting System using Blockchain

4.3 Use case Diagram:

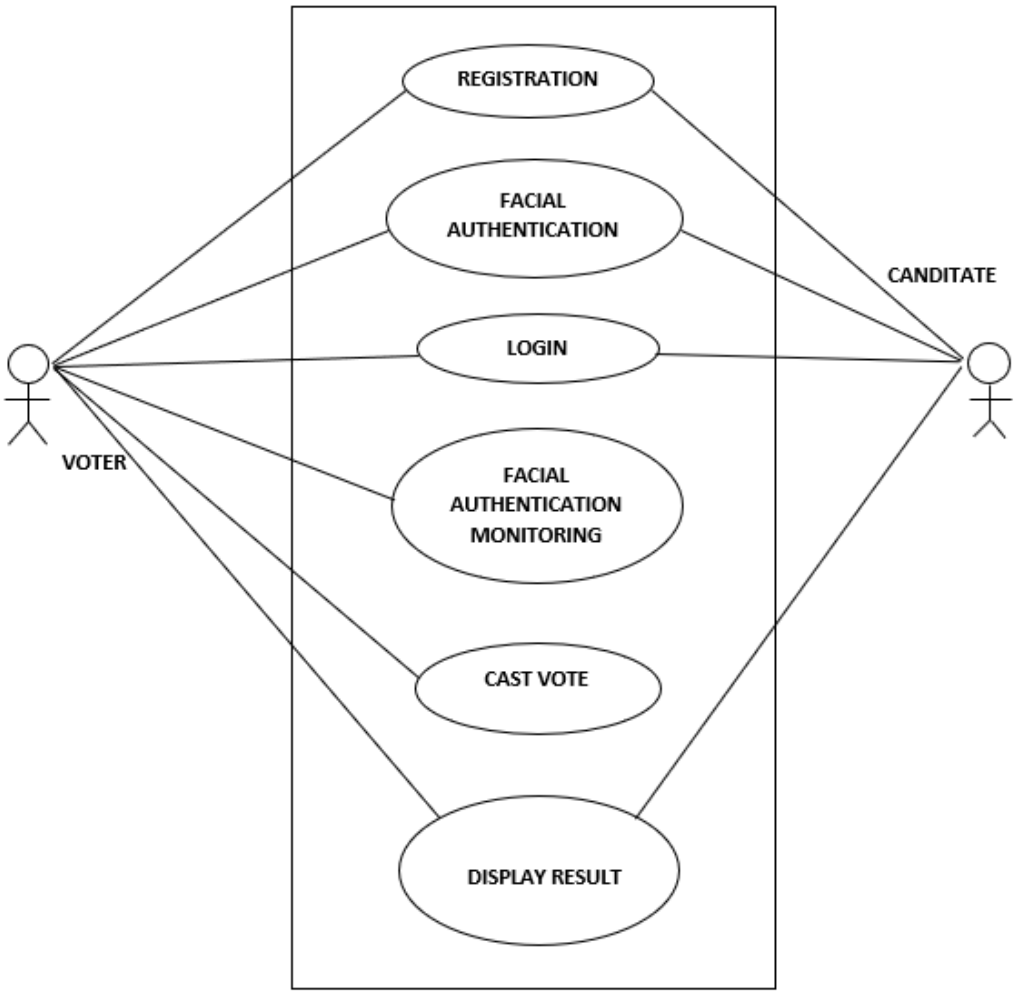


Fig 4.3: Use Case Diagram of Smart Voting System using Blockchain

4.4 Technology Used:

1. Hardware

- i5 processor
- 8 GB RAM
- 1TB HDD

2. Software

- Operating System: Windows 10
- VS code
- Node.js
- Metamask (Browser Extension)
- Nextjs and React

3. Modules

- Ethersjs
- Hardhat
- Tailwind css

4.5 Project timeline

Gantt chart

Project Start	14-Jul-2022		
TASK	PROGRESS	START	END
Guidelines about project and Topic Selection	100%	14-Jul-2022	20-Jul-2022
Title Selection	100%	21-Jul-2022	27-Jul-2022
Study of Literature	100%	28-Jul-2022	3-Aug-2022
Dscussions on architecture, wireframes and data flow models.	100%	4-Aug-2022	10-Aug-2022
Discussion and development of UML diagrams.	100%	18-Aug-2022	24-Aug-2022
Finalization of the Technology Stack	100%	25-Aug-2022	31-Aug-2022
Discussion and development of SRS	100%	1-Sep-2022	7-Sep-2022
Implementation: Discussion on the User Interface	100%	8-Sep-2022	14-Sep-2022
the tools and other editors	100%	15-Sep-2022	21-Sep-2022
Implementation: UI Design using Figma	100%	22-Sep-2022	28-Sep-2022
Implementation: UI Design using Figma	100%	29-Sep-2022	5-Oct-2022
Implementation: UI Design in Flutter	100%	6-Oct-2022	12-Oct-2022
Implementation: UI Design in Flutter	100%	13-Oct-2022	19-Oct-2022
Study of existing database of Terna Hospital	100%	20-Oct-2022	26-Oct-2022
Discussions on implementing the security policies	100%	27-Oct-2022	29-Oct-2022

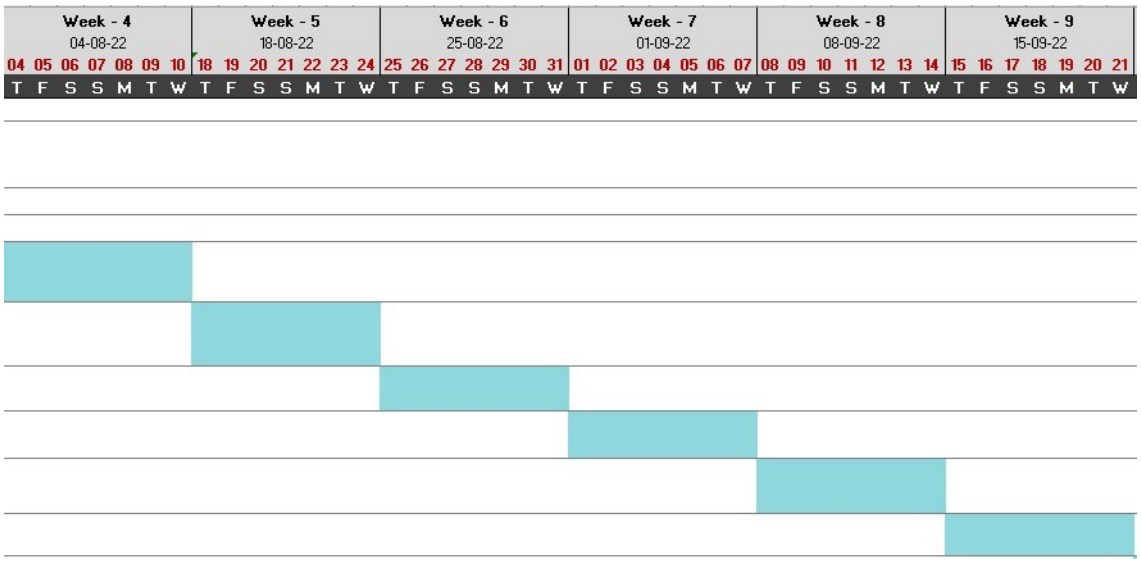


Figure 4.5: Gantt Chart

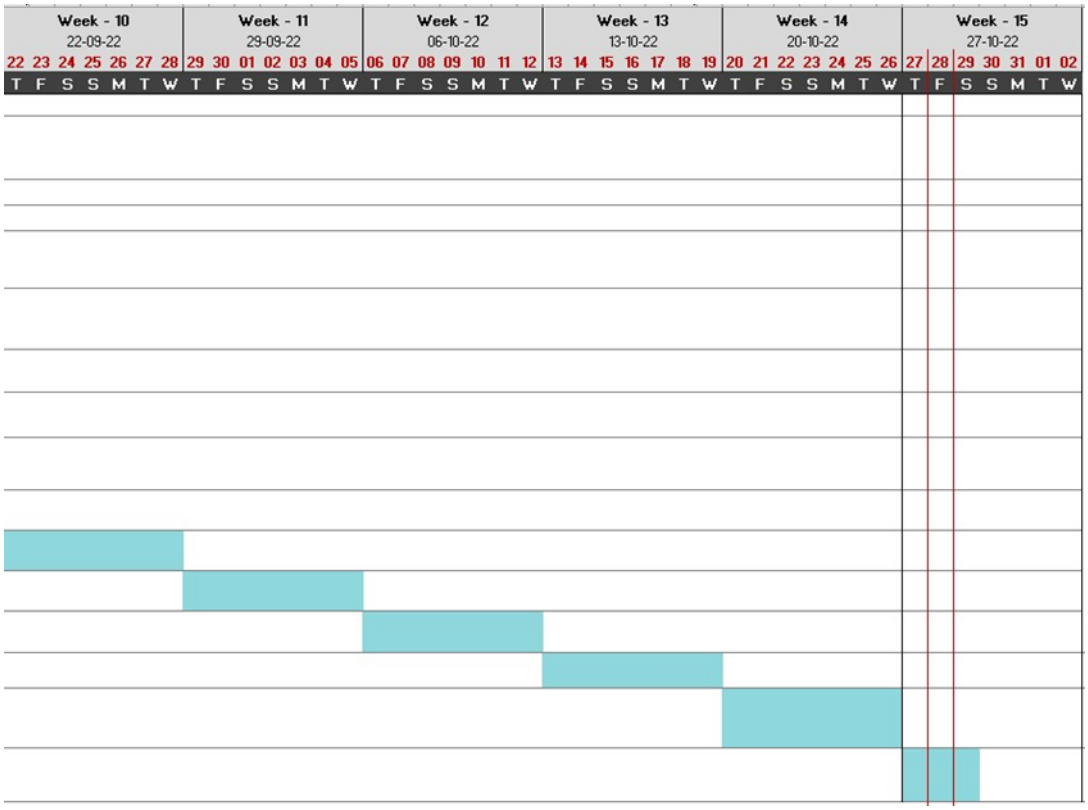


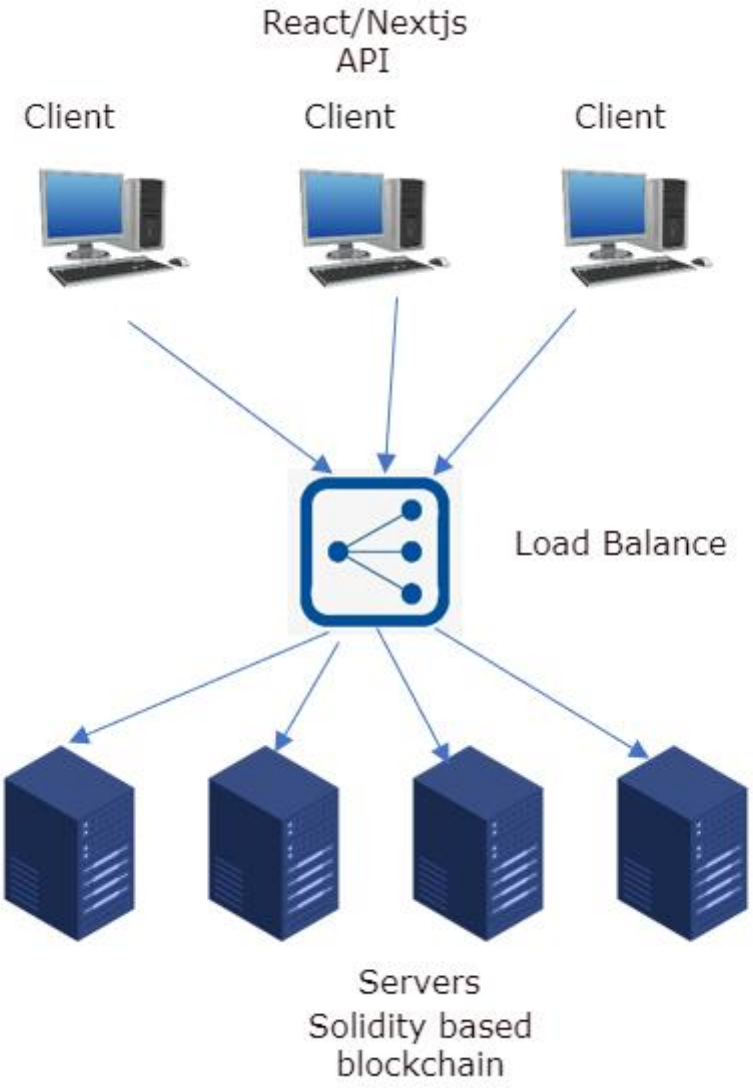
Figure 4.5: Gantt Chart

Chapter 5

Design, Analysis & Implementation

5.1 System design-

User Interface: We are going to build our UI with React and Nextjs as a Web Application. The web application will contain the option such as registration for both voters and candidates. There will be option such as Name, password, phone number and email address if needed, then facial authentication for storing the face data, After the registration, the next page will be about login, Once the voter successfully login's then the facial monitoring will start and he/she will be monitored during the voting process, after the voting process is done, it will show the message that the voting is successfully done and the voter will logout, after the voting period is over, result will be displayed to both candidates and voters.



5.2 Software analysis-

5.2.1 Waterfall model/Agile Model

The Waterfall Model was the first Process Model to be introduced. It is also referred to as a linear-sequential life cycle model. It is very simple to understand and use. In a waterfall model, each phase must be completed before the next phase can begin and there is no overlapping in the phases.

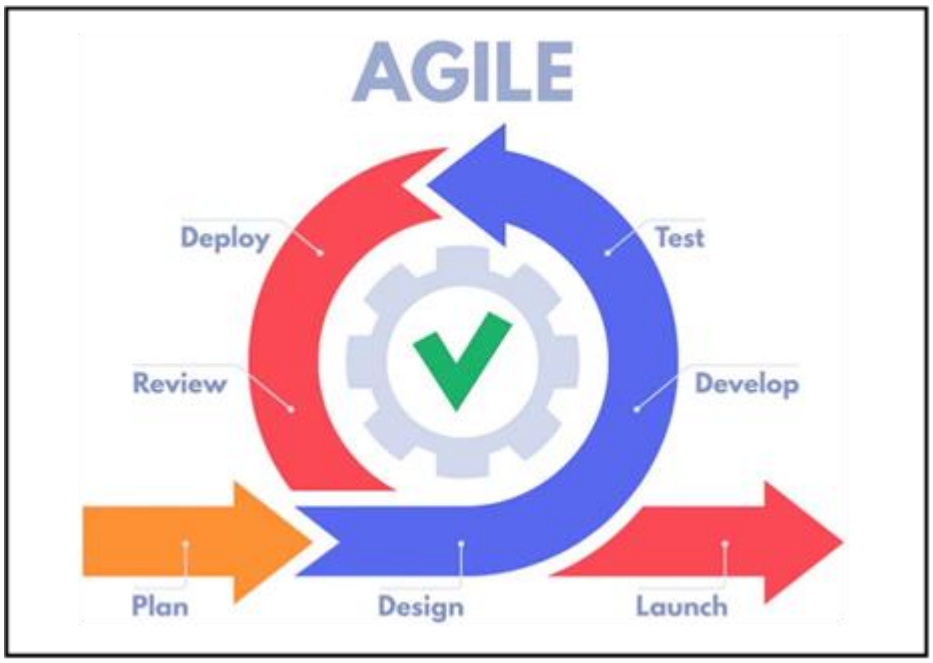


Figure 5.2.1: Agile Methodology

5.2.2 Phases

1. Requirements gathering: In this phase, you must define the requirements. You shouldexplain business opportunities and plan the time and effort needed to build the project. Based on this information, you can evaluate technical and economic feasibility.

2. Design the requirements: When you have identified the project, work with stakeholders to define requirements. You can use the user flow diagram or the high-levelUML diagram to show the work of new features and show how it will apply to your existing system.

3. Construction / iteration: When the team defines the requirements, the work begins. Designers and developers start working on their project, which aims to deploy a working product. The product will undergo various stages of improvement, so it includes simple, minimal functionality.

4. Testing: In this phase, the Quality Assurance team examines the product's performanceand looks for the bug.

5. Deployment: In this phase, the team issues a product for the user's work environment.

6. Feedback: After releasing the product, the last step is feedback. In this, the teamreceives feedback about the product and works through the feedback.

5.2.3 Implementation code

```
//SPDX-License-Identifier: MIT
pragma solidity ^0.8.4;

contract VoteSession {
  event CandidateRegistered(address _address, string _name, uint8 _id);
  event VotingStarted(uint256 _timestamp);
  event VotingEnded(uint256 _timestamp);

  error NotOwner();
  error NotEnoughCandidates();
  error NotEnoughVotes();
  error AlreadyVoted();
  error CandidateDoesNotExist();
  error AlreadyRegistered();
  error RegistrationIsClosed();
  error VotingIsClosed();
  error CannotRegisterInVotingPeriod();

  struct Candidate {
    string name;
    uint8 id;
    uint256 voteCount;
  }

  enum RegistrationStatus {
    OPEN,
    CLOSED
  }

  enum VotingStatus {
    OPEN,
    CLOSED
  }

  RegistrationStatus public registrationStatus;
  VotingStatus public votingStatus;
  mapping(uint8 => Candidate) public candidates;
  mapping(address => bool) public alreadyVoted;
  mapping(address => bool) public voters;
  uint8 public numberOfCandidates;
  uint256 public totalVotes;
  address public owner;

  constructor() {
    registrationStatus = RegistrationStatus.CLOSED;
    votingStatus = VotingStatus.CLOSED;
    owner = msg.sender;
  }

  function registerCandidate(uint8 _id, string memory _name)
    public
    isCandidateRegistrationOpen
  {
    if (candidates[_id].id != 0) {
      revert AlreadyRegistered();
    }

    numberOfCandidates++;
    candidates[_id] = Candidate({
      name: _name,
      id: numberOfCandidates,
      voteCount: 0
    });

    emit CandidateRegistered(msg.sender, _name, numberOfCandidates);
  }

  function registerVoter() public {
    if (voters[msg.sender] == true) {
      revert AlreadyRegistered();
    }

    voters[msg.sender] = true;
  }

  function registrationStart() public isCandidateRegistrationOpen checkOwner {
    if (votingStatus == VotingStatus.OPEN) {
      revert CannotRegisterInVotingPeriod();
    }
    registrationStatus = RegistrationStatus.OPEN;
  }

  function registrationStop() public checkOwner {
    registrationStatus = RegistrationStatus.CLOSED;
  }

  function votingStart() public checkOwner {
    if (numberOfCandidates < 2) {
      revert NotEnoughCandidates();
    }
```

```
    if (registrationStatus == RegistrationStatus.OPEN) {
      registrationStop();
    }
    votingStatus = VotingStatus.OPEN;
  }

  function votingStop() public checkOwner {
    registrationStatus = RegistrationStatus.CLOSED;
  }

  function vote(uint8 _candidateId) public isVotingOpen {
    if (alreadyVoted[msg.sender]) {
      revert AlreadyVoted();
    }
    if (_candidateId > numberOfCandidates) {
      revert CandidateDoesNotExist();
    }

    alreadyVoted[msg.sender] = true;
    candidates[_candidateId].voteCount++;
    totalVotes++;
  }

  function declareResult() public checkOwner returns (Candidate[] memory) {
    if (numberOfCandidates < 2) {
      revert NotEnoughCandidates();
    }
    if (totalVotes < 1) {
      revert NotEnoughVotes();
    }
    (, uint256[] memory voteCounts) = getCandidates();
    uint256 mostVotes;
    Candidate memory tempCandidate;
    for (uint8 i = 0; i < numberOfCandidates; i++) {
      if (voteCounts[i] > mostVotes) {
        mostVotes = voteCounts[i];
      }


    }

    // if size of winners is greater than 1
    // then its a draw
    Candidate[] memory winners = new Candidate[](numberOfCandidates);
    for (uint8 i = 0; i < numberOfCandidates; i++) {
      if (voteCounts[i] == mostVotes) {
        winners[i] = candidates[i];
      }
    }

    return winners;
  }

  function getCandidates()
    public
    view
    returns (string[] memory, uint256[] memory)
  {
    string[] memory names = new string[](numberOfCandidates);
    uint256[] memory voteCounts = new uint256[](numberOfCandidates);
    for (uint8 i = 0; i < numberOfCandidates; i++) {
      names[i] = candidates[i].name;
      voteCounts[i] = candidates[i].voteCount;
    }
    return (names, voteCounts);
  }

  function getCandidate(uint8 _candidateId)
    public
    view
    returns (Candidate memory)
  {
    return candidates[_candidateId];
  }

  function getTotalVotes() public returns (uint256) {
    return totalVotes;
  }

  function checkVoterIsValidOrNot() public view returns (bool) {
    return voters[msg.sender];
  }

  modifier isCandidateRegistrationOpen() {
    if (registrationStatus != RegistrationStatus.OPEN) {
      revert RegistrationIsClosed();
    }
    _;
  }
```

```
180
181     modifier isVotingOpen() {
182       if (votingStatus != VotingStatus.OPEN) {
183         revert VotingIsClosed();
184       }
185       _;
186     }
187
188     modifier checkOwner() {
189       if (msg.sender != owner) {
190         revert NotOwner();
191       }
192       _;
193     }
194   }
195
```

Chapter 6

Output/Result

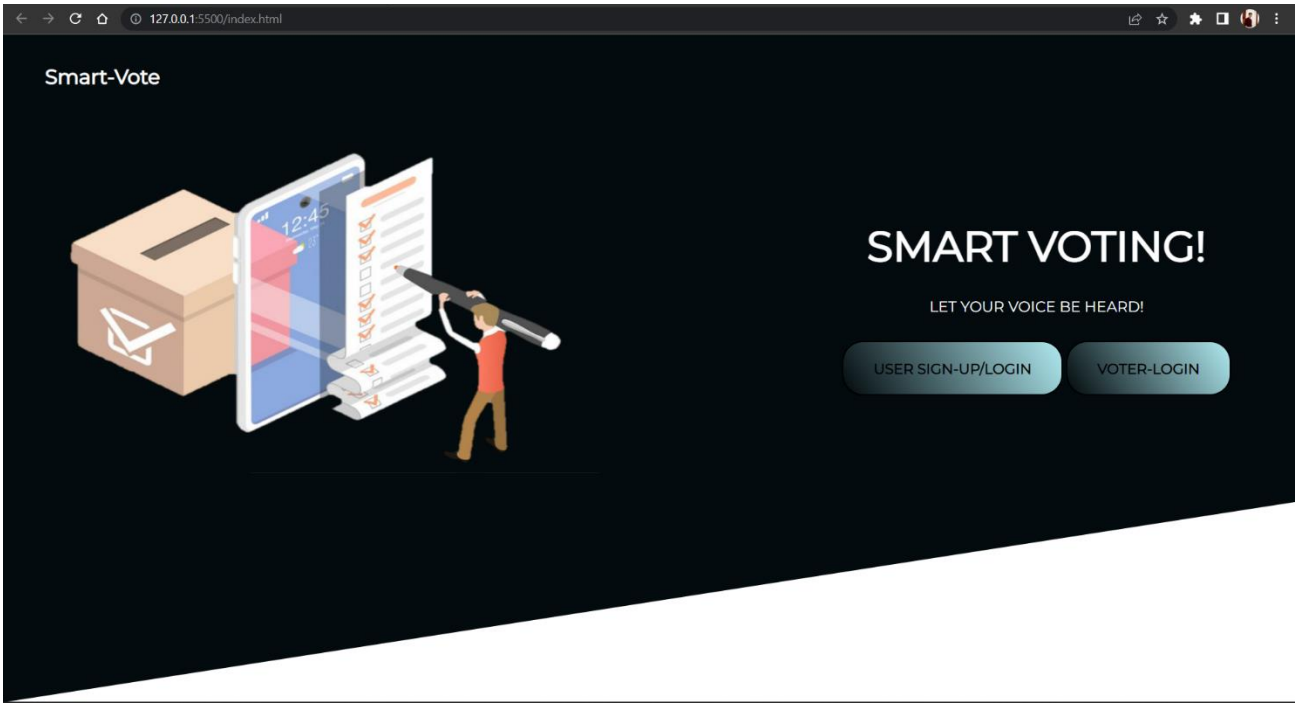


Fig 6.1: Front End (Main Page)

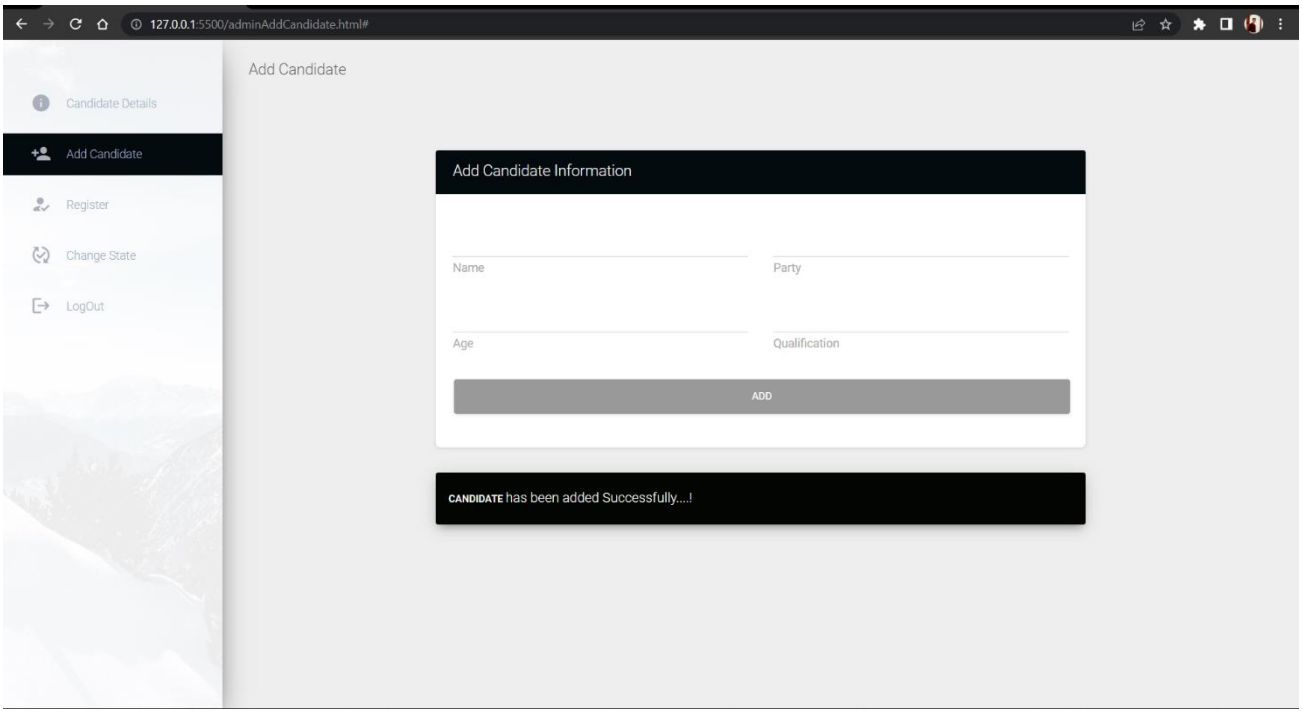


Fig 6.2: Registration Page for Candidates

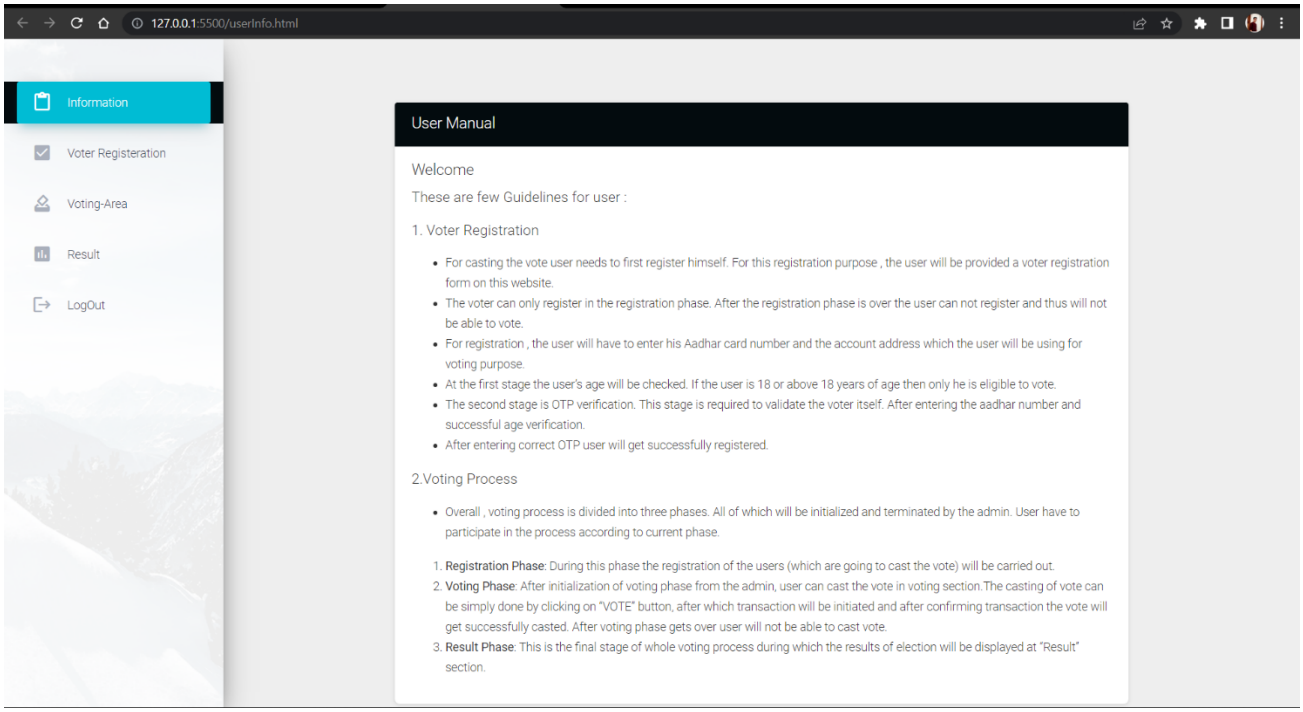


Fig 6.3: User (Voter) Manual for voters before the Voting

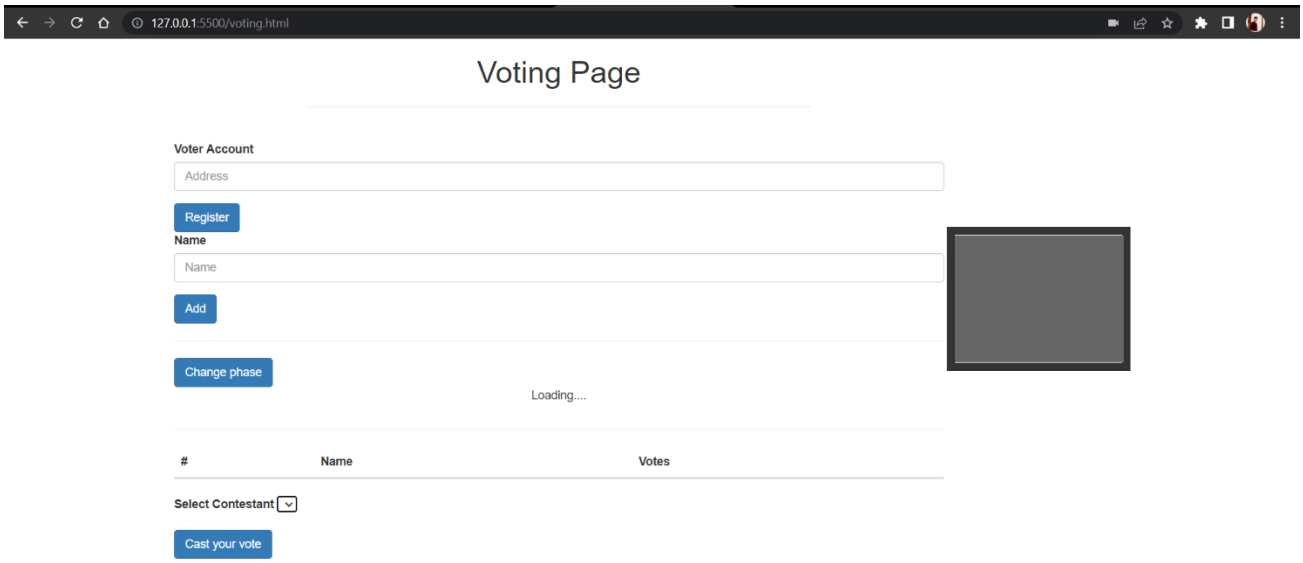


Fig 6.4: Voting Page while voting the candidate. That black box in right is camera

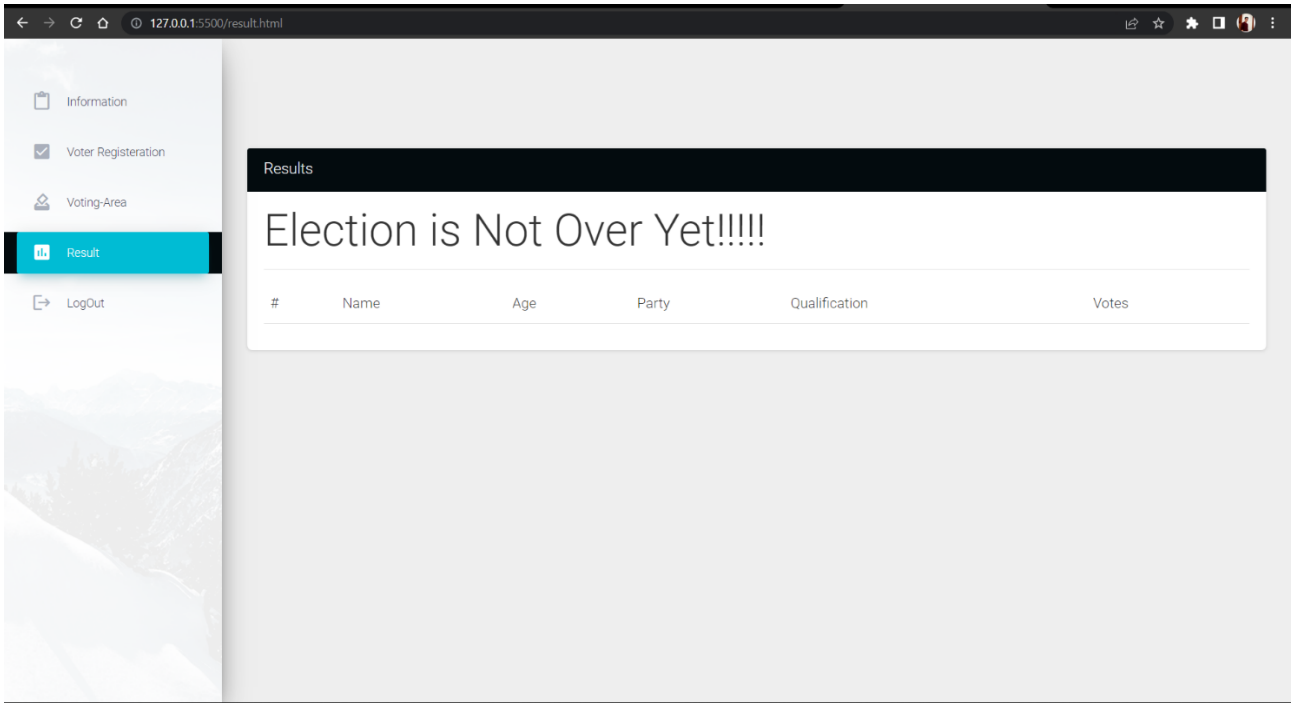


Fig 6.5: Result Page

Chapter7

Conclusion

Voting is an very important part of any flied such as politics, electing local community leader, school president, college president and so on. For which there is a traditional process of voting, and there are lot of loopholes in traditional process, that are really easy to exploit. With this system, we aim to eliminate the shortcomings of the traditional ballot system and provide the citizens of India an easy and secure access to voting. Our system would also help to conserve paper, indirectly saving trees which is the need of the hour as it is used in huge amounts in traditional voting. With the technology available today, we aim to strive for a brighter future for voting

Chapter 8

References

[1] E. Febriyanto, Triyono, N. Rahayu, K. Pangaribuan and P. A. Sunarya, "Using Blockchain Data Security Management for E-Voting Systems," 2020 8th International Conference on Cyber and IT Service Management (CITSM), 2020, pp. 1-4, doi: 10.1109/CITSM50537.2020.9268847
[2] J. Thakkar, N. Patel, C. Patel and K. Shah, "Privacy-Preserving E-voting System through Blockchain Technology," 2021 IEEE International Conference on Technology, Research, and Innovation for Betterment of Society (TRIBES), 2021, pp. 1-6, doi: 10.1109/TRIBES52498.2021.9751618.
[3] G. Rathee, R. Iqbal, O. Waqar and A. K. Bashir, "On the Design and Implementation of a Blockchain Enabled E-Voting Application Within IoT-Oriented Smart Cities," in IEEE Access, vol. 9, pp. 34165-34176, 2021, doi: 10.1109/ACCESS.2021.3061411.
[4] Z. Khudoykulov, U. Tojiakbarova, S. Bozorov and D. Ourbonalieva, "Blockchain Based E-Voting System: Open Issues and Challenges," 2021 International Conference on Information Science and Communications Technologies (ICISCT), 2021, pp. 1-5, doi: 10.1109/ICISCT52966.2021.9670245.
[5] "What Are Smart Contracts? A Beginner's Guide to Smart Contracts", Blockgeeks, 2016. Available at: https://blockgeeks.com/guides/ smart-contracts/
[6] Salanfe, Setup your own private Proof-of-Authority Ethereum network with Geth, Hacker Noon, 2018. Available at: https://tinyurl.com/ y7g362kd.
[7] Geth.ethereum.org. (2018). Go Ethereum. Available at: https://geth. ethereum.org/
[8] Vitalik Buterin. (2015). Ethereum White Paper Available at: https: //github.com/ethereum/wiki/wiki/White-Paper.
[9] Ethdocs.org. (2018). What is Ethereum? — Ethereum Homestead 0.1 documentation. [online] Available at: http://ethdocs.org/en/latest/ introduction/what-is-ethereum.html
[10] Agora (2017). Agora: Bringing our voting systems into the 21st century Available at: https://agora.vote/Agora_Whitepaper_v0.1.pdf [11] Patrick McCorry, Siamak F. Shahandashti and Feng Hao. (2017). A Smart Contract for Boardroom Voting with Maximum Voter Privacy Available at: https://eprint.iacr.org/2017/110.pdf
[11] Y. Zhang, Y. Li, L. Fang, P. Chen and X. Dong, "Privacy-protected Electronic Voting System Based on Blockchin and Trusted Execution Environment," 2019 IEEE 5th International Conference on Computer and Communications (ICCC), Chengdu, China, 2019, pp. 1252-1257, doi: 10.1109/ICCC47050.2019.9064387.
[12] Xiao S., Wang X.A., Wang W., Wang H. (2020) Survey on Blockchain-Based Electronic Voting. In: Barolli L., Nishino H., Miwa H. (eds) Advances in Intelligent Networking and Collaborative Systems. INCoS 2019. Advances in Intelligent Systems and Computing, vol 1035. Springer, Cham. https://doi.org/10.1007/978-3- 030-29035-1_54

Smart Voting System Using Blockchain

ORIGINALITY REPORT

6%	6%	4%	%
SIMILARITY INDEX	INTERNET SOURCES	PUBLICATIONS	STUDENT PAPERS

PRIMARY SOURCES

1	myfik.unisza.edu.my Internet Source	6%

Exclude quotes	Off	Exclude matches	Off
Exclude bibliography	Off		