Project Report

on

Decentralized E Voting with Smart Contracts

Submitted as partial fulfillment for the award of

BACHELOR OF TECHNOLOGY

DEGREE

Session 2022-23
in

Computer Science & Engineering

By
ROHIT KUMAR & LAVI BADWAL
1900320100131 & 1900320100080

Under the guidance of
MS. SANDHYA AVASTHI

ABES ENGINEERING COLLEGE, GHAZIABAD

AFFILIATED TO
DR. A.P.J. ABDUL KALAM TECHNICAL UNIVERSITY, U.P., LUCKNOW
(Formerly UPTU)

Project Report

on

Decentralized E Voting with Smart Contracts

Submitted as partial fulfillment for the award of

BACHELOR OF TECHNOLOGY

DEGREE

Session 2022-23
in

Computer Science & Engineering

By
ROHIT KUMAR & LAVI BADWAL
1900320100131 & 1900320100080

Under the guidance of
MS. SANDHYA AVASTHI

ABES ENGINEERING COLLEGE, GHAZIABAD

AFFILIATED TO
DR. A.P.J. ABDUL KALAM TECHNICAL UNIVERSITY, U.P., LUCKNOW
(Formerly UPTU)

STUDENT'S DECLARATION

We hereby declare that the work being presented in this report entitled **DECENTRALIZED EVOTING WITH SMART CONTRACTS** is an authentic record of my / our own work carried out under the supervision of **Ms. SANDHYA AVASTHI.**

The matter embodied in this report has not been submitted by us for the award of any other degree.

Signature of Students: **Dated:** May 30, 2023

Rohit Kumar (1900320100131) &

Lavi Badwal (1900320100080)

Department:

This is to certify that the above statement made by the candidates is correct to the best of my knowledge.

Signature of Supervisor **Signature of Project Coordinator** **Signature of HOD**

Name: **Prof. (Dr.) Divya Mishra**

Designation: **HOD-CSE**

Department: Computer Science & Engineering Department

CERTIFICATE

This is to certify that Project Report entitled **Decentralized E Voting with Smart Contracts** which is submitted by **Rohit Kumar & Lavi Badwal** in partial fulfillment of the requirement for the award of degree B. Tech. in Department of Computer Science & Engineering of Dr. A.P.J. Abdul Kalam Technical University, formerly Uttar Pradesh Technical University is a record of the candidate own work carried out by him/them under my supervision. The matter embodied in this thesis is original and has not been submitted for the award of any other degree.

Supervisor Signature

Name:

Designation:

Date:

ACKNOWLEDGEMENT

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

We also take the opportunity to acknowledge the contribution of Professor (Dr.) Divya Mishra, Head, Department of Computer Science & Engineering, ABESEC Ghaziabad for her full support and assistance during the development of the project.

We also do not like to miss the opportunity to acknowledge the contribution of all faculty members of the department for their kind assistance and cooperation during the development of our project. Last but not the least, we acknowledge our friends for their contribution in the completion of the project.

Signature of Students(s):

Name(s): Rohit Kumar & Lavi Badwal

Roll No.(s): 1900320100131 & 1900320100080

Date: April 30, 2023

ABSTRACT

A democratic election is a pivotal act in any country which decides the future of that country for a particular term. Some of the old means of voting like Ballot Paper and EVM (Electronic Voting Machine) has their drawback like transparency, low voter turn-out, votes tempering, and many more.

As with the advancement of modern digital society, the online trend is getting accelerated, which further creates security and authenticity issues. The issues found in the Ballot system or EVM can be easily overcome by Blockchain technology and Smart Contracts.

Electronic Voting powered by Blockchain & Smart Contracts takes the miles over these old means of a voting system which securely delivers the results in less time and cost. With E-Voting using Blockchain costs can be reduced, the need for Polling stations and the use of resources like EVM, Ballot Paper can also reduce as well as security can also be enhanced by providing End to End Encryption and authenticity. This blockchain-powered e-voting can easily gain trust as the transaction is transparent, and immutable as well as not easily be changed once hosted due to smart contracts. The proposed method is a MERN-based web Application with lots of enhanced methods for authentication and authorization that can be achieved using OTP Verification and face verification.

This voting data is stored in the form of a transaction stored in a Blockchain-based ledger through Smart Contracts to enhance security features.

Through this above-mentioned system, we can conduct election online and with High Security.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

LIST OF ABBREVIATIONS

AWS	Amazon Web Services
S3	Simple Storage Services
MERN	MongoDB ExpressJS ReactJS and NodeJS
DB	Database
JS	JavaScript
HTML	Hypertext Markup Language
CSS	Cascading Stylesheet
EVM	Electronic Voting Machine
E Voting	Electronic Voting
API	Application programming interface
SMS	Short Message Service
NSDL	National Securities Depository Limited
CDSL	Central Depository Services Limited
SQL	Structured Query Language

CHAPTER 1

INTRODUCTION

Election plays an important role in a large Democratic country like India. In a country such as India where a large section of the marginalized population is illiterate or ignorant, election officials must read paper ballot signatures or thumb impressions to determine the legality of votes. Votes from vulnerable populations are effectively discarded because they are riddled with inaccuracies. EVM technology ensures that these groups vote in elections and that their ballots are counted correctly. But EVM has its challenge. That's why this issue arises which includes Votes Tempering, polling booth capture, EVM hacking, and votes Manipulation. These problems were captured in the traditional way of voting and by the means of this advanced System, we tried to take meals over them. Online Voting is the latest trend comprised of the conduction of election or poll voting that makes the work of voting easier and fast.

When utilized in elections, electronic voting methods must be legal, accurate, safe, and convenient. However, adoption may be hampered by potential issues with computerized voting systems. To solve these concerns, blockchain technology was developed, which includes decentralized nodes for electronic voting. It is used to create electronic voting systems due to the benefits of end-to-end verification. This system is an excellent solution for traditional electronic voting methods due to its distribution, non-repudiation, and security properties.

This E-Voting powered by Blockchain enables to cast votes online with the power Blockchain which enhances the security, authenticity, and end-to-end encryption of voting records such that Nobody can change or temper the records. These records are stored in a decentralized manner such that all the information is shared with each node connected in the network and if any changes occur in data that this information is shared with every node. Our Tool or Application enables Citizens to cast votes authorized by Admin without going to polling Booths which reduces election costs and increases voting percentage.

1. 1. Problem Introduction

Elections are conducted from ancient times when kings were chosen by voting from the People and the Ministers of the King come to vote for a decision. But in the present time the two most commonly used mediums of voting are:

- Ballot Paper
- E V M (Electronic Voting Machine) Voting

The Ballot Paper Mode of Election has drawbacks like Votes Tampering/Manipulation, Polling Booth Capture, and requires physical presence and need large amounts of funds for the conduction of the election. On the other side, the EVM Mode of Election can be Hacked and tampered with easily due to this it does not gain voters' trust.

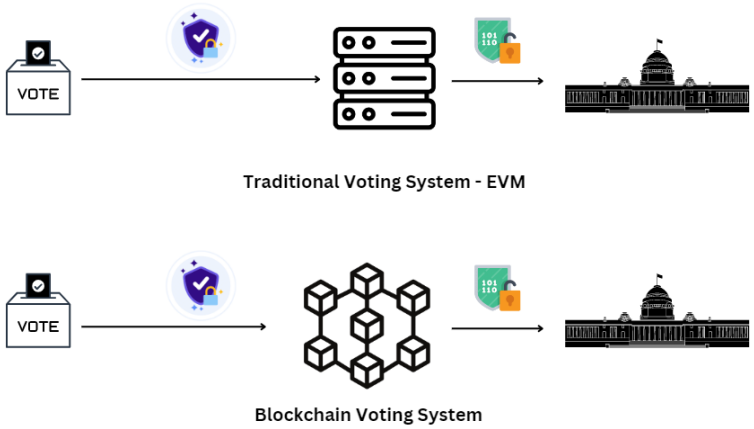

Figure 1: Comparison b/w EVM and Blockchain Mode of Voting

After going through the drawbacks of old mediums of voting we proposed an E-Voting System which reduces these drawbacks.

1. 1. 1. Motivation

The motivation for taking the next step towards a more decentralized voting structure comes from having a huge increase in the number of people and entities that have a vested interest in a particular outcome. In addition to current third-party involvement and the necessity for votes to be recorded and verified, this poses problems as a single party can manipulate the entire process by either altering the vote count or voting

1. 1. 2. Project Objective

To develop a trustworthy Secure Electronic Voting Solution that hides drawbacks of traditional voting medium and should be cost effective and free from any types of amendments and are highly secure.

1. 1. 3. Scope of the Project

The main scope of this Projects is to make step forward in direction of online voting by providing ways that compensates lackness of old voting mediums and provide an isolate way free from any type of dangers.

As you all know election like Govt. Elections, Polls, Society Elections plays a crucial role in judging a person based on the opinion of another person. In this case online election system are very useful with the help of this system users can cast their votes online which in immutable, highly secure and does not require additional setup and saves money too.

1. 2. Related Previous Work

The Popular E Voting Application build using Blockchain with their Limitations are:

1. Follow My Vote (https://followmyvote.com)

   Follow My Vote is a non-profit, independent election integrity and blockchain voting technology developer that is developing decentralized voting infrastructure using the Bitcoin blockchain.

   Follow My Vote has one major mission: to protect the integrity of the voting process and to ensure that every vote cast in an election is treated equally, even if one side or the other happens to win. In the case of an election, each side will win every time an election is true.

   Limitations:

   - During the voting process, if a vote is recorded incorrectly, the vote is invalid. In case of doubt, all votes are checked against a "kill-switch."

- Vote counts are NOT final until 48 hours after the close of voting on Election Day.
- Every voter will also have to use a special app, not available on the App Store, that they have to download to their phone and enter their personal information to prove their identity.

2. Ballotchain

Ballotchain allows for an online process with the same guarantees of a public election.

The fundamental idea of the Ballotchain solution is to match a Bitcoin transaction to a vote cast by an elector in support of the candidate selected by the voter. Every vote therefore benefits from the characteristics of a Blockchain transaction, namely: It is non-modifiable; It is non-repudiable; It cannot be registered in multiple ways; All nodes possess a valid copy. In practice, a voter casts their vote by giving a Ballot coin (a small amount of cryptocurrency as desired) to the wallet of their candidate.

Limitations:

- There is no centralized authority which can perform the processing. Voting systems all have a central authority which is in charge of data integrity and account encryption and/or security.
- Ballotchain is also not designed to work across state or country borders.
- Ballotchain voting solutions do not fix the most significant problem with elections, which is access and participation

3. NSDL e-Voting System (https://www.evoting.nsdl.com)

e-Voting is voting through an electronic system where members/shareholders can vote on resolutions of companies requiring members/shareholders consent. The need for e-Voting arises when a company wishes to pass a resolution by Postal Ballot/AGM/EGM which requires members/shareholders consent.

Ministry of Corporate Affairs has authorized NSDL for setting up an electronic platform to facilitate members/shareholders to cast vote in electronic form. Accordingly, NSDL has set-up an electronic infrastructure to facilitate members/shareholders to cast votes in electronic form through internet.

Limitations:

- Limited to Certain groups of people belongs from an organization
- Not Totally based on Blockchain
- Deals with Elections related to shareholders or organization structure

4. CDSL e-Voting System (https://www.evotingindia.com)

The e-Voting platform aims to improve transparency and Corporate Governance standards and also helps in reducing the administrative cost associated with Postal Ballot while facilitating declaration of results immediately after the close of the voting.

Limitations:

- Limited to Certain groups of people belongs from an organization
- Not Totally based on Blockchain
- Deals with Elections related to shareholders or organization structure

5. Polys (https://polys.votee)

Polys is online voting platform based on blockchain technology and backed with transparent crypto algorithms.

It provides facilitate help governments, businesses, educational institutions, and communities dispense with paperwork and facilitate efficient online elections.

Limitations:

- No more than 20 voters can participate in the free version of Polys.
- Polys functionality does not allow voting on multiple issues within one voting session.
- Presently It cannot delete or hide the voting even after it ends.

6. B-Voting

B-Voting (Blockchain-Voting) is the innovative electronic voting system engineered and developed by Net Service, integrated with one or more electoral event management procedures (system set-up, distribution of credentials, voting, collection of ballot papers, counting of preferences, publication of results).

Limitations:

- Not Scalable
- Costly

7. Trustworthy Electronic Voting Using Adjusted Blockchain Technology - Basit Shahzad Raju, Jon Crowcroft in the year 2019

As per this Paper, new hashing Algorithm was introduced which increases user security and it also introduces some concept of Block-sealing and block-creation. A framework is suggested in this System which used hashing method.

Table 1: Table for Comparison with Existing System

	Decentralized	**Specific Groups**	**Immediate Result**	**Voting ID**	**Free**	**Central Authority**
Our application	✓	✗	✓	✓	✓	✓
Follow my Vote	✓	✗	✗	✗	✗	✓
Ballot chain	✓	✗	✓	✗	✗	✗
NSDL eVoting System	✗	✓	✗	✓	✓	✓
CSDL eVoting System	✗	✓	✗	✓	✓	✓
Polys	✓	✗	✓	✗	✗	✓
B- Voting	✓	✗	✓	✗	✗	✓

CHAPTER 2

SOFTWARE REQUIREMENT SPECIFICATION

2. 1. Product Perspective

The E-Voting System (CastMyVote) is a self-contained system that helps in conduction of elections online due to that helps it helps in increasing voting percentage and saves a large amount of money of election organizing authority. It becomes the task of various stakeholder easy and in a convenient way i.e stakeholders requires a better medium of election and wants that large amount of voters involved in the election and voters also needs a convenient way to cast their votes online without going to Booths and no need to time spend.

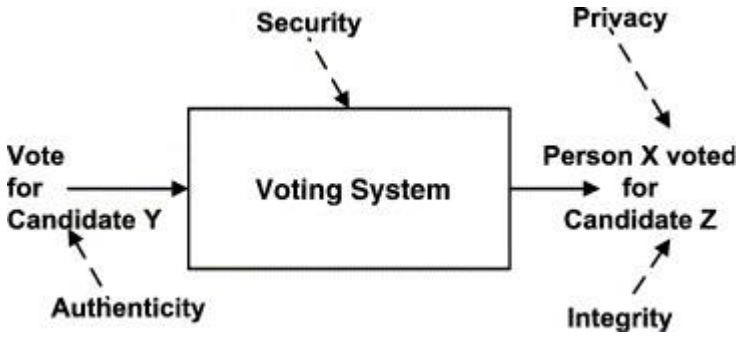

Figure 2: Product Perspective

The following subsections describe how the software operates inside various constraints.

2. 1. 1. System Interfaces

The various systems are used in order to fulfill the requirements of our system. These are:

(i) NodeMailler

NodeMailler is basically a NodeJS based library that is used to Email through SMTP API through this our work becomes easy to setup Emails Communication with the user.

(ii) Twilio

Twilio is a platform that is used to send SMS alert to the user. It permits message sender to only verified user without purchase plan

(iii) MongoDB

MongoDB is a Non-SQL (Key-Value) based database which we used in our system to maintains records. Here we have use MongoDB Atlas cluster

(iv) NodeJS Server

NodeJS Server is written in JavaScript scripting language. It helps in linking all other system in one place i.e Database, NodeMailler etc. Here we have API which is called by Client side as a response of those api a task is executed.

(v) Amazon Web Services (AWS) S3

AWS S3 is used to store files. Here we have used this to keep files that is linked in Database.

(vi) Ganache

The local Blockchain that provides some free Web 3 Account with some gas or ethers.

2. 1. 2. Hardware Interfaces

The system has no hardware interface requirements.

2. 1. 3. Software Interfaces

2.1.4.1 NodeJS Server v16.16.0. The system must use NodeJS Server which helps in maintaining connection database. This server also facilitates Email and SMS Communication and it also helps in generation of PDF Cards using PDF-Lib library. Communication with the DB is through API Calls. The system must provide No-SQL database through Mongoose Library.

2.1.4.2 Client (Web Browser). The client (voter as well admin) requires a latest web browser which helps in communication with the server through API calls.

2.1.4.3 MongoDB Database. This system used MongoDB NO-SQL (key-value based) database which are linked with server through mongoose.

2. 1. 4. Communications Interfaces

This system used https protocol for the communication between client and server. Our server is hosted in a NodeJS Web Hosting which performs task when the API Calls.

When a client wants to execute a task then web browser (from client) calls a respective API as the result of that respective API server can do a task and a response to returned to client web browser and a valid result is shown on browser window as an output.

2. 1. 5. Memory Constraints

This system has certain memory constraints which were:

i. For the System it requires a client latest web browser latest run properly.
ii. AWS S3 has limitations of 10 GB Files Storage
iii. MongoDB Atlas has a storage limit of upto 500MB
iv. Size of NodeJS Server should be less than 1GB.

2. 1. 6. Operations

The normal and special operations required by the user such as:

(1) User must wait for server to becomes active and status is available to them.
(2) The user must login properly in that System and Login session will be of only 15min within that user must complete their tasks.
(3) User must check their mails specially in Junk Mails.
(4) User will follow on-screen instructions to cast their votes.

2. 1. 7. Site Adaptation Requirements

The system is deployed in Online Hosting Provider which does not require any adaptation and additional hardware. This server is hosted in online Hosting provider which does not require server hardware management and no space and cooling of server is required.

Record entry does not require a special procedure it inserts records based on the DB Schema and no any other purchase software is required.

2. 2. Product Functions

This system level is divided into two view (i) Voter View and (ii) Admin View

The major functions available in Voter's view are:

i. Signup and Login
ii. Apply for new Voter and Track its Status
iii. Download their voter ID card
iv. Check His/her voting History

v. Cast their Votes
vi. Check Result of the election once election completed
vii. View Election Information.

The major functions available in Admin's View are:

i. Voter Management (i.e Voter addition and voter approval)
ii. Election Management (Add, Update and delete)

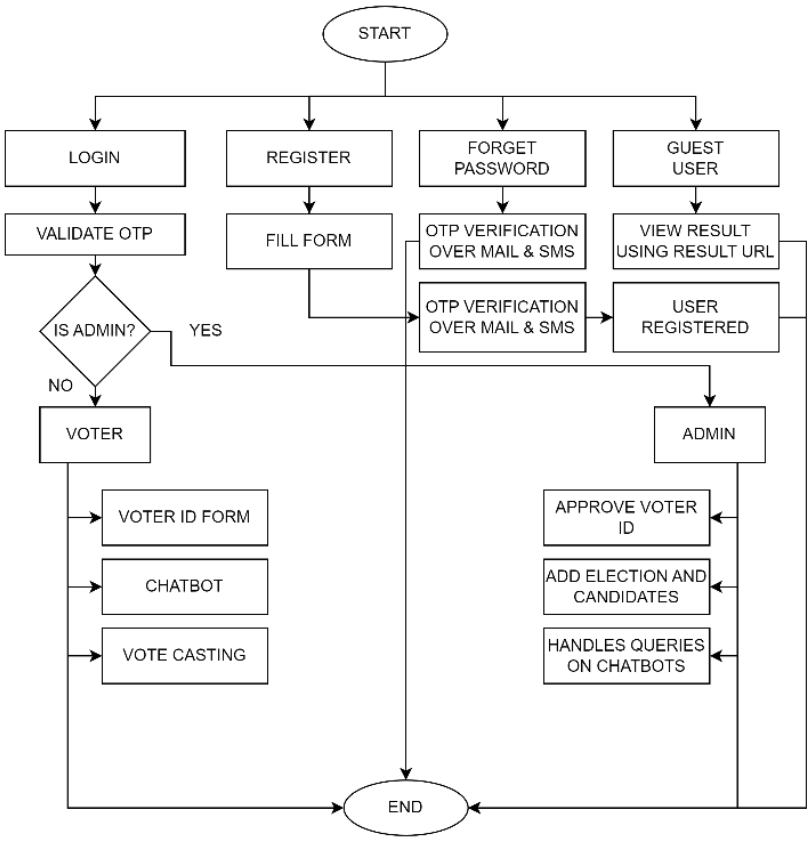

Figure 3: System Functions

2. 3. User Characteristics

To use this system voter/ User must have basic requirements or characteristics:

i. User must be a registered voter (Generated after approval from the admin)
ii. The User must have a valid email and phone for communication.
iii. The user must have basic knowledge about computer fundamentals.

2. 4. Constraints

The main constraints in this system are:

(1) The Free database (MongoDB) has a certain limit of 512 MB.
(2) The file storage AWS S3 has also limit of 10GB.
(3) The Free server hosting makes our server sleep so that it takes some time in seconds to start server till than API calls does not work.
(4) The admin panel is accessed by Admin Email Address with Password and OTP.

2. 5. Assumptions and Dependencies

The above SRS requirements are also affected by various factors. While considering that this system has some dependencies and assumptions taken into consideration.

These are:

Assumptions:

i. Each voter has a valid Email and Phone no for authentication.
ii. Each voter has an internet accessed Device (i.e., Mobile Phone or a computer)
iii. Voter has a good Internet Connectivity.
iv. The system should be user-friendly so that it is easy to use for the users
v. This Application is Error Free

Dependencies:

i. Database and File Storage is limited so it can't be scalable but it can be scalable with Money.
ii. SMTP Mail API has a limit of 200 Mails per Day.
iii. Free Mobile SMS API has a certain limit and can be send to registered mobile number.
iv. AWS S3 Free Plan is valid only for 12 Months after that it starts money.
v. Free NodeJS Server hosting has also limits of 750hrs (Render Hosting) and it makes server offline to save time.

2. 6. Apportioning of Requirements.

The basic requirements that may be delayed until future versions of the system but if I look at the projects plan are:

i. Proper functioning of elections (like as a Lok Sabha and a Vidhan Sabha elections)
ii. Proper decentralized model of projects

2. 7. Use case Diagram

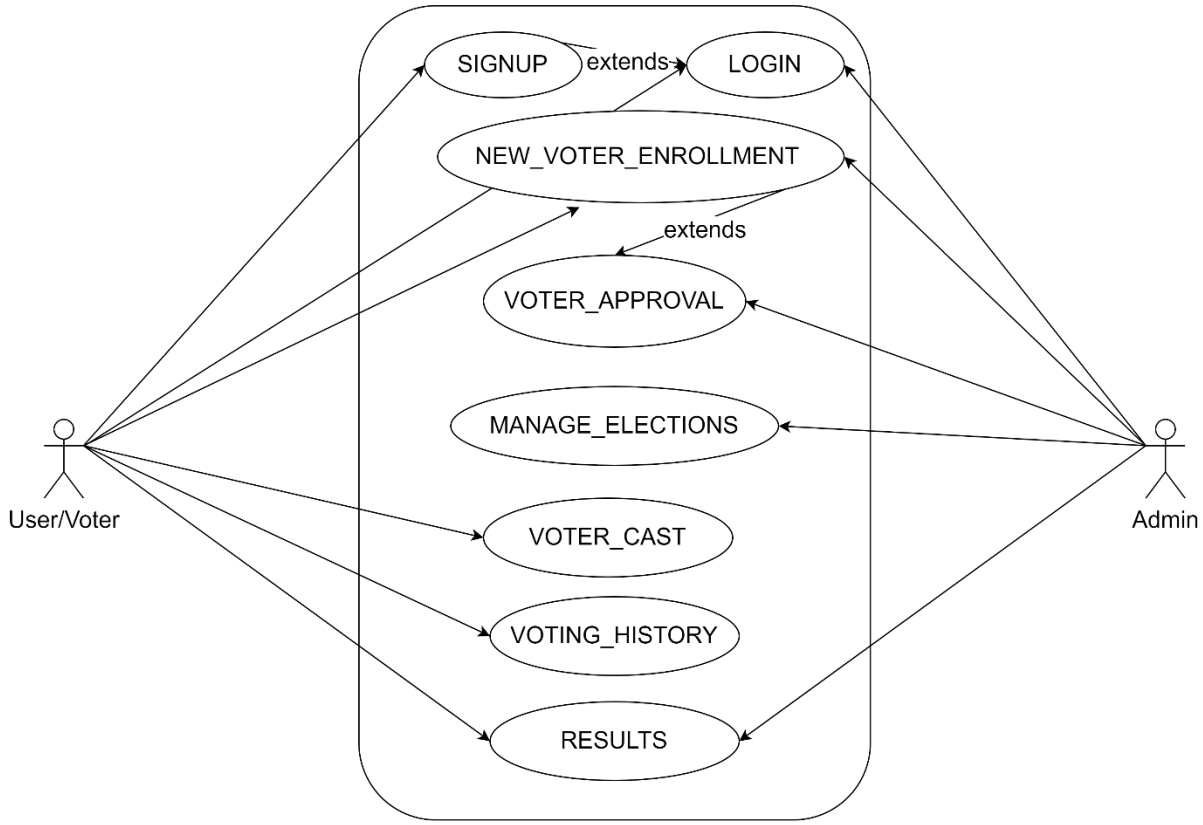

Figure 4: Use Case Diagram for E Voting

The main stakeholder of this system are:

- Election authority: It includes authority who has given task to conduct free & fair election. There may be other statutory bodies as well, such as the legislative institutions themselves, security organizations, or local governments that have some responsibility to support election preparations.

- The contestants: It includes person who may take direct part in the election which may be belong from a party as well as a group and the main reason to conducting election is to choose a good candidate among all contestants.
- The electorate or Voter: It includes the person who have right to votes. It also includes the election authority members as well as contestants. These have a certain eligibility criterion those who have passed that criteria will have the right to vote.

2. 8. Sequence Diagrams

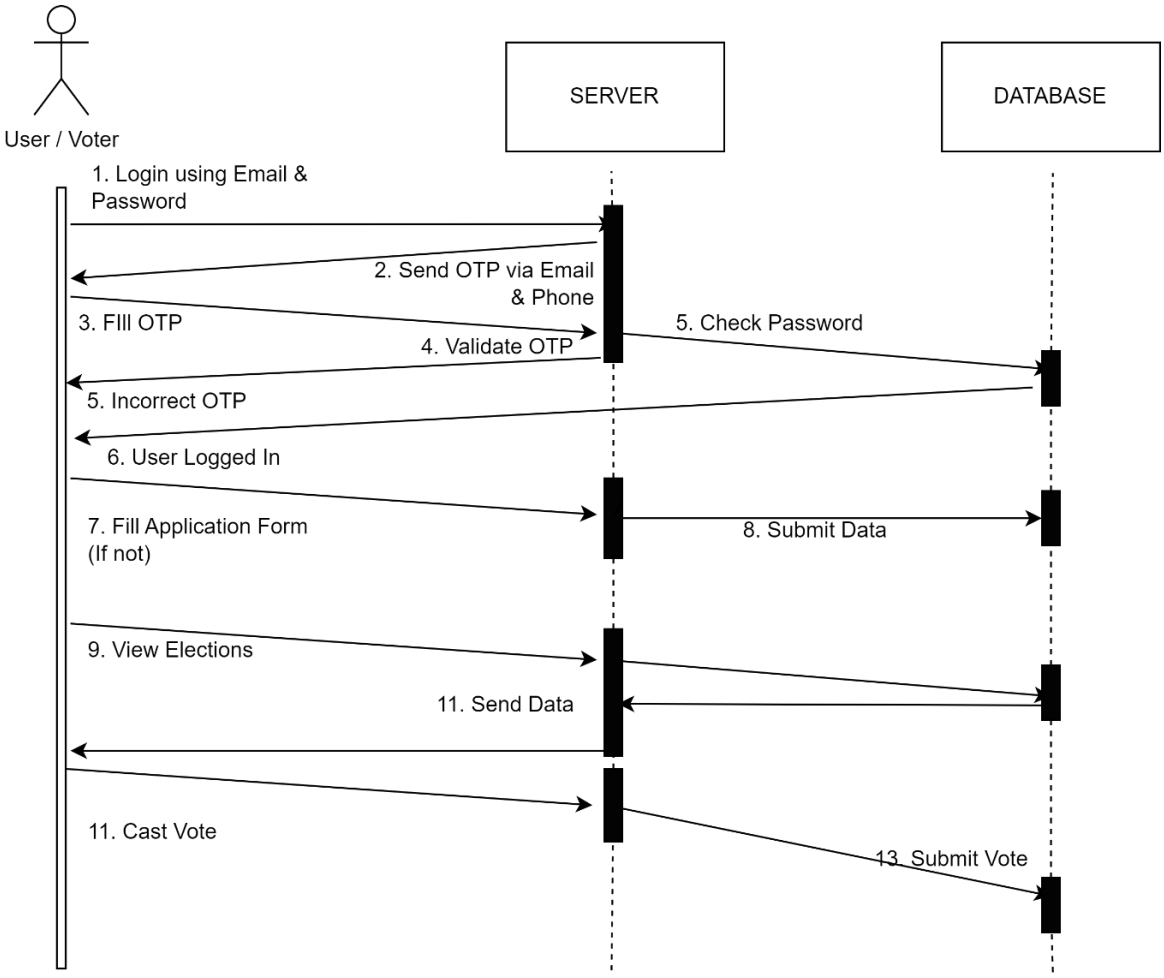

Figure 5: Sequence Diagram for Voter

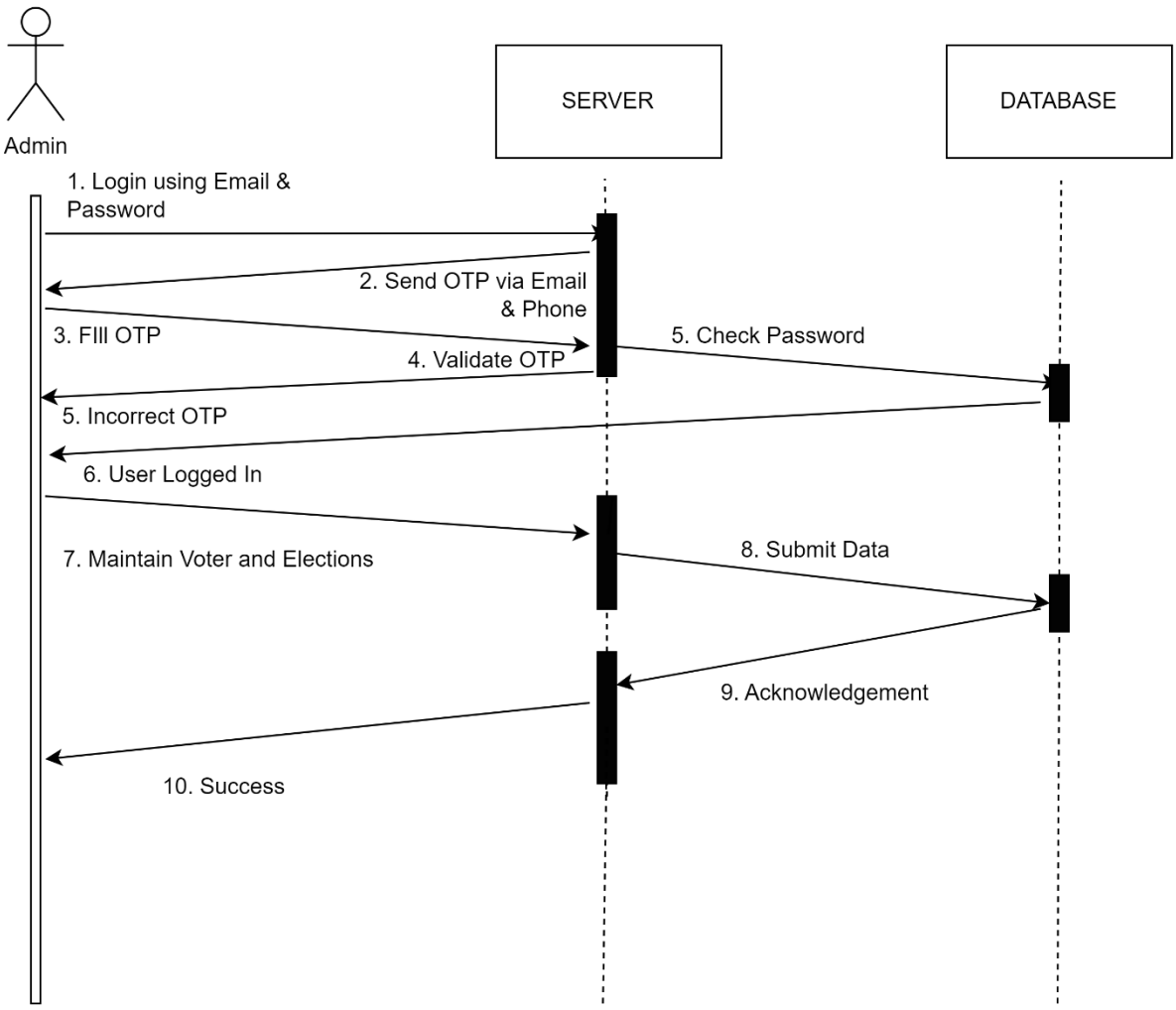

Figure 6: Sequence Diagram for Admin

CHAPTER 3

SYSTEM DESIGN

3. 1. Architecture Diagrams

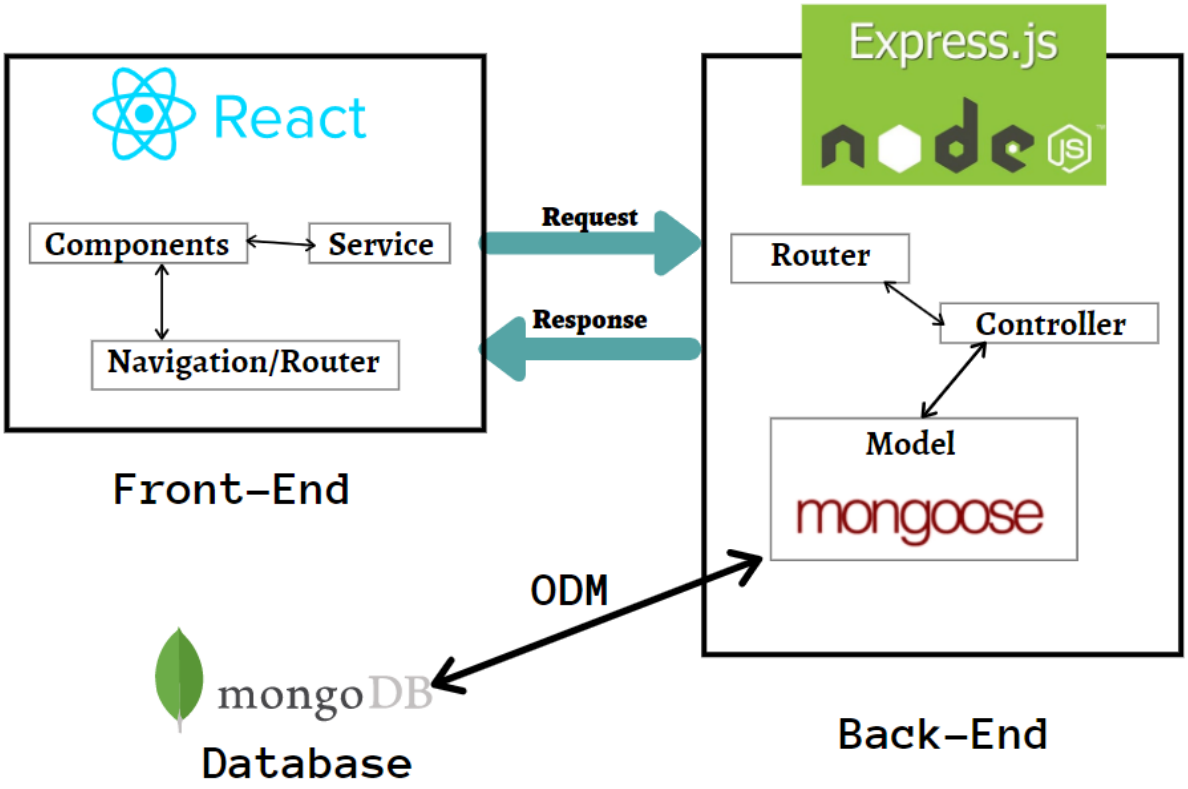

Figure 7: Architecture Diagram for E Voting

3. 2. Data Flow Diagram

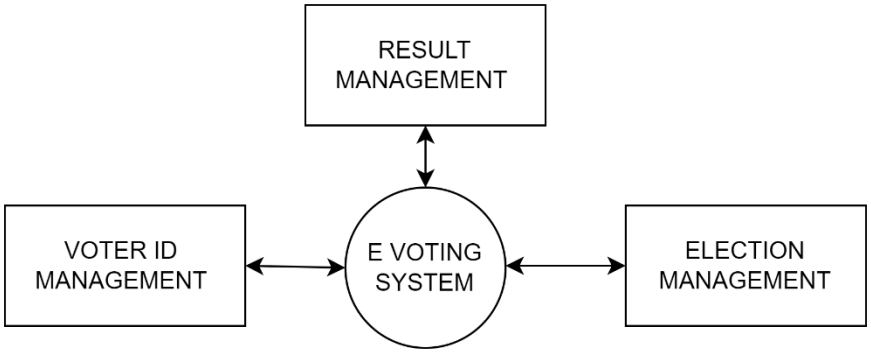

Figure 8: Zero Level DFD for E Voting

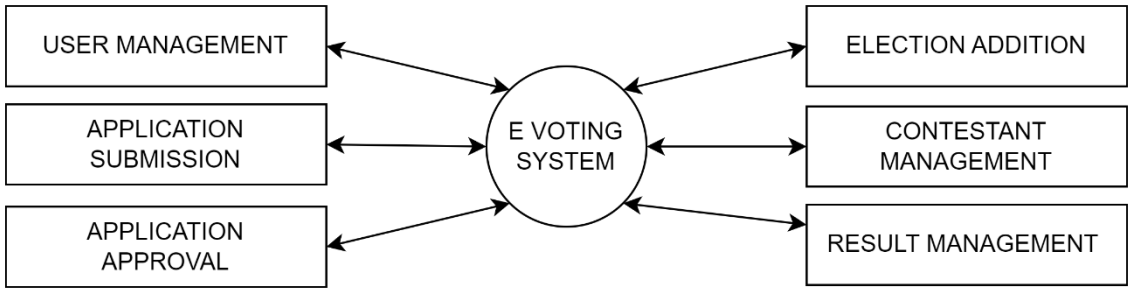

Figure 9: One Level DFD for E Voting

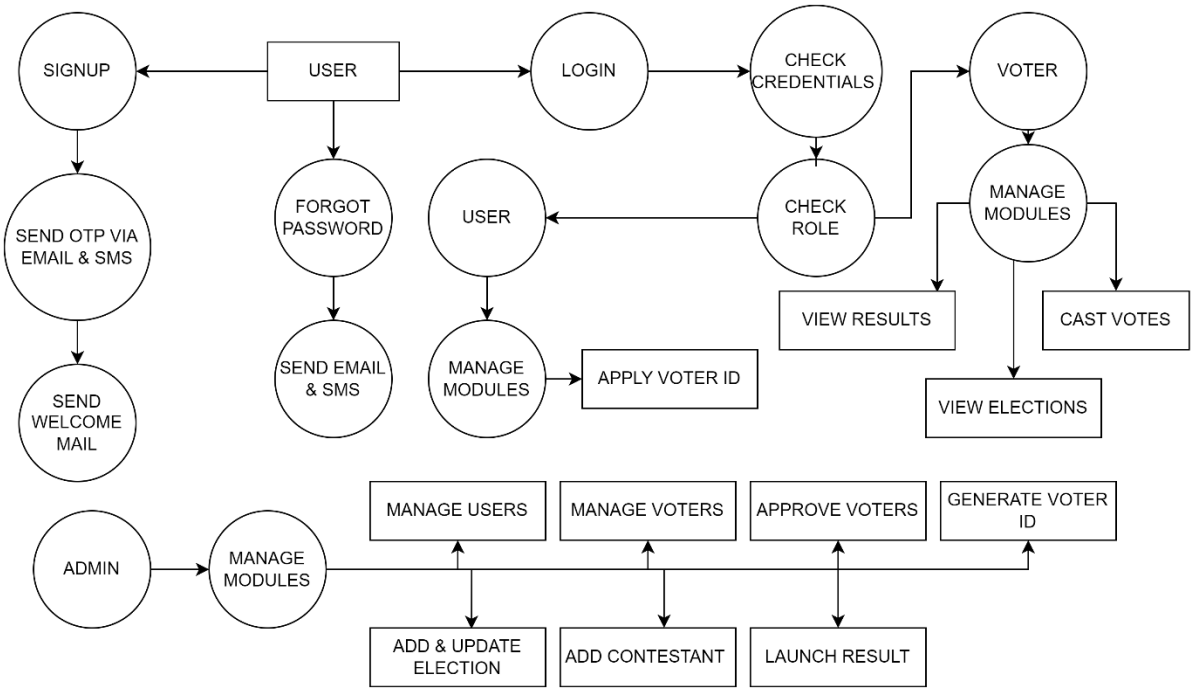

Figure 10: Second Level DFD for E Voting

3. 3. Activity Diagram

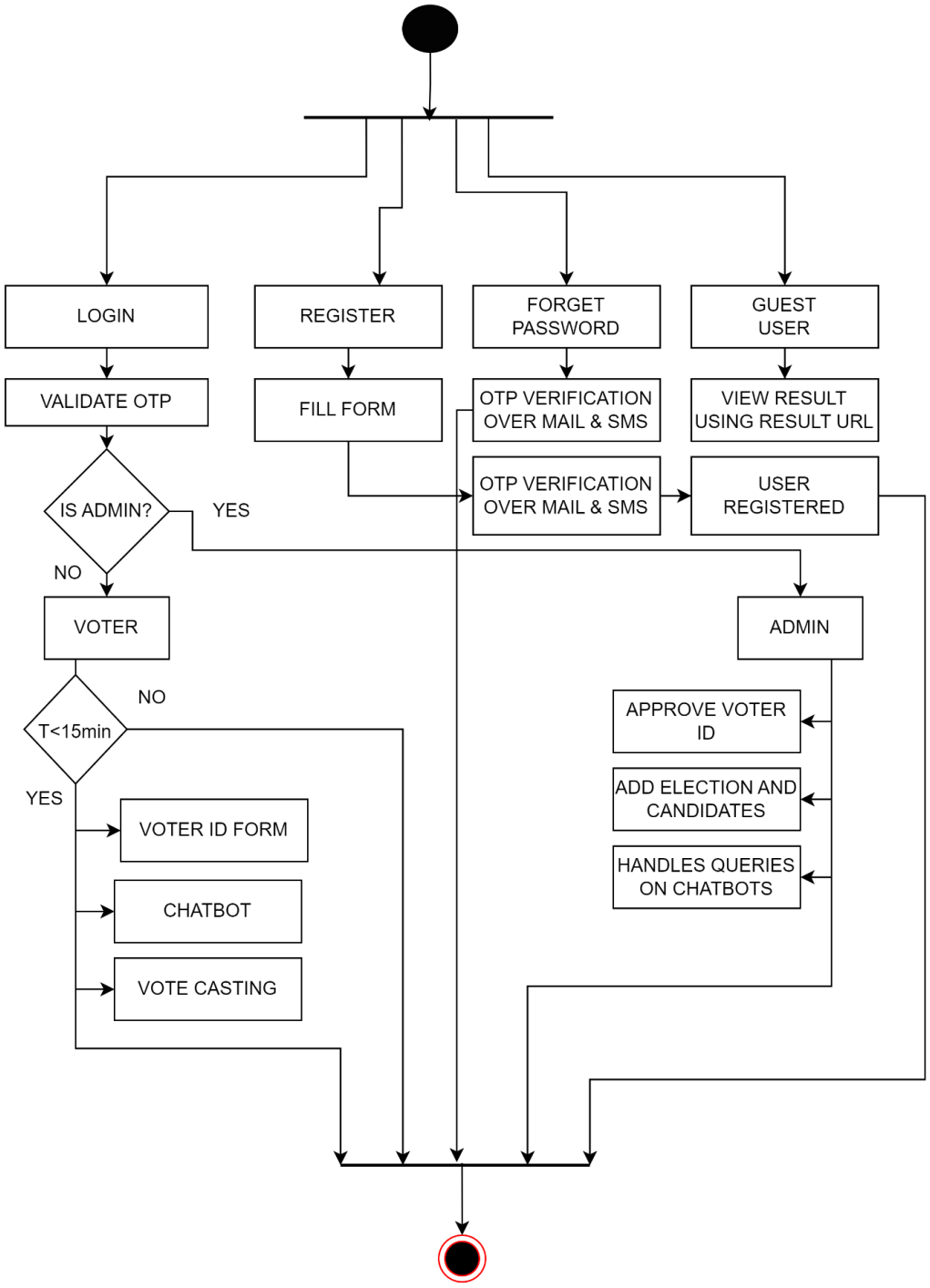

Figure 11: Activity Diagram for E Voting

3. 4. ER Diagrams

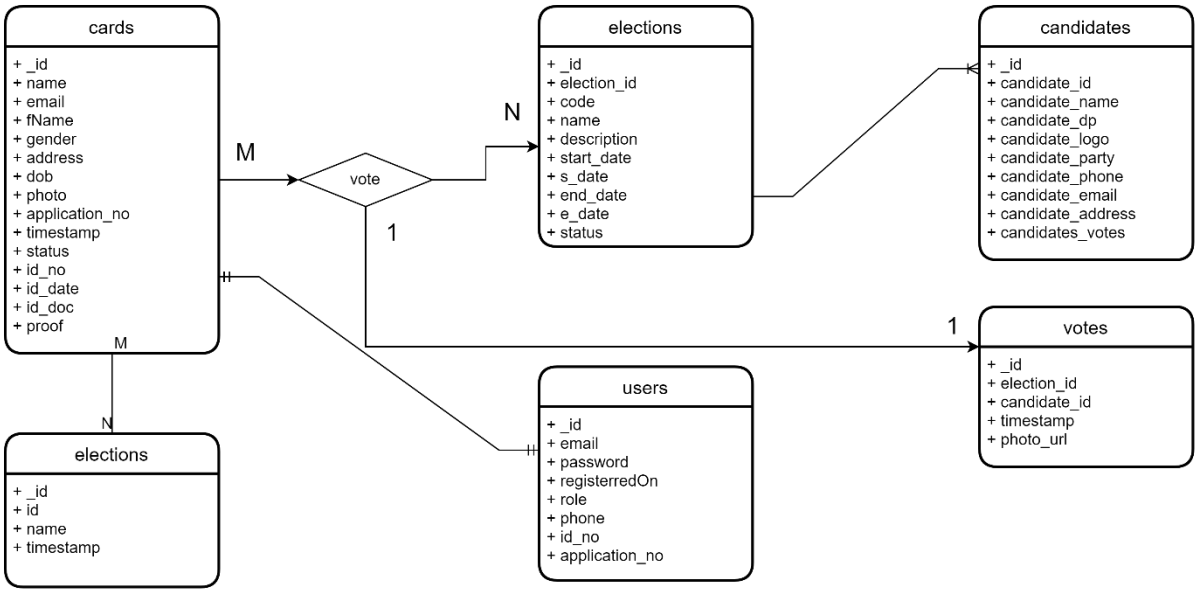

Figure 12: ER Diagram for E Voting System

3. 5. Database schema diagrams

3. 5.1 Users

```
name: { type: String, },
phone: { type: String, },
email: { type: String},
password: { type: String, },
registeredOn: { type: String },
role: { type: String },
id_no: { type: String },
application_no: { type: String },
```

Figure 13: Database Schema for Users Entity

3. 5.2 Cards

```
name: { type: String },
email: { type: String },
fName: { type: String },
gender: { type: String },
phone: { type: String },
address: { type: String },
dob: { type: String },
photo: { type: String },
application_no: { type: String },
timestamp: { type: String },
status: { type: String },
id_no: { type: String },
id_date: { type: String },
id_doc: { type: String },
proof: { type: String },
elections: [
  {
    id: { type: String },
    name: { type: String },
    timestamp: { type: String },
  },
],
```

Figure 14: Database Schema for Cards Entity

3. 5.3 Votes

```
election_id: { type: String },
candidate_id: { type: String },
timestamp: { type: String },
snapshot: { type: String },
```

Figure 15: Database Schema for Votes Entity

3. 5.4 Elections

```
election_id: { type: String },
code: { type: String },
name: { type: String },
description: { type: String },
start_date: { type: String },
s_date: { type: String },
end_date: { type: String },
e_date: { type: String },
status: { type: String },
candidates: [
  {
    candidate_id: { type: String },
    candidate_name: { type: String },
    candidate_dp: { type: String },
    candidate_logo: { type: String },
    candidate_party: { type: String },
    candidate_phone: { type: String },
    candidate_email: { type: String },
    candidate_address: { type: String },
    candidate_votes: { type: Number },
  },
],
```

Figure 16: Database Schema for Elections Entity

CHAPTER 4

IMPLEMENTATION AND RESULTS

4. 1. Software and Hardware Requirements

Hardware Requirements:

This Basic Hardware Required to develop and run this trustworthy system are:

- Processor: Minimum 1.6 GHz; Recommended 2GHz or more
- Ethernet connection (LAN) OR a wireless adapter (Wi-Fi)
- Hard Drive: Minimum 32 GB; Recommended 64 GB or more
- Memory (RAM): Minimum 1 GB; Recommended 4 GB or above
- Hosting:
- Backend: Recommended Online Server with NodeJS Environment like Localhost, Heroku, Cyclic etc.
- Frontend: Recommended Online Hosting like Vercel, Firebase, GitHub Pages, Firebase etc.

Software Requirements:

This Basic Hardware Required to develop and run this trustworthy system are:

- Operating System:

  OS X El Capitan (10.11+)

  Windows 8.0, 8.1 and 10, 11 (32-bit and 64-bit)

  Linux (Debian): Ubuntu Desktop 16.04, Debian 9

  Linux (Red Hat): Red Hat Enterprise Linux 7, CentOS 7, Fedora 34

- Web Browser (Latest version of Google Chrome, Microsoft Edge, Mozilla Firefox etc.).
- A Code Editor (Recommend VS Code by Microsoft)
- Git or GitHub Desktop
- Ganache PC App
- NodeJS Installed on System.

4. 2. Assumptions and dependencies

Assumptions:

- Each voter has a valid Email and Phone no for authentication.
- Each voter has an internet accessed Device (i.e., Mobile Phone or a computer)
- Voter has a good Internet Connectivity.
- The system should be user-friendly so that it is easy to use for the users
- This Application is Error Free

Dependencies:

- Database and File Storage is limited so it can't be scalable but it can be scalable with Money.
- SMTP Mail API has a limit of 200 Mails per Day.
- Free Mobile SMS API has a certain limit and can be send to registered mobile number.
- AWS S3 Free Plan is valid only for 12 Months after that it starts money.
- Free NodeJS Server hosting has also limits of 750hrs (Render Hosting) and it makes server offline to save time.

4. 3. Constraints

The main constraints in this system are:

- The Free database (MongoDB) has a certain limit of 512 MB.
- The file storage AWS S3 has also limit of 10GB.
- The Free server hosting makes our server sleep so that it takes some time in seconds to start server till than API calls does not work.
- Ethereum Blockchain used is based on Ganache
- The admin panel is accessed by Admin Email Address with Password and OTP.

4. 4. Implementation Details

In this section design, implementation, and functioning of the E-Voting application are discussed. This application can be accessed by both admin and user for their specific needs.

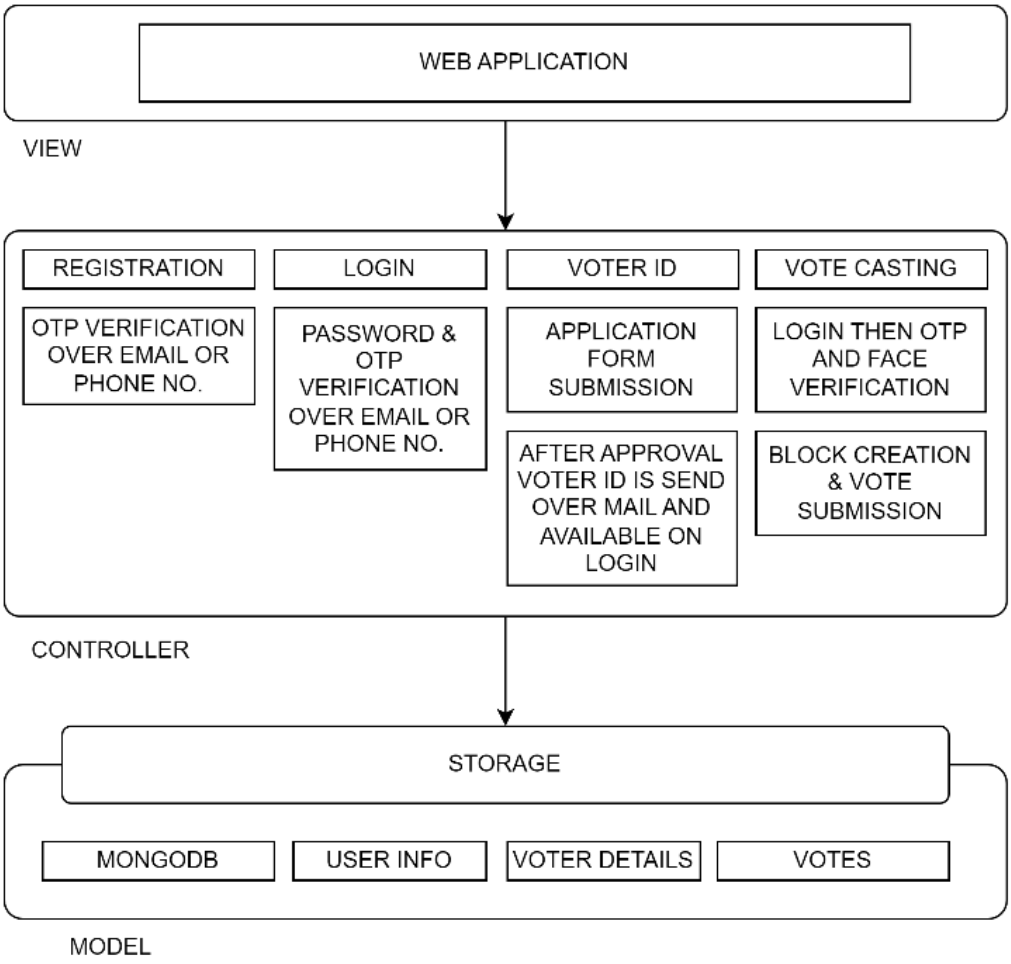

Figure 17: MVC Architecture of System

Both user and admin can access the application from where it is hosted and the user can cast their votes as well register him/her as a first-time voter and the admin can assign election/poll to the registered user and accept an application for first-time voters.

User-Driven Modules:

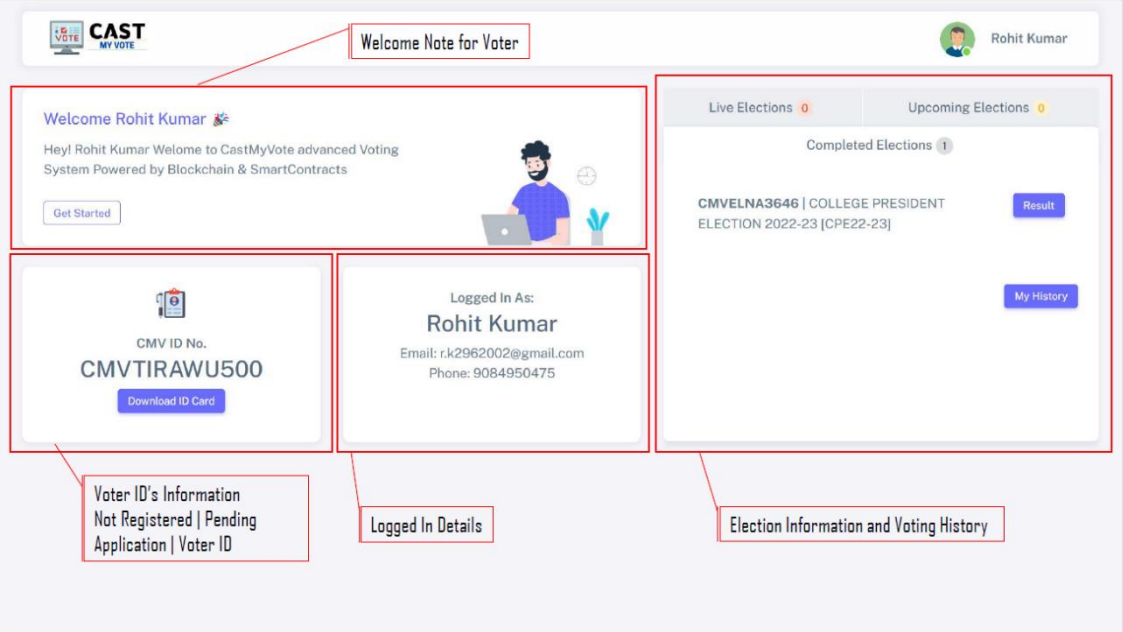

Figure 18: User Dashboard

Signup Module: In this module, a voter can first register themselves through basic details like Name, Phone No, Email & Password after signup Voter can receive OTP over email once a verified Voter account is created into the System.

Voter Registration Module: In this Module, voters can request their Voter ID by filling out a form with basic fields like DOB, Address, and Photo. After submitting of form Voter will receive an Application ID that shows the status of the application and the voter will receive mail for the same and the same is also displayed on the Voter Dashboard.

Login Module: In this Module, Voters can sign in to their accounts to access all services, the voter will enter their email and password an OTP is sent to the registered Email, Once OTP verified Voter can enter into its dashboard and a login token is created with a limit of 15min after that voter will sign-out automatically.

Voter ID Module: In this Module, voters can download their Voter ID Card. ID Card can be available when the Registration form is approved by the admin. This Voter Card

is in the .pdf format with Name, Photo, and basic details mentioned on it and with a Voter ID Card Bar Code mentioned for further use.

Voting Module: In this Module, voters can cast their vote just by choosing Election through Name and a list of Participants is available to him/her. Voter Just chooses Choice and submits voter with OTP verification and Face Verification with a Registration Photo and Voter Selfie is uploaded at the time of submission of the vote. After the successful submission of the Vote, a thank you mail was sent to the Voter.

User Login > Choose Election (with Live Status) > OTP Verification over Mail and by SMS > Choose Candidate > Submit > Success.

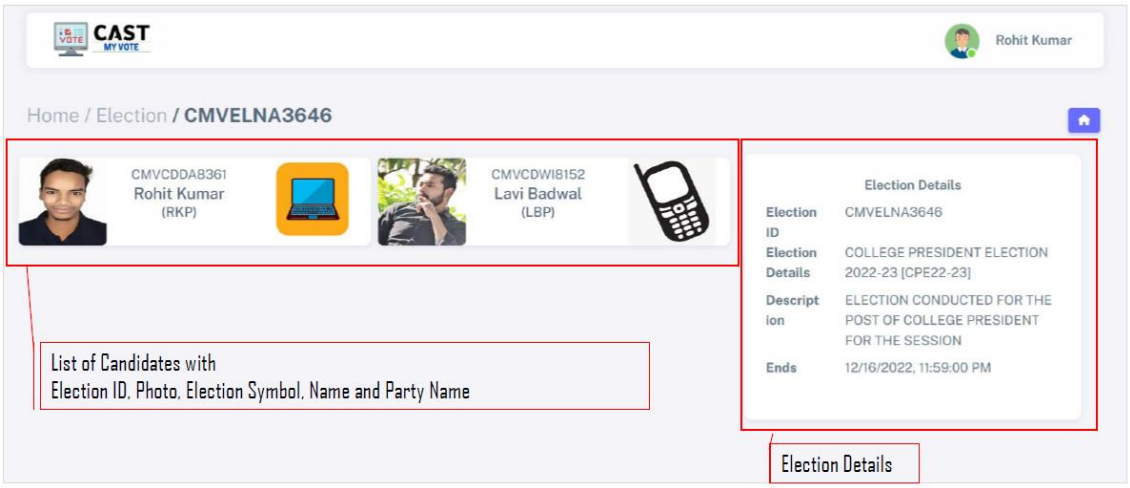

Figure 19: Voting Module

Admin-Driven Modules:

In the following section, some admin-driven modules are described.

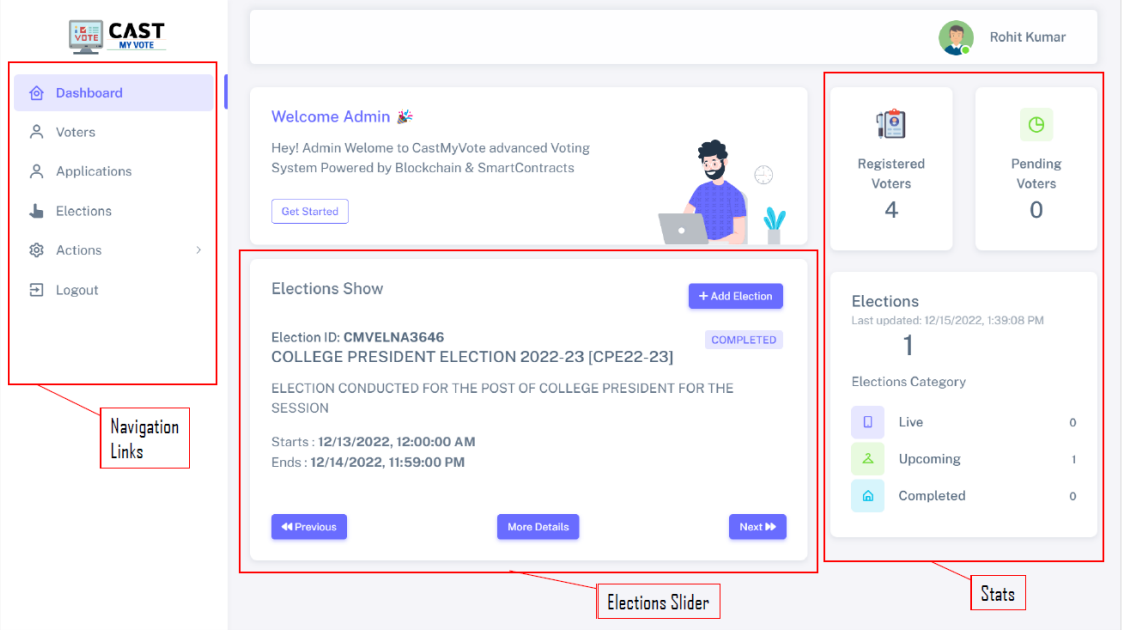

Figure 20: Snapshot of Admin Dashboard

Application Module: In this module, The Applications submitted by Users will be displayed from where Admin can view the submitted form with supporting doc. From here user can Accept or discard the application. Once Discarded a discard mail is sent to the user with the discard reason and Once approved New Voter ID is issued to the application an SMS with details and an email with the Voter ID attached as an attachment is sent to the User and the User will then become a voter.

Election Module: In this module, Admin can create a new Election with a name and a Unique Code after addition a Voting Invitation mail is sent to all the Voters with some basic details about the Election and a link to the tentative list of Candidate. After Election addition Admin can add, Update and delete candidates till Election starting time. Once Election started Anybody cannot change details like Election Info. Candidates etc.

Figure 21: Snapshot of Election Module

Common Modules:

Result Module: In this module, the result of the election is prepared Once Election got finished the System will check all the Blocks on the Blockchain Network and count the Votes. After the Counting of all Votes, the Result is available on Admin and Voter Login and a Result mail is sent to Everyone and Result is also displayed on Results URL send on the Election thank you mail.

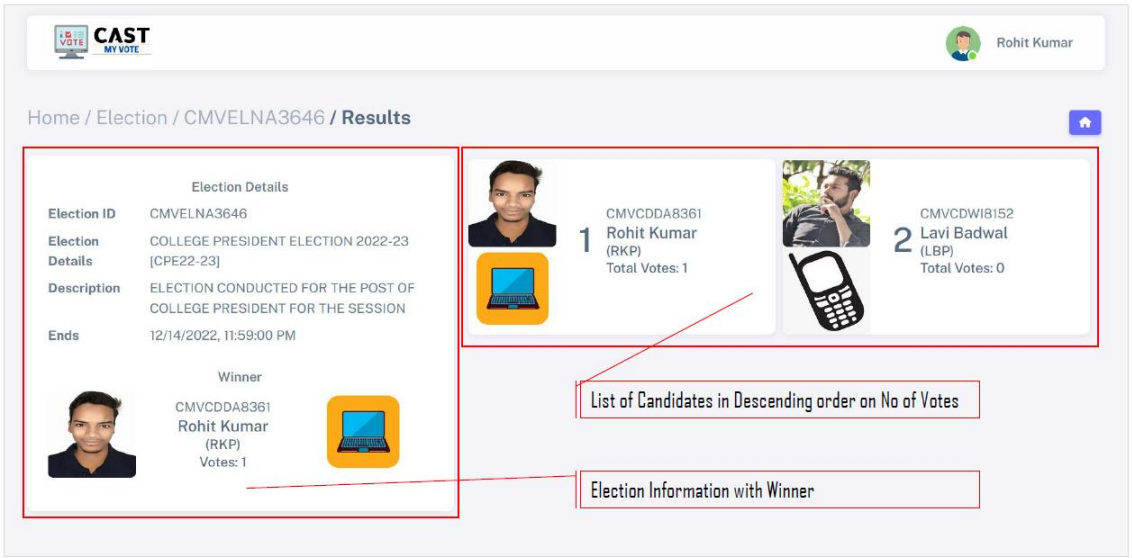

Figure 22: Results Module

Database: All the records whether User Login Records, Voter IDs, or Election Details are stored in MongoDB Atlas (Non-SQL Database)

Authentication: User, as well as Admin Login, can be verified through OTP via Email and SMS and this OTP creates a layer of security in the system OTP is required in various operation in our system

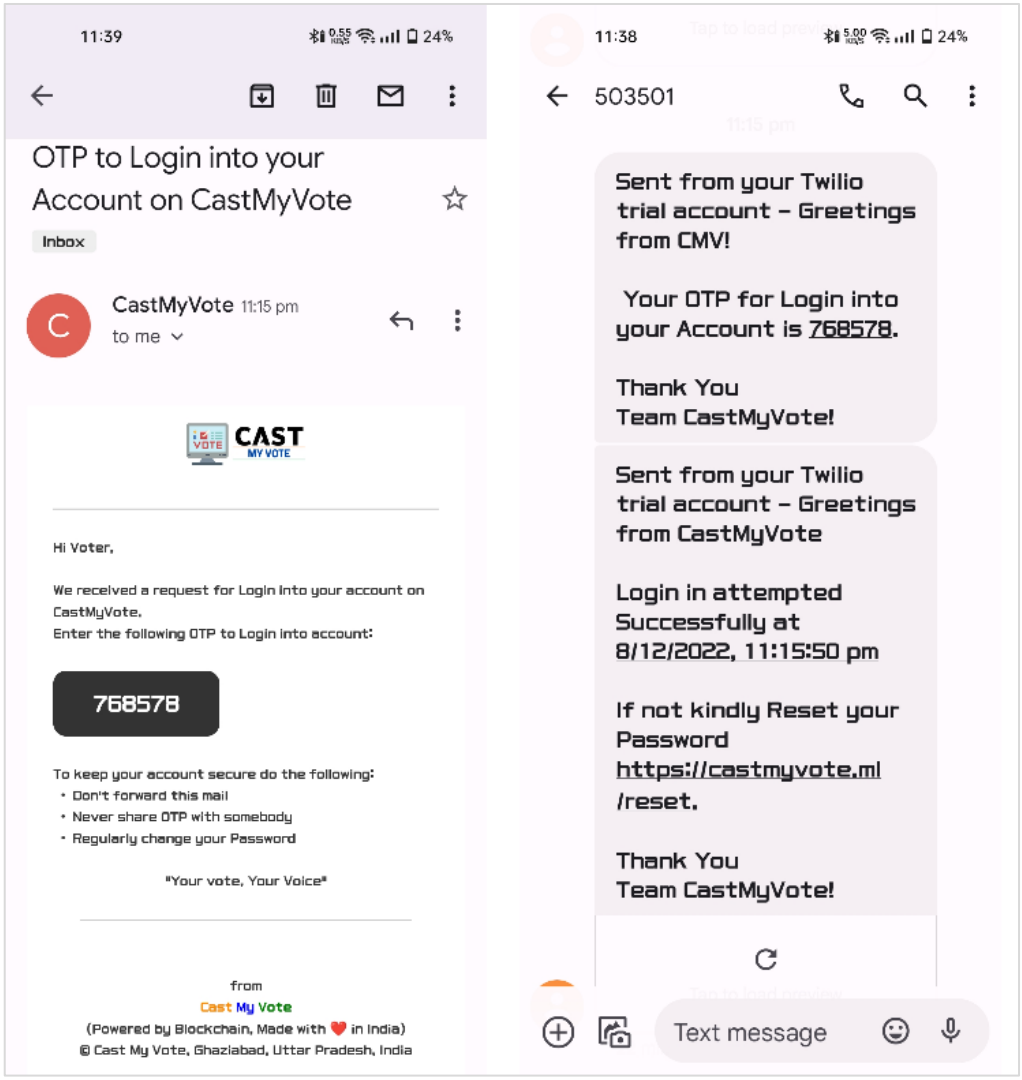

Figure 23: OTP Authentication

Blockchain Network: All the Voting data are stored in Blockchains which are deployed in a decentralized Network.

4. 4.1 Snapshots of Interfaces

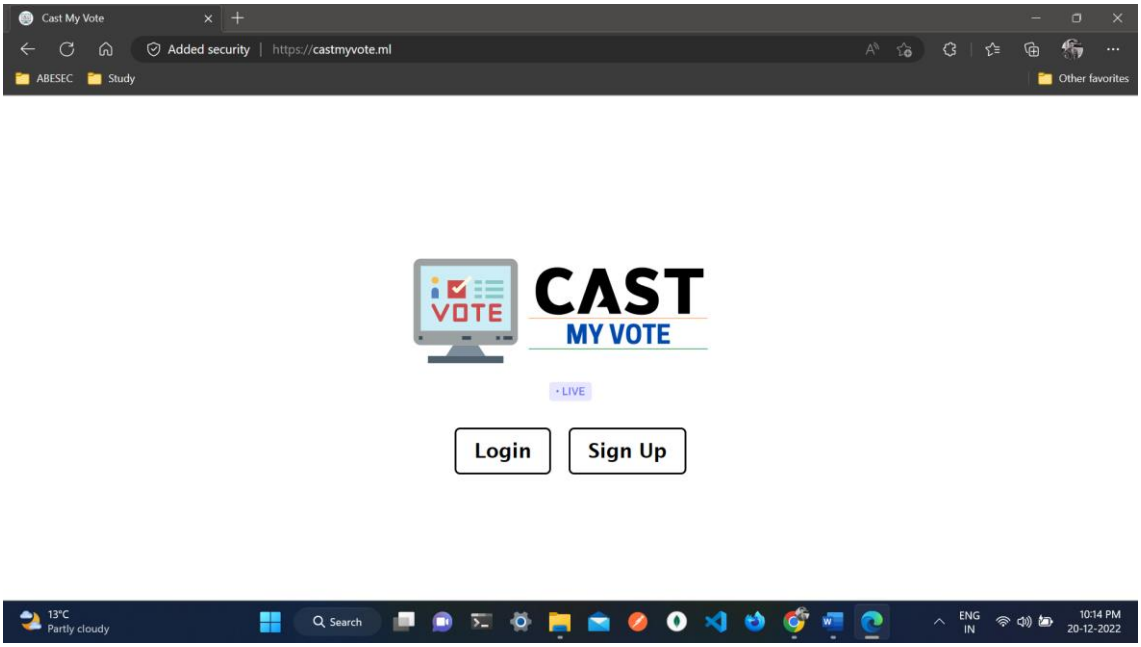

Figure 24: Snapshot for Home Page

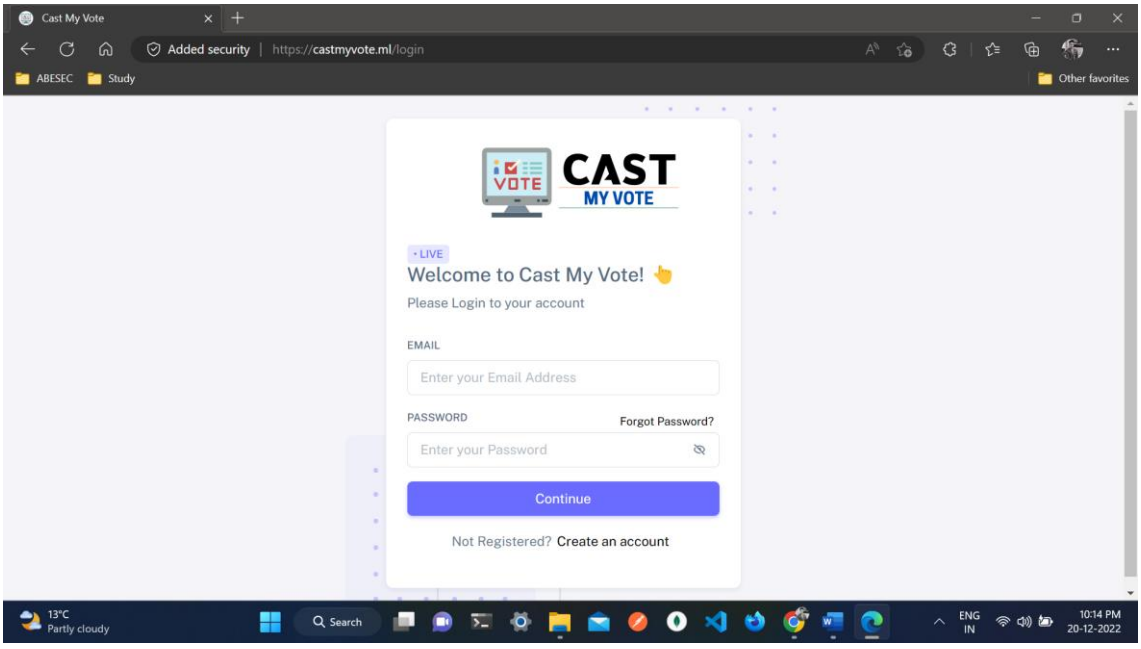

Figure 25: Snapshot for Login Page

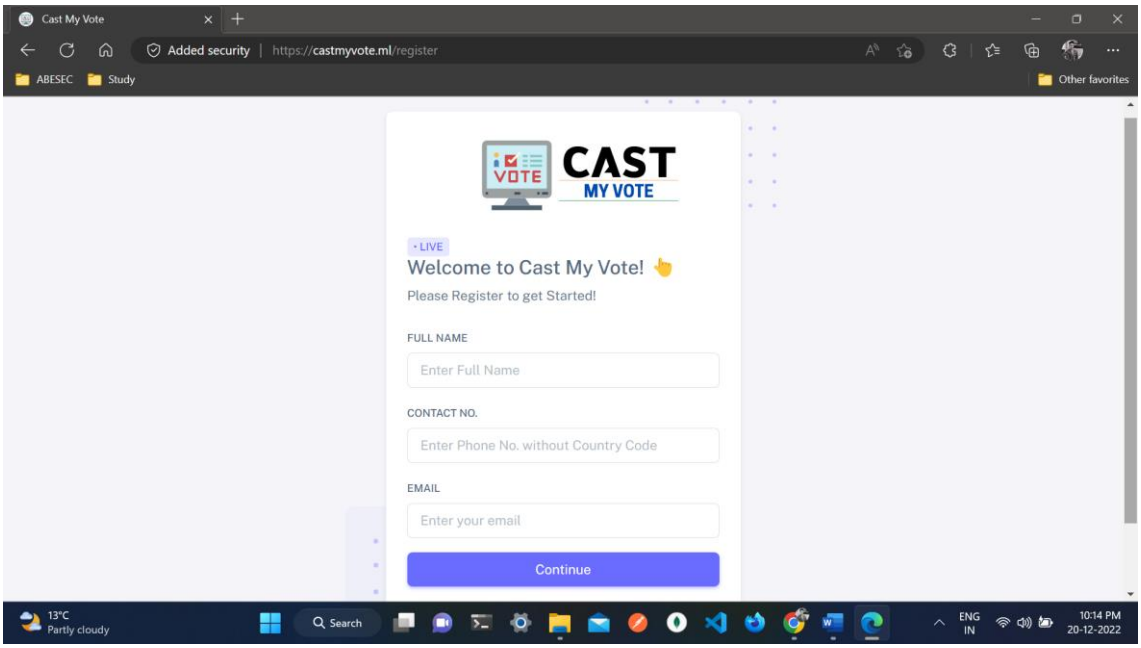

Figure 26: Snapshot for Signup Page for Voter

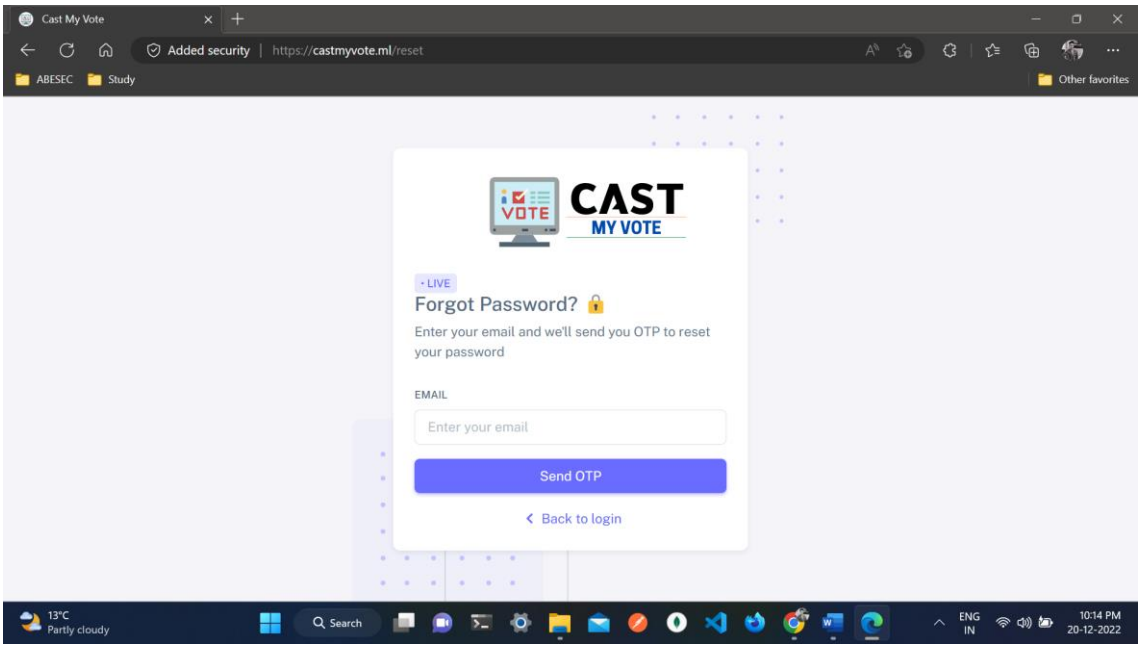

Figure 27: Snapshot for Forget Password Page

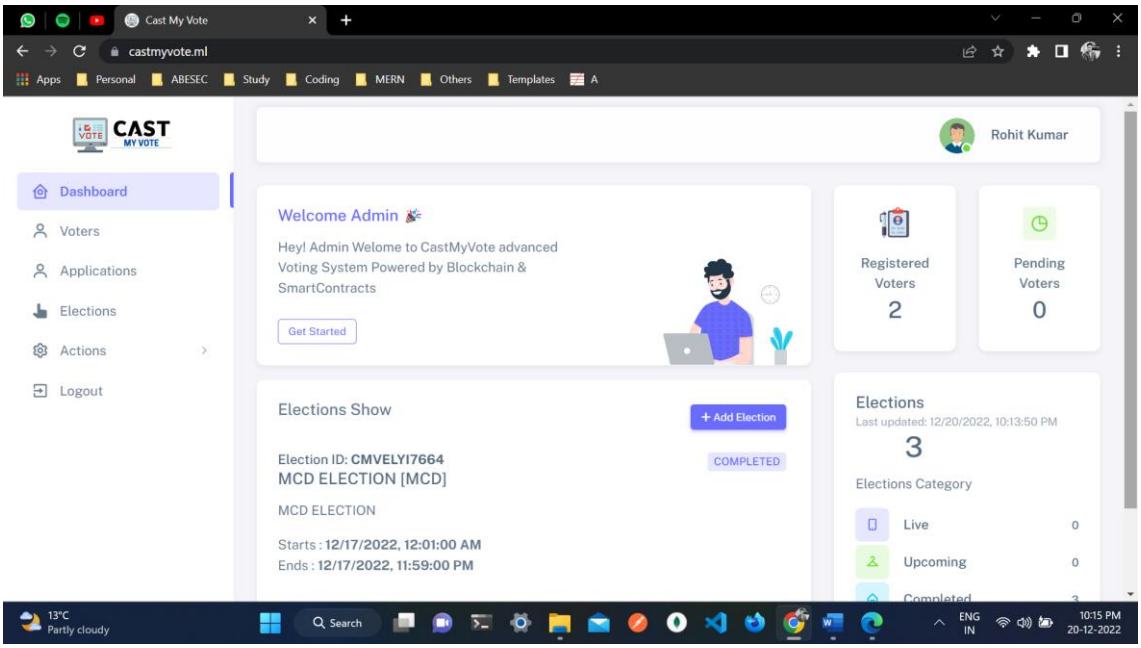

Figure 28: Snapshot for Admin Dashboard

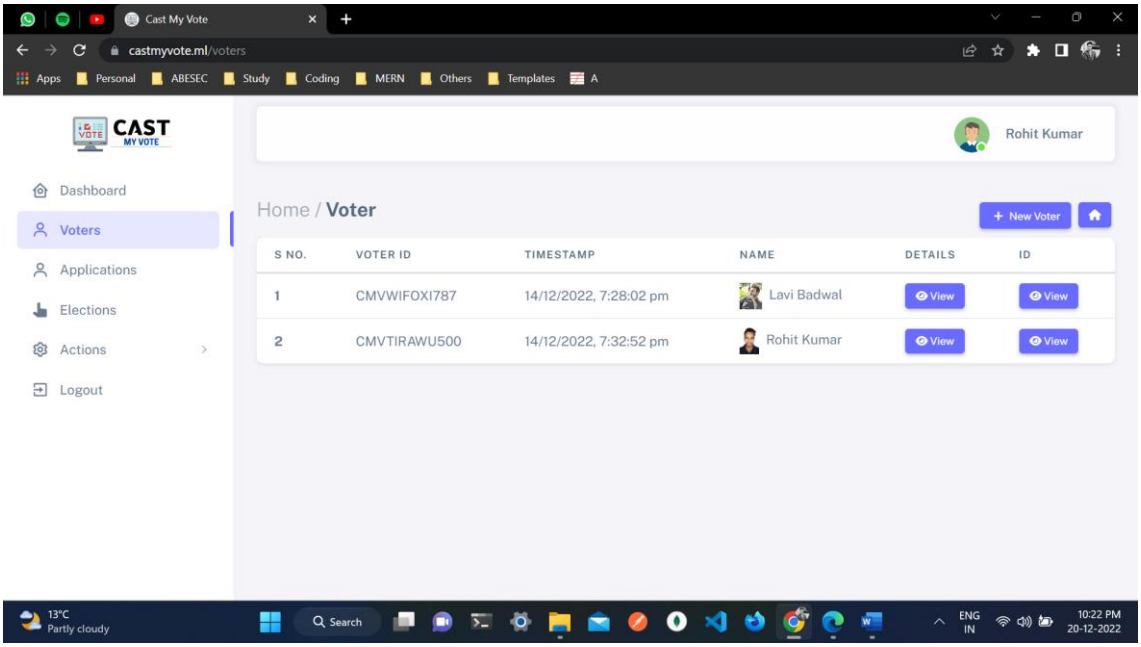

Figure 29: Snapshot for Voter's Record

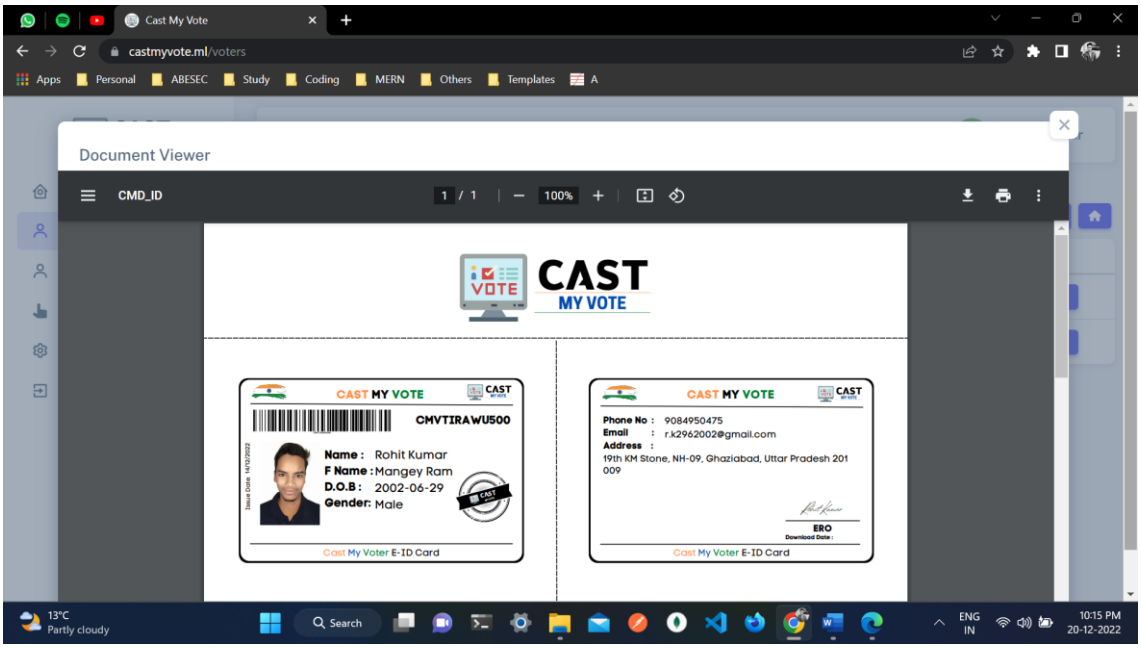

Figure 30: Snapshot for Voter ID Card Popup

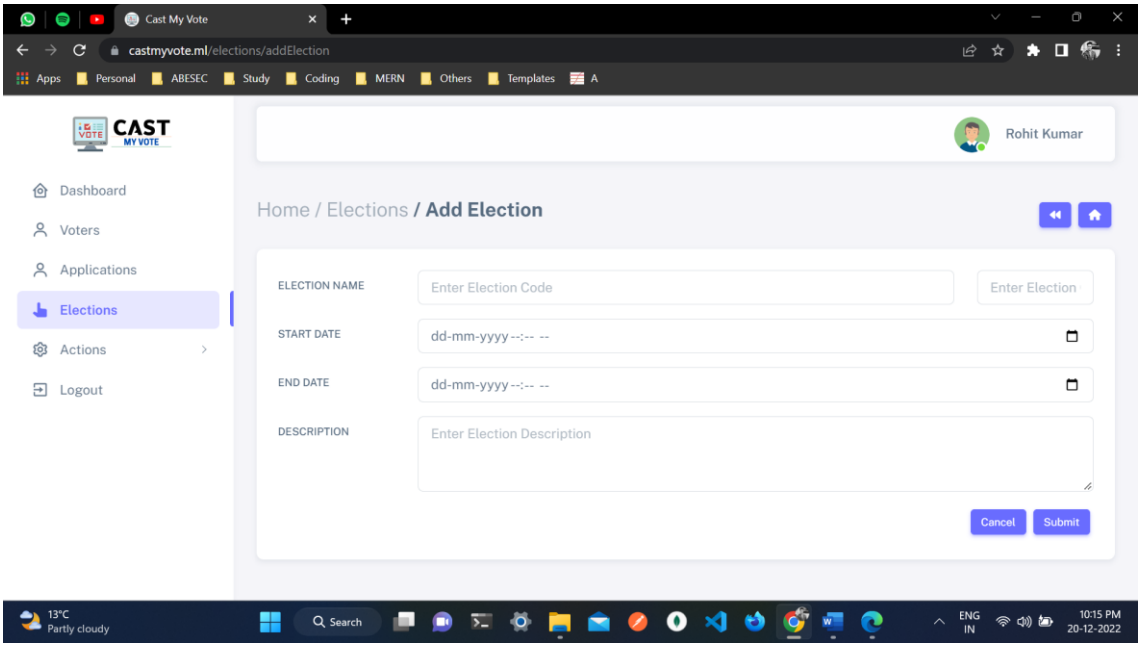

Figure 31: Snapshot for New Election Form Page

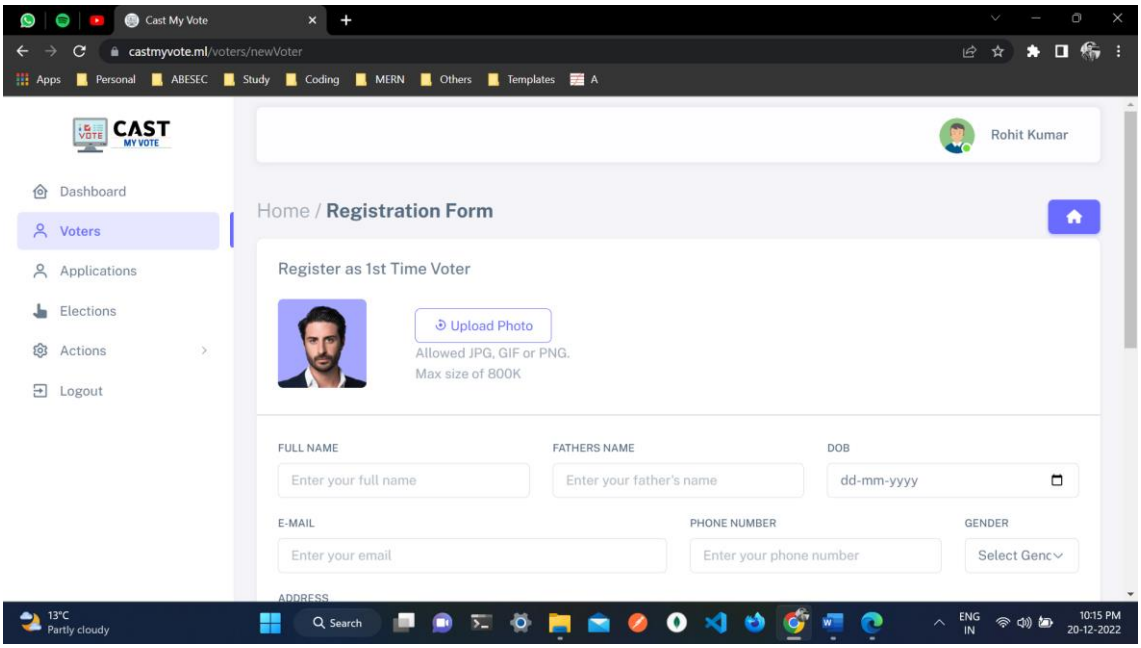

Figure 32: Snapshot for New Voter Form through Admin

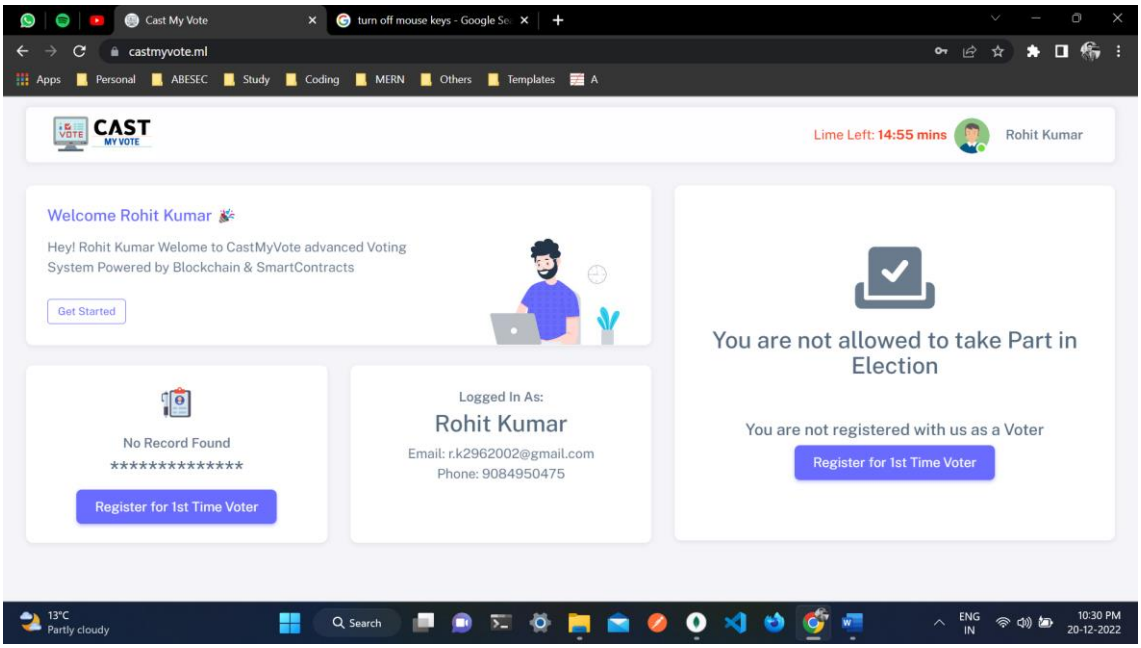

Figure 33: Snapshot for User Dashboard when Signup

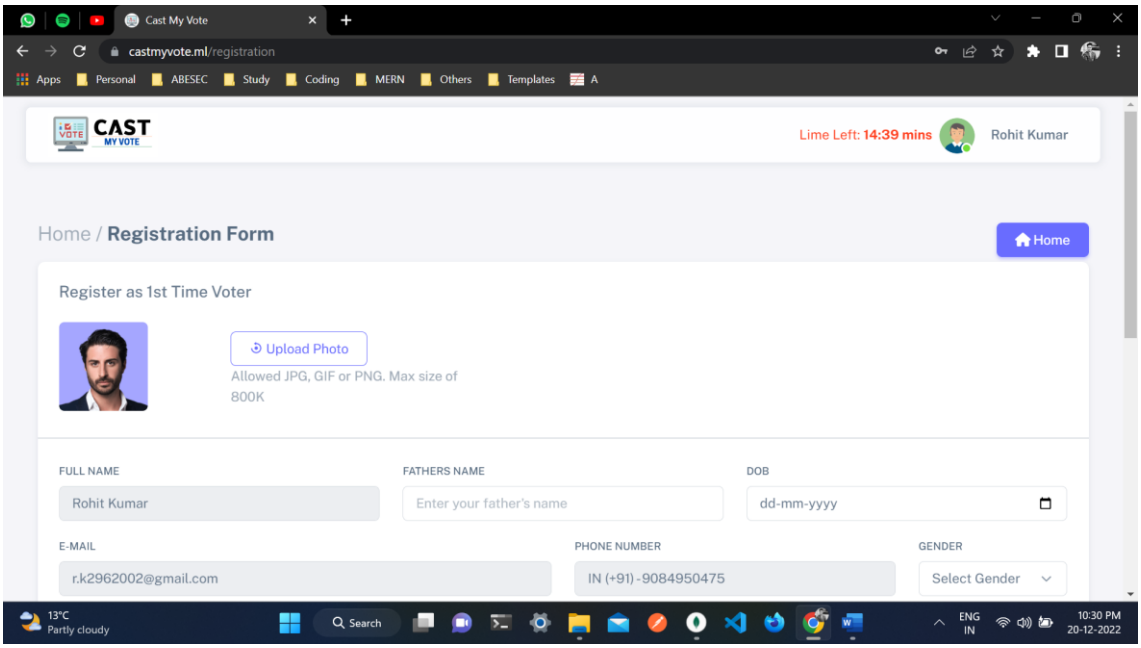

Figure 34: Snapshot for New Voter Enrollment Form

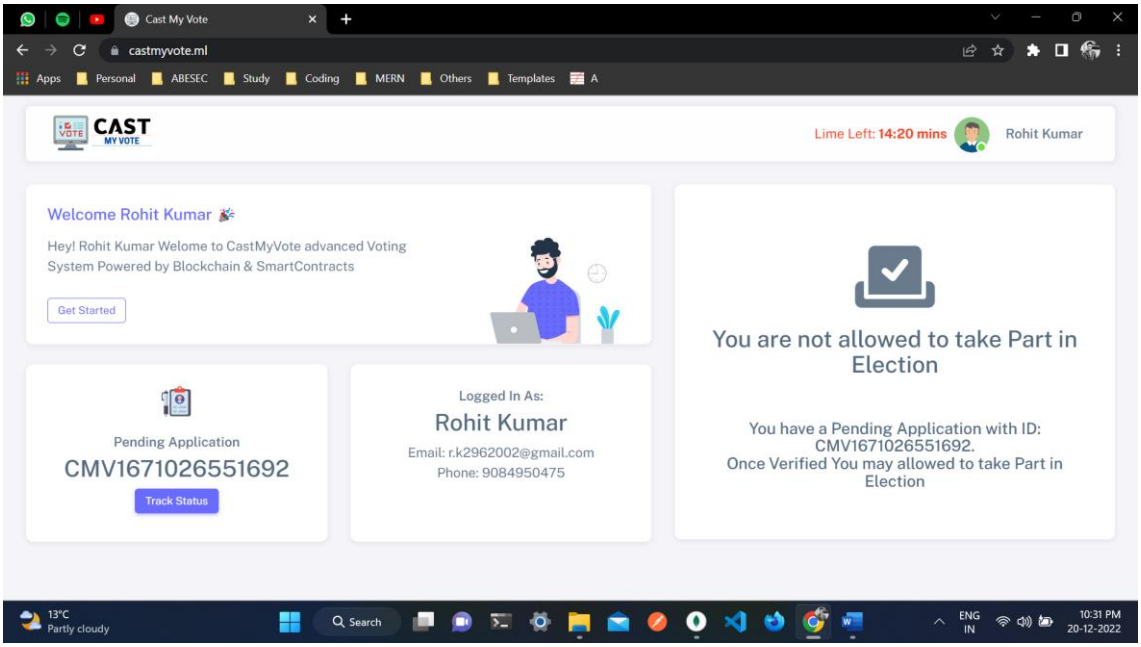

Figure 35: Snapshot for User Dashboard with New Voter Application

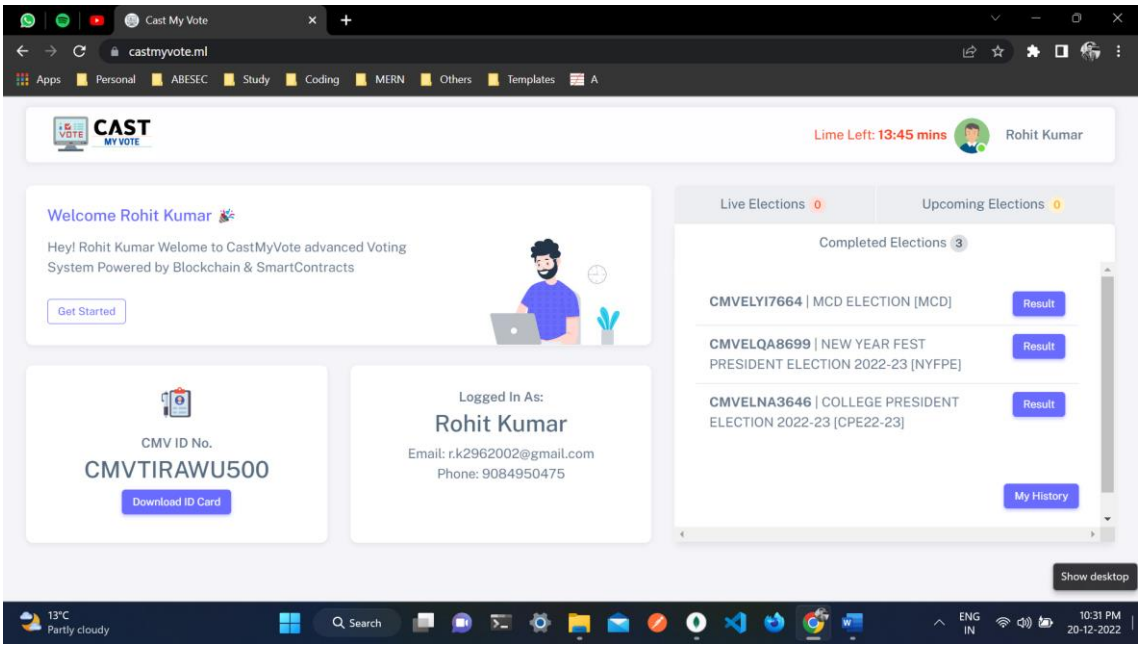

Figure 36: Snapshot for User Dashboard with Voter ID

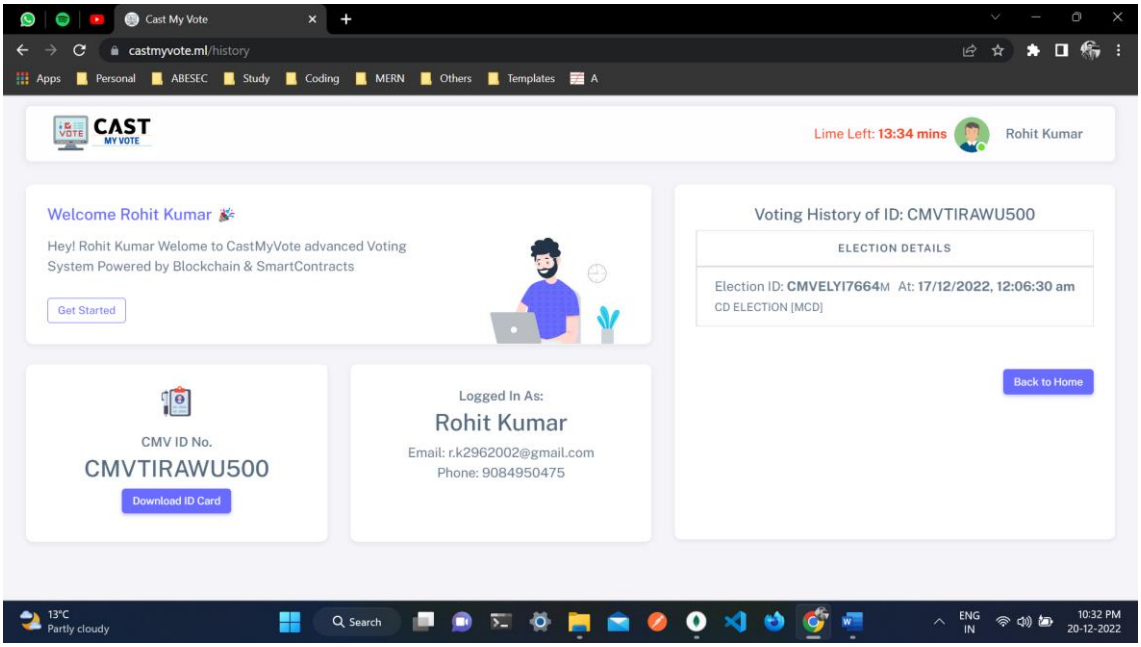

Figure 37: Snapshot for User Voting History

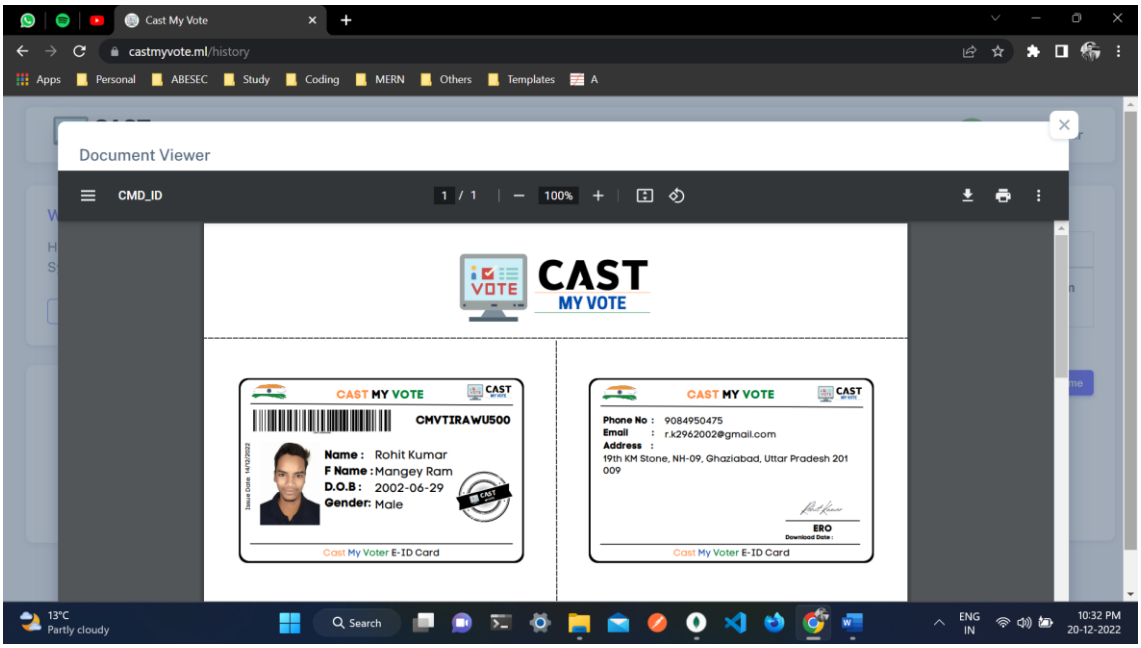

Figure 38: Snapshot for User Voter ID Card

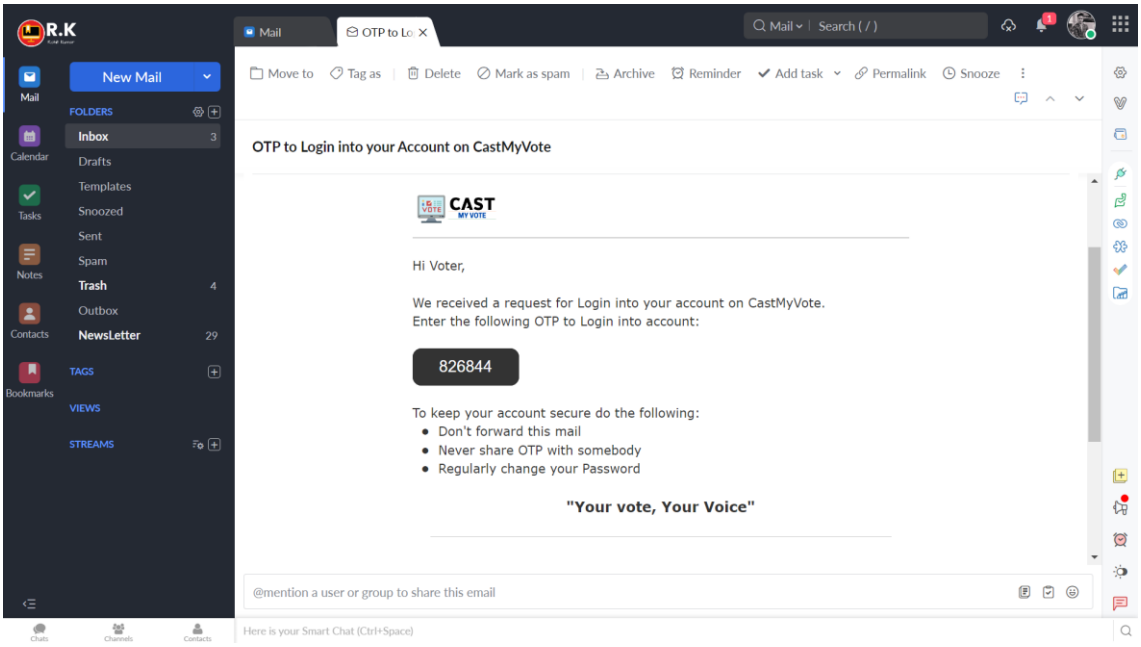

Figure 39: Snapshot for OTP Mail

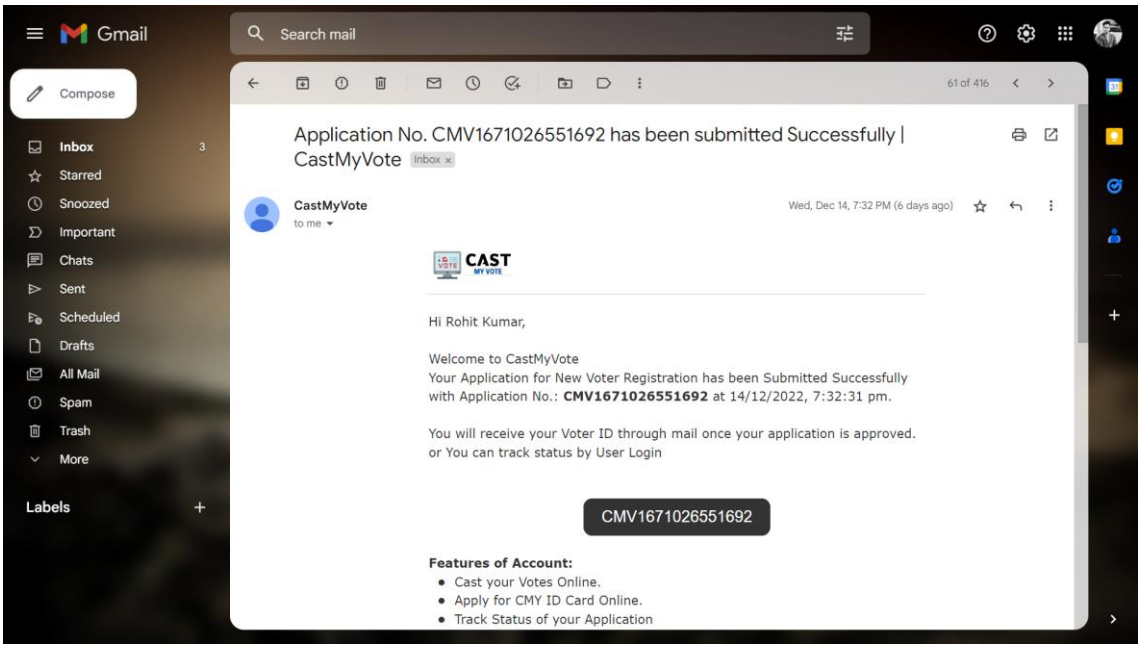

Figure 40: Snapshot for Application Submission Mail

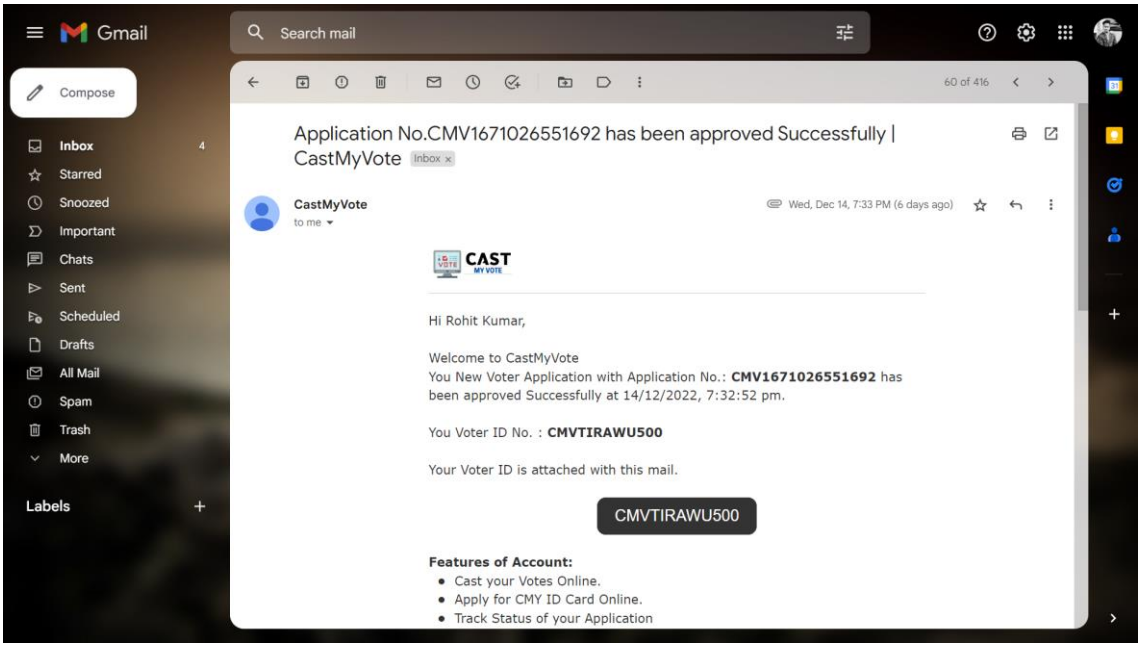

Figure 41: Snapshot for Application Approval Mail (1)

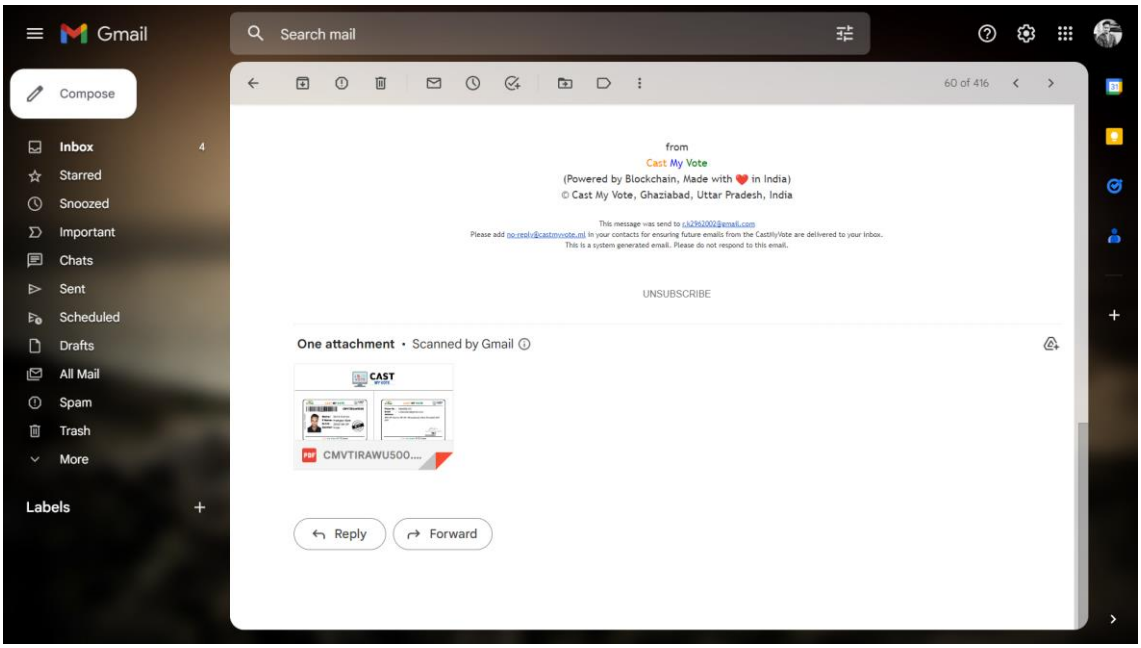

Figure 42: Snapshot for Application Approval Mail (2)

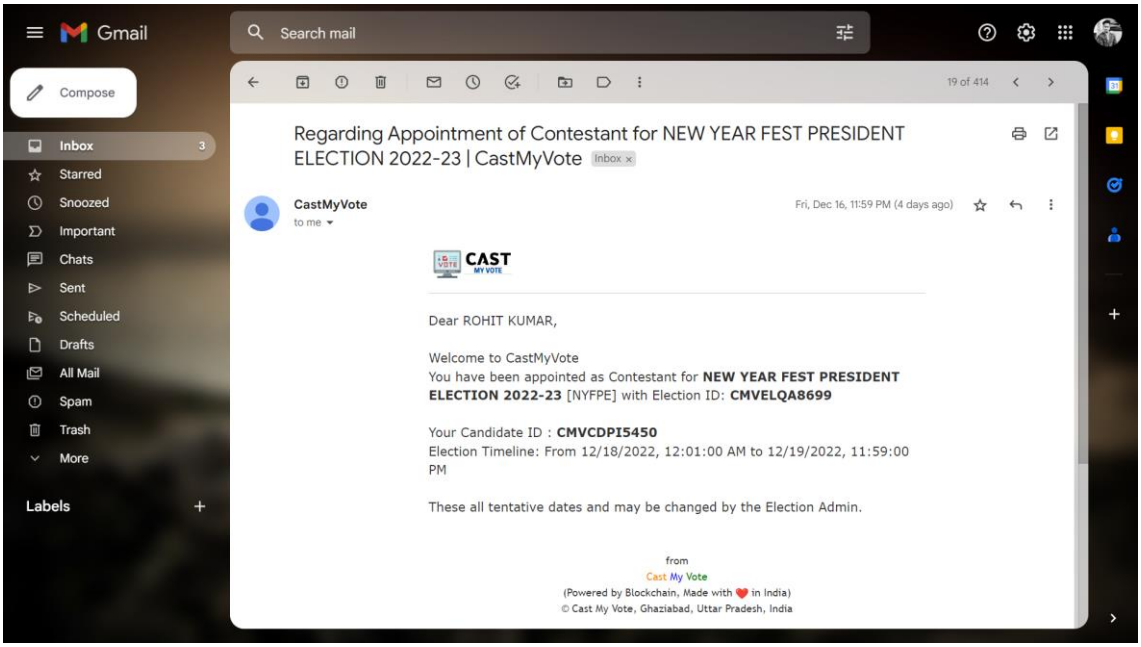

Figure 43: Snapshot for Contestant Appointment Mail

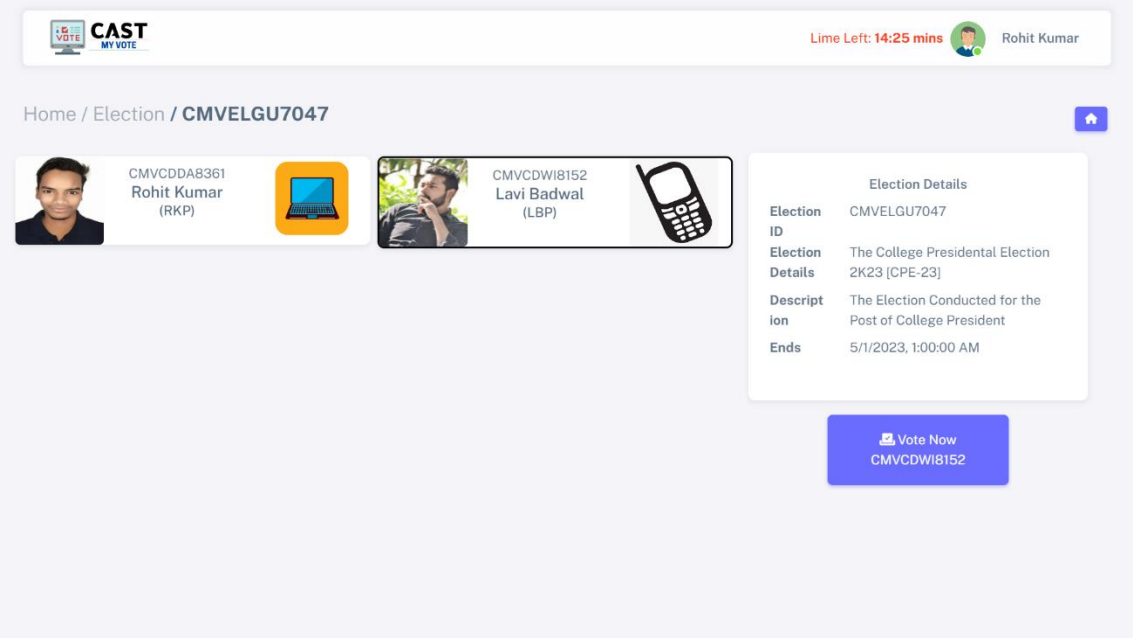

Figure 44: Snapshot for Voter Ballot

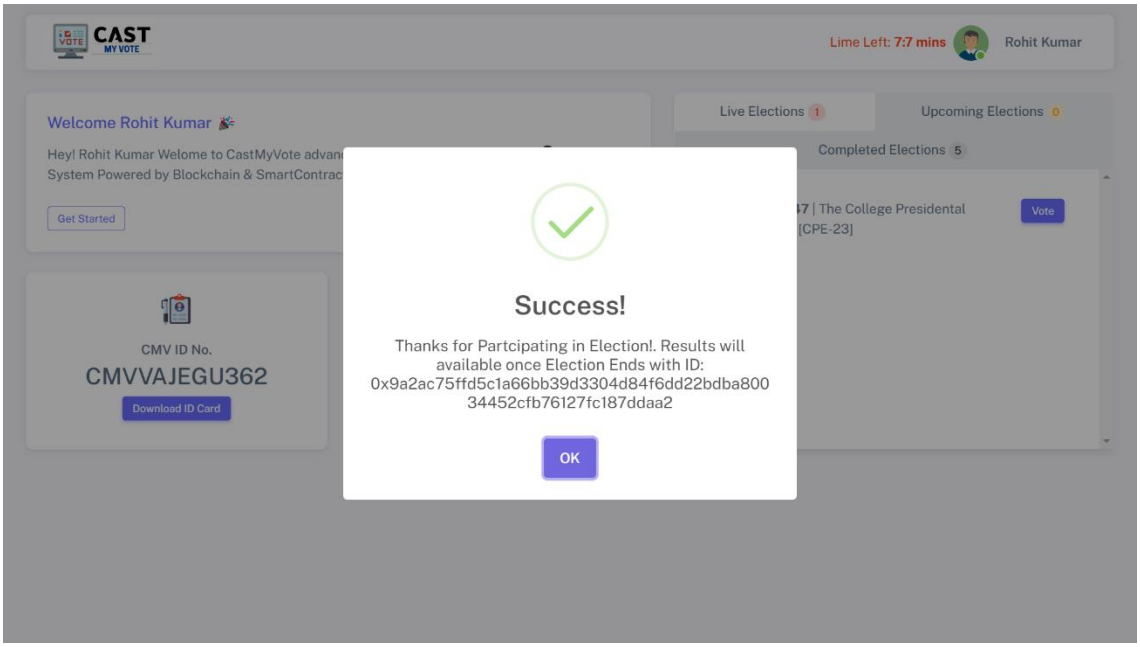

Figure 45: Snapshot for Vote Submission Alert

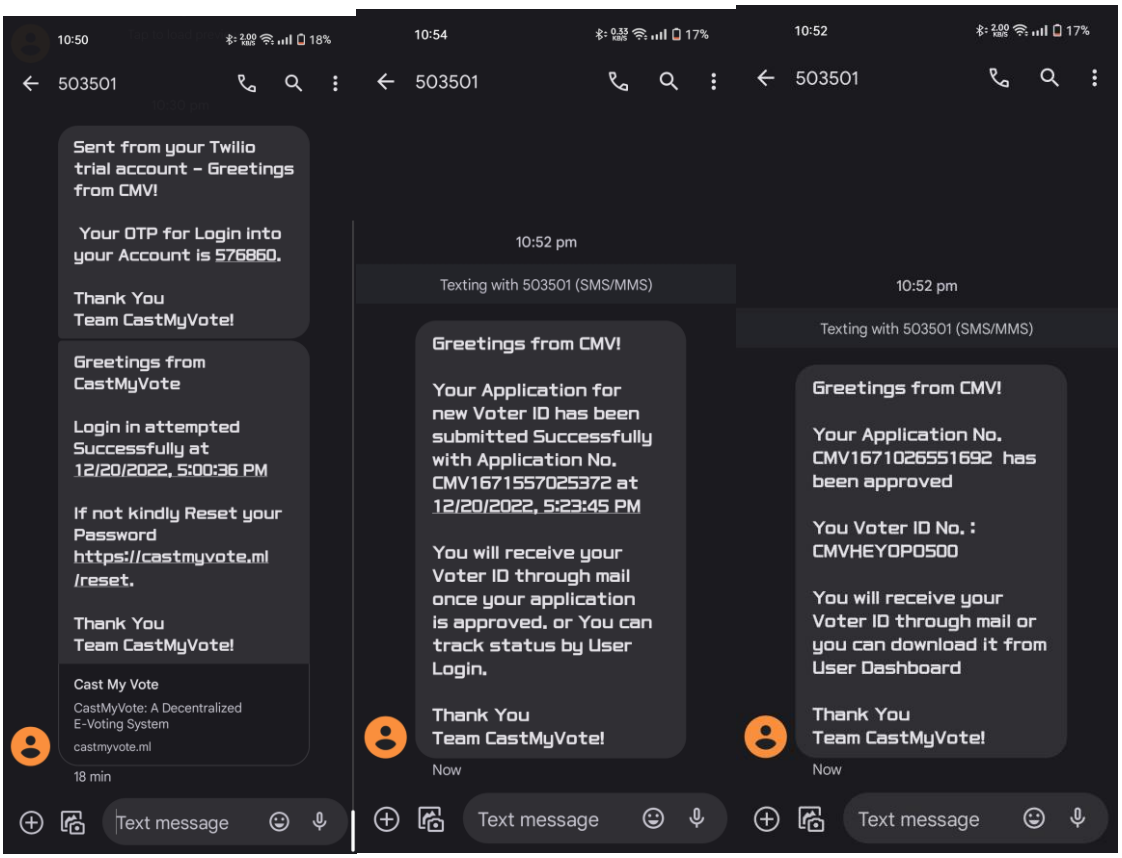

Figure 44: Snapshot of SMS Send to User/Voter

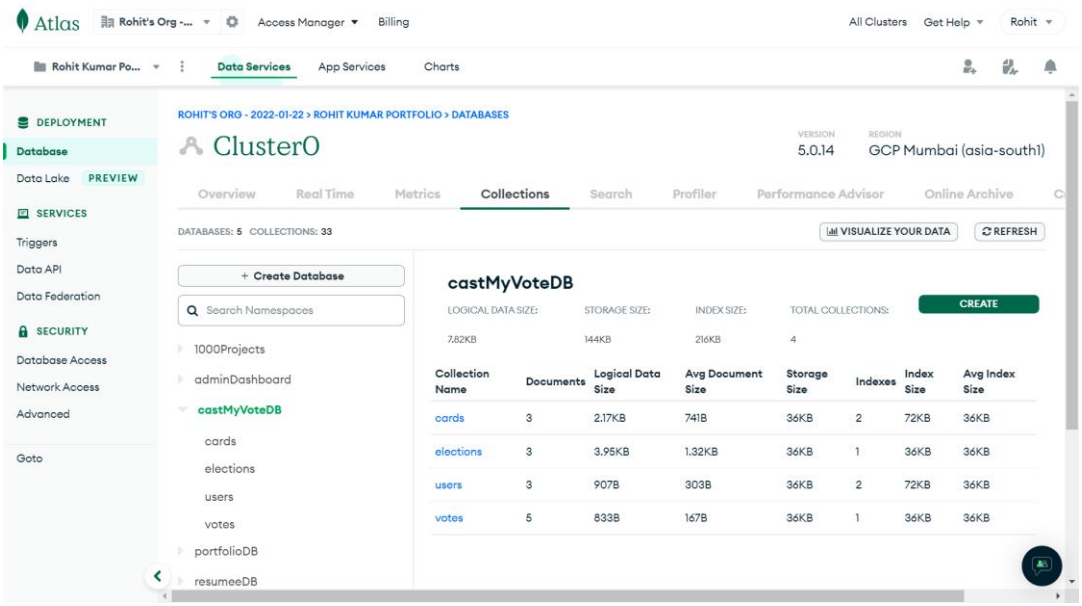

Figure 45: Snapshot for MongoDB Console

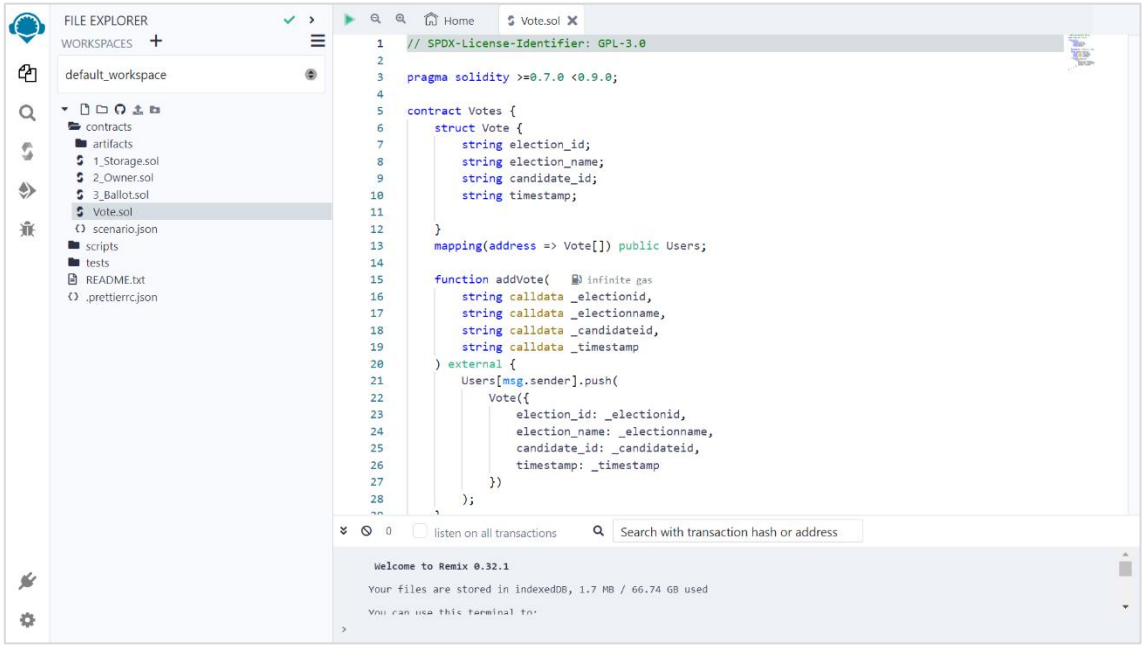

Figure 46: Snapshot for Vote Smart Contract on Remix IDE

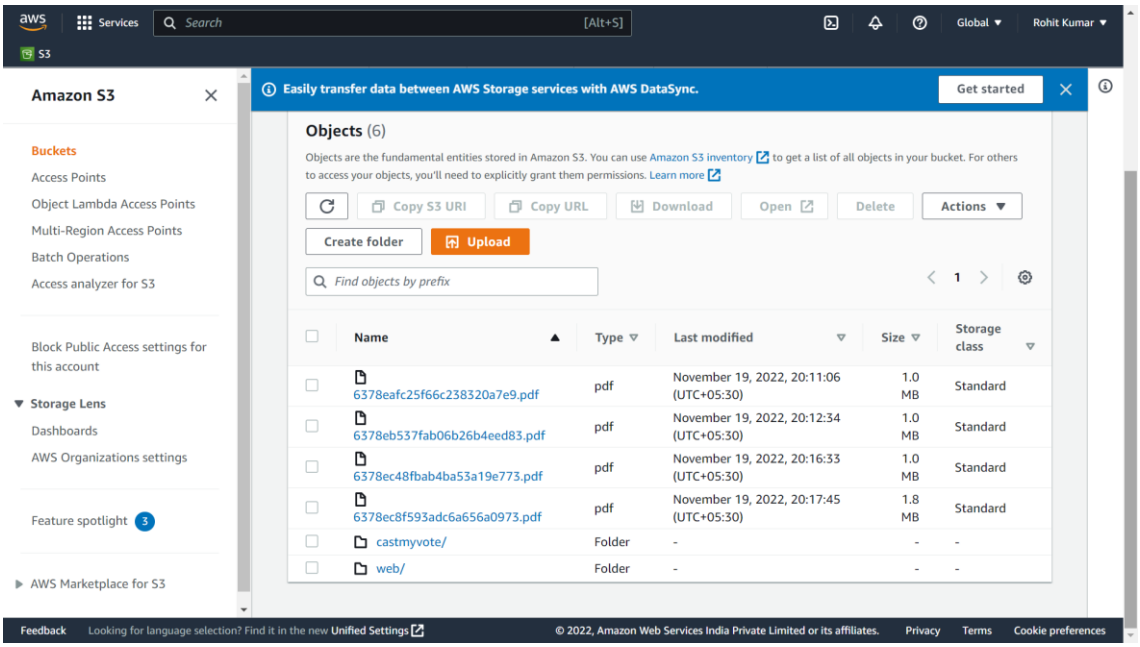

Figure 47: Snapshot for AWS S3 Bucket

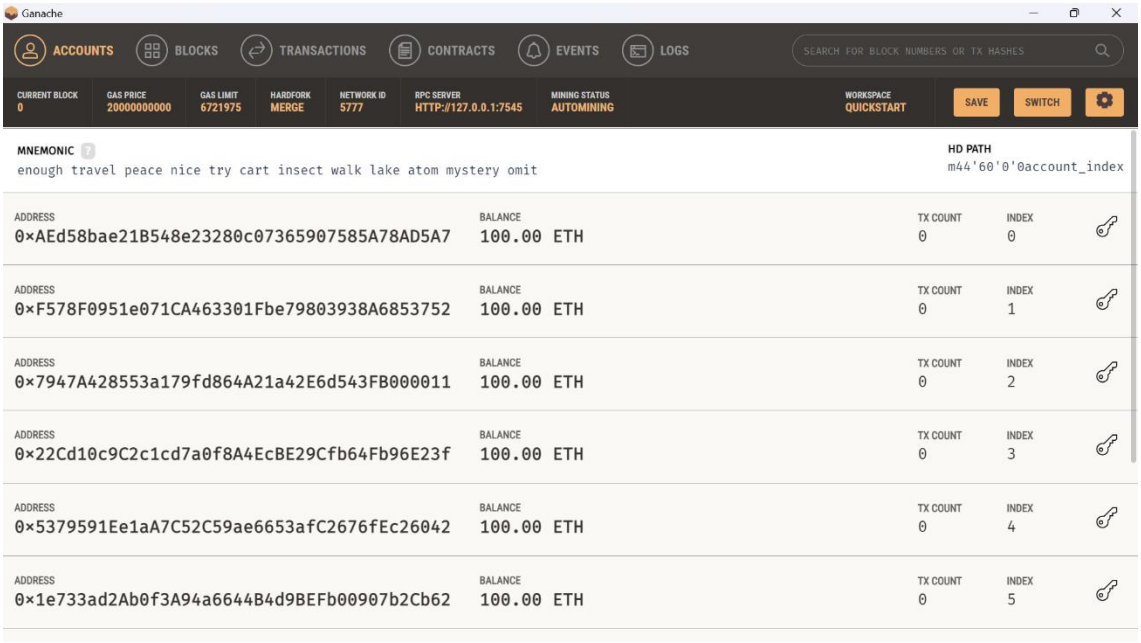

Figure 48: Snapshot for Ganache

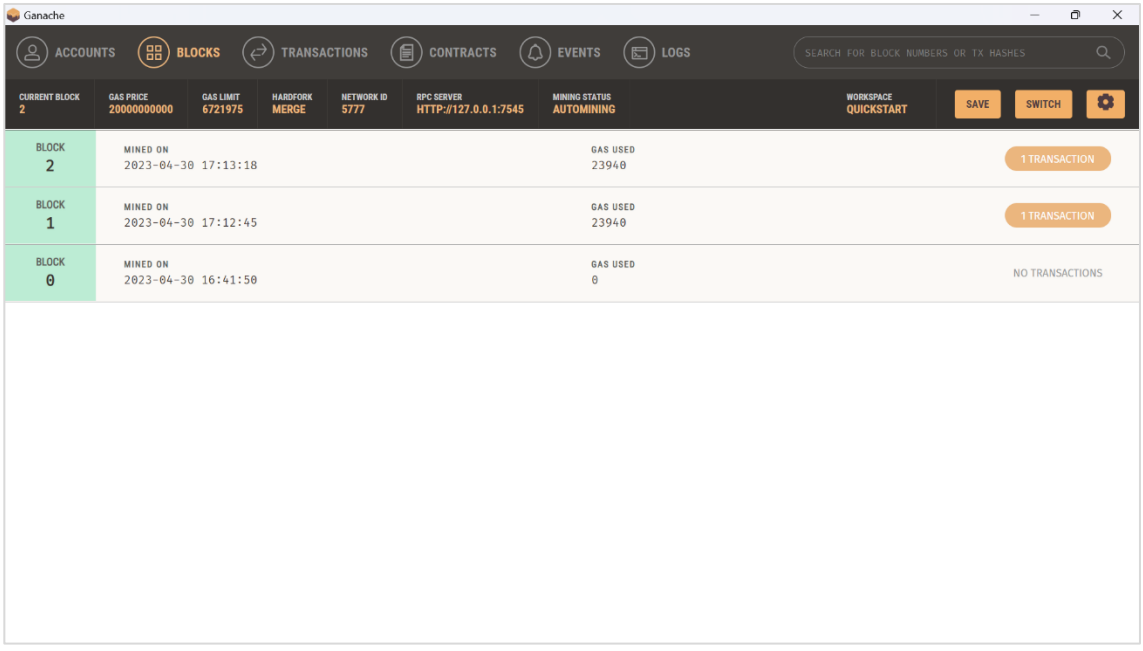

Figure 49: Snapshots for List of Blocks

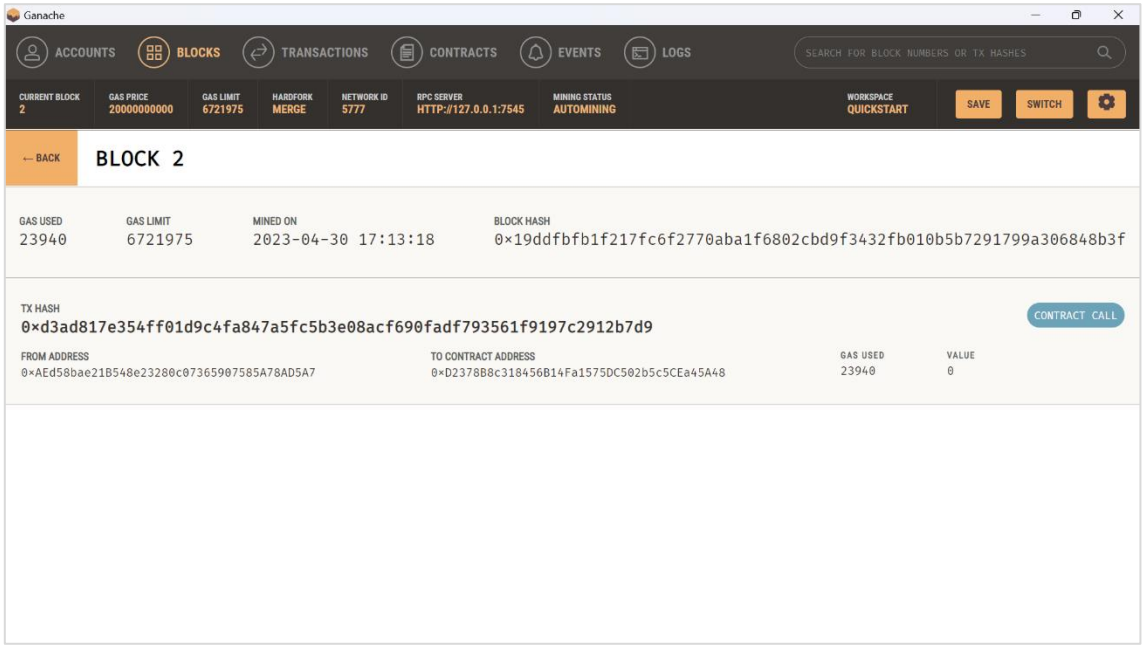

Figure 50: Snapshot for a Blockchain Block

4. 4.2 Test Cases

Each module underwent unit testing upon successful completion. Prior to being integrated with the entire system, each module was evaluated independently. Integration testing was carried out to ensure that each module was properly integrated with the system after integration.

Black-box testing of these entire system was done after all integrations were finished to make sure it functions properly. Black box testing of the system's primary functions

1. Login & Signup Process
   - Case (i) Voter entered unregistered email in login form.
     The error message “User not Found” appears on the screen as Output.
   - Case(ii) User entered an Incorrect OTP
     The error message “Invalid OTP” displayed on the screen.
   - Case (iii) Voter is registering through already registered Email Address
     The error message “User Already Exists!” displayed on the screen.

2. Voter Section

   Case (i) The voter has no voter ID

   The System shows error that you are not registered and also shows link for voter registration.

   Case (ii) The voter has no voter ID but have a pending application

   The System shows that user have a pending application with application no shown on the screen and also shows status on that.

   Case (iii) Voter try to submit vote again.

   The System will show alert that you have already voted and prevent duplicate vote.

3. Admin Section

   Case (i) The admin tries to change running or completed election.

   The system did not permit to change election once it goes live or completed.

   Case (ii) Election Table Buttons

   The System will show equivalent button with respect to election state.

4. 4.3 Results

At the end the system will working fine and we have successfully hosted an election that it all works fine.

CHAPTER 5

CONCLUSION

In this project report, a Blockchain-powered E-Voting system is discussed that users/voters can use to cast their votes without going to the polling booth. An innovative electronic voting system based on blockchain technology is described in detail that ensures voter privacy while facilitating secure and economical elections. After going through a detailed study on recent Blockchain Technology, it is concluded that Blockchain offers democratic countries a new opportunity to replace the old means like EVM and Ballot system with this E-Voting system that is cost- and time-effective, while also enhancing the security features of the existing system and presenting new opportunities for transparency. This research paper presents a blockchain-based online voting framework based on MERN that uses smart contracts to keep voter privacy intact while keeping the process of elections safe and affordable. In addition to current blockchain-based voting systems, the blockchain offers the ability to significantly improve electronic voting as well as prospective future research avenues. Many experts feel that blockchain might be an excellent fit for a decentralized electronic voting system.

5.1. Performance Evaluation

This Project works fine as per discussed in this report and some evaluation is as below:

iii. Increase User Satisfaction with Best Features and Best UI Interface.

iv. Best Loading Speed and work on low Bandwidth.

v. We have data backup we any data is deleted we can recover the data

vi. It is available 24*7

5.2. Comparison with existing State-of-the-Art Technologies

This project is best because it used modern technologies like ReactJS. NodeJS which is in current trend and does not require additional treatment and these technologies are highly secure because it use 3-tier model.

5.3. Future Directions

This project is extended to higher level if we have a time and budget in the following ways:

iv. AI Implementation during Voting Procedure.

v. Candidates Enrollment through it.

vi. To make this system Highly scalable so that a large-scale election can be conducted on that.

APPENDIX – A

CODE SAMPLES

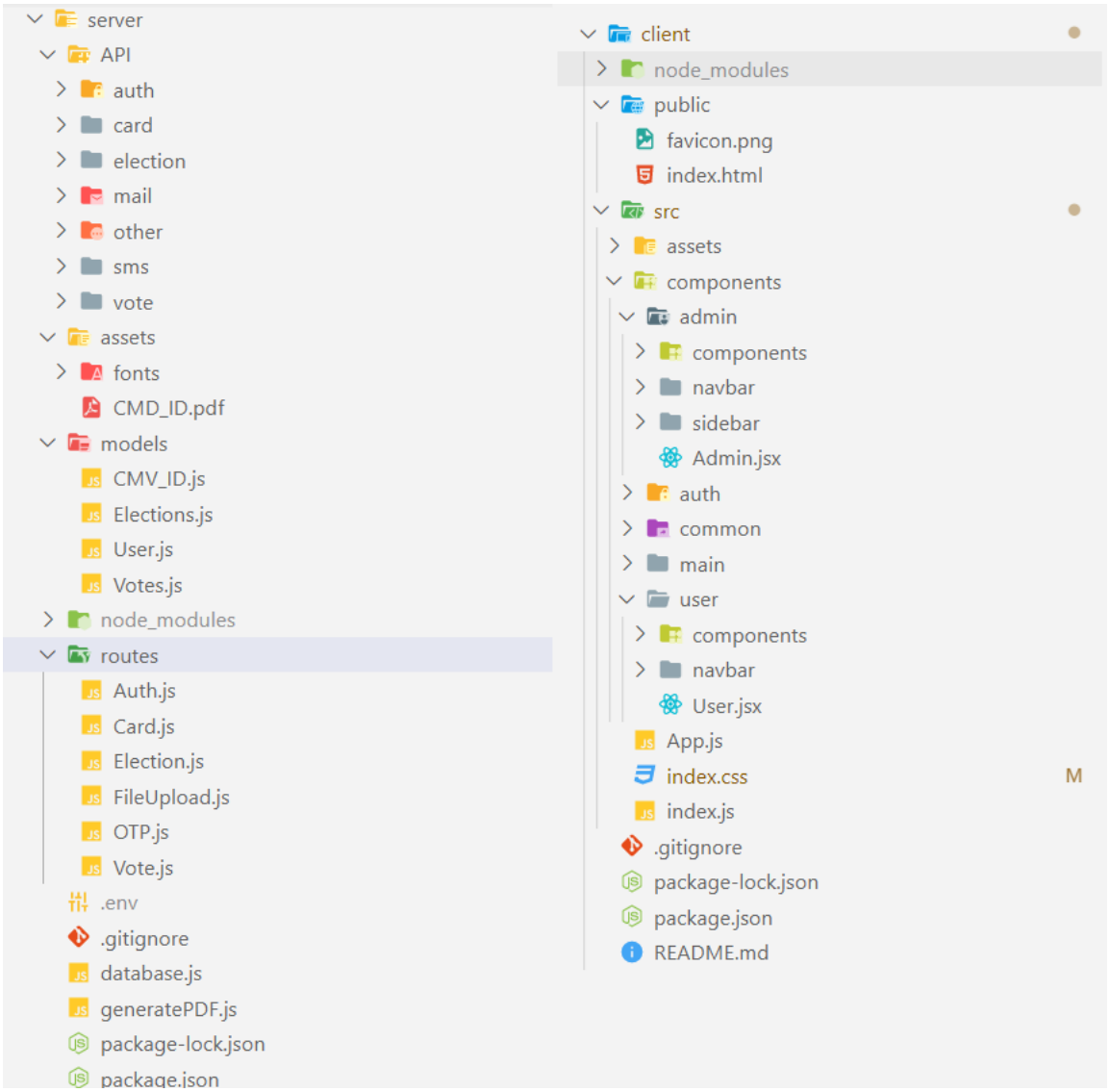

Figure 51: Directory Structure of Our System

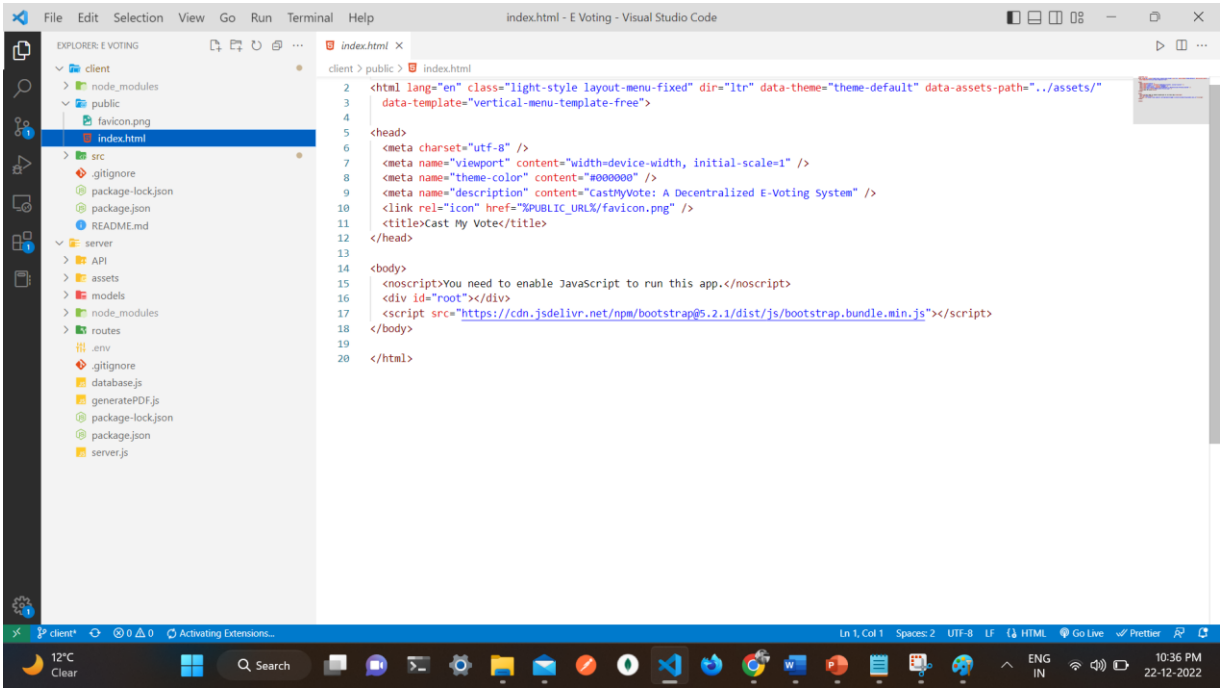

Figure 52: Snapshot for our main .html file

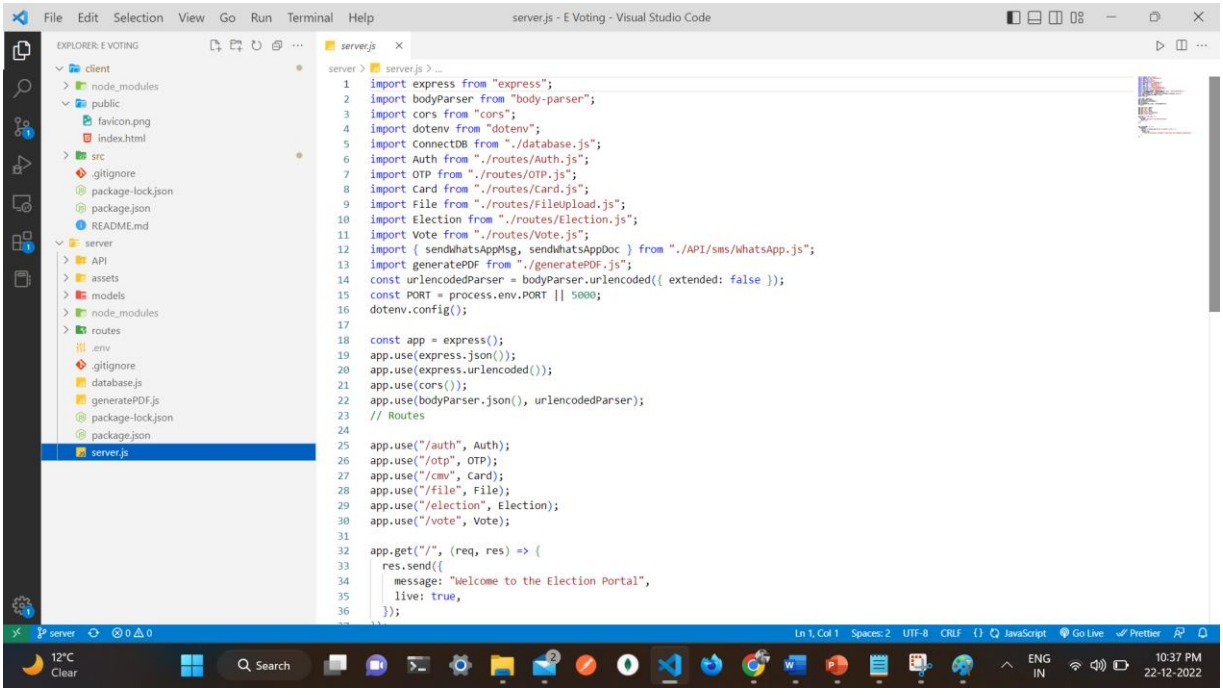

Figure 53: Snapshot for Our Server.js File

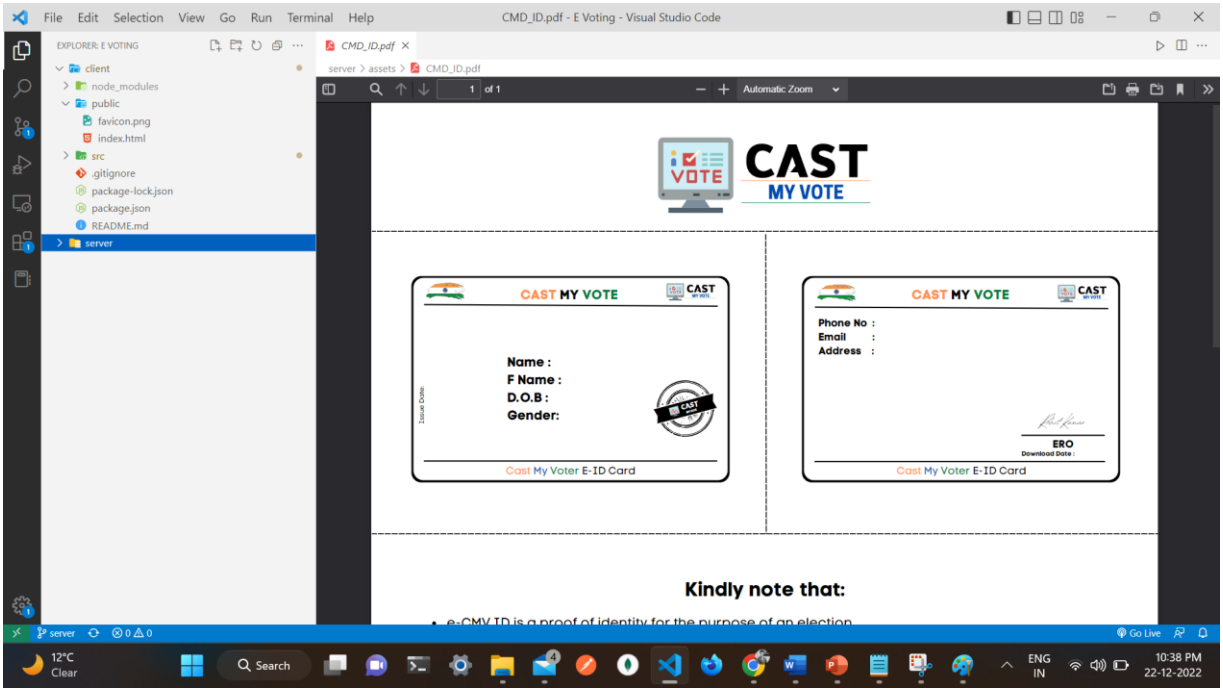

Figure 54: Snapshot for Our Empty Voter ID Card

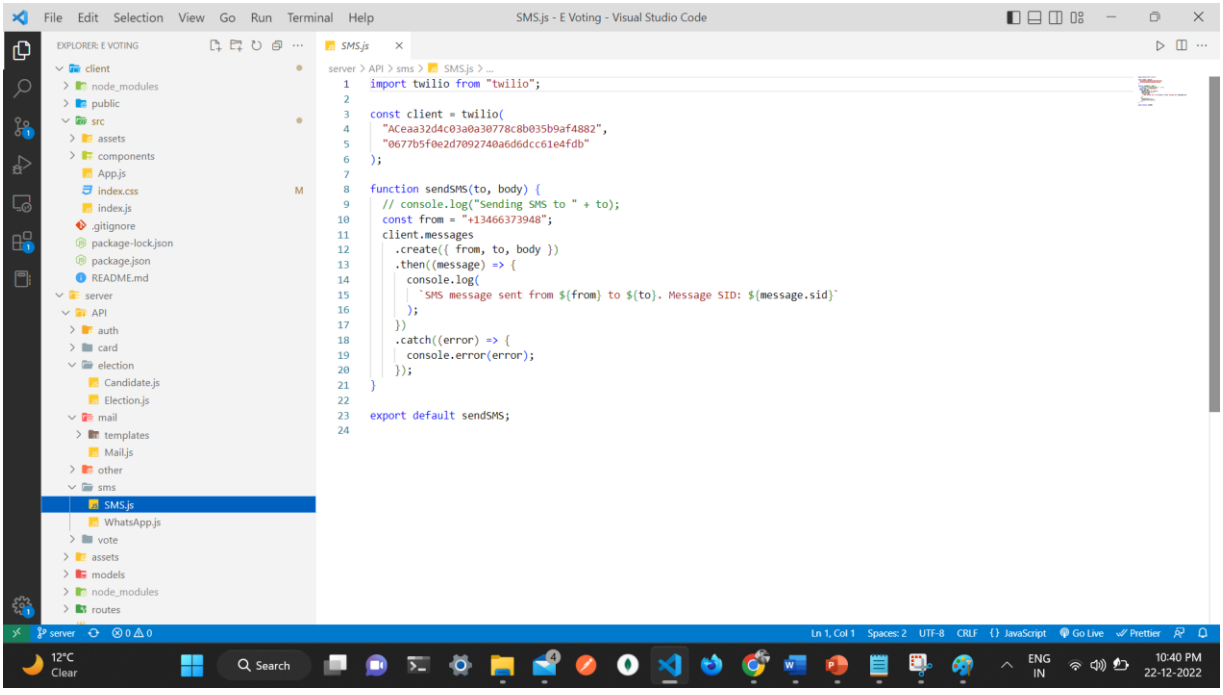

Figure 55: Snapshot for SMS Sender Function

Figure 56: Snapshot for Mail Sender Function

Figure 57: Snapshot for User and Admin Components

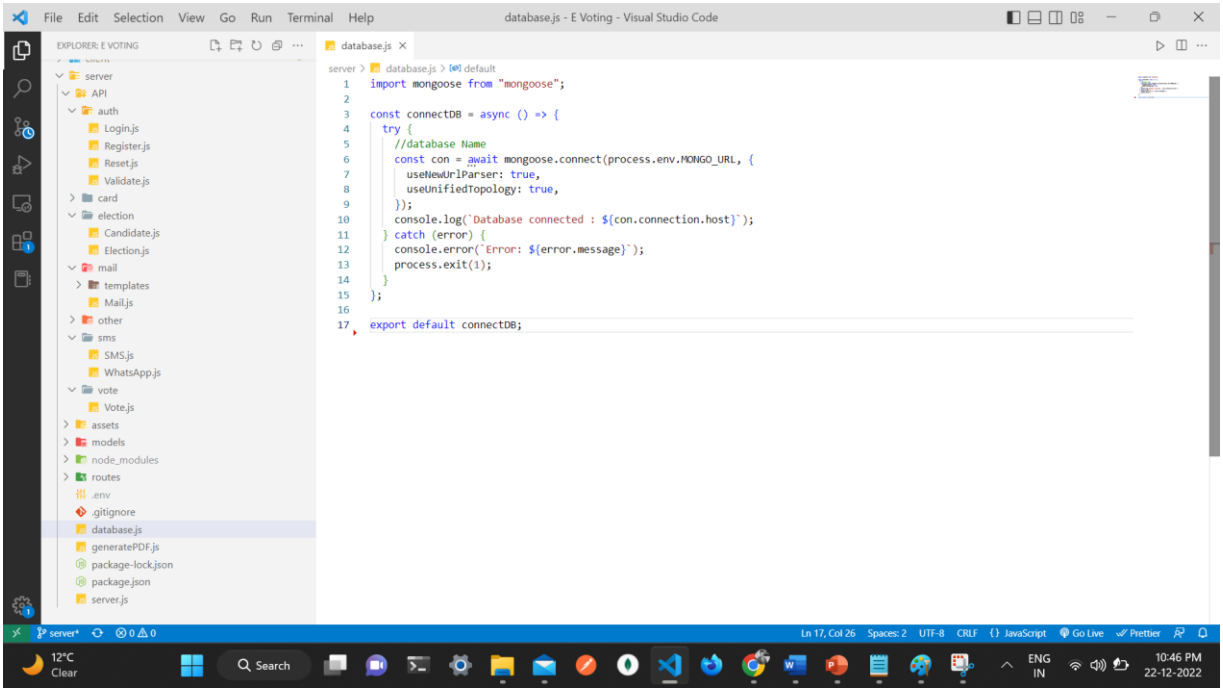

Figure 58: Snapshot for Database Connection File

REFERENCES

(i) R. Kumar, L. Badwal, S. Avasthi and A. Prakash, "A Secure Decentralized E-Voting with Blockchain & Smart Contracts," 2023 13th International Conference on Cloud Computing, Data Science & Engineering (Confluence), Noida, India, 2023, pp. 419-424, doi: 10.1109/Confluence56041.2023.10048871.

(ii) Yadav, Abhishek. (2020). E-Voting using Blockchain Technology. International Journal of Engineering Research and. V9. 10.17577/IJERTV9IS070183.

(iii) Jafar, U.; Aziz, M.J.A.; Shukur, Z. Blockchain for Electronic Voting System—Review and Open Research Challenges. Sensors 2021, 21, 5874. *https://doi.org/10.3390/ s21175874*.

(iv) Benny, Albin, Blockchain based E-voting System (July 11, 2020). Available at SRN: https://ssrn.com/abstract=3648870 or http://dx.doi.org/10.2139/ssrn.3648870.

(v) S. Nakamoto, “Bitcoin: a peer-to-peer electronic cash system”, [Online]. Available: https://bitcoin.org/bitcoin.pdf

(vi) Nir Kshetri, Jeffrey Voas, “Blockchain-Enabled E-Voting”.

(vii) Ali Kaan Koç, Emre Yavuz, Umut Can Çabuk, Gökhan Dalkılıç “Towards Secure E-Voting Using Ethereum Blockchain”.

(viii) E. Maaten, “Towards remote e-voting: Estonian case”, Electronic Voting in Europe-Technology, Law, Politics and Society, vol. 47, pp. 83-100, 2004.