# E-VOTING SYSTEM USING BLOCKCHAIN

**A Project Report**
Submitted in partial fulfillment of the
Requirements for the award of the Degree of

**BACHELOR OF SCIENCE (COMPUTER SCIENCE)**

**By**

**Santosh Vishwakarma**
**545**

**Under the esteemed guidance of**

**Mrs. Elizabeth Leah George**

**Assistant Professor**


**NAGINDAS KHANDWALA COLLEGE(Autonomous)**

***(Affiliated to University of Mumbai)***

**MUMBAI, 400 064**

**MAHARASHTRA**

**2022-23**

**(Original Copy of the Approved Proforma of the Project Proposal)**

***(Note: All entries of the proforma of approval should be filled up with appropriate and* complete information. Incomplete proforma of approval in any respect will be summarily rejected.)**

PRN No**.:** **2018016401126116** Roll no**:** **545**

1. Name of the Student: **Santosh Vishwakarma**
2. Title of the Project: **E-Voting System Using Blockchain**
3. Name of the Guide: **Mrs. Elizabeth Leah George**
4. Teaching/Industry experience of the Guide: **7 Years**
5. Is this your first submission? **Yes**

Signature of the Student: Signature of the Guide:

Date: …… Date: …….

Signature of the Coordinator:

Date: ……

**NAGINDAS KHANDWALA COLLEGE(Autonomous)**

*(Affiliated to University of Mumbai)*

**MUMBAI, 400 064**

**MAHARASHTRA**

**DEPARTMENT OF COMPUTER AND INFORMATION SCIENCE**

# CERTIFICATE

This is to certify that the project titled, **"E-Voting System Using Blockchain"**, is bonafied work of **Santosh Vishwakarma** bearing Seat.No: **545** submitted in partial fulfillment of the requirements for the award of degree of BACHELOR OF SCIENCE in COMPUTER SCIENCE from University of Mumbai.

**Internal Guide** **Coordinator**

**External Examiner**

**Date:** **College Seal**

# Abstract

Democratic voting is a crucial and serious event in any country, the current voting scheme in any country is through ballot paper or by use of EVM. These processes have many drawbacks such as transparency, low voter turn-out, tampering of votes, distrust in the election body, forging of unique Id (voter id card), delay in giving out results and the most important is security issues. Security of digital voting is always the biggest concern when considering to implement a digital voting system. With such monumental decisions at stake, there can be no doubt about the system's ability to secure data and defend against potential attacks. One way the security issues can be potentially solved is through the use of blockchain technology.

Blockchain technology offers infinite number of applications. Blockchain is a distributed ledger technology that allows digital assets to be transacted in a peer-to-peer decentralized network. A distributed ledger technology is an exciting advancement in this regard. Block is a collection of all the transactions. Blockchain possess salient features such as immutability, Decentralization, Security, Transparency and anonymity. Blockchain with smart contracts emerges as a promising candidate for building a safer, secure and transparent E-voting systems.

We have implemented and tested a sample e-voting application as a smart contract for the Ethereum network using the blockchain technology through wallets and the Solidity language. Limited amount of token(gas) is given in the wallet which is exhausted when the user votes thus preventing duplicity of votes. This paper also highlights the pros and cons of using blockchain technology and also demonstrates a practical system by showcasing a webapp for voting and its limitations.

# ACKNOWLEDGEMENT

I would like to thank all those who are involved in this endeavor for their kind cooperation for its successful completion. At the outset, I wish to express my sincere gratitude to all those people who have helped me to complete this project in an efficient manner.

I am thankful to Prof. Dr. Moushumi Datta, Principal, Nagindas Khandwala College, Malad (W), for her kind support in all respects during my study. I would like to thank Dr. Sindhu PM, Coordinator & Assistant Professor, Department of CS, Nagindas Khandwala College, Malad (W), who gave the opportunity to do this project.

I offer my special thanks to my Project Guide Mrs. Elizabeth Leah George, Assistant Professor, Department of CS, Nagindas Khandwala College, Malad (W), without whose help & support throughout this project, this project would not have been a success.

Most of all & more than ever, I would like to thank my family members for their warmness, support, encouragement, kindness & patience. I am really thankful to all my friends who always advised & motivated me throughout the Course.

# DECLARATION

I hereby declare that the project entitled, **"E-Voting System Using Blockchain"** done at **Nagindas Khandwala College, Malad(West)**, has not been in any case duplicated to submit to any other university for the award of any degree. To the best of my knowledge other than me, no one has submitted to any other university.

The project is done in partial fulfillment of the requirements for the award of degree of **BACHELOR OF SCIENCE (COMPUTER SCIENCE)** to be submitted as final semester project as part of our curriculum.

**Santosh Vishwakarma**

# TABLE OF CONTENTS

# List of Tables

# CHAPTER 1: INTRODUCTION

## 1.1 Background

This project is a smart way to manage the progress of Voting through Blockchain.

It contains 2 portals & 2 homepages:

Portals/Profilers

1. Voters

2. Admin

Homepages

1. Master Homepage

2. Drives Homepage

Profilers

1. Voters

It Contains the Voters details and The Form to cast the vote to the candidate which is Later Fed to the Admin Profile for Verification.

2. Admin

The Responsibility of Admin is to Creation a Poll and Verify the Details of the Voters and they can Update the details of the Voters and Candidates.

Homepage

1. Master Homepage

It is the Main Homepage where the Different Voters Logins are Contained along with a link to the Drives Homepage.

2. Drives Homepage

This Page is for the voters to Look at the Current Polling Drives, and Polling Results.

# 1.2 Objectives

Our main motivation in this project is to provide a secure voting environment and show that a reliable e-voting scheme is possible using blockchain. Because, when e-voting is available for everyone who has a computer, or a mobile phone, every single administrative decision can be made by people and members; or at least people's opinion will be more public and more accessible by politicians and managers. This will eventually lead humanity to the true direct democracy. It's important for us since elections can easily be corrupted or manipulated especially in small towns, and even in bigger cities located in corrupt countries.

Plus, large-scale traditional elections are very expensive in the long term, especially if there are hundreds of geographically distributed vote centers and millions of voters. Also, the voter turn-out at the voting centers is relatively low as the person might not be staying at the address his name is enrolled in the list, or he might be out for vacation or any other work. E-voting will be able solve these problems, if implemented carefully. The concept of e-voting is significantly older than blockchain. So that, all known examples so far used means of centralized computation and storage models.

# 1.3 Applicability

This project, is applicable for small-scale polls and elections such as college election. A larger voting with millions of voters may have different problems to address. It manages Voters information in the Secure Database with Blockchain Security. It improves the existing system. It has the facility of maintaining the details of the Voter, thereby reducing the manual work. It will save time and energy spent in making report or Result and collecting data. E-Voting System using Blockchain can be accessed throughout the location where the Voters are there.

The Ethereum network's scalability is still unknown and needs further research, that's why we cannot suggest use of these contracts for nation-wide elections, at least for now. Our contracts are executed in the Ethereum blockchain, so wherever browser can be run (location, platform, device, etc.), our voting application can be used, too.

A fundamental problem of blockchain based e-voting systems is to provide anonymity for voters without compromising the transparency of the general voting process. In detail, all the transactions (money transfers, votes etc.) are essentially written to the blocks of the blockchain as plaintext. So that, a vote from wallet address A to wallet address B can be seen by anyone who has access to the chain. Which is, of course, a big disadvantage. And, it is not possible to use such a system for official/critical elections. Providing this anonymity is also a major challenge in the current state-ofthe-art works. proposed a solution based on the Diffie-Hellman process, which also implies the use of public/private key pairs and random numbers, so that a "two-round" referendum can supposedly be held with some ballot privacy.

# CHAPTER 2: GAP ANALYSIS/ DRAWBACK OF EXISTING SYSTEM

While current election systems are far from perfect, security risks can persist in Internet- and blockchain-based voting systems, says a study by researchers from Massachusetts Institute of Technology (MIT).

The paper titled "Going from Bad to Worse: From Internet Voting to Blockchain Voting," comes at a time when news reports of possible foreign interference in elections, of unauthorised voting, of voter disenfranchisement, and of technological failures have called into question the integrity of elections worldwide.

Internet- and blockchain-based voting would greatly increase the risk of undetectable, nation-scale election failures, said the MIT paper.

According to the researchers, claims that "voting over the Internet" or "voting on the blockchain" would increase election security have been found wanting and "misleading".

For the study, Institute Professor Ronald Rivest of MIT's renowned Computer Science and Artificial Intelligence Laboratory (CSAIL) and his colleagues analysed prior research on the security risks of online and electronic voting.

They showed that "not only do these risks persist in blockchain-based voting systems, but blockchains may introduce additional problems for voting systems."

The paper pointed out that prior studies had shown that online voting may have little to no effect on turnout in practice, and it may even increase disenfranchisement.

"More importantly: given the current state of computer security, any turnout increase derived from with Internet- or blockchain-based voting would come at the cost of losing meaningful assurance that votes have been counted as they were cast, and not undetectably altered or discarded," the researchers wrote.

# CHAPTER 3: REQUIREMENTS AND ANALYSIS

## 3.1 Problem Definition

Every eligible voter must participate in the election process, and Election Commission (EC) tries to make available various reliable, trustworthy, and secure platforms so no one can stay away from their fundamental rights. The new voting platforms must have to satisfy all the legal constraints of the existing and traditional voting process. The complete election process is separated into several tasks. Dimitris elaborated the system requirements for secure e-voting. Software Engineering principles assist in identifying these functional and nonfunctional requirements of the e-voting process.

## 3.2 Requirements Specification

A. Functional Requirements (FR):-

In this, we are specifying the various services or functionalities that are offered by the system. These functional requirements specify the inputs and their behavior as per the inputs provided by the user to the system. The abstract view of all e-voting related functional requirements are described as given below:

1) **Voter Registration [FR1]:** Every voter has to register herself in front of election authority with necessary documents. The screening criteria help to identify the eligibility of the voter.

2) **Provide the Authentication Credentials to Voter [FR2]:** Each voter, after validation of the identity of voter, gets credential details in the form of either user id and password or public-private key pair. The credential hides voters' real identity from the rest of the world. At the time of vote casting, every voter authenticated using credentials and this will help to prevent double-voting.

3) **Prepare the Digital Ballot Paper [FR3]:** As the candidates are finalized for the election, the next phase is to prepare the ballot paper to cast the vote. The ballot paper consists of the name of all the candidates. As the start of vote casting phase, every voter received a ballot paper to cast a vote via email or notification on client-side software.

4) **Casting the Vote [FR4]:** Each voter, select one of the candidates from the list of candidate mentioned on the ballot paper. No link between voter and vote is available to prove the way a vote is

cast by a voter. For this, each vote must be encapsulated by a double layer. First by using the voter's digital signature and then vote encrypted by one of the encryption techniques. The votes cast by voters are verified against the already available votes to avoid the double-voting problem.

5) **Vote Tallying [FR5]:** After the end of the vote casting phase, the system starts to count the votes stored on the blockchain, after counting the result is stored on the blockchain for the audit purpose, and then declare the results of an election. In this phase, all the votes are fetched from the blockchain.

B. Non-Functional Requirements:-

Non-functional requirements define systems quality attributes that affect the user experience. If unable to fulfill the non-functional requirements, then the system will not satisfy the user's expectations. There are several non-functional requirements but, in this we are targeting security related quality attributes. E-voting is a confidential process and must protect against security attacks, such as Sybil attack, Denial-of-Service (DoS) attack, Man-in-the-middle attack, etc. These non-functional requirements are listed below:

1) **Privacy:** This requirement satisfies that no one can be traced out to whom the vote is cast by the elector. It is not possible to reveal the candidate chosen by the voter in the vote casting phase, and all votes remain concealed. Voter name, elector's public/private key-pair, voters' documents, etc., will not be shared with unauthorised users.

2) **Data Integrity:** The adversary should not access or tamper with the data stored on the system. Also, unauthorized users should not alter the vote cast by electorates and the electorates personal information. Cryptographic hash algorithms are one-way functions, so the integrity of data can remain intact.

3) **Confidentiality:** means the system will not reveal the true identity of the electorate to their public-key infrastructure. An authorized user may have access to the system data. This confirms that the system has quality attributes such as coercion-resistance and privacy.

4) **Authentication:** The legal and eligible voters only access the system and cast the vote, and it helps in preventing the system from data or information leakage.

5) **Authorization:** Different stakeholders of the system are provided with a wide range of functionalities. These functionalities help in preventing illegal access and tampering with the data.

# 3.3 Planning and Scheduling

Following is the gantt chart showing the schedule for carrying out the various activities involved in this project.

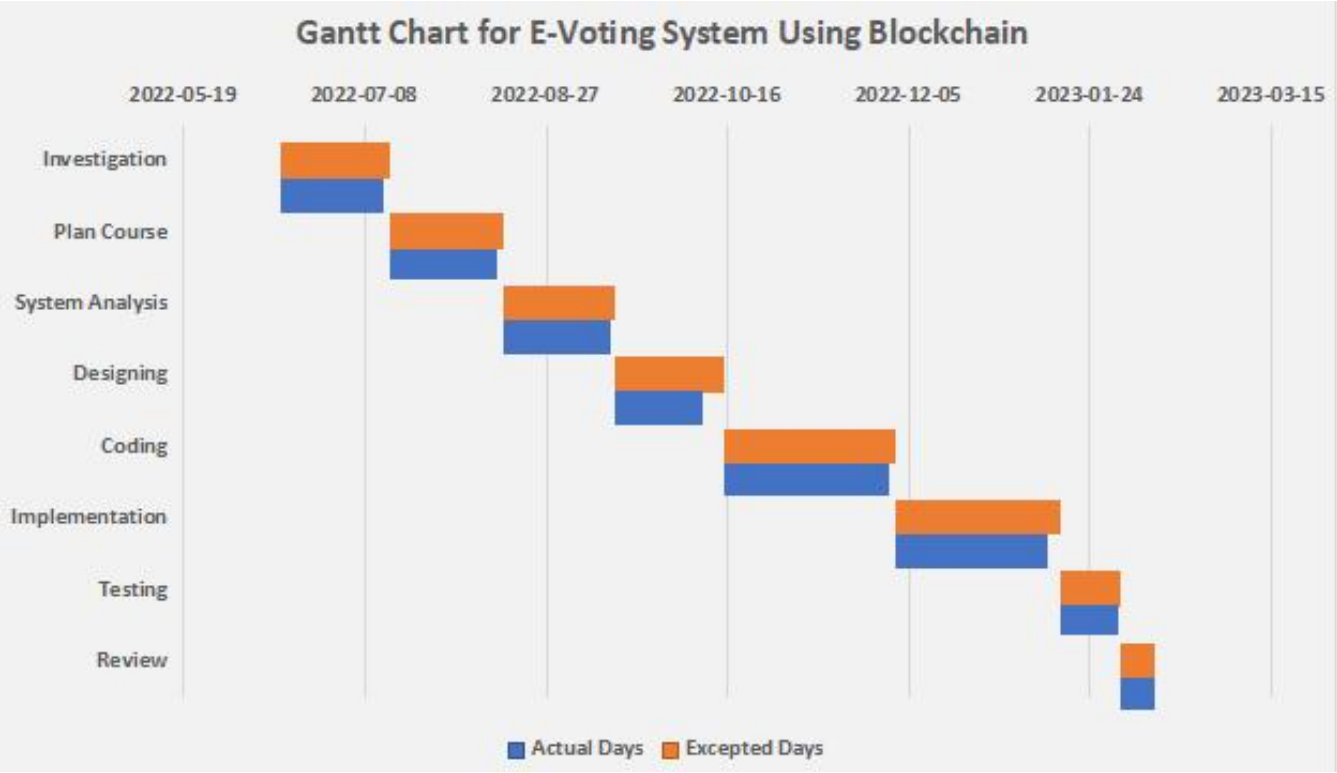



Fig. 1 - Gantt Chart

# 3.4 Software and Hardware Requirements

**Software Requirements**

1. Visual Studio Code (All Files Editing)
2. XAMPP (Apache Server & MySQL Database)
3. Ganache (Truffle)
4. Windows 7 or higher (Operating System)

**Hardware Requirements**

1. Processor Intel Dual Core or Above
2. 4GB RAM
3. 100 GB Hard Disk

# 3.5 Preliminary Product Description

The proposed e-voting system architecture and has been divided into several layers to achieve modular design. These layers are described below;

1. User Interaction and Front-end Security layer is responsible for interacting with a voter (to support vote casting functions) and the administrator (to support functions pertaining to administering the election process). It encapsulates two key functions i.e. authentication and authorization of the users (voters and administrators) to ensure that the access to the system is restricted to legitimate users in accordance with the predefined access control policies. A number of different methods can be applied to achieve this function ranging from basic username/password. Therefore these are rendered specific to individual implementation of the proposed architecture. Overall, this layer serves as the first point of contact with the users and is responsible for validating user credentials as governed by the system-specific policies.

2. Access Control Management layer is envisaged to facilitate by providing services required to achieve their expected functions. These services include roles definition, their respective access control policies and voting transaction definitions. The role definition and management provides core support for the access control functions implemented whereas the voting transaction definitions support the Blockchain based transaction mapping and mining performed. Overall, enables a coherent function of the proposed system by providing the foundations required by individual layers of system.

3. E-Voting Transaction Management layer is the core layer of the architecture where the transaction for e-voting constructed at Role Management / Transactions is mapped onto the Blockchain transaction to be mined. This mapped transaction also contains the credentials provided by a voter for authentication. This data is then used to create the Cryptographic hash and contributes towards creating the transaction ID. The verification of such credentials is envisioned to be achieved at User Interaction and Front-end Security layer. A number of virtual instances of nodes are involved in the process of mining to get this transaction finally enter into the chain.

4. Ledger Synchronization layer synchronizes Multi-chain ledger with the local application specific database using one of the existing database technologies. Votes cast are recorded in the data tables at the backend of the database. Voters are able to track their votes using the unique identifier provided to them as soon as their vote is mined and added into the Blockchain ledger. The security considerations of the votes are based on block-chain technology using Cryptographic hashes to secure end-to-end communication. Voting results are also stored in the application's database with the view to facilitate auditing and any further operations at a later stage.

# CHAPTER 4: SYSTEM DESIGN

## 4.1 Basic Modules

1. **Registration Phase:-** The voter has to Register. Itself first with its unique id and attributes such as name, electrol roll no and mobile number. All this data is stored in the database.

2. **Login:-** The voter after registration tries to login themselves to cast a vote. In this phase voter first login After successful login, to cast their vote. Voter has to authenticate themselves. for real time authentication is used for enhanced security.

3. **Blockchain Technology:-** This Technology is. mainly for its security features. Blockchain provides a secure and Transparent environment. Blockchain encrypts the voter message (casted vote) using Assymetric encrypting algorithms. A public key is provided by Blockchain and private key with host. Public key is used for verification purpose by ledger.

4. **Database:-** User database u stored in database details like name, gender, unique Id. are stored in Database. My SQL is the proposed database to be used.

5. **Ethereum Network:-** Ethereum network provides framework for Blockchain a creation and storage. Every block is created and its details are stored in an encrypted ledger. These created blocks are distributed among nodes which provides high fault tolerance to the system.

6. **Result Phase:-** The processing and tallying of votes is done in Result phase. Result are generated and displayed on website. User can verify their voter their oven public key. This provides transparency to the voting system.

# 4.2 Schema Design

## Data Modeling:-

Data Dictionary of Tables which are created under the Database of this Project:

Table 1:- Candidate

This Table keeps the information about Candidate.

candidate

| Column | Type | Null | Default | Links to | Comments | Media type |
|---|---|---|---|---|---|---|
| id *(Primary)* | int(11) | No | | | | |
| name | varchar(255) | No | | | | |
| pollId | int(11) | Yes | *NULL* | poll -> id | | |

It is a Data Dictionary of Candidate table which contains 5 column and 3 rows.

Fig. 2 - Candidate Table

Table 2:- Migrations

This Table keeps the information about Data Migration of users or Voters their timestamp and name.

migrations

| Column | Type | Null | Default | Links to | Comments | Media type |
|---|---|---|---|---|---|---|
| id *(Primary)* | int(11) | No | | | | |
| timestamp | bigint(20) | No | | | | |
| name | varchar(255) | No | | | | |

It is a Data Dictionary of Migration table which contains 3 column and 3 rows.

Fig. 3 - MigrationTable

Table 3:- Poll

This Table keeps the information about Poll.

poll

| Column | Type | Null | Default | Links to | Comments | Media type |
|---|---|---|---|---|---|---|
| id *(Primary)* | int(11) | No | | | | |
| name | varchar(255) | No | | | | |

It is a Data Dictionary of Poll table which contains 3 column and 2 rows.

Fig. 4 - Poll Table

Table 4:- User

This Table keeps the Data or Information of Users and Admin.

user

| Column | Type | Null | Default | Links to | Comments | Media type |
|---|---|---|---|---|---|---|
| id *(Primary)* | int(11) | No | | | | |
| name | varchar(100) | No | | | | |
| citizenshipNumber | varchar(255) | No | | | | |
| email | varchar(180) | No | | | | |
| password | varchar(255) | No | | | | |
| admin | tinyint(4) | No | | | | |
| verified | tinyint(4) | No | 0 | | | |

It is a Data Dictionary of Candidate table which contains 3 column and 7 rows.

Fig. 5 - User Table

# 4.3 UML Diagrams / Block Diagram

**Class Diagram:-**

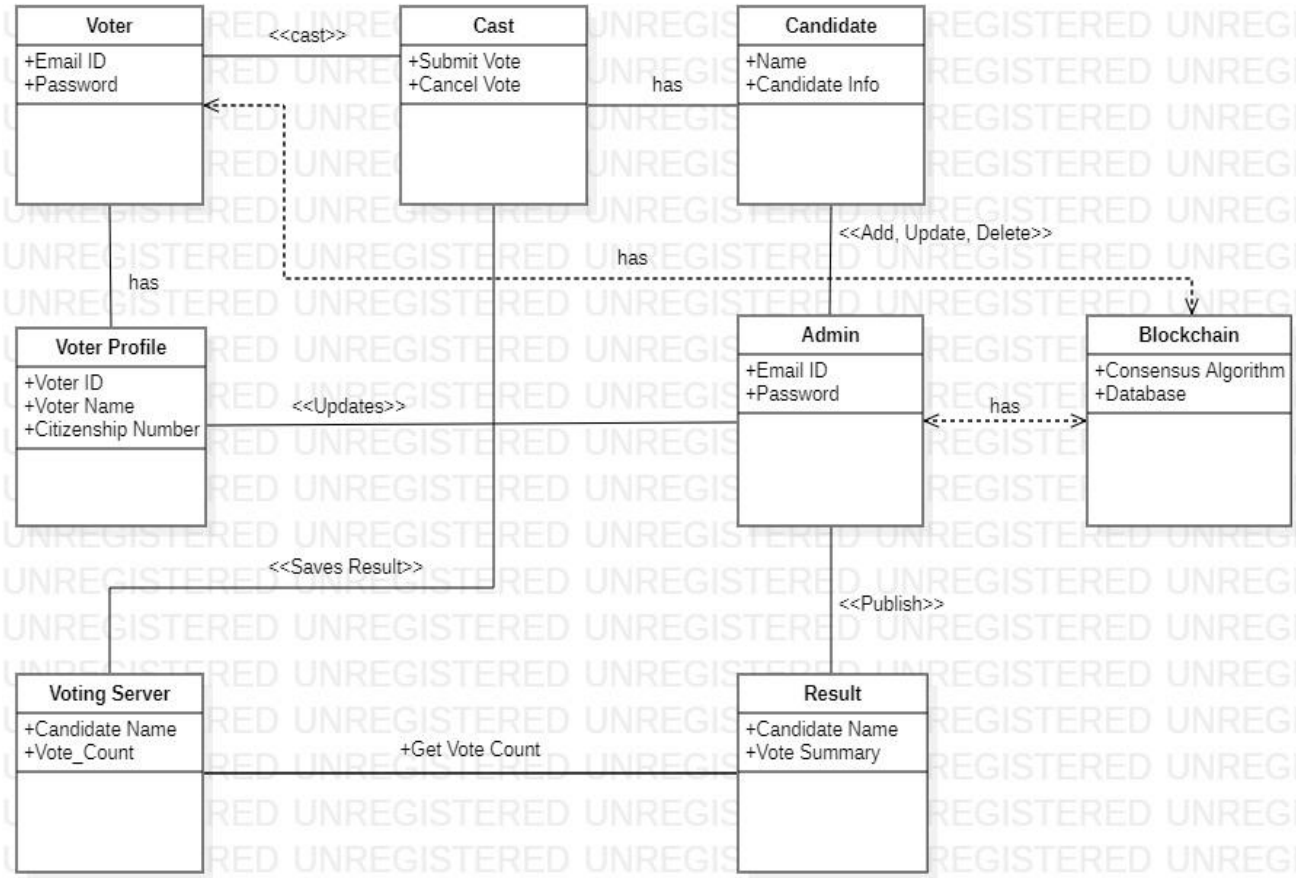



Fig. 6 - Class Diagram

**Data Flow Diagram:-**

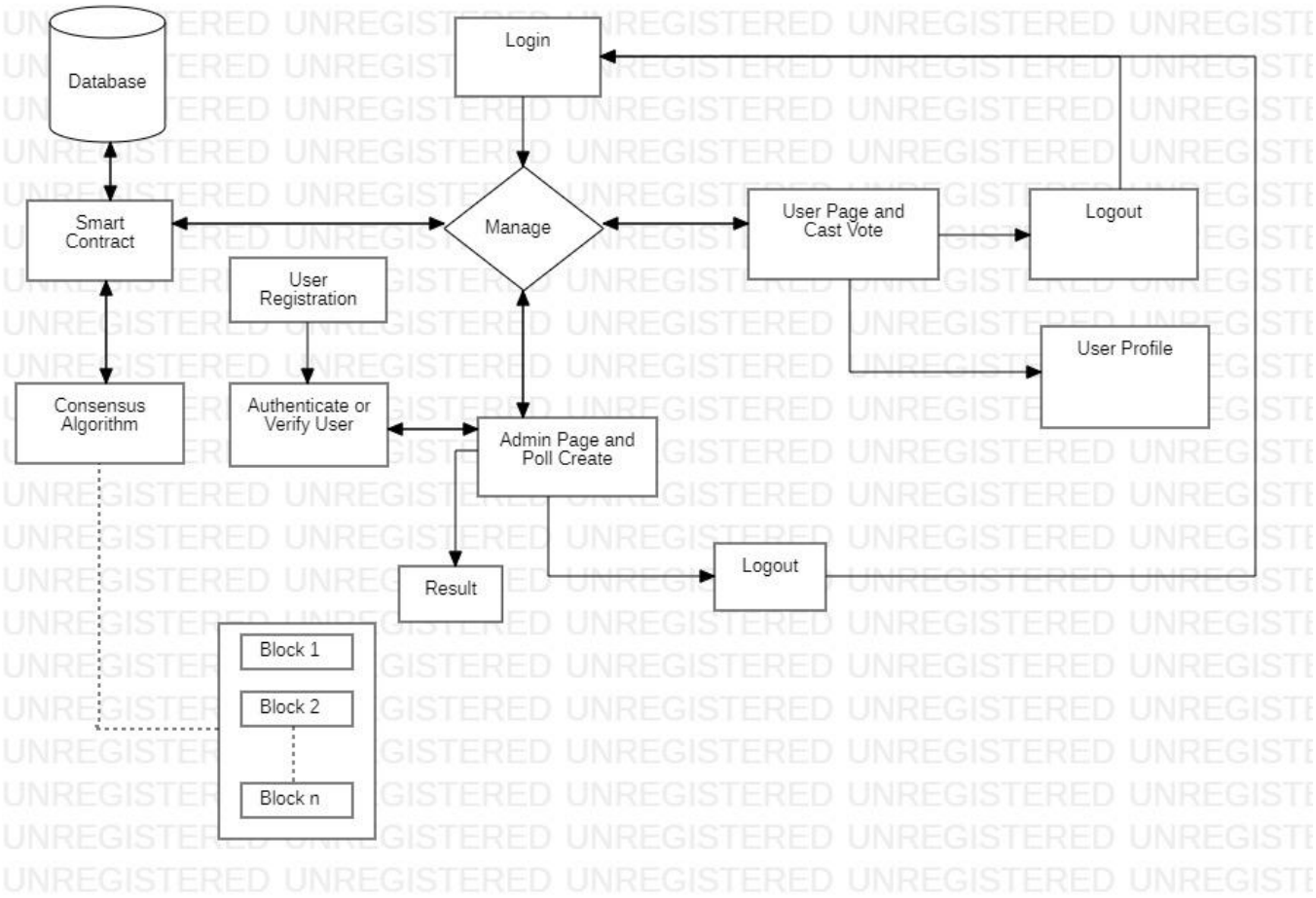



Fig. 7 - Data Flow Diagram

**Use Case Diagram:-**

Admin and User End Use Case Diagram:-

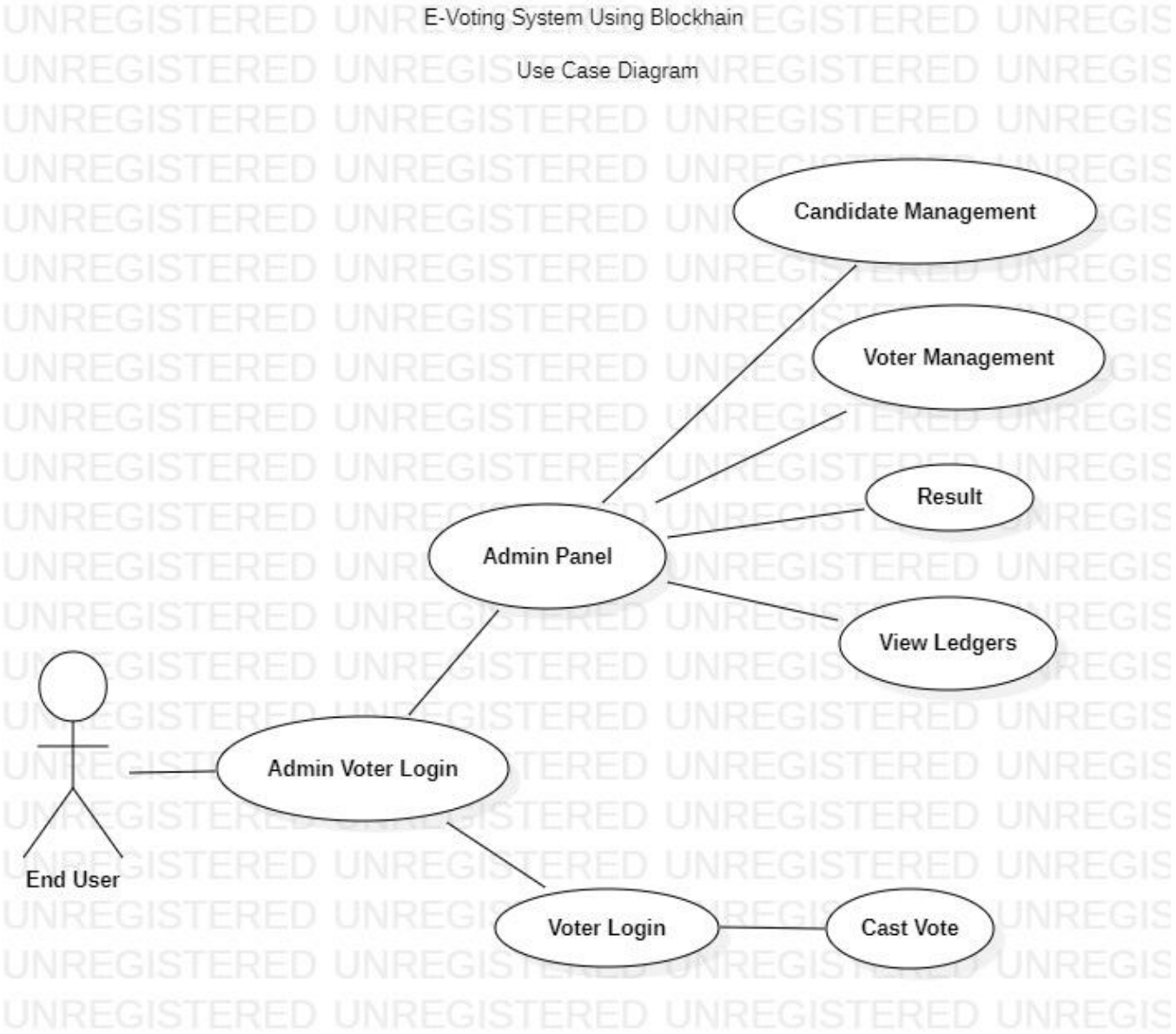



Fig. 8 - Use Case Diagram

**Sequence Diagram:-**

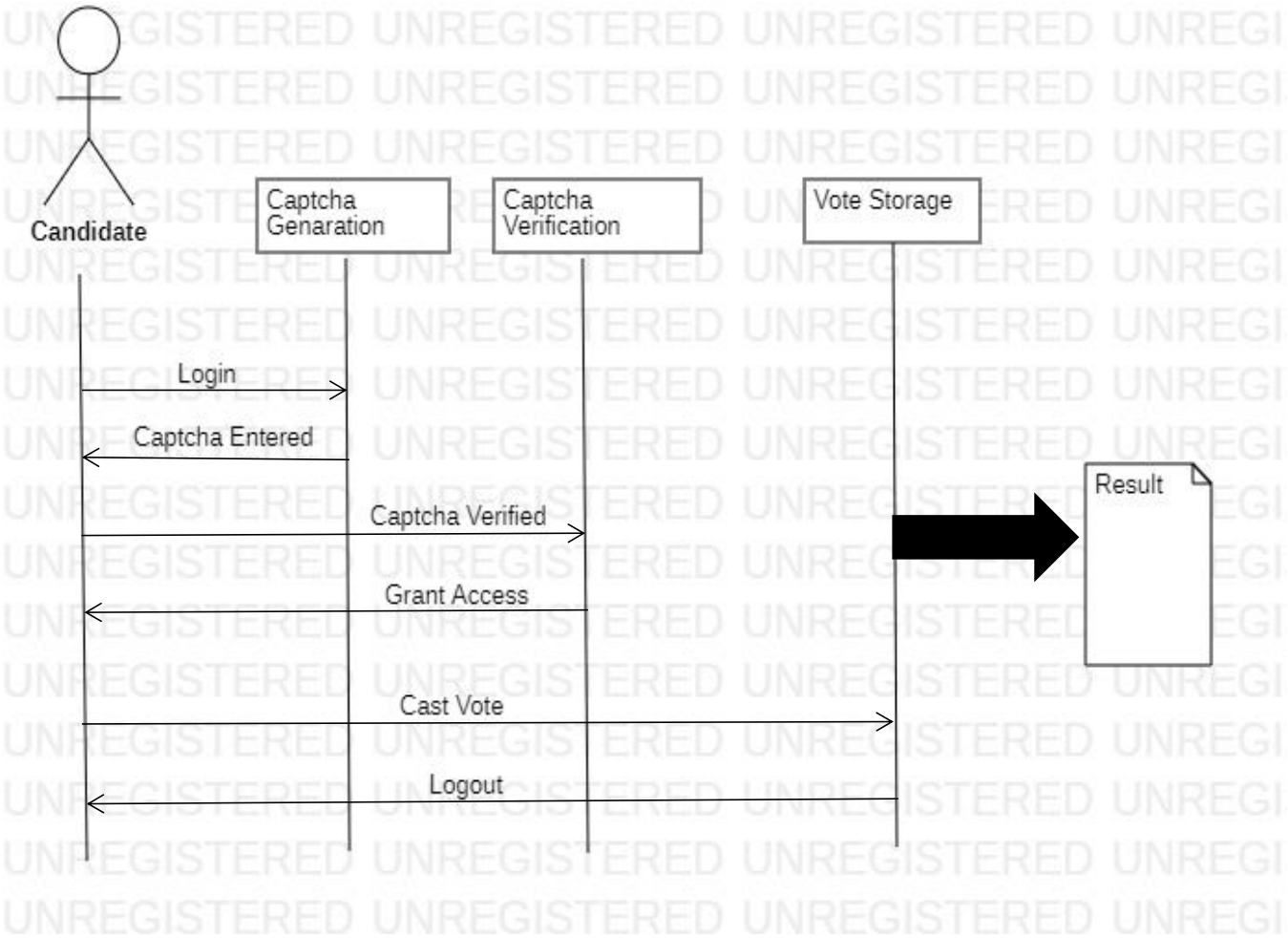



Fig. 9 - Sequence Diagram

**ER Diagram:-**

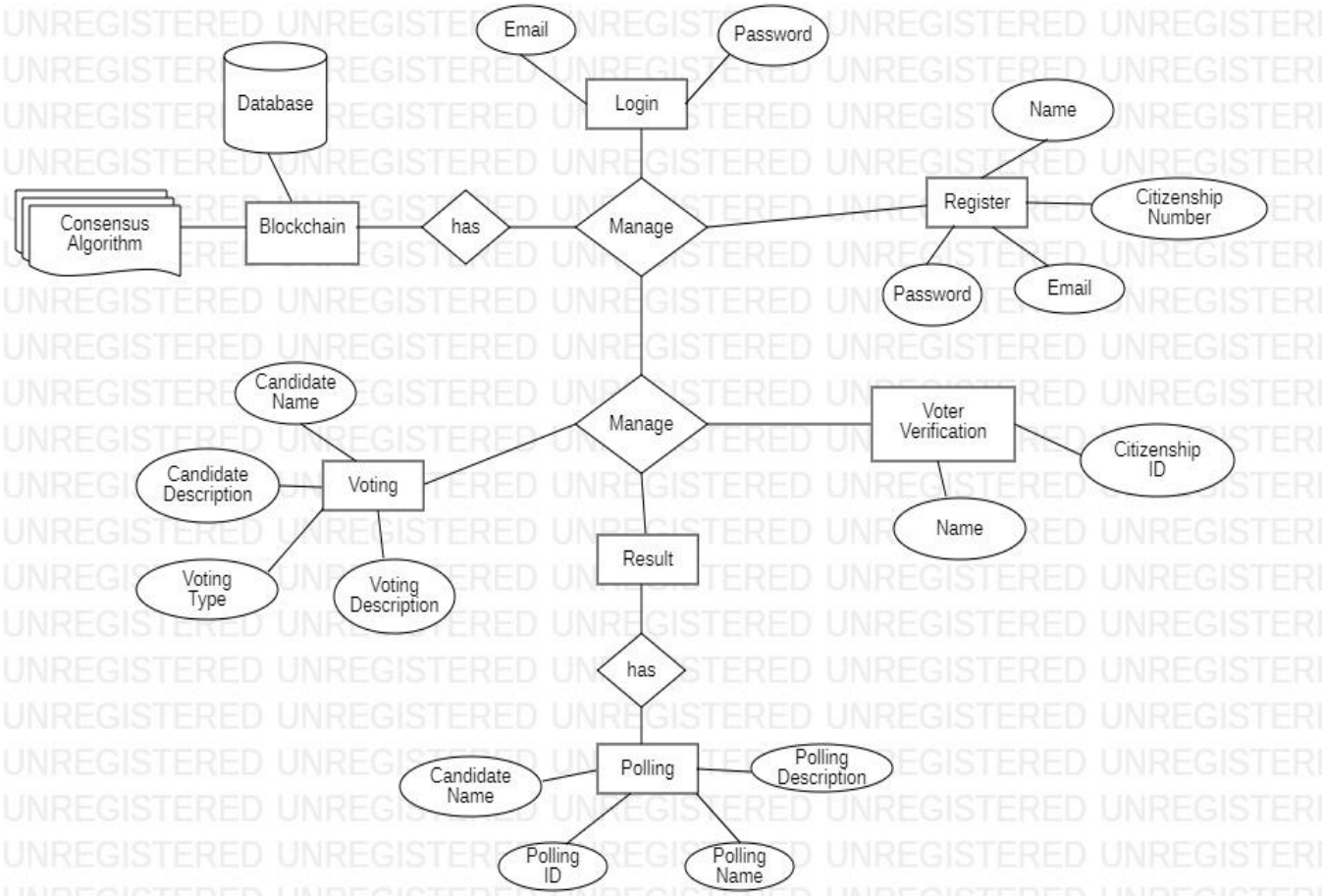
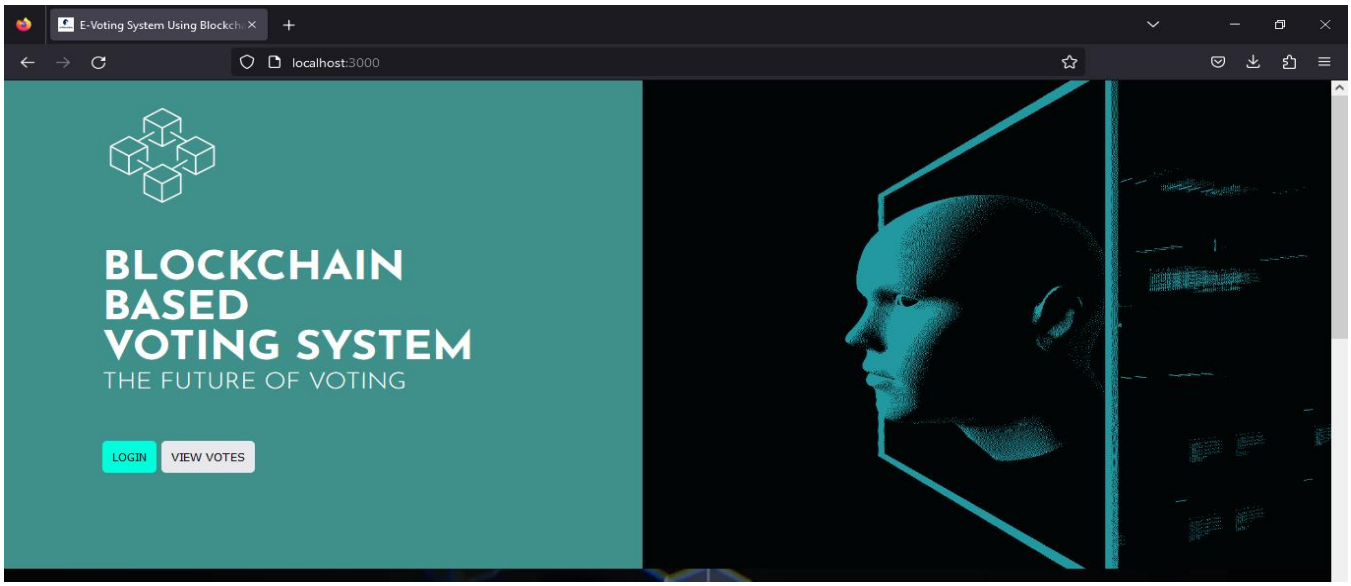



Fig. 10 - ER Diagram

# 4.4 User interface design

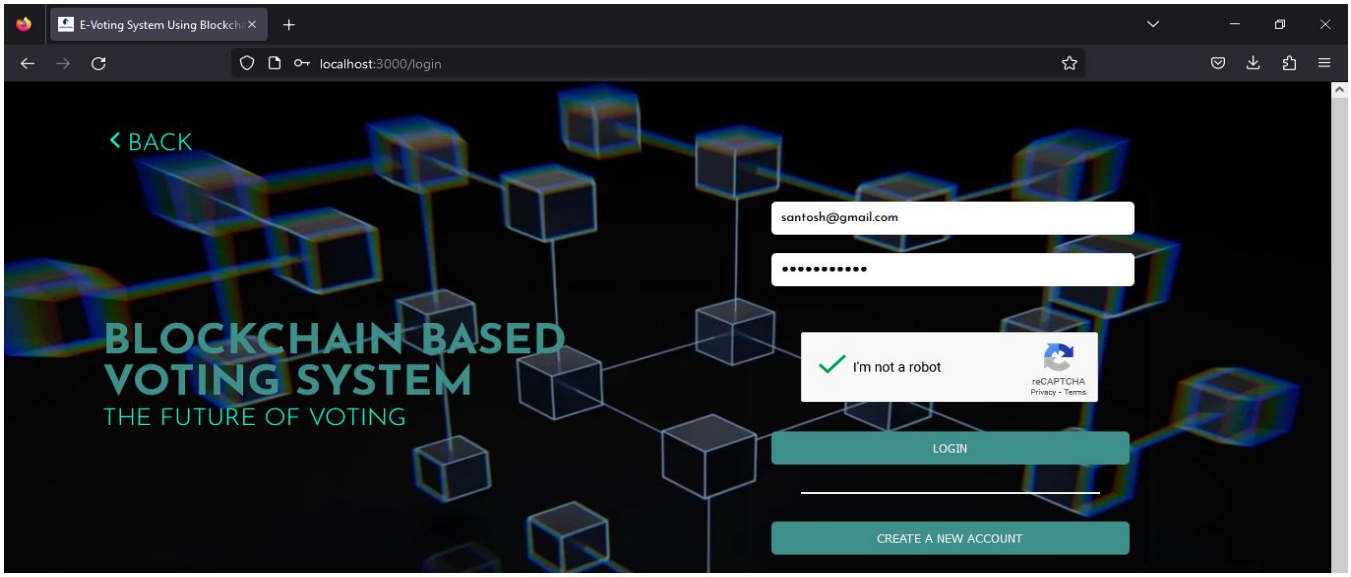
The Main page is shown in Fig. 11.




Main Page it is Landing Page of E-Voting System

Fig. 11 - Main Page

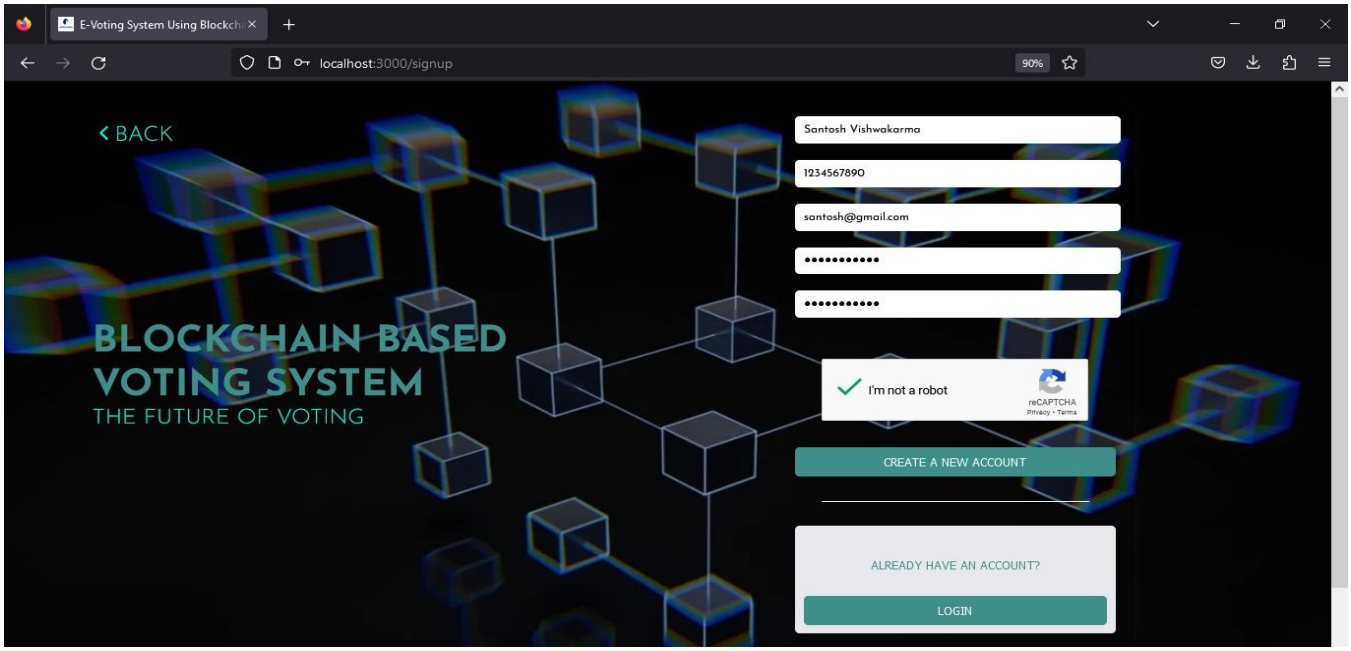
The Voters Login page is shown in Fig. 12.




It is a Login Page Admin and Voter Login with their Credentials and Entered into their Account.

Fig. 12 - Voters Login Page

Voters Registration Page is shown in Fig. 13.




It is a Registration Page where any Voter can come and create their Account.

Fig. 13 - Voters Registration Page

Admin Page for Poll Creation is shown in Fig. 14.


It is a Admin Page where Admin can Create a Poll of as many Candidate as they want.

Fig. 14 - Poll Creation Page

Voter Page for Vote Cast is shown in Fig. 15.


It's a Voters Page where Voter can Cast their valuable vote(that is only one vote) to any Candidate they want.

Fig.15 - Voter Casting Vote

After Casting a Vote is shown in Fig. 16.


It is a Voter Page when they Casted their Vote to Candidate after that they should not Vote any else.

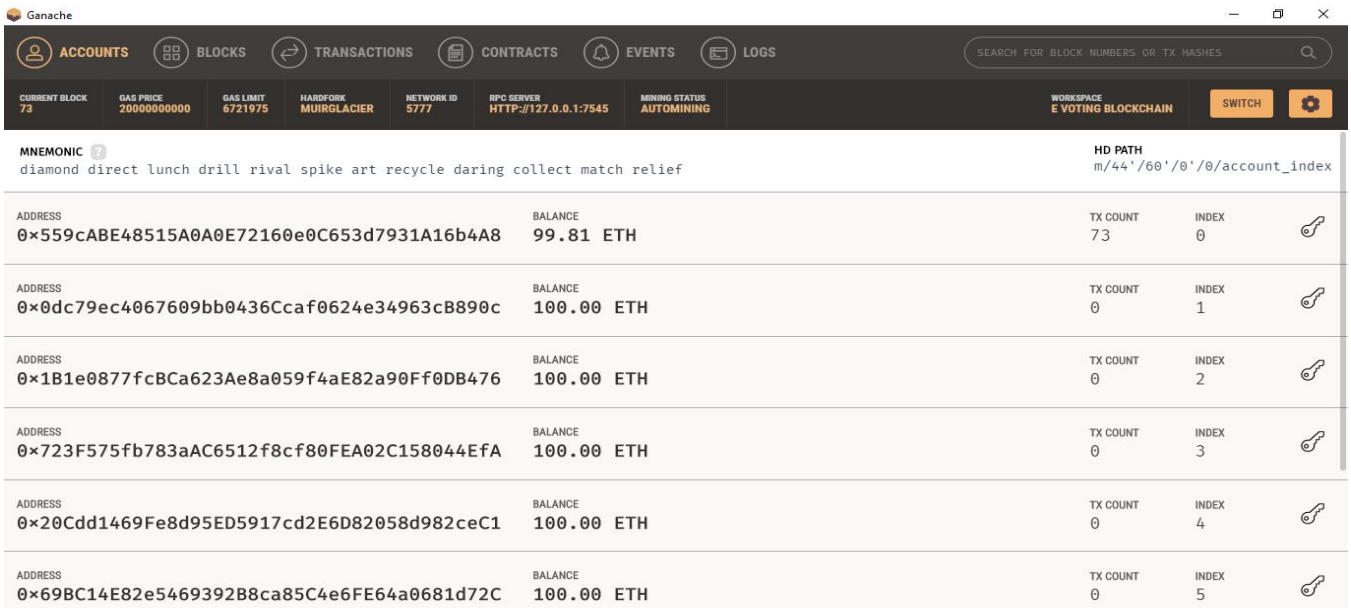
Fig. 16 - Casted Vote

Result Page of Poll is shown in Fig. 17.


It is a Result Page where Votes are showing to Admin how much votes are getting a Candidate.

Fig. 17 - Result Page of Poll

Truffle Migrate is connected to Backend shown in Fig. 18.

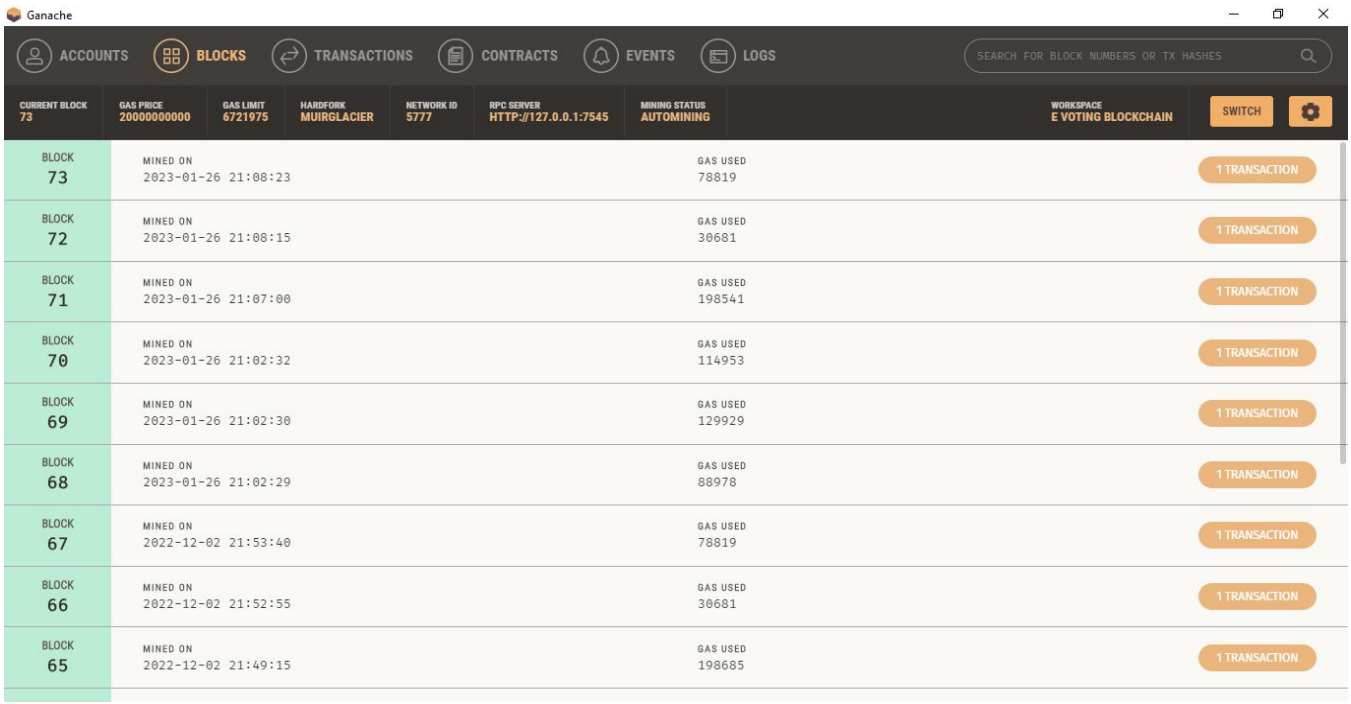



It is a Account Page of Truffle where Truffle Migration connection with Backend.

Fig. 18 - Truffle Migrate Connection

Inside Ganache Truffle Migrate Blocks shown.

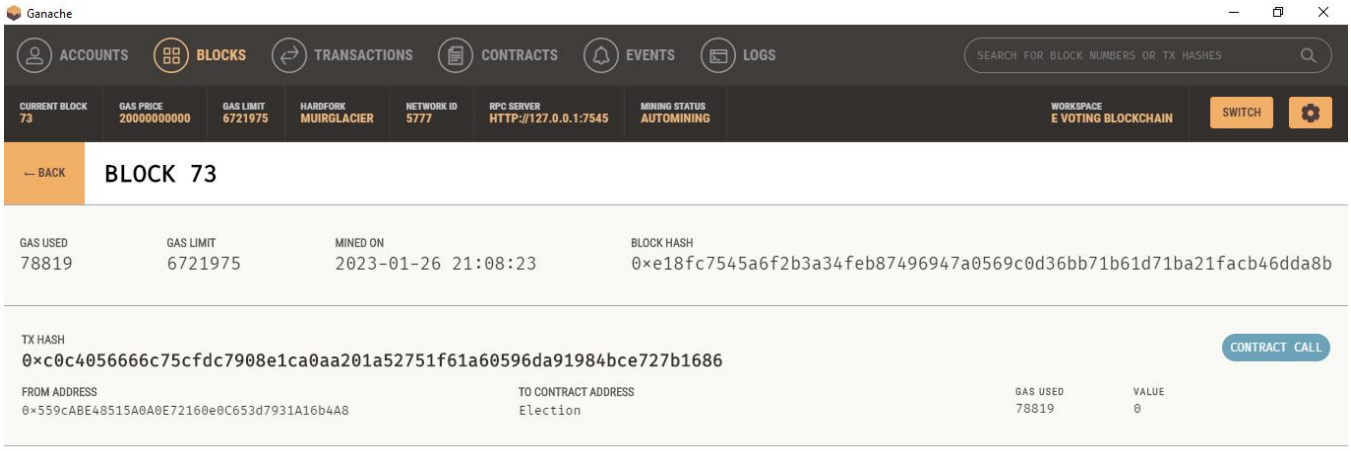
Step1:- Blocks are connected with each other through their Hash Files is shown in Fig. 19.




It is a Blocks Page where how Blocks are Connected with each other through their Hash Files.

Fig. 19 - Truffle Migrate Block

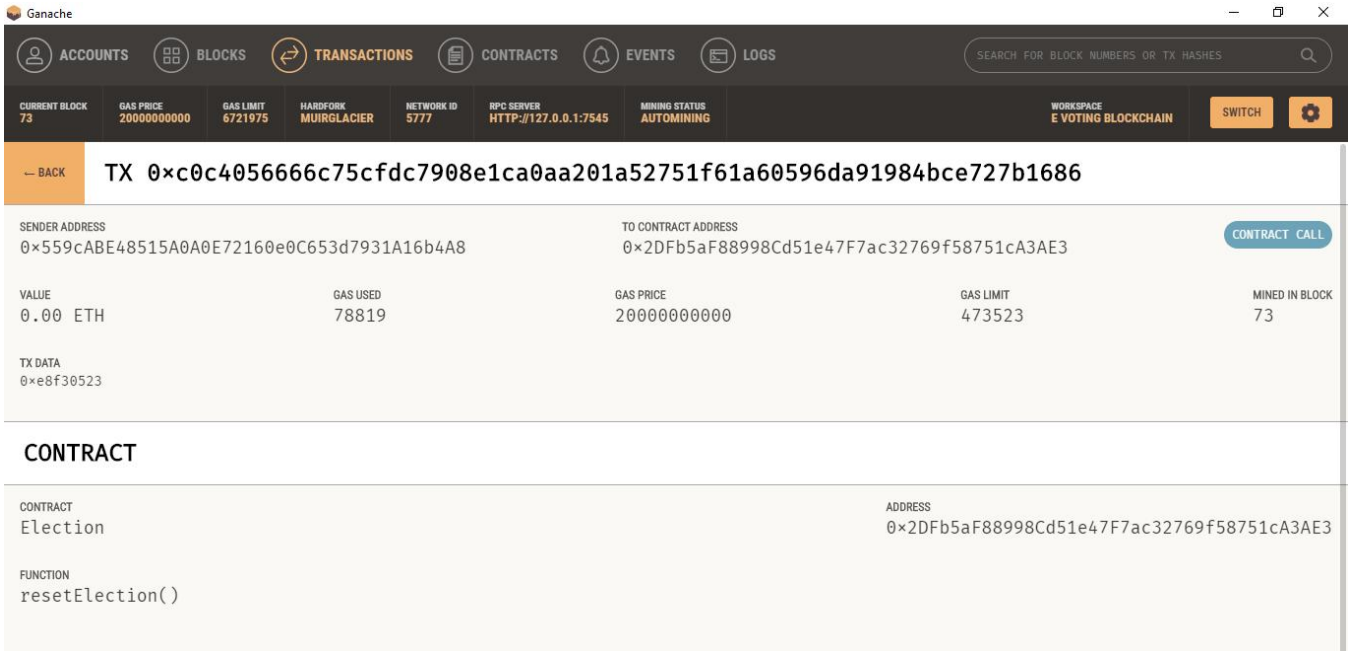
Step2:- Inside Block73 with their Hash file shown in Fig. 20.




It is an Inside of Block73 showing their Hash Files.

Fig. 20 - Block73

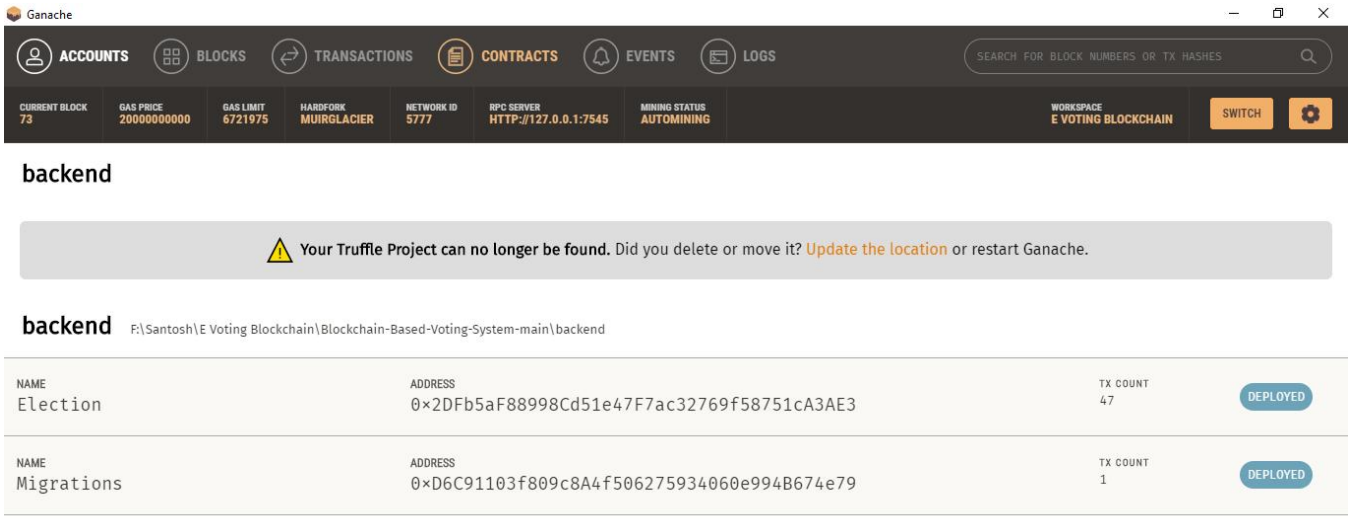
Step3:- Inside Hash File of Block73 shown in Fig. 21.




It is a Inside of Hash File showing their Transactions Data.

Fig. 21 - Hash File Transaction Data

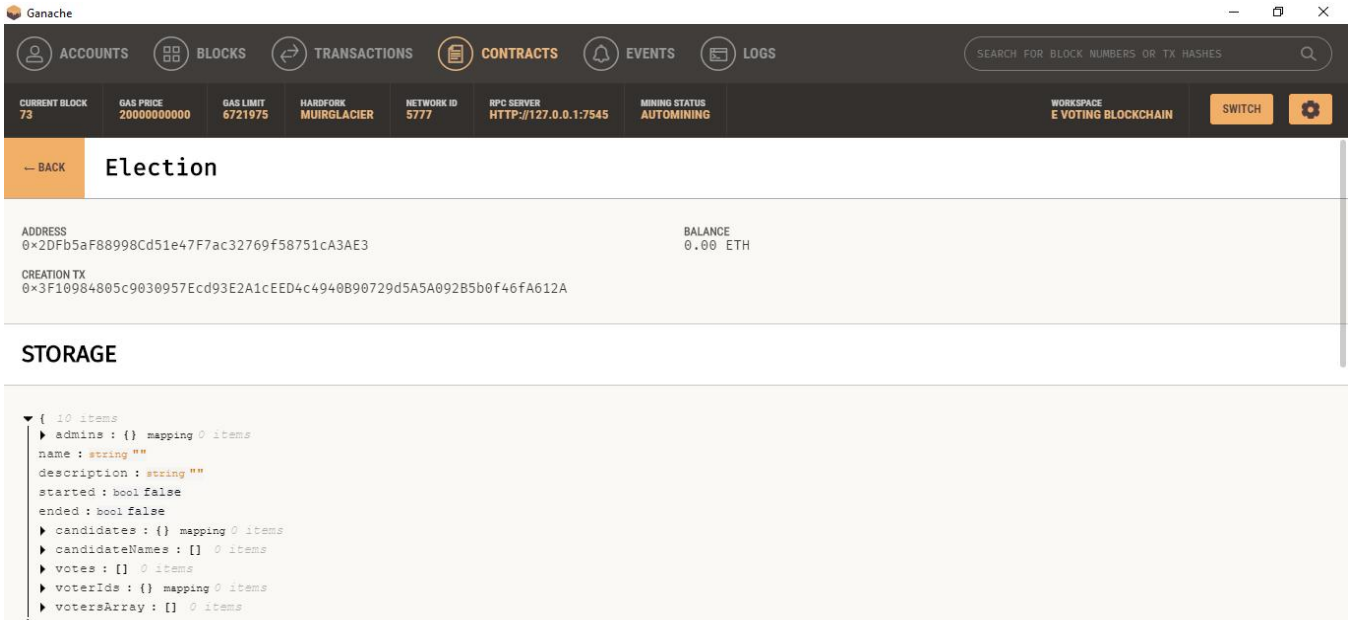
Truffle Migrate Contract of Election deployment and Migration deployment shown in Fig. 22.




It is a Contract Page where how Migration is been deployed with Backend Server of System.

Fig. 22 - Truffle Migrate Contract

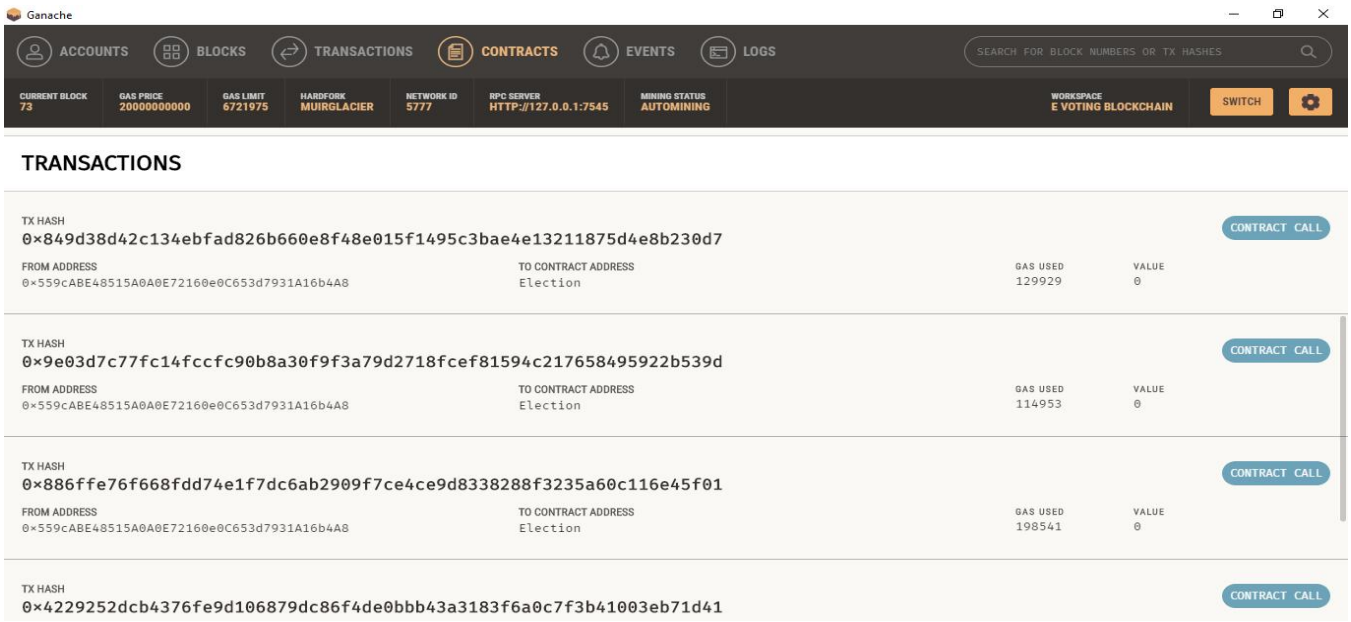
Inside Contracts Election deployment with Backend shown in Fig. 23.




Inside of the Election Contract Page of Backend Server showing their Address.

Fig. 23 - Election Deployment

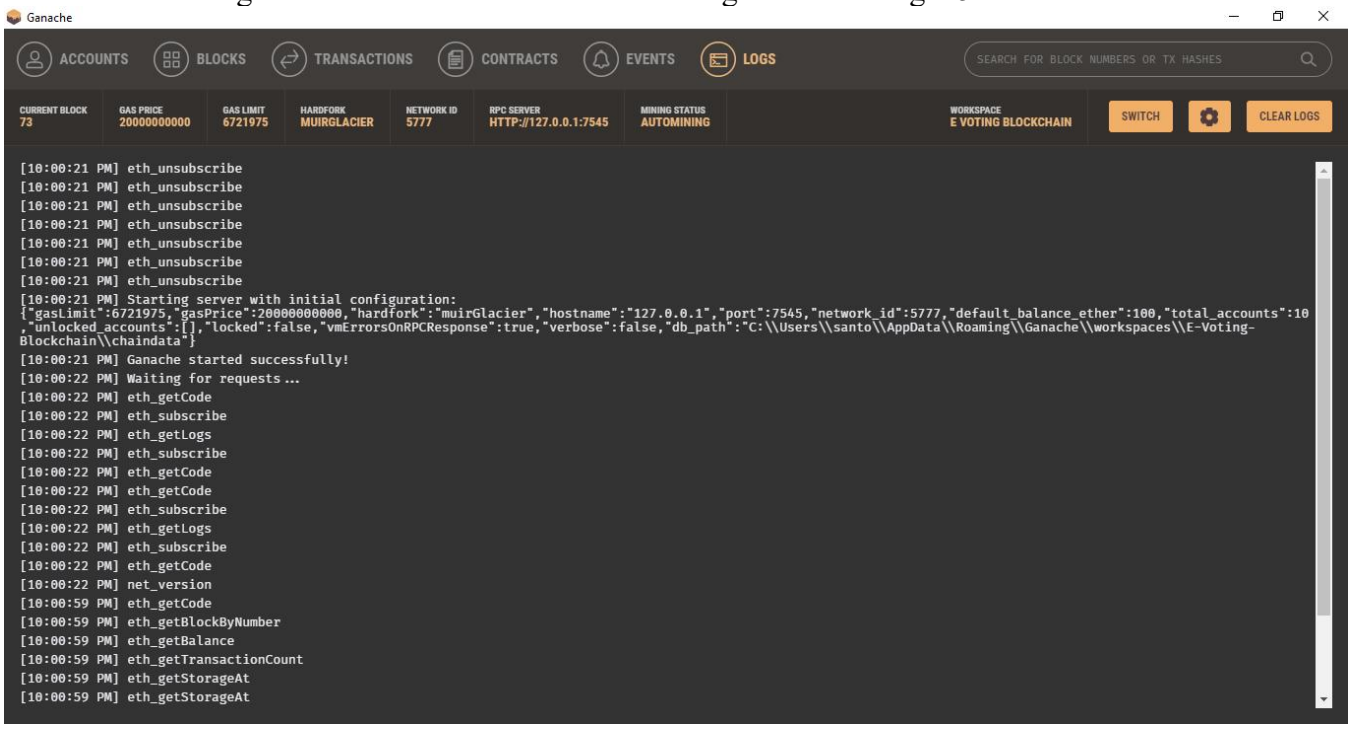
Inside Contracts Migration deployment with Backend shown in Fig. 24.




It is Transactions of Contract Migration Deployment with Backend Server.

Fig. 24 - Migration Deployment

Inside Truffles Log full transactions saved with their Logs shown in Fig. 25.




It is Insides of Log Page showing full Details of Transactions when Backend Server is connected with Truffle and when mining is Start.

Fig. 25 - Truffles Log

# 4.5 Security Issues

This system does not have a Two Step Authentication Feature Yet. Hence, if another user were to gain your User ID & Password, they would have unauthorized access to resources & could masquerade as you throughout their session. In the voting process relies on voters email address that can be hacked or manipulated easily. To be obvious, there will be always some people who registers to the system using someone else's mail address and votes on behalf of them. For example, a grandson may open an email address for his grandparents from different devices, and cast their votes. This method guarantees none of the required qualifications such as security, data integrity or privacy that an e-voting system has system. For such a system, stealing votes or changing votes are totally.

# CHAPTER 5: IMPLEMENTATION AND TESTING

## 5.1 Code

- Following is the code snippet for the validation of the Voters Home page.

```
import React, { useContext, useEffect, useState } from "react";
import axios from "../../axios";
import Chart from "../../components/Polls/Chart";
import Finished from "../../components/Polls/Finished";
import Panel from "../../components/Polls/Panel";
import Running from "../../components/Polls/Running";
import Waiting from "../../components/Waiting";
import { AuthContext } from "../../contexts/Auth";

const User = () => {
  const [voteState, setVoteStatus] = useState<
    "finished" | "running" | "not-started" | "checking"
  >("checking");
  const [loading, setLoading] = useState(true);
  const [data, setData] = useState({ name: "", description: "", votes: {} });
  const [votable, setVotable] = useState("");

  const authContext = useContext(AuthContext);

  useEffect(() => {
    console.log("called here ?");

    axios
      .get("/polls/status")
      .then((res) => {
        setVoteStatus(res.data.status);
        setLoading(false);
      })
      .catch((error) => console.log({ error }));
  }, []);

  useEffect(() => {
```

```
    if (voteState !== "checking") {
      axios.get("/polls/").then((res) => {
        setData(res.data);
        console.log(res);
        setLoading(false);
      });

      axios
        .post("/polls/check-voteability", {
          id: authContext.id.toString(),
        })
        .then((res) => {
          setVotable(res.data);
        })
        .catch((err) => console.log(err));
    }
  });

  if (loading || voteState === "checking") return <div></div>;

  if (voteState === "not-started") return <Waiting />;

  return (
    <Panel name={data.name} description={data.description}>
      <>
        {voteState === "running" ? <Running /> : <Finished />}

        <Chart
          enableVote={votable === "not-voted"}
          userId={authContext.id}
          userName={authContext.name}
          votes={data.votes}
        />
      </>
    </Panel>
  );
};

export default User;
```

- Following is the code snippet for the validation of the Voters Login page.

```
import React, { useState, useContext } from "react";
import ReCAPTCHA from "react-google-recaptcha";
import { useNavigate } from "react-router";
import { Formik } from "formik";
import { RouteProps } from "react-router";
import LoginLayout from "../layouts/Login";
import * as Yup from "yup";
import axios from "../axios";
import { AuthContext } from "../contexts/Auth";

const schema = Yup.object().shape({
  email: Yup.string().email("Invalid email").required("Required"),
  password: Yup.string().min(3).required("Required"),
});

const Login = (props: RouteProps): JSX.Element => {
  const navigate = useNavigate();
  const authContext = useContext(AuthContext);

  const [error, setError] = useState<any>("");
  const [verified,setVerified] = useState(false)

  function onChange() {
    console.log("Captcha value:", (useState));
    setVerified(true);
  }

  return (
    <div>
      <LoginLayout error={error}>
        <div className="form-container">
          <Formik
```

```
33                initialValues={{
34                  email: "",
35                  password: "",
36                }}
37                validationSchema={schema}
38                onSubmit={(values) => {
39                  axios
40                    .post("/auth/login", { ...values })
41                    .then((res) => {
42                      authContext.authenticate(res.data.user, res.data.accessToken);
43                    })
44                    .catch((err) => {
45                      let error = err.message;
46                      if (err?.response?.data)
47                        error = JSON.stringify(err.response.data);
48                      setError(error);
49                    });
50                }}
51              >
52                {({ errors, touched, getFieldProps, handleSubmit }) => (
53                  <form onSubmit={handleSubmit}>
54                    <div className="input-container">
55                      <input
56                        id="email"
57                        type="email"
58                        placeholder="Email"
59                        {...getFieldProps("email")}
60                      />
61                      <div className="form-error-text">
62                        {touched.email && errors.email ? errors.email : null}
63                      </div>
64                    </div>
```

```
65
66                  <div className="input-container">
67                    <input
68                      id="password"
69                      type="password"
70                      placeholder="Password"
71                      {...getFieldProps("password")}
72                    />
73                    <div className="form-error-text">
74                      {touched.password && errors.password
75                        ? errors.password
76                        : null}
77                    </div>
78                  </div>
79
80                  <div className="form-info-text">
81                  <ReCAPTCHA sitekey="6LdWkG0kAAAAAMi-ZVhJwn-UJkd6Gq7-1AovYDP8" onChange={onChange}/>
82                  </div>
83
84                  <button className="login-button button-primary" type="submit" disabled={!verified}>
85                    Login
86                  </button>
87                </form>
88              )}
89            </Formik>
90
91            {/* <div className="form-info-text">Forgot Password?</div> */}
92
93            <hr />
94
95            <button
96              onClick={() => navigate("/signup")}
```

```
97                    className="button-secondary"
98                  >
99                    Create a New Account
100                 </button>
101               </div>
102             </LoginLayout>
103           </div>
104       );
105   };
106
107   export default Login;
```

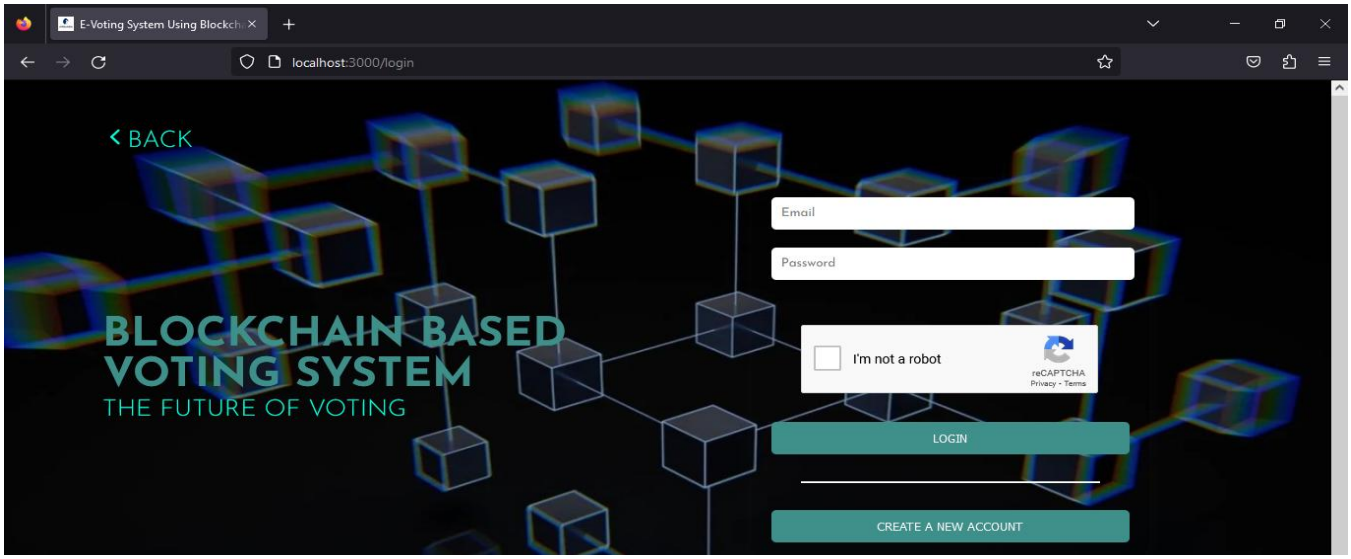
- Following is the code snippet for the validation of Admin Home Page.

```
import React, { useEffect, useState } from "react";
import { RouteProps } from "react-router";
import axios from "../../axios";
import StartPage from "./Start";
import PollsPage from "./Polls";
import ResultPage from "./Result";

const Home = (props: RouteProps): JSX.Element => {
  const [loading, setLoading] = useState<boolean>(true);
  const [status, setStatus] = useState<"not-started" | "running" | "finished">(
    "not-started"
  );

  useEffect(() => {
    setLoading(true);
    axios
      .get("/polls/status")
      .then((res) => {
        setStatus(res.data.status);
        setLoading(false);
      })
      .catch((error) => console.log({ error }));
  }, []);

  if (loading) return <div></div>;

  if (status === "finished") return <ResultPage />;
  if (status === "running") return <PollsPage />;

  return <StartPage />;
};
export default Home;
```

# 5.2 Testing Approach and Test Cases

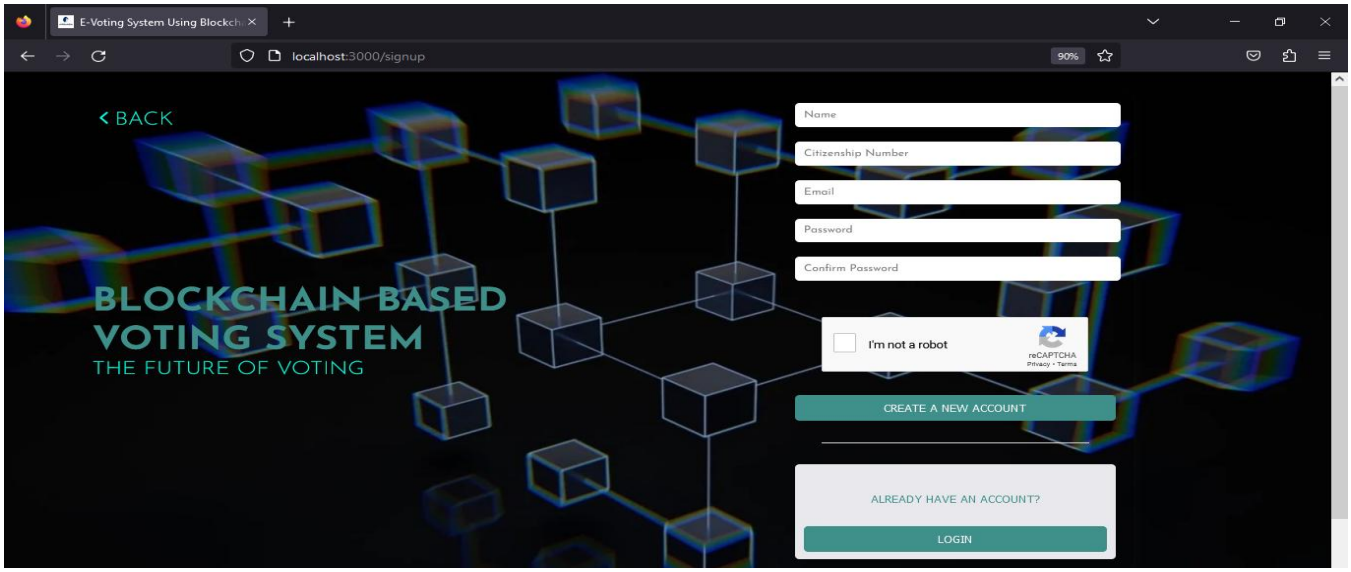
- **Black Box Testing:**

E-Voting System Using Blockchain test. In this module, Voter's name, username, password and other necessary information are supposed to be input.




When you click on Login Button directly without checking the reCaptcha Box it will not Login.

Fig. 26 - reCaptcha CheckBox Error and Login Button Testing

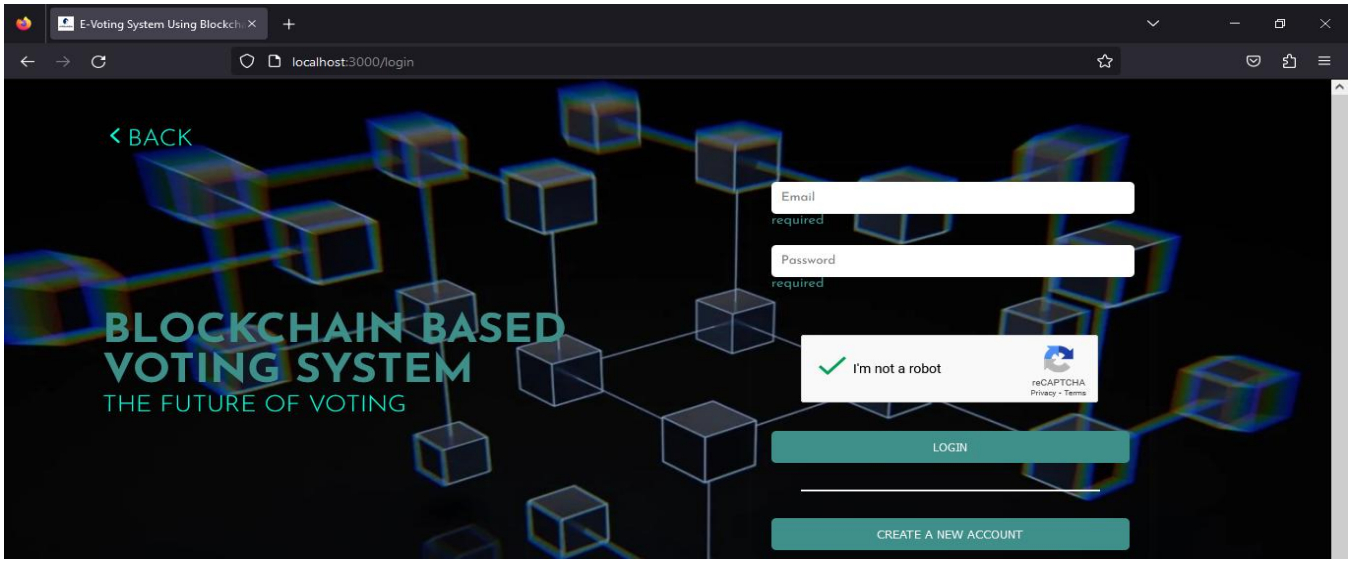



When you click on Crate a New Account Button it will not create any Account without checking reCaptcha Box.

Fig. 27 - reCaptcha CheckBox Error and Create a new Account Button Testing

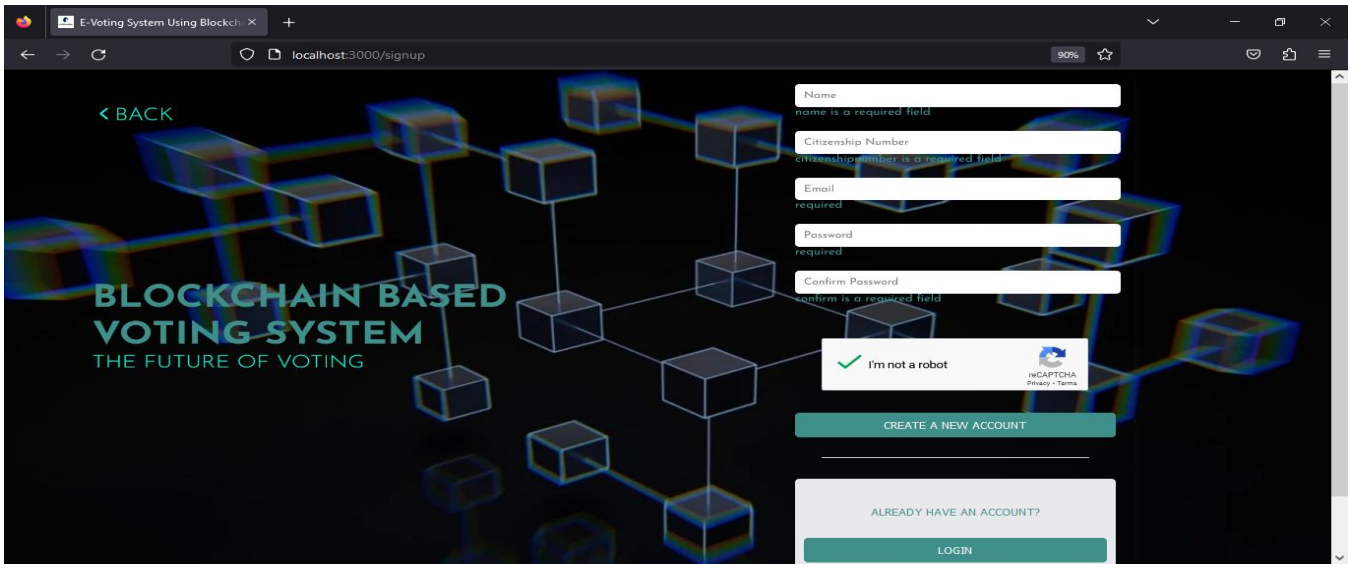
- **White Box Testing:**

E-Voting System Using Blockchain test. This part is designed to make it easier for administrator to add, query, modify and delete Voters information, and ensures the accuracy and security of input information. Avoiding errors of Voters information due to a large number of Voters.




If we not entered any credentials like Email and Password and we checked reCaptcha Box and then click Login Button it will throw an Error for Required to Fill.

Fig. 28 - Required to Fill Error Testing in Login Page.




If we not Entered any Details of New Voter and checked the reCaptcha Box and then click on Create New Account Button it will throw an Error for Required to Fill.

Fig. 29 - Required to Fill Error Testing in Create a New Account Page.

# CHAPTER 6: RESULTS AND DISCUSSION

## 6.1 Test Reports

In this Project formalize the system model of self tallying voting systems based on blockchain in decentralized IoT. An electronic voting system based on smart contracts to protect voters' privacy. The test results suggested that the system can effectively reduce the cost of using smart contracts to vote, and protect the privacy of voters with practicability. An e-voting system based on blockchain that eliminates some of the limitations in existing voting systems. The presented implementation is suitable for small scale elections like inside corporate houses, board rooms etc. The implementation uses smart contract from Ethereum. Truffle framework is used for development, testing and deploying smart contracts. Ganache is used as Ethereum client for testing. A distributed blockchain electronic voting scheme with self-tally function by using blockchain. In the Project, the voter's voting privacy information is protected by two rounds of zero knowledge proof protocol, but the voting scheme only allows voters to select one candidates, which can satisfy the situation of multiple candidates in one voting process.

In this testing Report we conducted a Voting Process the Candidate and Voters Information available under the Ganache Truffle Contracts and also one voter successfully casted their vote to their favourite Candidate.

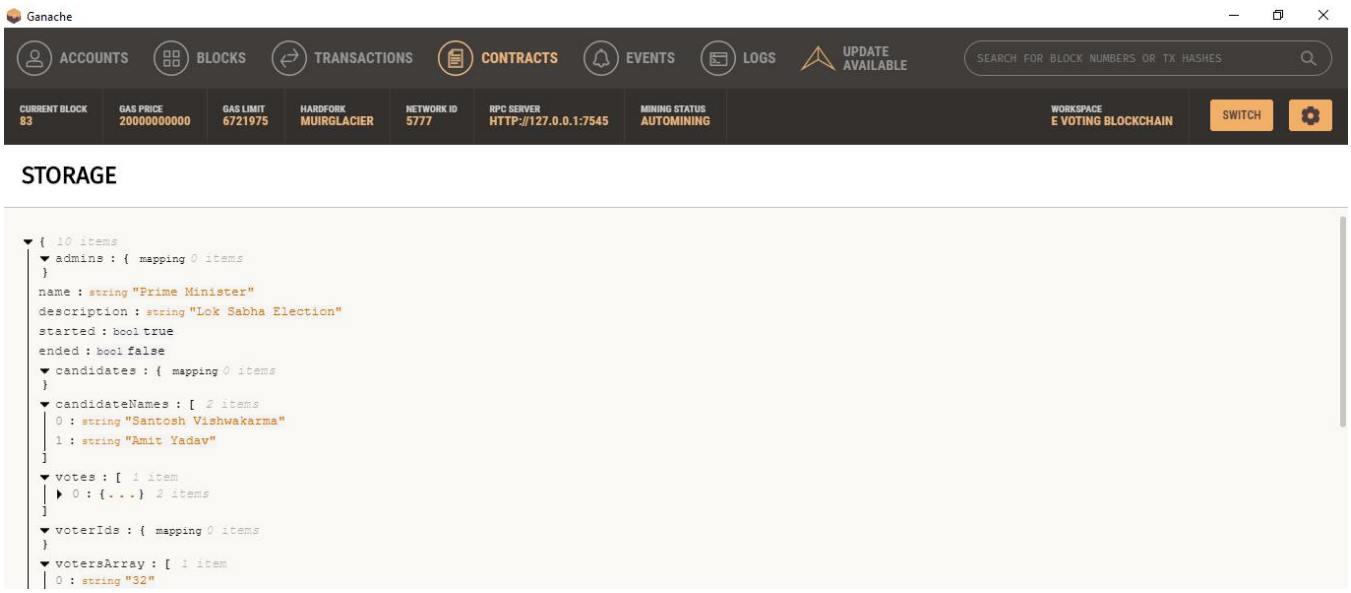



Fig. 30 - Candidate and Voters Information under the Blockchain Network.

In this Report showing admin info who have all the control of Voting Process.

```
MariaDB [bbvs]> select * from user where admin=1;
+----+---------------------+------------------+-----------------+--------------------------------------------------------------+-------+----------+
| id | name                | citizenshipNumber | email           | password                                                     | admin | verified |
+----+---------------------+------------------+-----------------+--------------------------------------------------------------+-------+----------+
|  4 | Santosh Vishwakarma | 9874563210       | santosh@gmail.com | $2b$10$EH4vmFqAq7p/5XT3AVrYWe51QJK9Y2qnUq1rW/1JBHqPvxsZ2x7Na |     1 |        1 |
|  5 | Amit Yadav          | 1234567892       | amit@gmail.com  | $2b$10$7vLrsJD7J8j6Q3nWVmsi7.aXVpH0zK0guJEpd7axD7pzGiqb0FmLW |     1 |        1 |
+----+---------------------+------------------+-----------------+--------------------------------------------------------------+-------+----------+
2 rows in set (0.001 sec)
```

In this Report 2 Admins are their they can create a poll, verifying the voters.

Fig. 31 - Admin Information.

In this Report showing Users or Voters info who can cast their votes under the Voting Phase.

```
MariaDB [bbvs]> select * from user where admin=0;
+----+----------------------+------------------+---------------------+--------------------------------------------------------------+-------+----------+
| id | name                 | citizenshipNumber | email               | password                                                     | admin | verified |
+----+----------------------+------------------+---------------------+--------------------------------------------------------------+-------+----------+
|  1 | John                 | 9860777906       | john@gmail.com      | $2b$10$6sdkothEwAguhA0FytsGF.gcWPmTDB5hosif6rGX5FFJK8PdBgRHu |     0 |        1 |
|  2 | Liza                 | 9860777907       | liza@gmail.com      | $2b$10$70yLw0dPhAD0py/iiGUInO7kklGUmbMfa5BmXKGCXEID1ufTsqSQ6 |     0 |        1 |
|  3 | Ben                  | 9860777908       | ben@gmail.com       | $2b$10$1DsQFSqUs3ufyDDRBd9wYuU5i9ihbnYR4GCYJsI3IzGXamwFWnr4S |     0 |        1 |
|  6 | ANKIT                | 72838888         | ankit@gmail.com     | $2b$10$UNplYvaQaUdRynYSpWj5Eefo9jjdU86SNBmcfucwv9u4qs5lQwjze |     0 |        1 |
|  9 | Aman Rai             | 12345678901      | aman@gmail.com      | $2b$10$IMYhv0Zk/WvTg8/yjCJWM.z43vHH2CtfjOw1fCUCp8UCKJ4FqPGOW |     0 |        1 |
| 11 | Anmol Kaur           | 12345678902      | anmol@gmail.com     | $2b$10$eH2h6y5tBzCS0W6PaED/bOEtC8saqwu2SDRDaRPqXG86t73trPaJK |     0 |        1 |
| 12 | Suraj Yadav          | 12345678903      | suraj@gmail.com     | $2b$10$hyl5lLN2oVebYdStjiD8aeap0L7thRRvY28XBb6F/79GpE02VPrZi |     0 |        1 |
| 13 | Nikhil Dubey         | 12345678904      | nikhil@gmail.com    | $2b$10$ZUlq3YCRjhmxsxgjCweeUerfwljtitCCiEqEbXw9TIlHB7x44QWpO |     0 |        1 |
| 14 | Priya Yadav          | 12345678905      | priya@gmail.com     | $2b$10$G/q6z9L.TATHrzByet3JOO0nGLl4e0GCiQcKW3qOoQ.iYsCzRqTCS |     0 |        1 |
| 15 | Marylou Pereria      | 12345678906      | marylou@gmail.com   | $2b$10$IbFlFp0KqwVkNF3r8nFZ1.rDLOssrj5LxZ5oEFJKUkox12FPNI0am |     0 |        1 |
| 16 | Dhaval Mali          | 12345678907      | dhaval@gmail.com    | $2b$10$54AMP2MTvjgzqV5M76QXZuSOZdUFGbDI6OtHHcejnL0UEEZ5tK456 |     0 |        1 |
| 17 | Nidhi Shetty         | 12345678908      | nidhi@gmail.com     | $2b$10$JxseThDkTBtxY3dtldCAMuEW2ToDSDaFYFw1hVRzVb.FixcDtoaMe |     0 |        1 |
| 18 | Vrinda Nair          | 12345678909      | vrinda@gmail.com    | $2b$10$J4o7/PRo4h6AUc3Uv6ICieVf8D4QP20LZHIOvRql9zmm4eUMWtoHC |     0 |        1 |
| 19 | Naveen Anandhan      | 12345678910      | naveen@gmail.com    | $2b$10$.3LkGVjp/3DHEOFZXCvBo.e2LSf9U/wvodhJE7FDn4TRPD5MvayHe |     0 |        1 |
| 21 | Ashish Pal           | 12345678911      | ashish@gmail.com    | $2b$10$pjgxNo22jnR2ZaMH2r6weu603ohA2idOUZiMhdzj.OmT0pJuV.7G6 |     0 |        1 |
| 23 | Niki Thakur          | 12345678913      | niki@gmail.com      | $2b$10$M5pmXLg5ArDGjFUuOLMrdemyt2nCkAMX6P/DfbUbqWhsB3aGfm/PG |     0 |        1 |
| 24 | Purvi Rawal          | 12345678915      | purvi@gmail.com     | $2b$10$6nSRKDv8et5C/pOhd5ktmuK4Q2FvRfIk6bsTyS6Q6k1eEupFs1md6 |     0 |        1 |
| 25 | Elizabeth Leah George | 12345678916      | elizabeth@gmail.com | $2b$10$eM5528YOkDauC6zefm2znOUgOcIwmYQP8BZvhD8Ok/BcDkKFYB7Km |     0 |        1 |
| 26 | Ashish Modi          | 12345678917      | ashishmodi@gmail.com | $2b$10$Dil2b46l2gUsFP2y9GrWQOsk/uUCTijb5963MLMCCHhYoWbIBuAh6 |     0 |        1 |
| 27 | Vishakha Bagwe       | 12345678918      | vishakha@gmail.com  | $2b$10$LRlYyxCIaa0TmY6jLfFWYOdah2p4.vf5K9F05zv0vSp2KL/.O7RGS |     0 |        1 |
| 28 | Mayuresh Shelke      | 12345678919      | mayuresh@gmail.com  | $2b$10$sTEOiWPY71PIeLkfc/xD1updk4RrdoIYbm4lEBfZQFAyXdKKHQCrm |     0 |        1 |
| 29 | Roshni Singh         | 12345678920      | roshni@gmail.com    | $2b$10$1.qLWl84/65oCmHMs0wjfuiOF1I3J5YtoKTrQkkOsVq.vMy/x4hRm |     0 |        1 |
| 30 | Khushi Kawa          | 12345678921      | khushi@gmail.com    | $2b$10$sv.8c.HneVyyTvpKZo9/WOeznuhuFBUuH7kMHzgXpveVVAJfVMGUW |     0 |        1 |
| 31 | Akhtar Ansari        | 12345678922      | akhtar@gmail.com    | $2b$10$ga6NSR3txh06wrQURcdxBuhAZuwmDeJWVh8UotU6fX7.6IjKYvH.m |     0 |        1 |
| 32 | Sanya Shaikh         | 9874589000       | sanya@gmail.com     | $2b$10$d0YBNRp26E1oSH14ZB0z7OeSN6..vuJO2t5O/gbMmj7RB4nEs74Xy |     0 |        1 |
+----+----------------------+------------------+---------------------+--------------------------------------------------------------+-------+----------+
25 rows in set (0.001 sec)
```

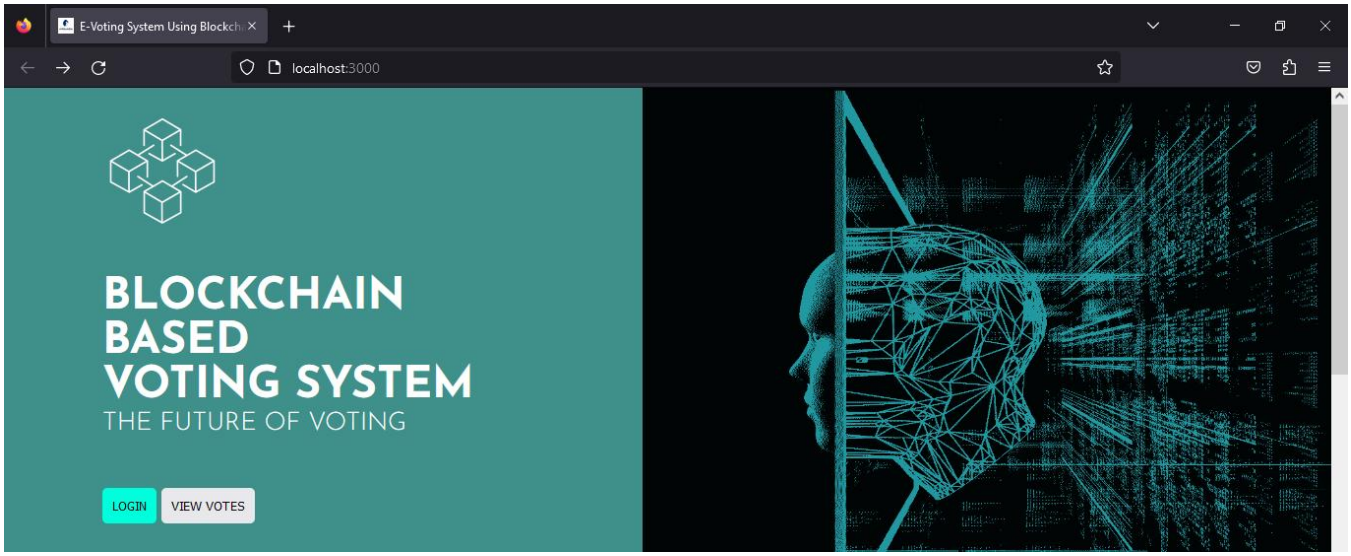
In this Report there are 25 Verified Voters who can cast one vote to their Favourite one Candidates under the Voting Process.

Fig. 32 - Voters Information.

# 6.2 User Documentation

## 1. Registration process:-

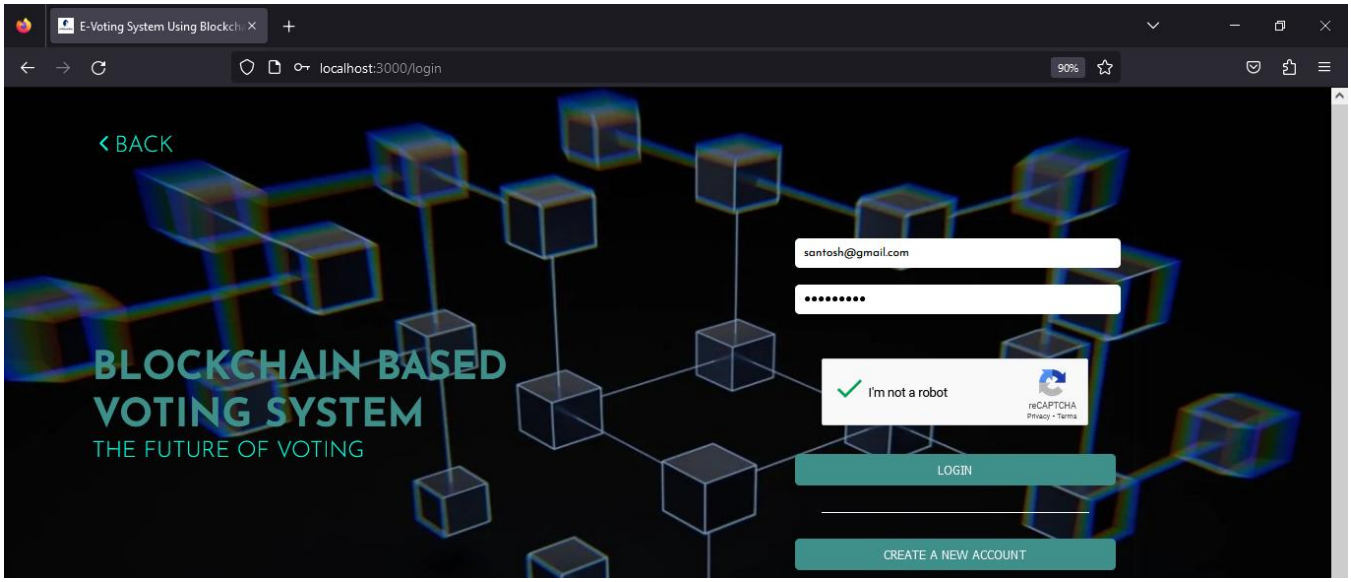
Step 1:- Firstly click on Login Button.




Voter have to login first by clicking on Login Button from Home Page.

Fig. 33 - Home Page.

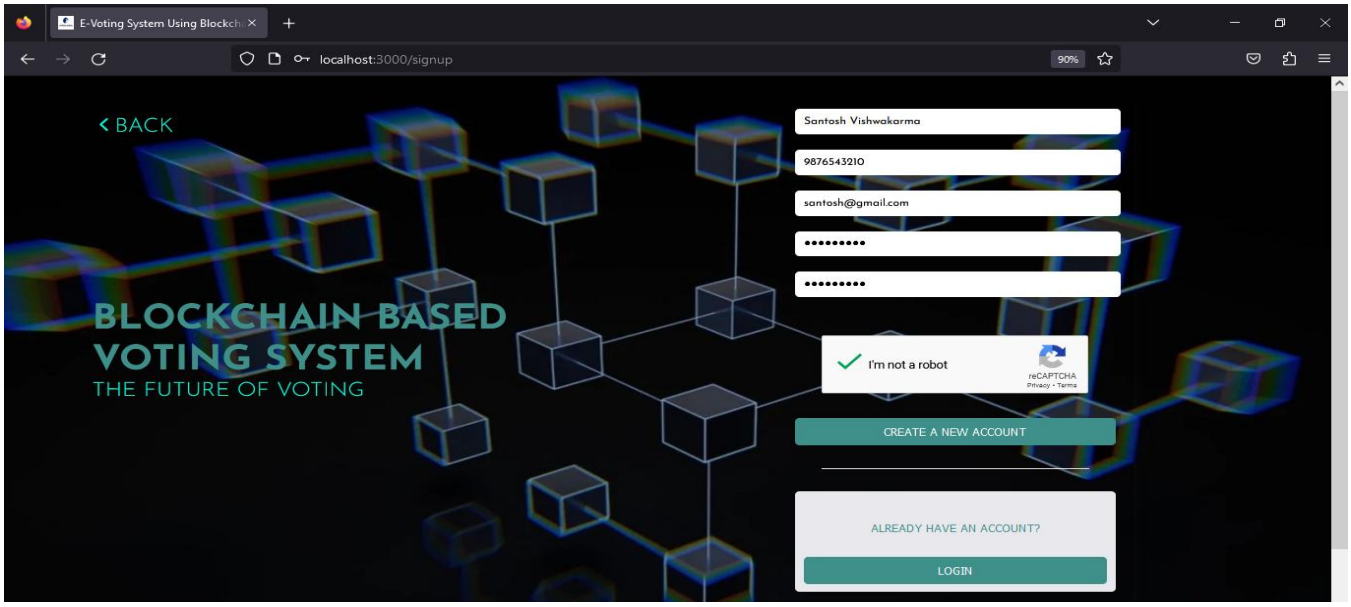
Step 2:- Enter your Login Details.




If you have Credential like Email and Password you can procced by entering the details if you don't have Email and Password just click on Create a New Account.

Fig. 34 - Login Page.

Step 3:- If you don't have login details just Create a New Account from this Page.




Just Enter your Details Like Name, Citizenship Number, Email, Password and click on Create a New Account Button and then after Verification Go to Login Page Enter your Credentials and Cast Vote.

Fig. 35 - Create New Account Page.

## 2. Verification Process:-

User after Creating a New Account their Account is in under process until the Admin Verify's their Account.

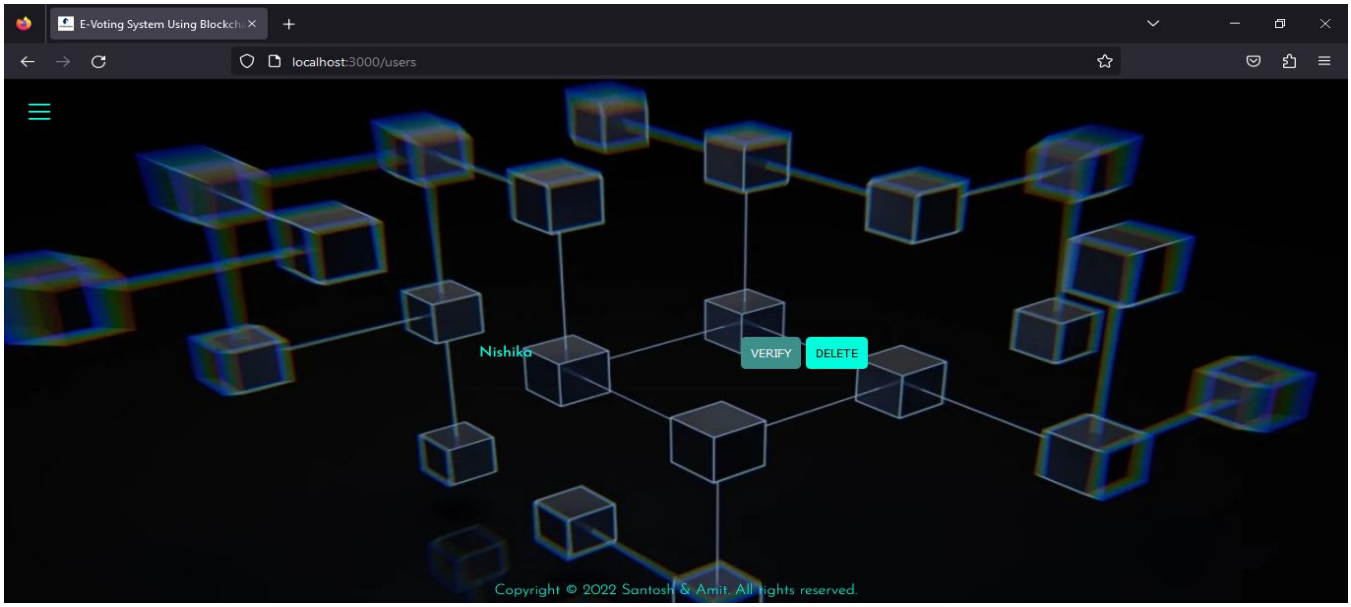



Fig. 36 - Verification Page.

## 3. Voting Process:-

After the Account Verification Voter or User can Cast a Vote.

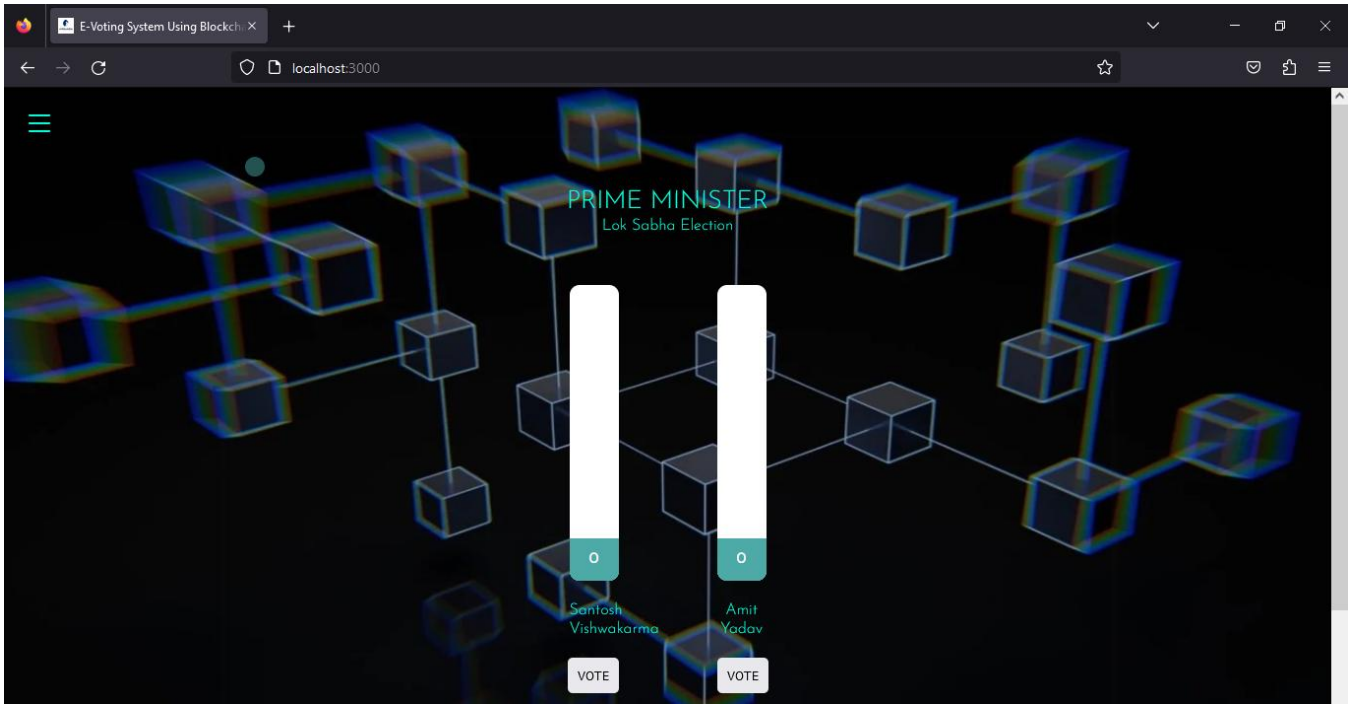



Fig. 37 - Vote Casting Page.

# CHAPTER 7: CONCLUSIONS

## 7.1 Conclusion

In this Project we analyzed and examined around the conventional voting framework additionally the focal points of execution blockchain based E-voting framework that employments different blockchain based instruments and utilizing case think about of manual voting prepare. After that we saw the comparison between conventional voting framework utilized and the blockchain based e voting framework. This framework will utilize blockchain as a arrange as well as database to store voter's data or credentials which is planning to utilize for their verification.

we presented a interesting, blockchain based electronic voting framework that utilizes keen contracts to empower secure and cost-efficient decision whereas ensuring voters protection By comparison to past work, we have appeared that the blockchain innovation offers a modern plausibility for equitable nations to development from the write and paper race plot, to a more fetched- and time effective race conspire, whereas expanding the security measures of the todays plot and offer unused conceivable outcomes of straightforwardness.

Voting is still a questionable point inside both political and logical circles. In spite of the presence of many exceptionally great cases, most of which are still in use; many more endeavors were either fizzled to supply the security and protection highlights of a conventional race or have genuine convenience and adaptability issues.

On the opposite, blockchain based e-voting arrangements, counting the one we have actualized utilizing the keen contracts and the Ethereum organize, address nearly all of the security concerns, like protection of voters, astuteness, confirmation and non-repudiation of votes, and straightforwardness of checking. However, there are too a few properties that cannot be tended to exclusively utilizing the blockchain, for case confirmation of voters (on the individual level, not on the account level) requires extra components to be coordinates, such as utilize of biometric Components.

## 7.1.1 Significance of the System

An e-voting system using blockchain technology has significant potential to improve the security, transparency, and efficiency of the voting process. By leveraging the unique properties of blockchain technology, such as immutability, decentralization, and transparency, an e-voting system can provide a tamper-proof, verifiable, and auditable record of every vote cast.

One of the most significant benefits of an e-voting system using blockchain technology is its ability to prevent voter fraud and manipulation. Because every vote is recorded on a blockchain ledger that is visible to all participants in the network, it becomes virtually impossible for any one individual or entity to manipulate or alter the outcome of an election.

Another key benefit of an e-voting system using blockchain technology is increased efficiency and speed. By leveraging blockchain's decentralized and distributed nature, the voting process can be streamlined, reducing the time and resources required to manage and conduct an election. This can lead to faster results and a more timely resolution of any disputes or challenges that may arise.

Finally, an e-voting system using blockchain technology can improve voter participation and accessibility. With the ability to vote online from anywhere, at any time, voters are no longer constrained by traditional voting methods, such as in-person voting or mail-in ballots. This can make voting more accessible to individuals who may have difficulty accessing traditional voting methods, such as those with disabilities or those who live in remote or rural areas.

# 7.2 Limitations of the System

While an e-voting system using blockchain technology has significant potential to improve the security, transparency, and efficiency of the voting process, it also has several limitations that must be addressed in order to ensure the integrity and accuracy of the election results. Some of the limitations of the system are:

1. **Technical expertise:-** E-voting systems using blockchain technology require a high level of technical expertise to design, develop, and maintain. This can be a barrier for small or under-resourced organizations or governments that lack the technical expertise to implement such a system.

2. **Cost:-** Implementing an e-voting system using blockchain technology can be expensive, requiring significant investments in hardware, software, and infrastructure. The cost of implementing and maintaining such a system may be prohibitive for some organizations or governments.

3. **Voter anonymity:-** While blockchain technology provides a tamper-proof record of every vote, it also raises concerns about voter anonymity. Because every vote is recorded on a public ledger, it may be possible to link a vote to a specific voter, compromising the privacy of the voting process.

4. **Security vulnerabilities:-** Although blockchain technology provides a high level of security, it is not immune to cyber attacks or security breaches. An e-voting system using blockchain technology is still vulnerable to attacks on the network or the software, and malicious actors may attempt to exploit these vulnerabilities to manipulate the election results.

**5. Accessibility:-** While e-voting systems using blockchain technology have the potential to improve accessibility and participation in the voting process, they may also create new barriers for individuals who lack access to the internet or who are not technologically literate.

# 7.3 Future Scope of the Project

The future scope of an e-voting system using blockchain technology is vast, as it has the potential to transform the way we conduct elections and ensure the integrity and transparency of the voting process. Here are some potential future developments and areas of focus for this project:

1. **Improved user experience:-** One area of focus for the future development of an e-voting system using blockchain technology is to improve the user experience. This can be achieved by creating more intuitive and user-friendly interfaces, simplifying the registration process, and providing clear instructions for voters.

2. **Increased scalability:-** As more organizations and governments adopt e-voting systems using blockchain technology, there will be a need for increased scalability. Future developments may focus on creating more efficient and scalable blockchain networks, improving consensus mechanisms, and optimizing the storage and retrieval of data.

3. **Enhanced security:-** Another area of focus for the future development of an e-voting system using blockchain technology is enhanced security. This can be achieved through the use of advanced encryption algorithms, multi-factor authentication, and other security measures to prevent unauthorized access and ensure the integrity of the voting process.

The current system uses ethereum which is public blockchain. It is permission less in nature as nothing is standing in the way of participation and anyone is able to engage with consensus mechanism, scaling obstacles have been encountered and throughput is relatively weak. To avoid such issues consortium blockchain can be used which combines elements from both public as well as private blockchain. The current project is built for small organization, but in future we would build it as a national voting system. In addition to the present fingerprint module which is used for authorization a facial recognition module would be incorporated for better security.

# REFERENCES


1. https://www.researchgate.net/
2. https://www.irjet.net/
3. https://stackoverflow.com/
4. https://www.mysql.com/
5. https://www.apachefriends.org/
6. https://trufflesuite.com/ganache/
7. https://www.wikipedia.org/
8. https://ieeexplore.ieee.org/document/8457919
9. https://www.itm-conferences.org/articles/itmconf/pdf/2020/02/itmconf_icacc2020_03001.pdf
10. https://www.researchgate.net/publication/351723225_Blockchain_Based_E-Voting_System
11. https://www.ijert.org/research/implementation-of-secure-voting-system-using-blockchain-IJERTV9IS060974.pdf
12. https://www.ijsdr.org/papers/IJSDR1905104.pdf
13. https://www.ijert.org/research/e-voting-using-blockchain-technology-IJERTV9IS070183.pdf

# APPENDIX

Report:-

The below is a Dataset of a User and Admin. This Dataset is Extracted from MariaDB Database to CSV file. In this CSV File the ID, Name, Citizenship Number, Email, Password, Admin and Verified Column has shown or included. In this Dataset 25 Voters and 2 Admins details are feeded into the database including their name, id, email, password for Voting. If we consider the Password column the Password is in Encrypted form has displayed and under the Admin Column which is shown by 1 they are Admins and those who have 0 under the admin column they are not verified for Admin.

Dataset of Voters and Admin:-

| | A | B | C | D | E | F | G |
|---|---|---|---|---|---|---|---|
| 1 | id | name | citizenshipNumber | email | password | admin | verified |
| 2 | 1 | John | 9860777906 | john@gmail.com | $2b$10$6sdkothEwAguhA0FytsGF.gcWPmTDB5hosif6rGX5FFJK8PdBgRHu | 0 | 1 |
| 3 | 2 | Liza | 9860777907 | liza@gmail.com | $2b$10$70yLw0dPhAD0py/iiGUInO7kkIGUmbMfa5BmXKGCXEID1ufTsqSQ6 | 0 | 1 |
| 4 | 3 | Ben | 9860777908 | ben@gmail.com | $2b$10$1DsQFSqUs3ufyDDRBd9wYuU5i9ihbnYR4GCYJsI3lzGXamwFWnr4S | 0 | 1 |
| 5 | 4 | Santosh Vishwakarma | 9874563210 | santosh@gmail.com | $2b$10$EH4vmFqAq7p/5XT3AVrYWe51QJK9Y2qnUq1rW/1JBHqPvxsZ2x7Na | 1 | 1 |
| 6 | 5 | Amit Yadav | 1234567892 | amit@gmail.com | $2b$10$7vLrsJD7J8j6Q3nWVmsi7.aXVpH0zK0guJEpd7axD7pzGiqb0FmLW | 1 | 1 |
| 7 | 6 | ANKIT | 72838888 | ankit@gmail.com | $2b$10$UNpIYvaQaUdRynYSpWj5Eefo9jjdU86SNBmcfucwv9u4qs5IQwjze | 0 | 1 |
| 8 | 9 | Aman Rai | 12345678901 | aman@gmail.com | $2b$10$IMYhv0Zk/WvTg8/yjCJWM.z43vHH2CtfjOw1fCUCp8UCKJ4FqPGOW | 0 | 1 |
| 9 | 11 | Anmol Kaur | 12345678902 | anmol@gmail.com | $2b$10$eH2h6y5tBzCS0W6PaED/bOEtC8saqwu2SDRDaRPqXG86t73trPaJK | 0 | 1 |
| 10 | 12 | Suraj Yadav | 12345678903 | suraj@gmail.com | $2b$10$hyI5lLN2oVebYdStjiD8aeap0L7thRRvY28XBb6F/79GpE02VPrZi | 0 | 1 |
| 11 | 13 | Nikhil Dubey | 12345678904 | nikhil@gmail.com | $2b$10$ZUIq3YCRjhmxsxgjCweeUerfwIjtitCCiEqEbXw9TIIHB7x44QWpO | 0 | 1 |
| 12 | 14 | Priya Yadav | 12345678905 | priya@gmail.com | $2b$10$G/q6z9L.TATHrzByet3JOO0nGLI4e0GCiQcKW3qOoQ.iYsCzRqTCS | 0 | 1 |
| 13 | 15 | Marylou Pereria | 12345678906 | marylou@gmail.com | $2b$10$IbFIFp0KqwVkNF3r8nFZ1.rDLOssrj5LxZ5oEFJKUkox12FPNI0am | 0 | 1 |
| 14 | 16 | Dhaval Mali | 12345678907 | dhaval@gmail.com | $2b$10$54AMP2MTvjgzqV5M76QXZuSOZdUFGbDI6OtHHcejnL0UEEZ5tK456 | 0 | 1 |
| 15 | 17 | Nidhi Shetty | 12345678908 | nidhi@gmail.com | $2b$10$JxseThDkTBtxY3dtIdCAMuEW2ToDSDaFYFw1hVRzVb.FixcDtoaMe | 0 | 1 |
| 16 | 18 | Vrinda Nair | 12345678909 | vrinda@gmail.com | $2b$10$J4o7/PRo4h6AUc3Uv6ICieVf8D4QP20LZHIOvRqI9zmm4eUMWtoHC | 0 | 1 |
| 17 | 19 | Naveen Anandhan | 12345678910 | naveen@gmail.com | $2b$10$.3LkGVjp/3DHEOFZXCvBo.e2LSf9U/wvodhJE7FDn4TRPD5MvayHe | 0 | 1 |
| 18 | 21 | Ashish Pal | 12345678911 | ashish@gmail.com | $2b$10$pjgxNo22jnR2ZaMH2r6weu603ohA2idOUZiMhdzj.OmT0pJuV.7G6 | 0 | 1 |
| 19 | 23 | Niki Thakur | 12345678913 | niki@gmail.com | $2b$10$M5pmXLg5ArDGjFUuOLMrdemyt2nCkAMX6P/DfbUbqWhsB3aGfm/PG | 0 | 1 |
| 20 | 24 | Purvi Rawal | 12345678915 | purvi@gmail.com | $2b$10$6nSRKDv8et5C/pOhd5ktmuK4Q2FvRfIk6bsTyS6Q6k1eEupFs1md6 | 0 | 1 |
| 21 | 25 | Elizabeth Leah George | 12345678916 | elizabeth@gmail.com | $2b$10$eM5528YOkDauC6zefm2znOUgOclwmYQP8BZvhD8Ok/BcDkKFYB7Km | 0 | 1 |
| 22 | 26 | Ashish Modi | 12345678917 | ashishmodi@gmail.com | $2b$10$DiI2b46l2gUsFP2y9GrWQOsk/uUCTijb5963MLMCCHhYoWbIBuAh6 | 0 | 1 |
| 23 | 27 | Vishakha Bagwe | 12345678918 | vishakha@gmail.com | $2b$10$LRLYyxClaa0TmY6jLfFWYOdah2p4.vf5K9F05zv0vSp2KL/.O7RGS | 0 | 1 |
| 24 | 28 | Mayuresh Shelke | 12345678919 | mayuresh@gmail.com | $2b$10$sTEOiWPY71PleLkfc/xD1updk4RrdoIYbm4IEBfZQFAyXdKKHQCrm | 0 | 1 |
| 25 | 29 | Roshni Singh | 12345678920 | roshni@gmail.com | $2b$10$1.qLWI84/65oCmHMs0wjfuiOF1I3J5YtoKTrQkkOsVq.vMy/x4hRm | 0 | 1 |
| 26 | 30 | Khushi Kawa | 12345678921 | khushi@gmail.com | $2b$10$sv.8c.HneVyyTvpKZo9/WOeznuhuFBUuH7kMHzgXpveVVAJfVMGUW | 0 | 1 |
| 27 | 31 | Akhtar Ansari | 12345678922 | akhtar@gmail.com | $2b$10$ga6NSR3txh06wrQURcdxBuhAZuwmDeJWVh8UotU6fX7.6IjKYvH.m | 0 | 1 |
| 28 | 32 | Sanya Shaikh | 9874589000 | sanya@gmail.com | $2b$10$d0YBNRp26E1oSH14ZB0z7OeSN6..vuJO2t5O/gbMmj7RB4nEs74Xy | 0 | 1 |

Fig. 38 - Dataset of Voters and Admin