LONDON METROPOLITAN UNIVERSITY

Islington college
(इस्लिङ्टन कलेज)

Module Code & Module Title

CS6P05NI Final Year Project

Assessment Type & Weightage

25% FYP Interim Report

Semester

2022-23 Autumn

Project Title: E-Voting Using Blockchain

Student Name: Mihir Rauniyar

London Met ID: 20049173

College ID: NP01NT4S210132

Submitted to: External Supervisor – Prabesh Hada

Internal Supervisor – Subekshya Pradhan

Assignment Due Date: 28th December, 2022

Assignment Submission Date: 27th December, 2022

I confirm that I understand my coursework needs to be submitted online via Google Classroom under the relevant module page before the deadline in order for my assignment to be accepted and marked. I am fully aware that late submissions will be treated as non-submission and a mark of zero will be awarded.

Acknowledgement

I would like to express my sincere gratitude to all of those who have contributed to the creation of this report.

I would like to thank Subekshya Pradhan ma'am and Prabesh Hada sir for their guidance and support throughout the research process. Their insights and expertise have been invaluable in helping me to understand the topic and to develop my ideas.

I would also like to thank Shurachith Cooperative and Mr. Rajan Ghimire for accepting my project idea and becoming the client.

Finally, I would like to express appreciation to my friends and families for their encouragement and support throughout the project. Without their help and understanding, this report would not have been possible.

Abstract

A blockchain-based e-voting website allows users to vote electronically on a decentralized and secure platform. The e-voting website can provide a dependable and trustworthy voting system that protects the integrity and privacy of the voting process by using the features of blockchain, such as immutability, transparency, and security.

Users can register and verify their identity via a secure authentication mechanism on this e-voting website. They may visit the portal after registering to examine a list of candidates and vote. The votes are tamper-proof recorded on the blockchain, ensuring that the election results are accurate and cannot be changed.

In general, the incorporation of blockchain technology into an e-voting platform has a variety of advantages, including improved security, transparency, and efficiency. It offers a trustworthy and dependable platform for holding elections and has the power to completely alter how elections are held in the future.

Table of Contents

Table of Figures

Table of Tables

Chapter 1. Introduction

1.1 Introduction

The right to vote is a constitutional right given to every citizen of any democratic country and Citizens, as stakeholders of the state, elect their respective representative via elections (ROJAS, 2005). Voting is still being conducted through paper ballots, regardless of the numerous technological advancements in today's society making elections inefficient as well as outdated (Javier Díaz-Santiso, 2021).

Even though democracies have started to use electronic voting for national scale elections, there are still no ideal, dependable, and efficient electronic voting system for people since the election requires one or more authorities for both authentication and protecting voters' privacy, and it is also difficult for voters to believe in the public authority that, it will always follow the rules or never be breached (Wei-Jr Lai, 2021).

The blockchain is globally divided and fully decentralized, meaning it has no supervisor or someone that can be blamed or rewarded (Söze, 2017). The Blockchain is made up of multiple irreplaceable blocks, i.e., a chain system (whenever a new block is established and included to blockchain, it replicates itself on its system, which remains on the internet, then simply syncs the same information on all the nodes operating in blockchain) representing a single source of truth (Söze, 2017). These features qualify blockchain to host major events like elections in its network.

1.2 Problem Scenario

- Paper ballot voting requires cumbersome manual work, a huge budget, and a substantial amount time and manpower for the overall election process (Simkhada, 2012). For example, the Election Commission prints around 20 million ballot papers, for which the country spends around NPR 110 million only on paper supply (Rijal, 2022).
- The existing system cannot guarantee prevention from unlawful manipulation of the data before or after casting a vote (Javier Díaz-Santiso, 2021). For example, some people seized the polling stations, tore up the remaining ballot papers, and ran away with the ballot boxes in two polling stations in Rautahat (Rijal, 2022).

- Due to the complexity of the balloting system every election many votes are disqualified because of various human errors.
- It also affects voter engagement. In the context of Nepal, about 4 million Nepalese could not vote simply because they were in a foreign country during election (Rijal, 2022).
- The voting sometimes gets halted and does not proceed as planned due to various natural disasters and human interferences (Javier Díaz-Santiso, 2021). For example, in the previous election, 79 polling centers were halted, and voting was postponed in 28 municipalities of 16 districts due to various reasons (Rijal, 2022).

1.3 Project as a Solution

- E-voting is a faster and more economy efficient and can be the best alternative in countries like Nepal where the infrastructure is poor and the population highly dispersed (Simkhada, 2012).
- The use of blockchain network covers up the security problems occurring in a regular centralized e-voting system (Javier Díaz-Santiso, 2021).
- The proposed project increases the accessibility for the voters and simultaneously decreasing voter fraud (Javier Díaz-Santiso, 2021).
- The chances of voting being halted drops to negligible as the blockchain network would be unaffected even after failure of one or more nodes (Cabanac G, 2022).
- The e-voting system could provide sufficient transparency while not raising any privacy issues (Cabanac G, 2022).
- Through the proper use of smart contracts each ballot could be counted anonymously, correctly, and efficiently (Javier Díaz-Santiso, 2021).

1.4 Aim and Objectives

1.4.1 Aim

- The aim of this project is to create an electronic voting system that provides decentralization as well as immutability and improves the current voting systems in terms of performance and reliability.

1.4.2 Objectives

The objectives for the completion of the proposed project are:

- To learn more about HTML, CSS and JavaScript and be able to create a user-friendly website for e-voting system.
- To dive deeper into the study of blockchain network and its functionalities.
- To become familiar with truffle (coding environment for blockchain).
- To learn more about Web3.js libraries and its alternatives.
- To understand and become capable of implementing smart contracts as per the requirements of the project.

1.5 Report's Structure

1.5.1 Background

The background section is intended to set the context for the information that follows. It comprises a quick introduction of the issue at hand, as well as any important background information required for the reader to grasp the context of the report. The background section may also include a description of any research or data that has been gathered and processed, as well as a summary of any major results or conclusions made from this study.

1.5.2 Development to date

The development part of a report is intended to present and explain the project's significant results or development in greater depth. It may contain descriptions of any study or data gathered, as well as any analysis or interpretation of this data. In general, the development section is intended to give a more in-depth study of the project's progress as well as a deeper grasp of the issue under discussion.

1.5.3 Analysis of Progress

An analysis of progress entails reviewing the progress achieved toward the project's goals and objectives. This may entail evaluating any completed milestones, examining any data or metrics gathered, and assessing any obstacles or issues that have developed throughout the course of the project.

1.5.4 Further Work

The future work section is used to detail any planned or probable future work that will be required to continue progress toward the project's goals and objectives. This might involve talking about any more research or data collecting that will be required, as well as any new tactics or plans that have been recognized as necessary to fulfill the project's goals.

Chapter 2. Background/ Literature Review

2.1 Client's Description and Requirements

2.1.1 Client's Name, Description and Letter

- **Name of the Client:**

Shurachith Cooperative (Rajan Ghimire)

- **Client's Description:**

Shurachith Cooperative, a renowned co-operative formed in 2067 B.S., is the client for this final year project (FYP). The firm has a good reputation in the sector and a dedicated customer base. Shurachith Cooperative, based in Hetauda, provides standard banking services such as checking and savings accounts, as well as loans.

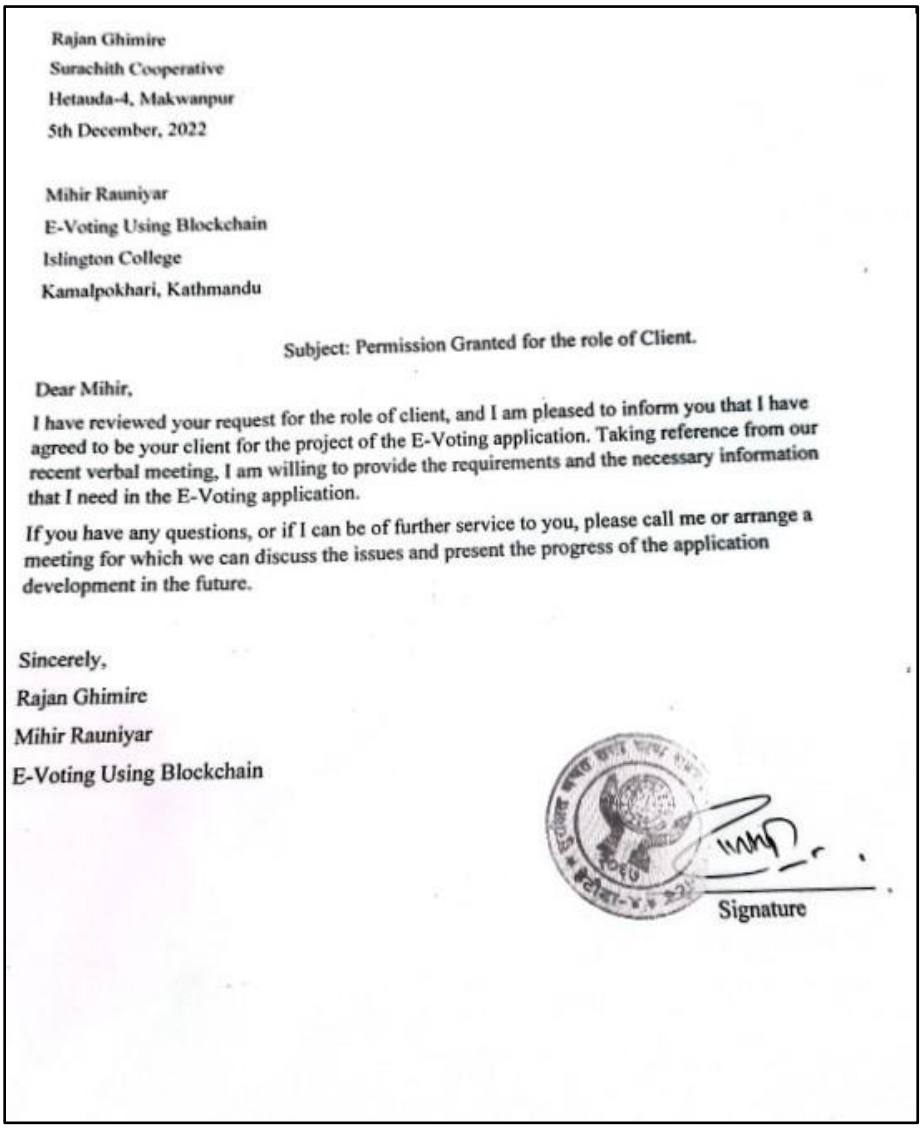
As part of the FYP, Mr. Rajan Ghimire representative of the Shurachith Cooperative's management will review and make suggestions as well as collaborate to create an e-voting website using blockchain technology and is willing to cooperate with the concepts and satisfy the requirements that have been specified.

2.1.2 Client's Requirements

- The website should be easy-to-use and have user-friendly UI.
- The project should be secure.
- The project should maintain vote and voter's privacy i.e., voters should be able to cast their ballots in private, without fear of reprisal or coercion.
- The project should maintain transparency and fairness i.e., voting process should be transparent, impartial, and free from undue influence.

- The project should be auditable and be accurate i.e., the results of the election should be able to be verified and audited to ensure the accuracy and integrity of the process.

2.1.3 Client's Approval Letter



Rajan Ghimire
Surachith Cooperative
Hetauda-4, Makwanpur
5th December, 2022

Mihir Rauniyar
E-Voting Using Blockchain
Islington College
Kamalpokhari, Kathmandu

Subject: Permission Granted for the role of Client.

Dear Mihir,

I have reviewed your request for the role of client, and I am pleased to inform you that I have agreed to be your client for the project of the E-Voting application. Taking reference from our recent verbal meeting, I am willing to provide the requirements and the necessary information that I need in the E-Voting application.

If you have any questions, or if I can be of further service to you, please call me or arrange a meeting for which we can discuss the issues and present the progress of the application development in the future.

Sincerely,
Rajan Ghimire
Mihir Rauniyar
E-Voting Using Blockchain

Signature

Figure 1: Client's Approval Letter

2.2 Understanding the Project

2.2.1 Project Elaboration

An Ethereum blockchain voting website is a web-based application that utilizes Ethereum's decentralized, open-source blockchain platform to enable secure and transparent voting in elections or other types of decision-making processes. The website would use a range of technologies and features to enable this functionality, including:

Smart contracts: Smart contracts are self-executing contracts with the terms of the agreement between buyer and seller being directly written into lines of code. The voting website would use smart contracts to automate the voting process, including the counting and reporting of votes.

User interface: The voting website would have a user interface that allows voters to cast their ballots electronically, and provides them with real-time updates on the voting process.

Overall, an Ethereum blockchain voting website would be a complex and sophisticated application that leverages the power of blockchain technology to enable secure and transparent voting. It would be designed to be user-friendly and easy to use, while also providing the necessary security and transparency to ensure the integrity of the voting process.

2.2.2 Project Deliveries

After the completion of the project, an e-voting website with the following features is expected:

- A decentralized E-voting system which is fair, accurate and trustworthy.
- The project will be built as a User-friendly website with easy-understandable UI.
- The project will include user registration, user login, and admin login.
- The application will store voter information so that voters may login and use their voting rights.
- At the time of registration, voters will be asked for their full name, age, mobile number, email address, and other information, after which they will be granted the right to vote.

- The user can vote for one of the candidates on the ballot. Voters can only vote for one candidate every election.
- The user can vote for one of the nominees. Voters can only vote for one candidate every election.
- Voters can also view a list of Candidates in their region via the website.

2.3 Summary of Pre-Survey's Responses

A survey was conducted to gather insights about the attitudes and behaviours of voters regarding e-voting using blockchain technology. The survey was conducted online and included 26 responses from different cities across the country.

Overall, the results of the survey showed that the majority of respondents (73.1%) at least heard of the blockchain technology and (26.9%) understood its potential benefits. The main motivations for using e-voting with blockchain cited by respondents included increased security, transparency, and the potential for faster and more efficient vote counting.

However, the survey also revealed that there were some concerns about the use of e-voting with blockchain. The main concerns cited by respondents included the potential for technical failures, a lack of trust in the about the proper implementation of electronic voting systems, and a preference for traditional in-person voting.

Overall, the survey results suggest that while there is strong interest in the potential of e-voting with blockchain, there are also some reservations about its adoption. Further research could be conducted to address these concerns and to explore ways to increase trust and confidence in the security and integrity of e-voting systems using blockchain technology.

For More Details, Click Here

2.4 Similar Projects

2.4.1 Project 1: Candida18/ Online-Voting-System

Author/s: Candida Noronha

Project Description: The term "ONLINE VOTING SYSTEM" refers to an online voting method. Authorized voters can use this technology to vote online instead of visiting to a polling place. A database is kept in which all the names of voters with comprehensive information are preserved.

Front-End: HTML, CSS, JAVASCRIPT and Bootstrap

Back-End: PHP

Database: My SQL

GitHub Link: https://github.com/Candida18/Online-Voting-System

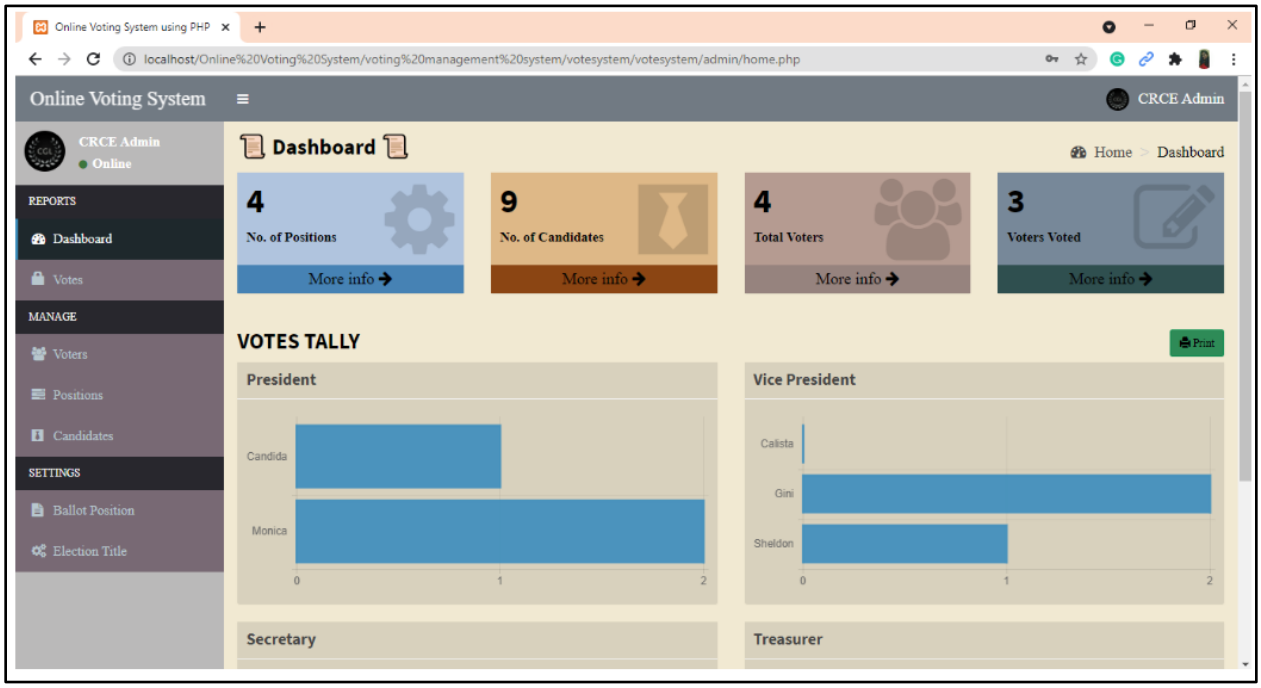


Figure 2: Project 1 (Noronha, 2021)

2.4.2 Project 2: vidhikhatwani / Polling-System-Java-Application

Author/s: Vidhi Khatwani

Project Description: The Online Polling System is a Java program with essential GUI and database attributes that dictate the software needs for this project. This project is intended for low-volume voting procedures such as college elections or feedback forms.

We will give an online facility for voting on various questions provided by the administrator/organizer in this project.

GitHub Link: https://github.com/vidhikhatwani/Polling-System-Java-Application

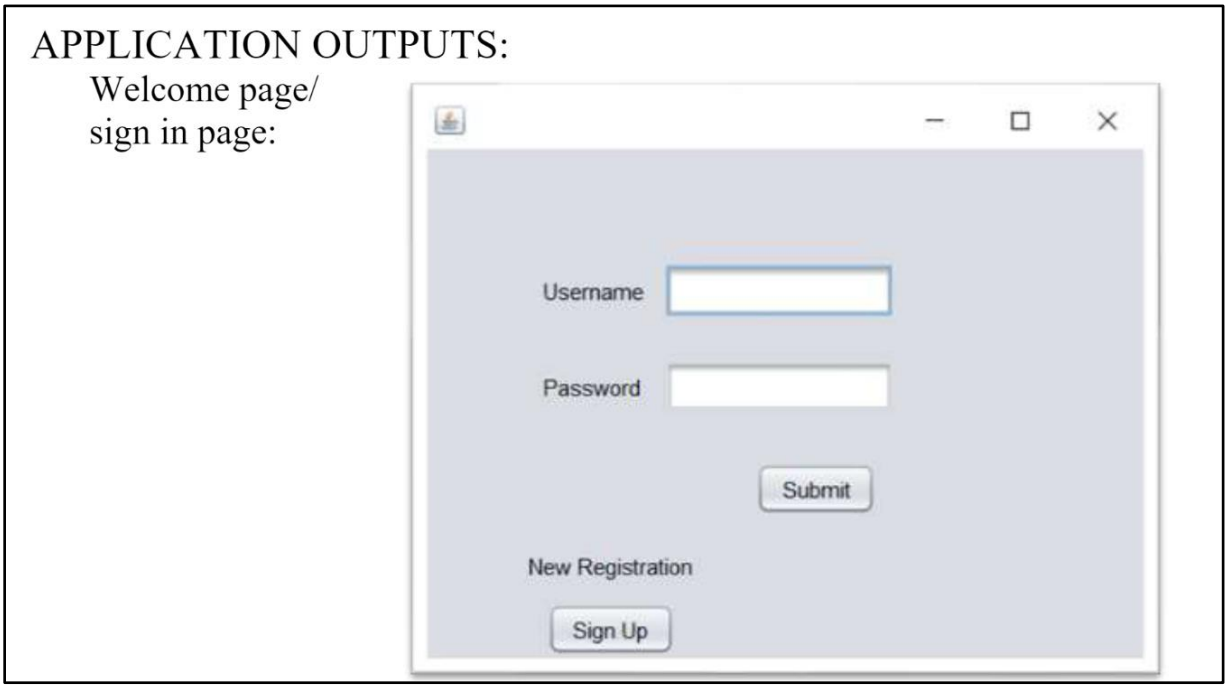


Figure 3: Project 2 (Khatwani, 2022)

2.4.3 Project 3: shah-deep / Online-Voting-System

Author/s: Deep Shah

Project Description: It's a desktop program built in Python that makes use of socket technology. It benefits from synchronous multithreading. Python socket programming might be used to build a system in which voters can connect to a server through the internet, cast their ballots, and have their votes recorded and tabulated in real-time. E-voting in Python utilizing socket programming is a simple and efficient way for voters to cast their ballots electronically without having to physically visit a polling location.

GitHub Link: https://github.com/shah-deep/Online-Voting-System

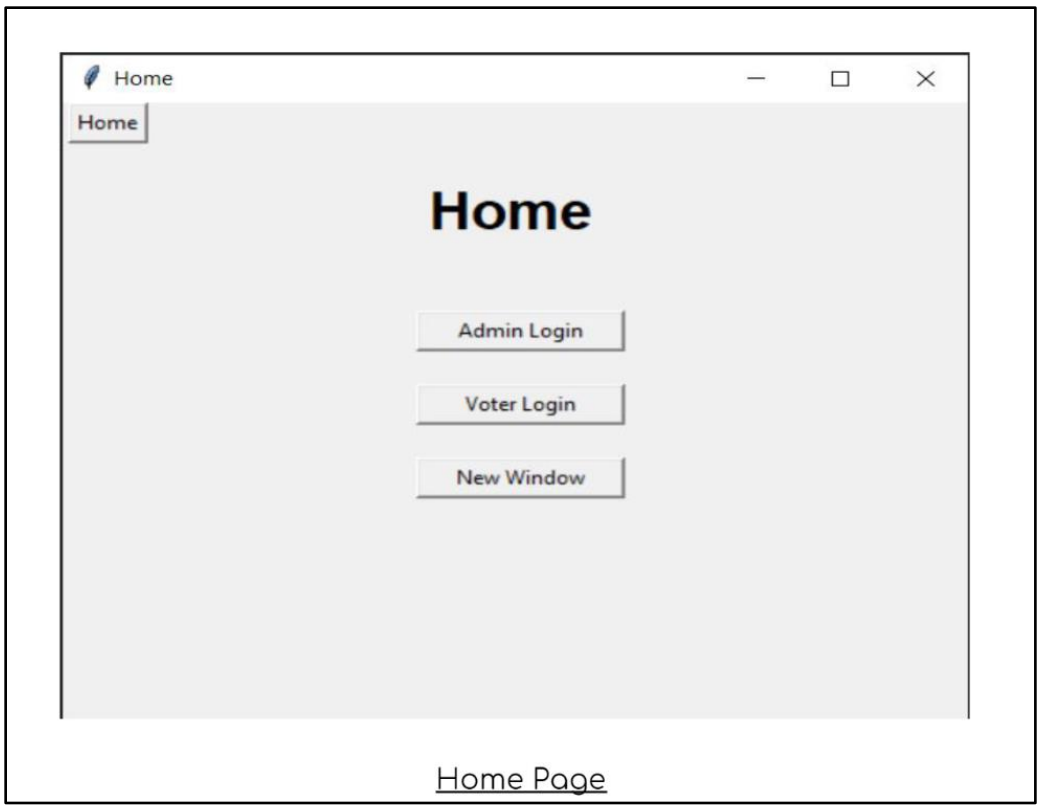


Figure 4: Project 3 (Shah, 2022)

2.5 Comparison Table

S.N.	Features	Project 1	Project 2	Project 3	This Project
1	Uses Blockchain	X	X	X	✓
2	Provides Authenticity	✓	✓	✓	✓
3	Provides Verifiability	✓	✓	X	✓
4	Is Widely Accessible	✓	✓	X	✓
5	Is Secure	X	X	X	✓
6	Is Scalable	✓	✓	X	✓
7	Maintains Vote Privacy	X	X	✓	✓

Table 1: Comparison Table

Chapter 3. Development to Date

3.1 Considered Methodologies

- Waterfall Methodology

The fundamental life cycle model for software development is the waterfall model. It is crucial since the conventional waterfall model serves as the foundation for all other software development life cycle models. The life cycle is divided into many segments according to the traditional waterfall paradigm. This model takes into account the possibility of starting a phase after the preceding phase has ended. That is, the input for the following phase will be the output of the previous one. As a result, the development process may be viewed as a waterfall's sequential flow. The phases do not overlap in this instance (Geeks for Geeks, 2022).

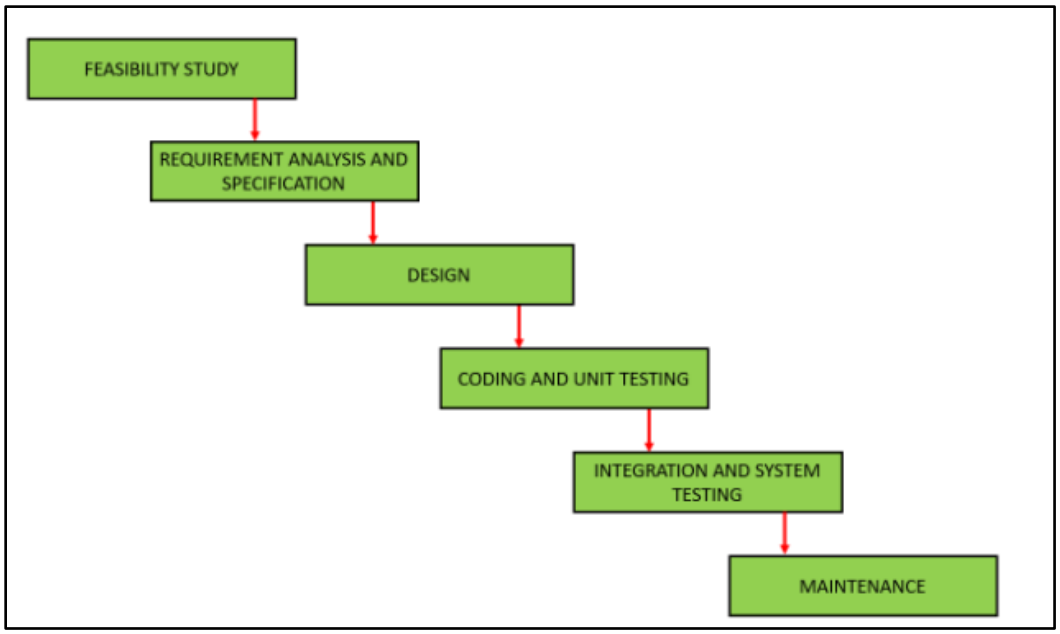


Figure 5: Waterfall Methodology (Geeks for Geeks, 2022)

- Agile Methodology

It is a process in which the need and its solution emerge via the collaborative activity of teams and the client. The activity is divided into numerous phrases, and continual

improvement and iteration are accomplished by communicating with stakeholders. Sprints are the names given to the divided portions (Young, 2013).

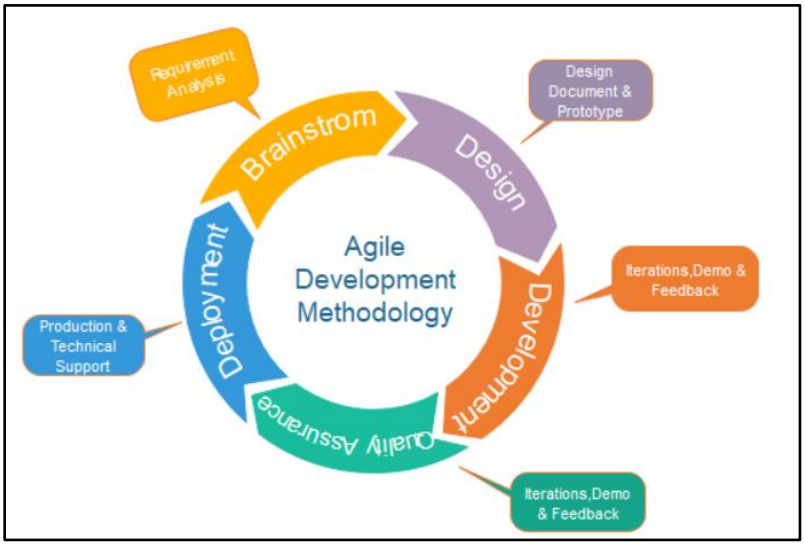


Figure 6: Agile Methodology (JavatPoint, 2022)

3.2 Selected Methodologies

The methodology chosen for the development and completion of this project is Prototype Methodology. It is explained below:

- Prototype Methodology

When users do not know the specific project needs ahead of time, the prototyping model is employed. In this approach, a prototype of the eventual product is created, tested, and improved based on consumer input until a final acceptable prototype is reached, which serves as the foundation for building the final product (Young, 2013).

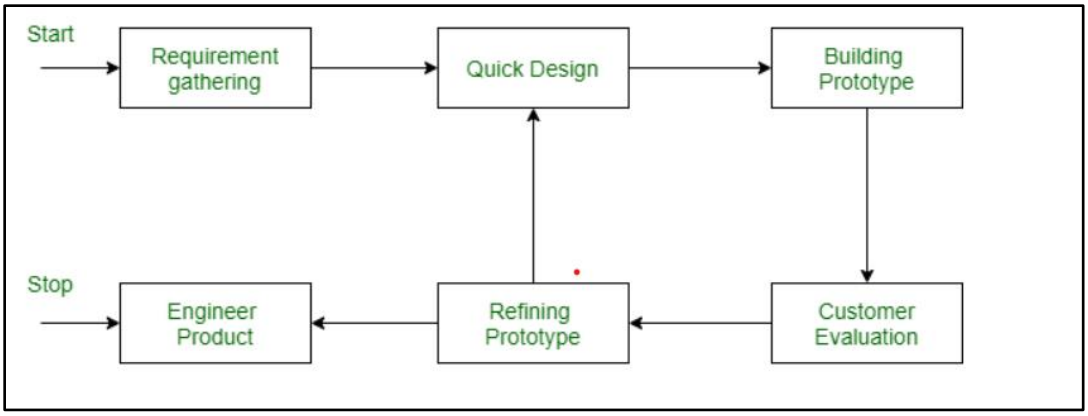


Figure 7: Prototype Methodology (Geeks for Geeks, 2022)

The advantage and disadvantages of Prototype methodology are:

Advantage: Allows designers and developers to test out new ideas and concepts fast and cheaply: Because prototypes are often simpler and less comprehensive than finished products, they may be manufactured and tested more rapidly and inexpensively than completely completed products. As a result, prototype technique is a versatile and effective tool for exploring many design possibilities and gathering feedback.

Disadvantages: Creating prototypes may be time-consuming, especially if numerous prototypes are required to properly test and perfect the concept. This may necessitate a large expenditure of time and resources, which may be detrimental to some initiatives.

3.3 Phases of Prototype Methodology for Development

The phases of the prototype methodology for the development of this project are explained below:

Planning and scoping: This step entail establishing the e-voting system's aims and objectives, as well as outlining any particular needs or restrictions that must be considered.

Design and development (Front-End): The design of the e-voting system is created and polished during this phase. This includes building wireframes, front-end codes, testing alternative design elements, and getting client input.

Design and development (Smart Contract): During this phase, the code for the e-voting system's smart contracts is written. This step also includes experimenting with various coding styles and selecting the best efficient code for the smart contract.

Testing: The prototype e-voting system is tested in this step to confirm that it is functional, secure, and user-friendly. This includes client's and developer's tests as well as evaluating the system's technical backbone.

Refinement: The prototype e-voting system is refined and enhanced as needed based on the results of testing. This may include changing the design, adding new features, or correcting any flaws or difficulties discovered during testing.

Deployment: The e-voting system is ready for use in an election after it has been improved and tested. This may include establishing the technological infrastructure required to support the system, as well as any voter registration and voting processes that may be required.

3.4 Work-breakdown Structure

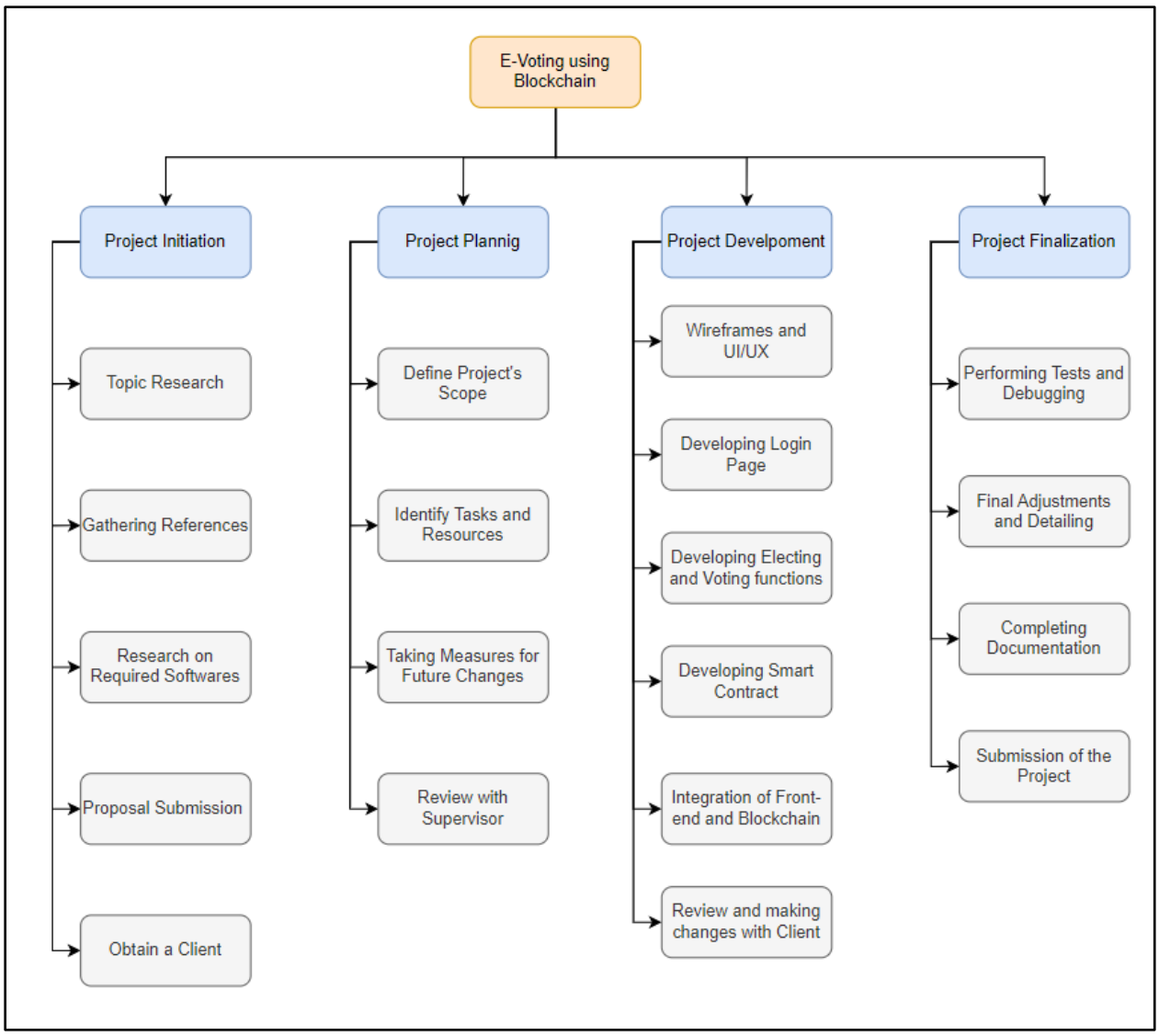

E-Voting using Blockchain
Project Initiation
Topic Research
Gathering References
Research on Required Softwares
Proposal Submission
Obtain a Client
Project Plannig
Define Project's Scope
Identify Tasks and Resources
Taking Measures for Future Changes
Review with Supervisor
Project Develpoment
Wireframes and UI/UX
Developing Login Page
Developing Electing and Voting functions
Developing Smart Contract
Integration of Front-end and Blockchain
Review and making changes with Client
Project Finalization
Performing Tests and Debugging
Final Adjustments and Detailing
Completing Documentation
Submission of the Project

Figure 8: WBS

3.5 Gantt Chart

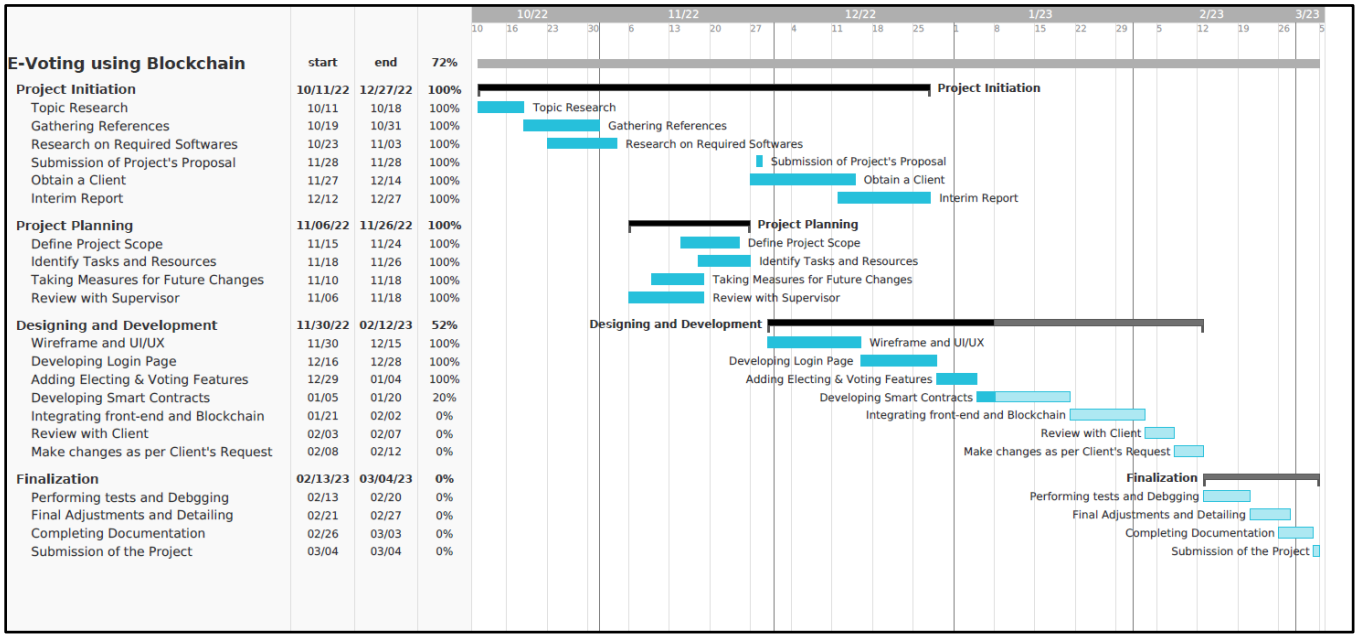


Figure 9: Gantt Chart

3.6 Milestones

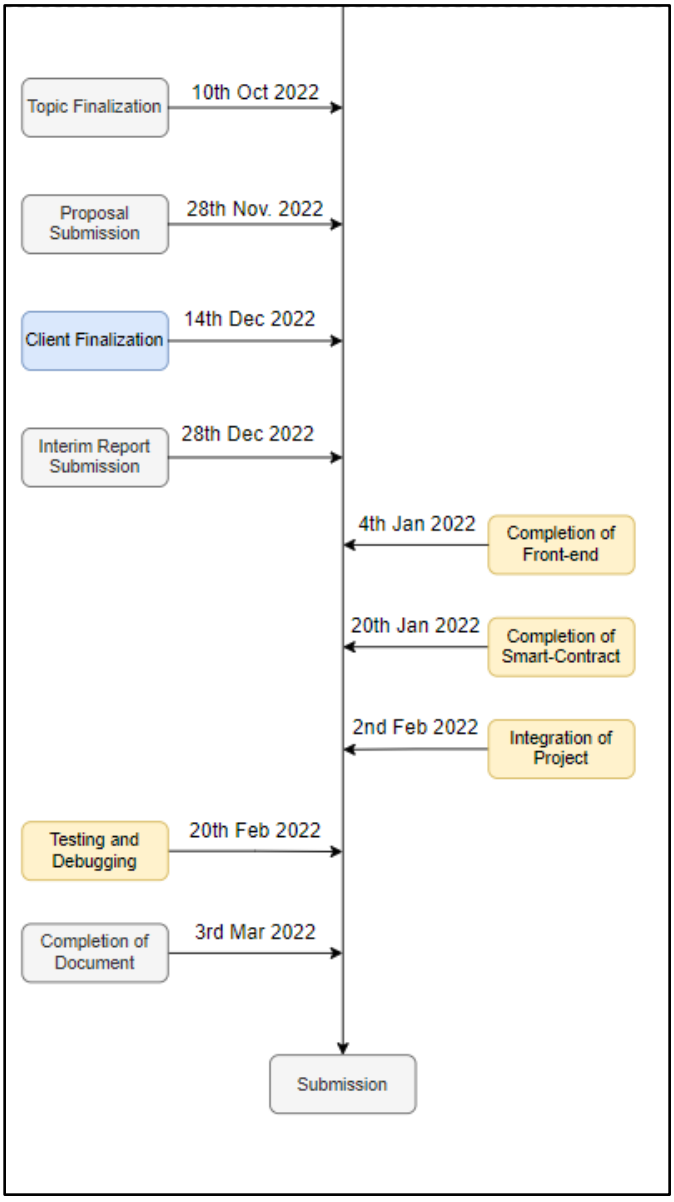


Figure 10: Milestones

Chapter 4. Analysis of Progress

4.1 Progress Table

SN	Tasks	Status	Progress (%)
1	Topic Selection	Completed	100%

2	Topic Research	Completed	100%
3	Gathering Reference	Completed	100%
4	Research on Required Software	Completed	100%
5	Proposal Submission	Completed	100%
6	Obtain a client	Completed	100%
7	Conduct a Survey	Completed	100%
8	Define Project's Scope	Completed	100%
9	Identify Tasks and Resources	Completed	100%
10	Taking Measures for Future Changes	Completed	100%
11	Wireframes and UI/UX	Completed	100%
12	Developing Front-end	Completed	100%
13	Developing Smart Contracts	Just Started	20%
14	Integration of Front-end and Smart Contract	Incomplete	0%
15	Review with Client	Incomplete	0%
16	Performing Tests	Incomplete	0%
17	Debugging	Incomplete	0%
18	Final Adjustments and Detailing	Incomplete	0%
19	Final Documentation	Incomplete	0%

Table 2: Progress Table

4.2 Progress Review

4.2.1 Project Plan, Design, and Requirements

Regarding the front-end development, the design and development of the user interface is functional and ready for testing. The wireframes for the system have also been completed, providing a clear and detailed blueprint for the overall system architecture.

Wireframe:

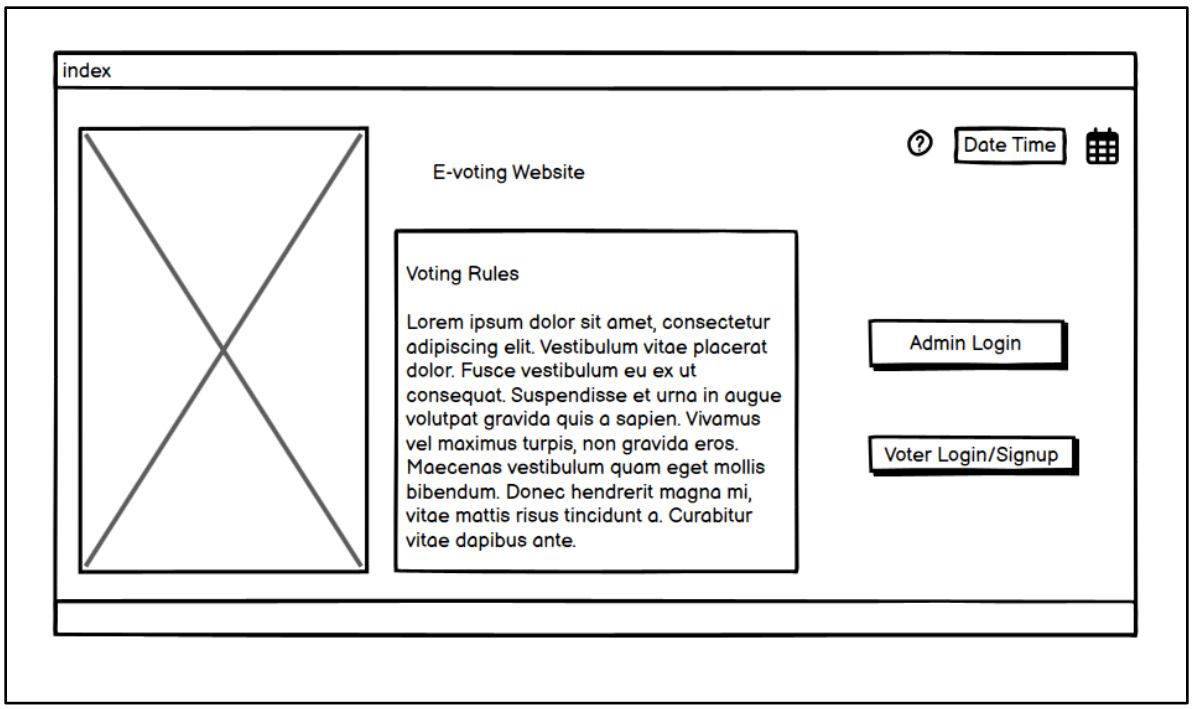


Figure 11: Wireframe: Login Page

Login Page:

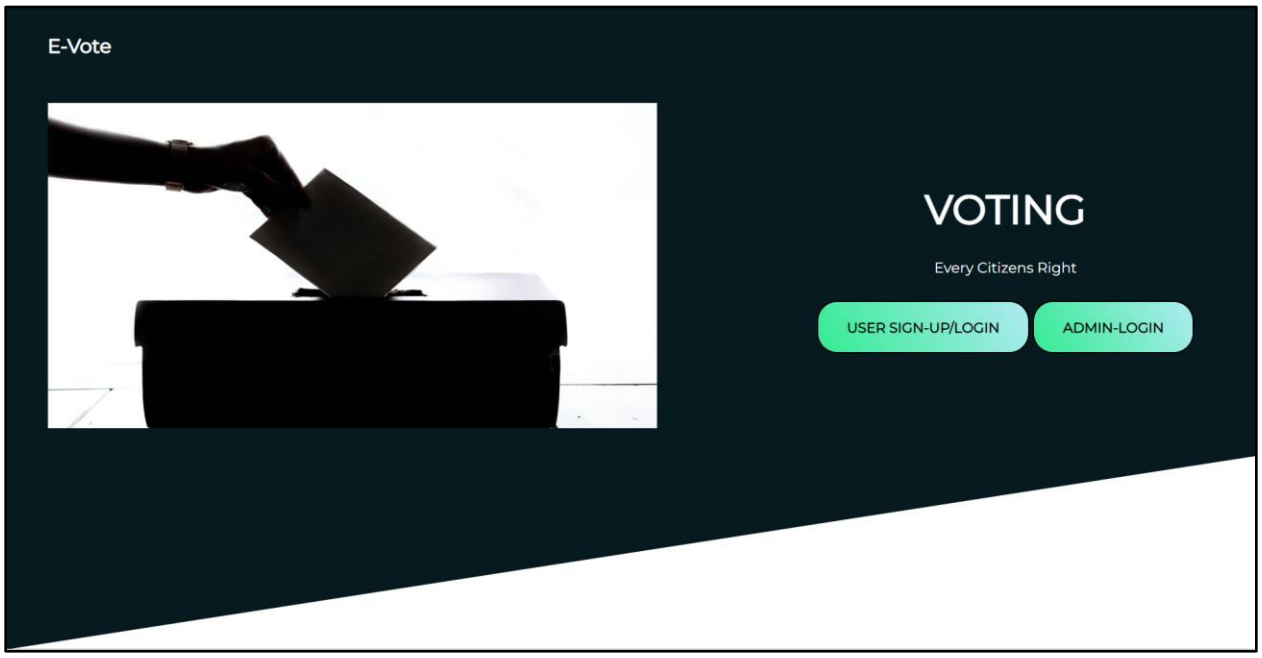


Figure 12: Login Phase

In terms of the smart contracts, a small portion of the necessary code is completed, and is currently being worked on. The smart contracts are a crucial component of the e-voting system, as they will be used to automate the voting process and ensure the security and transparency of the system.

Smart Contract:

```
// Import necessary libraries
import 'https://github.com/OpenZeppelin/openzeppelin-
solidity/contracts/math/SafeMath.sol';
import 'https://github.com/OpenZeppelin/openzeppelin-
solidity/contracts/utils/Address.sol';

// Use SafeMath library to ensure safe arithmetic operations
using SafeMath for uint256;

// Use Address library to perform checks on addresses
using Address for address;

// Smart contract events
event VoterRegistered(address voter);
event VoteCast(address voter, uint256 candidateId);
event ElectionClosed();
event ElectionResult(uint256 candidateId, uint256 votes);

// Smart contract variables
uint256 public totalVoters;
uint256 public totalCandidates;
uint256 public startTime;
uint256 public endTime;
bool public electionClosed;
```

Figure 13: Smart Contract

To See Other Images, Click Here.

4.2.2 Progress Timeline

The project has moved forward in accordance with the time estimations in the Gantt chart. The project's study and learning about blockchain and smart contracts began prior to the creation of the Gantt chart. Following that, wireframes were created. The pre-survey responses, client approval for the project, and client approval of the wireframe design were obtained prior to constructing the front-end. After gathering all of the essential processes for the project's start, the front-end portion of the website was created. The supervisor was then presented and approved it.

4.2.3 Action Plan

The action plan for future progress on the e-voting system using blockchain technology is as follows:

- Complete the remaining smart contracts and integrate the system with the blockchain.
- Establish a timeline for the completion of the smart contracts and the integration of the system with the blockchain.
- Assign specific tasks to team members and provide support as needed.
- Conduct testing and debugging of the system.
- Identify any issues or bugs in the system and work to resolve them.
- Conduct user acceptance testing to ensure that the system is fully functional and ready for deployment.
- Deploy the e-voting system on the blockchain.
- Follow the established deployment plan to ensure a smooth and successful rollout of the system.
- Monitor the system during the deployment process and address any issues that may arise.
- Launch the e-voting system.

Chapter 5. Further Work

5.1 Phases to Complete

- Smart Contract

The smart contract phase is a critical component of the development process for an e-voting system using blockchain technology. During the smart contract phase, it is required to develop the necessary code for the smart contracts that will be used in the e-voting system. This includes contracts for voting, tallying the votes, and recording the results of the election.

- Integration of Front-end and Smart Contract

During the integration phase, the work to connect the front-end of the e-voting system with the smart contracts that have been developed will be carried out. This involves creating a user interface that allows voters to cast their ballots electronically and view the results of the election, as well as integrating the system with the blockchain to ensure that the votes are recorded and counted accurately.

- Testing and Debugging

There are several different types of tests that are conducted during this phase i.e., unit testing, integration testing, and acceptance testing. Unit testing involves testing individual components of the system to ensure that they are working correctly, while integration testing involves testing how the different components of the system work together. Acceptance testing involves testing the system from the perspective of the end user to ensure that it meets the needs and expectations of the users.

- Documentation

Documentation is a portion in which extensive information about the project, such as its goal, aims, techniques, outcomes, and conclusions, is provided. It should also include any important project background information and context. It is critical to present a clear and comprehensive description of the project in this part, including its goals and strategies for achieving them.

References and Bibliography

Cabanac G, L. C., 2022. Retraction: A Secure Digital E-Voting Using Blockchain Technology.

Geeks for Geeks, 2022. *Geeks for Geeks.* [Online]
Available at: https://www.geeksforgeeks.org/software-engineering-prototyping-model/
[Accessed 25 November 2022].

Geeks for Geeks, 2022. *Geeks for Geeks.* [Online]
Available at: https://www.geeksforgeeks.org/software-engineering-classical-waterfall-model/
[Accessed 26 November 2022].

JavatPoint, 2022. *JavatPoint.* [Online]
Available at: https://www.javatpoint.com/software-engineering-agile-model
[Accessed 23 November 2022].

Javier Díaz-Santiso, P. F.-L., 2021. E-Voting System Using Hyperledger Fabric Blockchain and Smart Contracts. *Engineering Proceedings,* 7(11), p. 3.

Khatwani, V., 2022. *GitHub.* [Online]
Available at: https://github.com/vidhikhatwani/Polling-System-Java-Application
[Accessed 14 December 2022].

Noronha, C., 2021. *GitHub.* [Online]
Available at: https://github.com/Candida18/Online-Voting-System
[Accessed 14 December 2022].

Rijal, A., 2022. *Nepal Electronic Forum.* [Online]
Available at: https://nepaleconomicforum.org/electronic-elections-in-nepal-understanding-the-past-present-and-the-future/
[Accessed 24 November 2022].

ROJAS, H. S., 2005. A COMPARATIVE STUDY OF THE OVERSEAS VOTING LAWS. *Development Associates,* 17(1), p. 14.

Shah, D., 2022. *GitHub.* [Online]
Available at: https://github.com/shah-deep/Online-Voting-System
[Accessed 14 December 2022].

Simkhada, K., 2012. *E-Voting: Possibilities and Challenges in the Nepalese Context,* s.l.: s.n.

Söze, K., 2017. *BLOCKCHAIN Novice to Expert.* 1 ed. s.l.:s.n.

Vikram Dhillon, D. M. M. H., 2017. *Blockchain Enabled Applications.* 1st ed. s.l.:APRESS.

Wei-Jr Lai, J.-L. W., 2021. *An efficient and effective Decentralized Anonymous Voting System,* s.l.: s.n.

Young, D., 2013. Software Development Methodologies.

Appendix

Appendix 1: Pre-Survey Responses

Responders:

kamalthap58@gmail.com	thunderlordnik712@gmail.com
rauniyarsujal123@gmail.com	adhikarikushal4444@gmail.com
birajbhatta58@gmail.com	rabinpathak246@gmail.com
aidenxettri2001@gmail.com	anjanacharya.077@kathford.edu.np
baniyajitendra245@gmail.com	aasishstha09op1@gmail.com
kishortimalsina.17@gmail.com	paudyalpankaj26@gmail.com
024.shuvam@gmail.com	aayushbidari460@gmail.com
oshanr27@gmail.com	sanjelsuccess@gmail.com
mdrezatausif06@gmail.com	profile.archak@gmail.com
shresthapratik795@gmail.com	demonish100@gmail.com
dipeshdahal08@gmail.com	arushrijal7@gmail.com
nischalsilwalhtd@gmail.com	kunal.rocker2017@gmail.com
razztamang888@gmail.com	chaulagainking00@gmail.com

Responses:

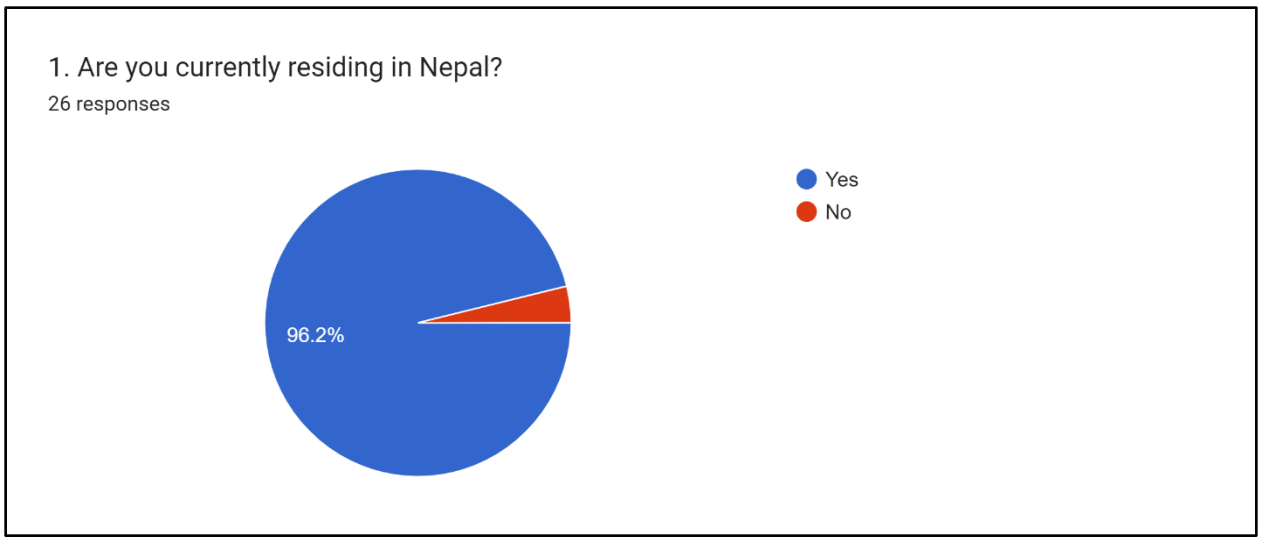


Figure 14: Q1

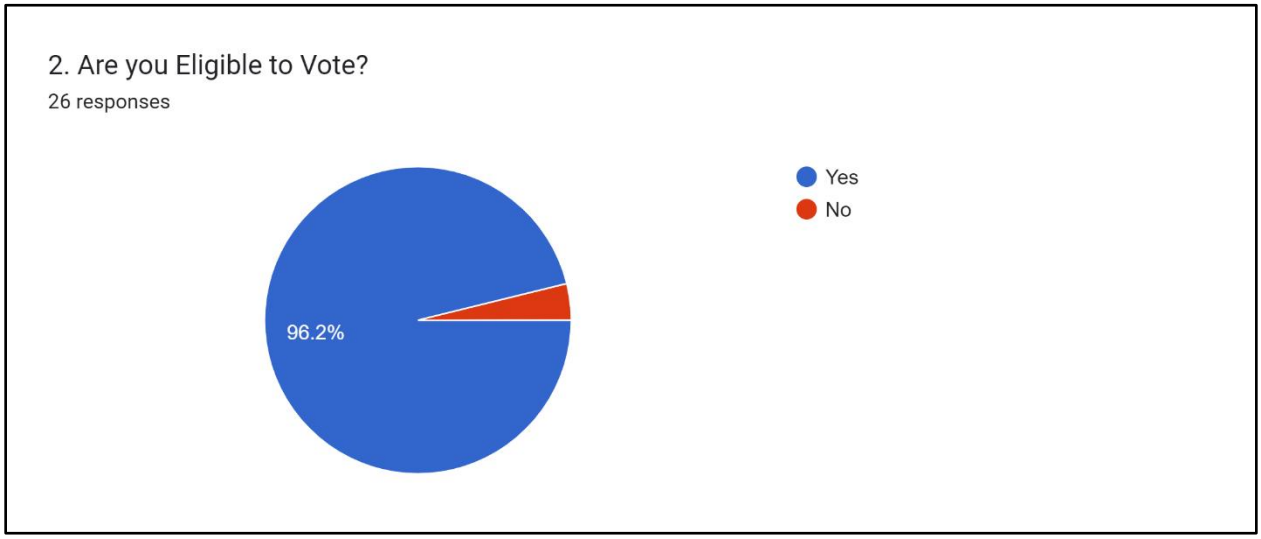


Figure 15:Q2

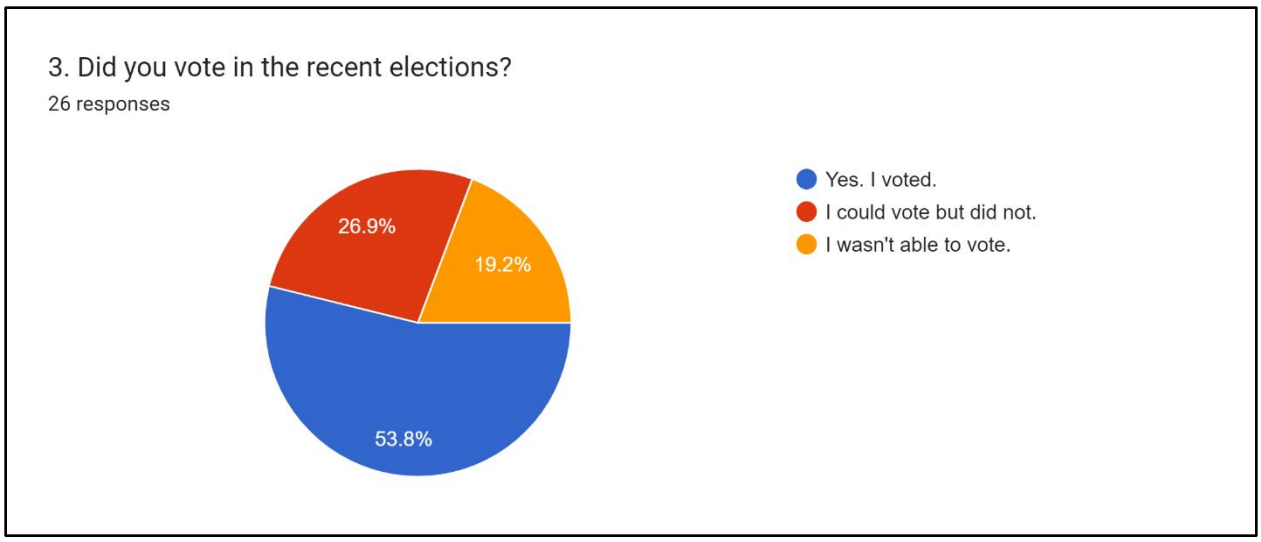


Figure 16: Q3

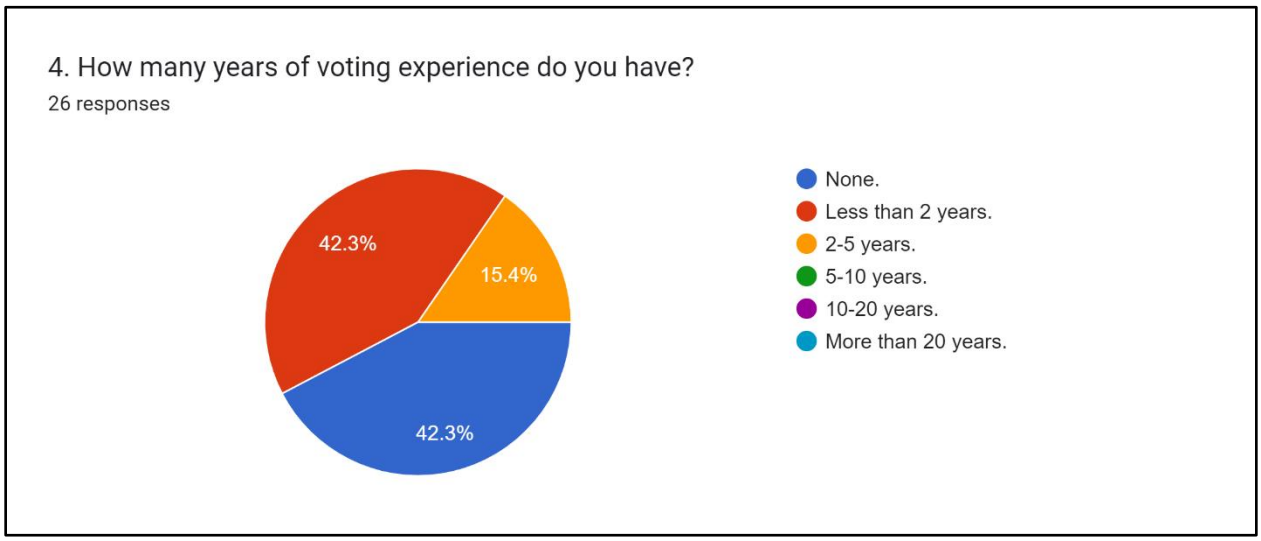


Figure 17: Q4

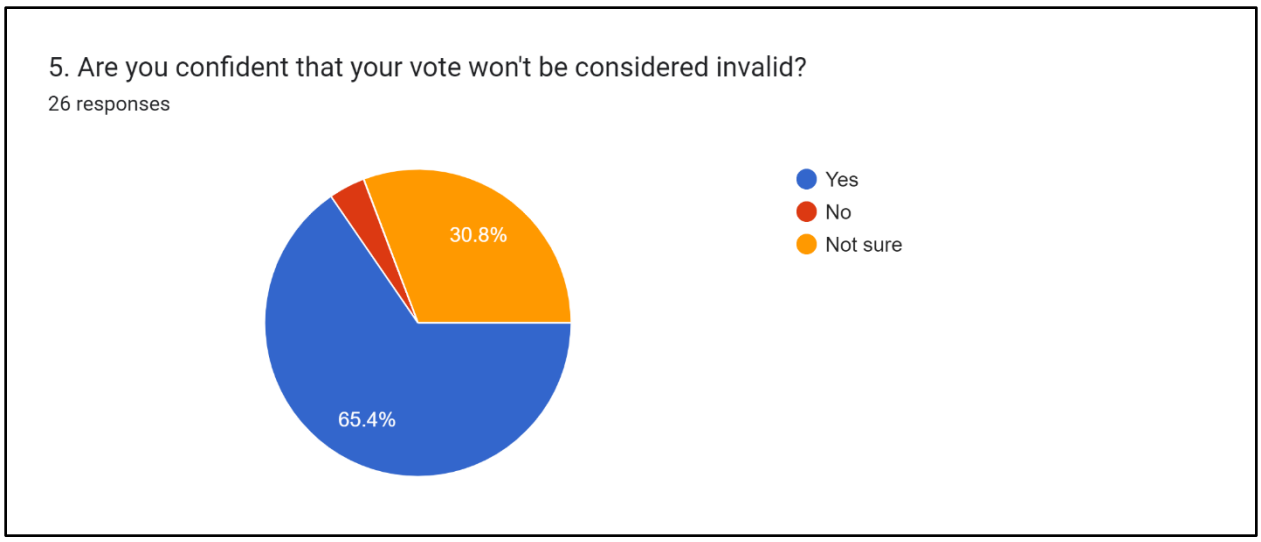


Figure 18: Q5

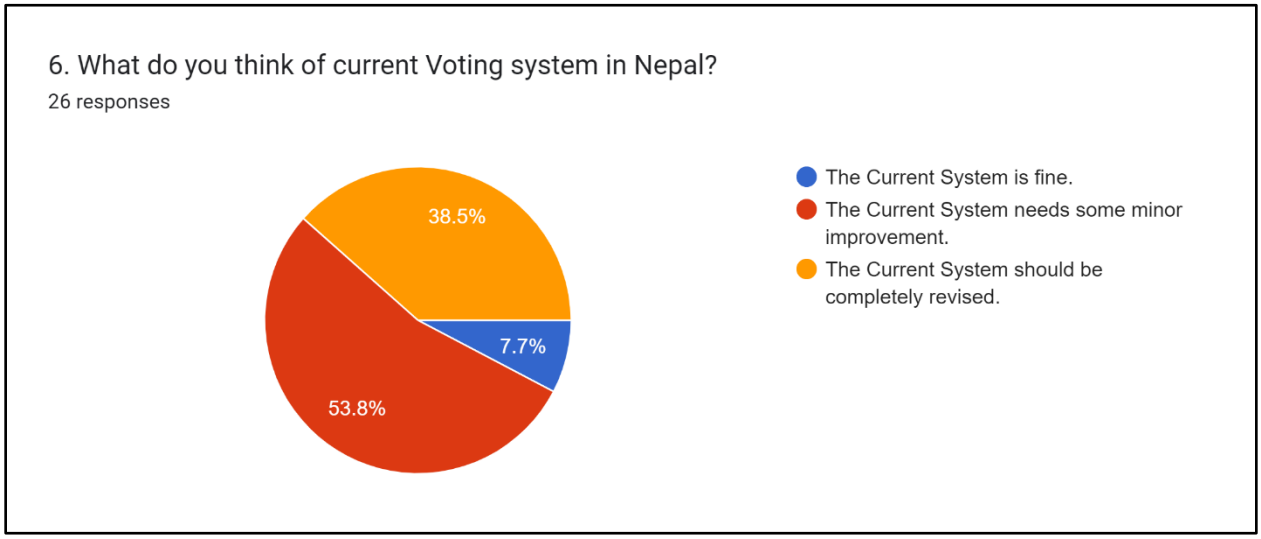


Figure 19: Q6

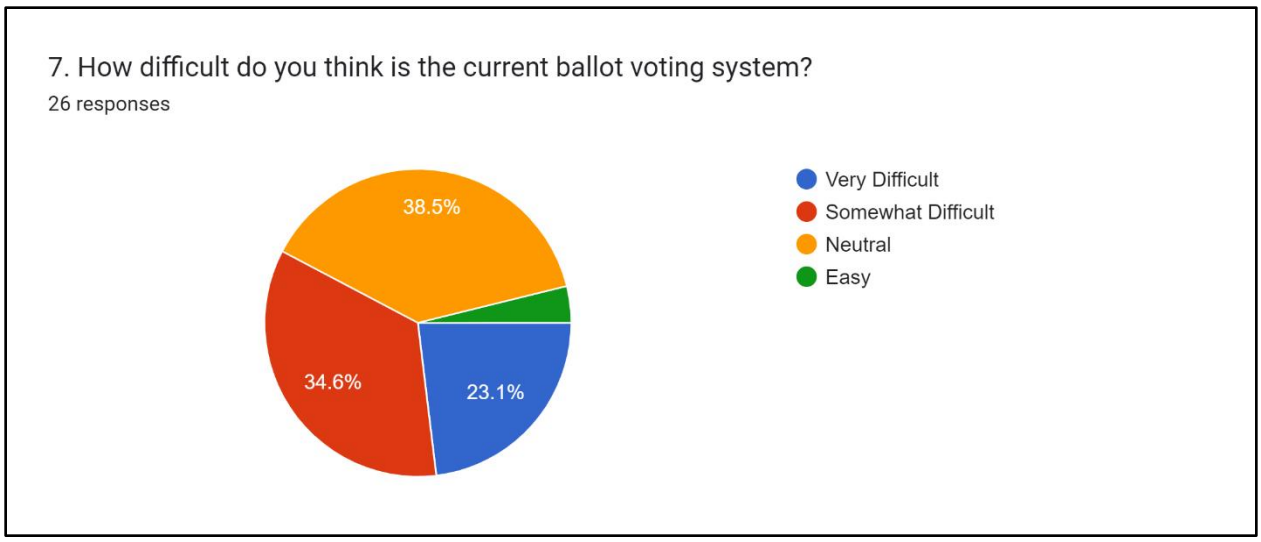


Figure 20: Q7

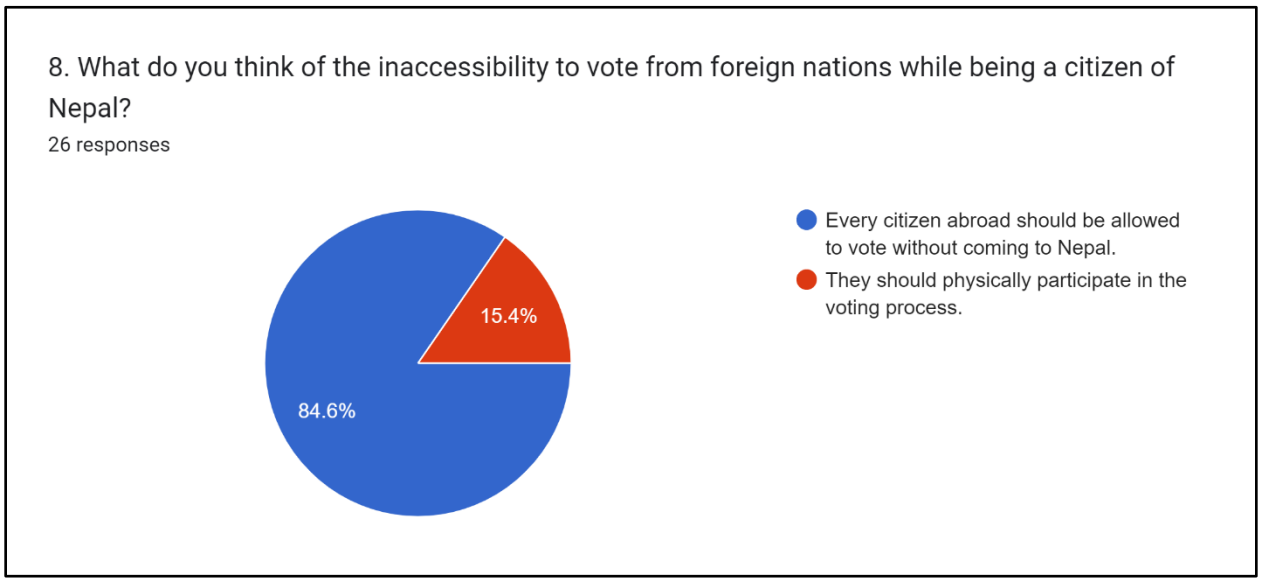


Figure 21: Q8

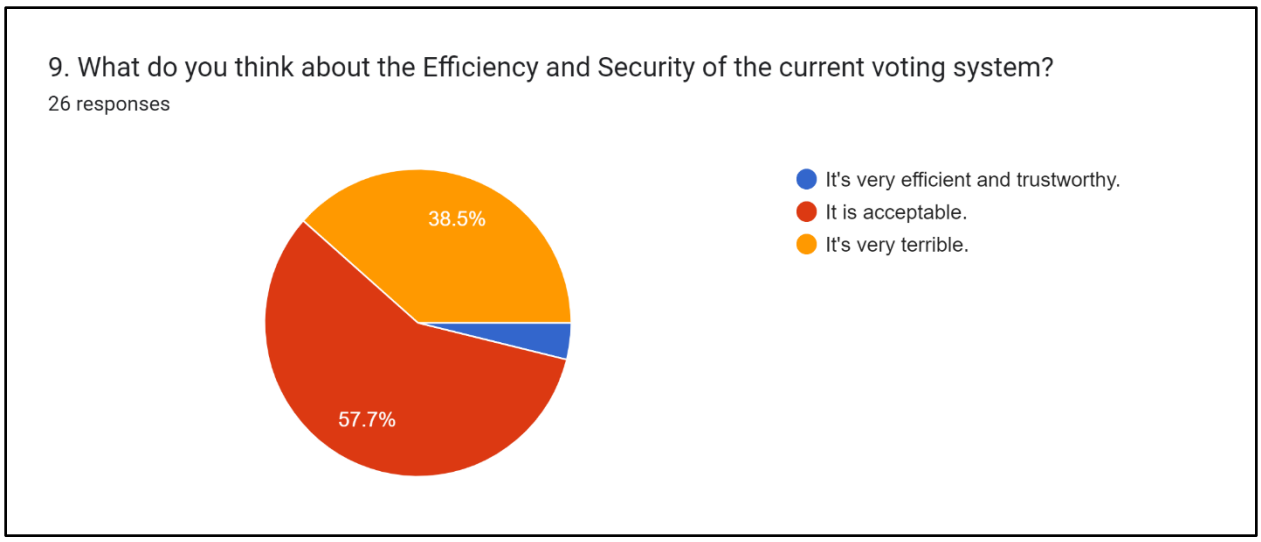


Figure 22: Q9

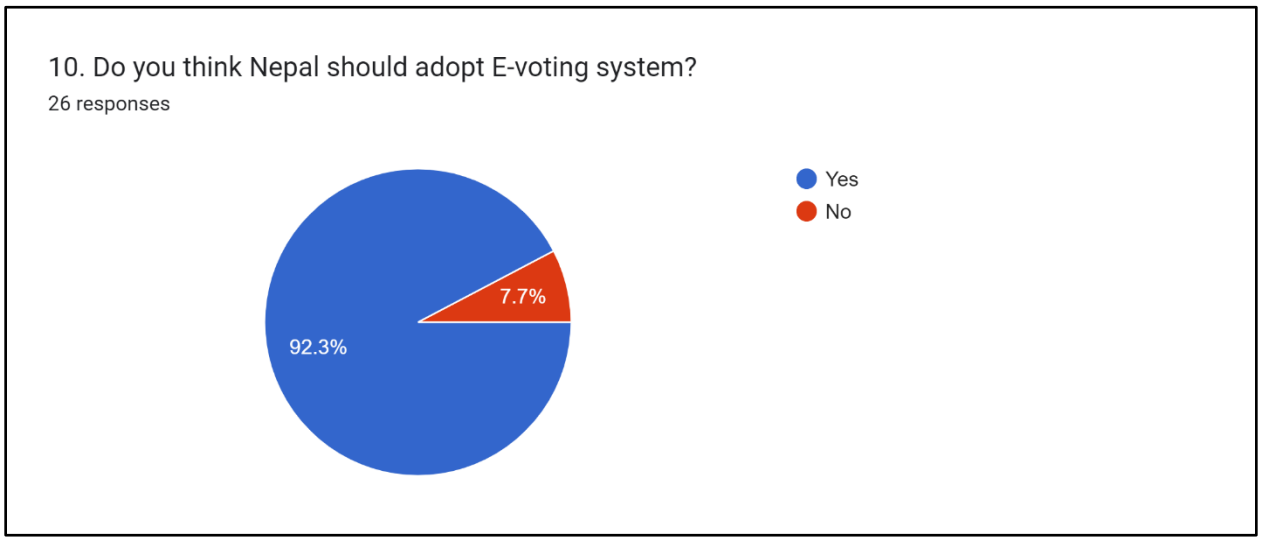


Figure 23: Q10

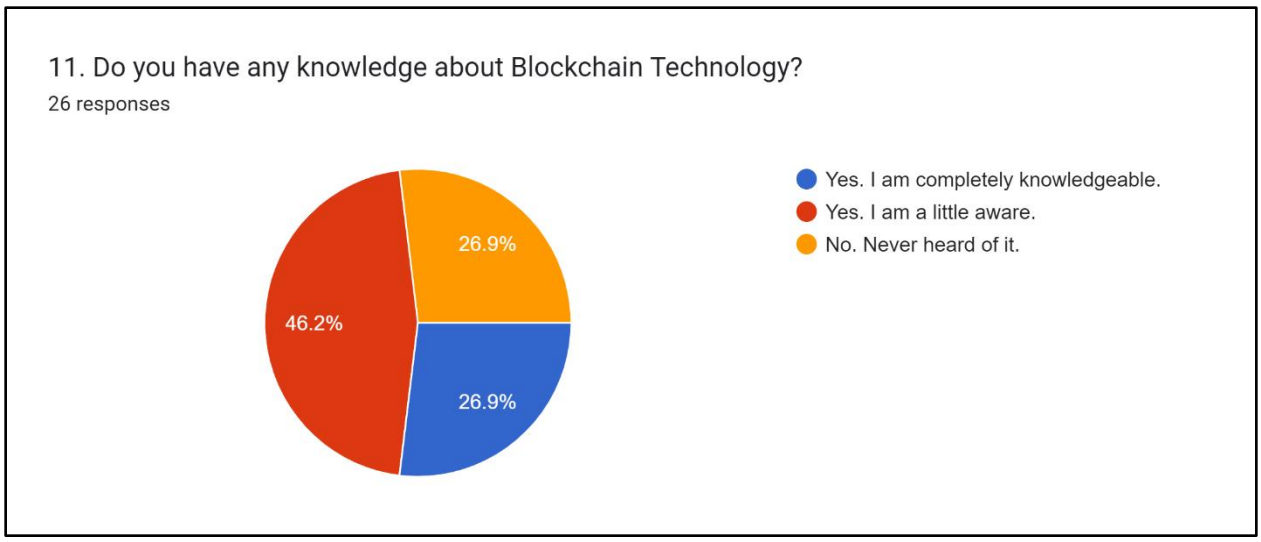


Figure 24: Q11

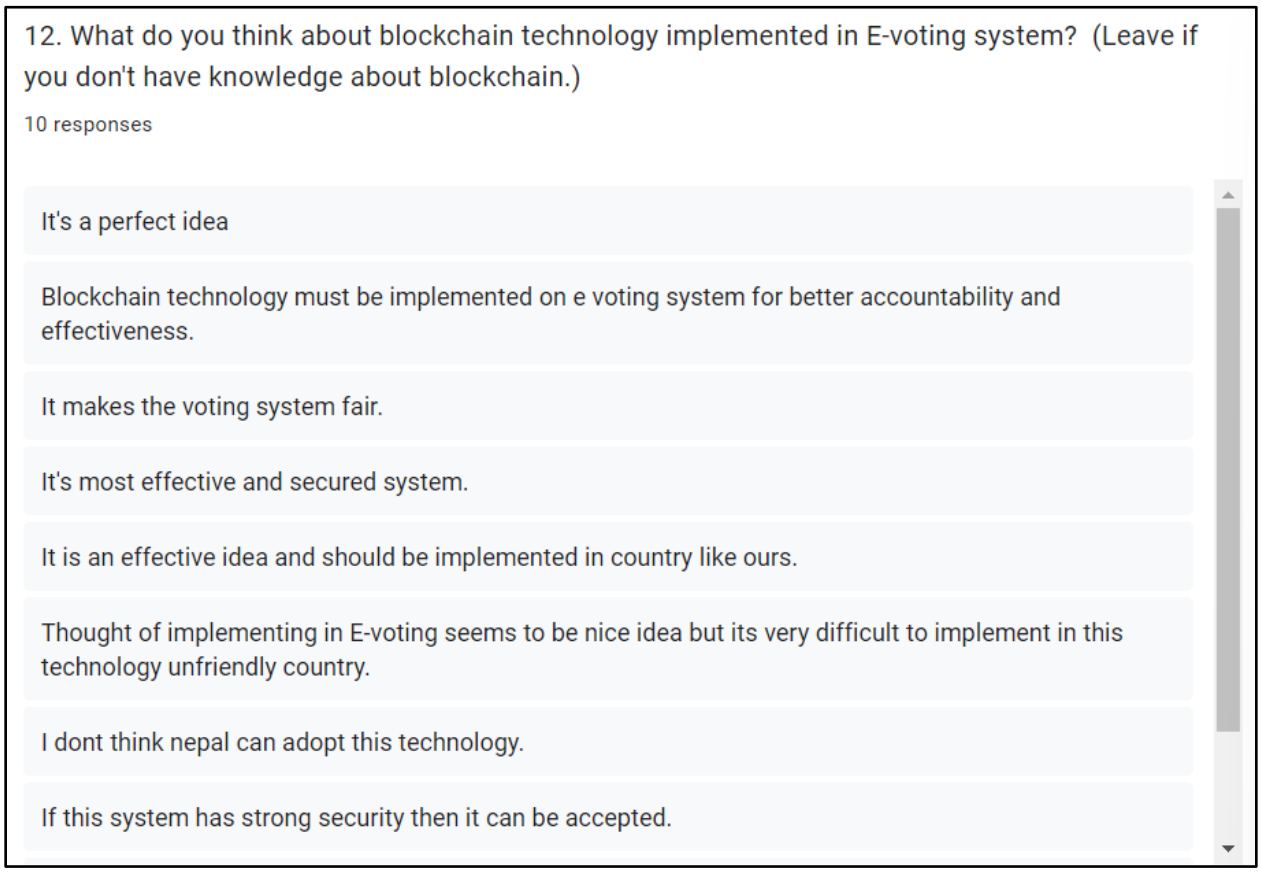


Figure 25: Q12

Appendix 2: Project Plan, Design and Requirements

Wireframes

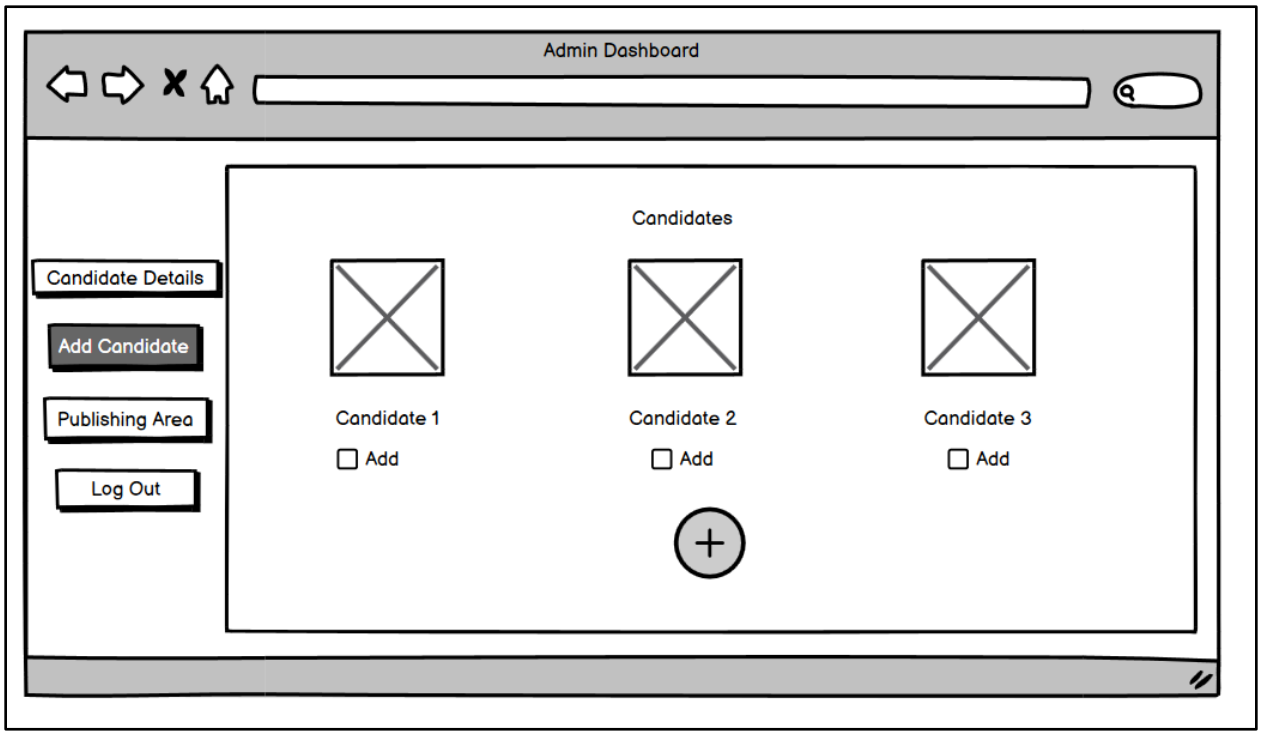


Figure 26: Wireframe: Add Candidate

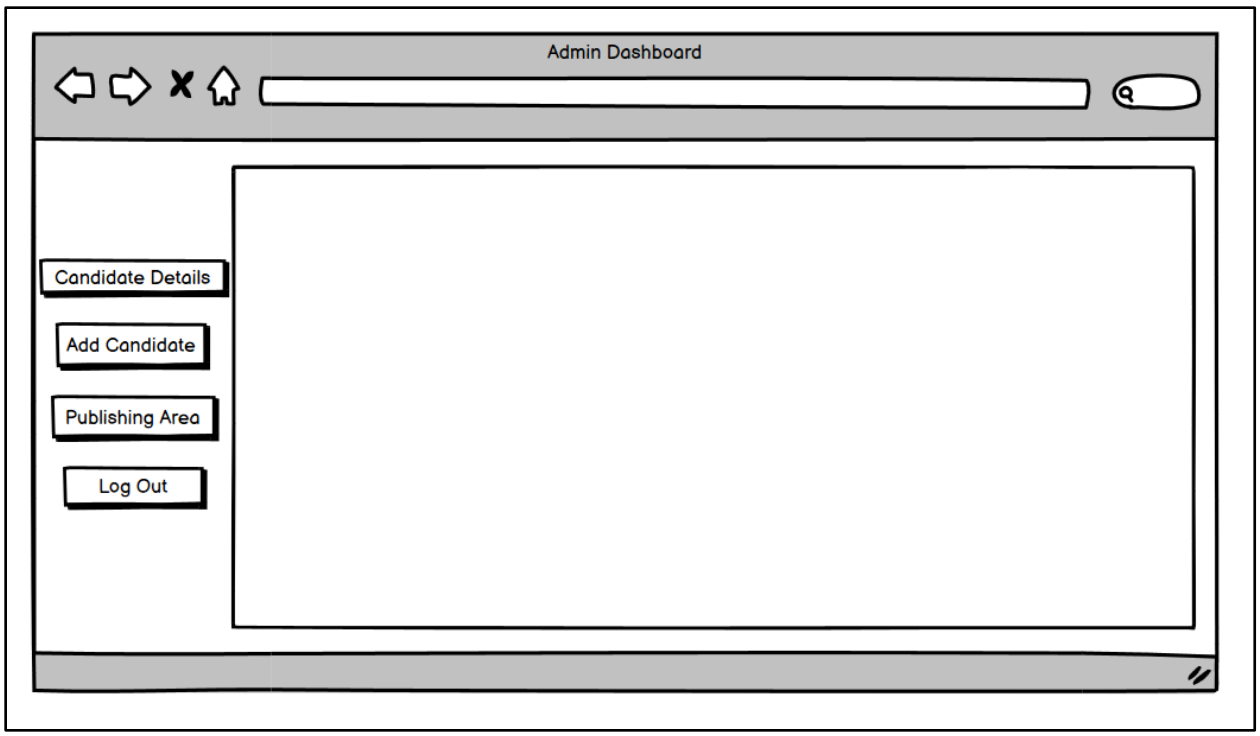


Figure 27: Wireframe: Dashboard

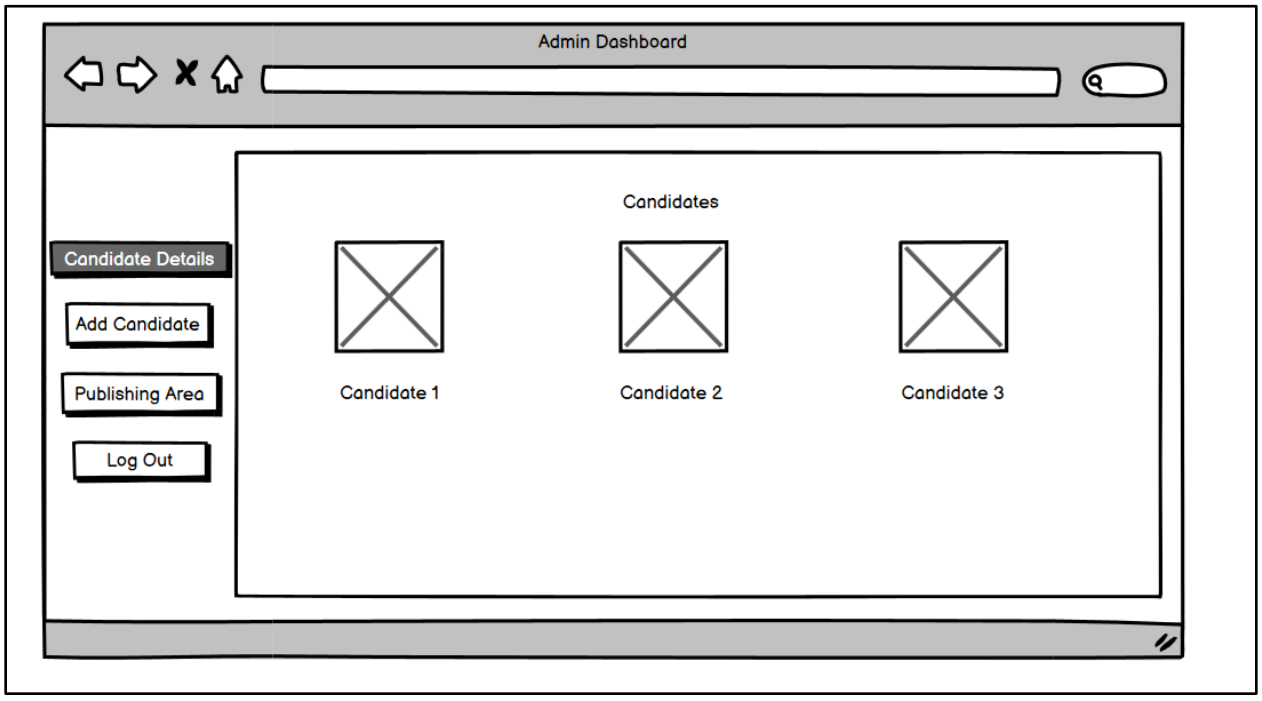


Figure 28: Candidate Details

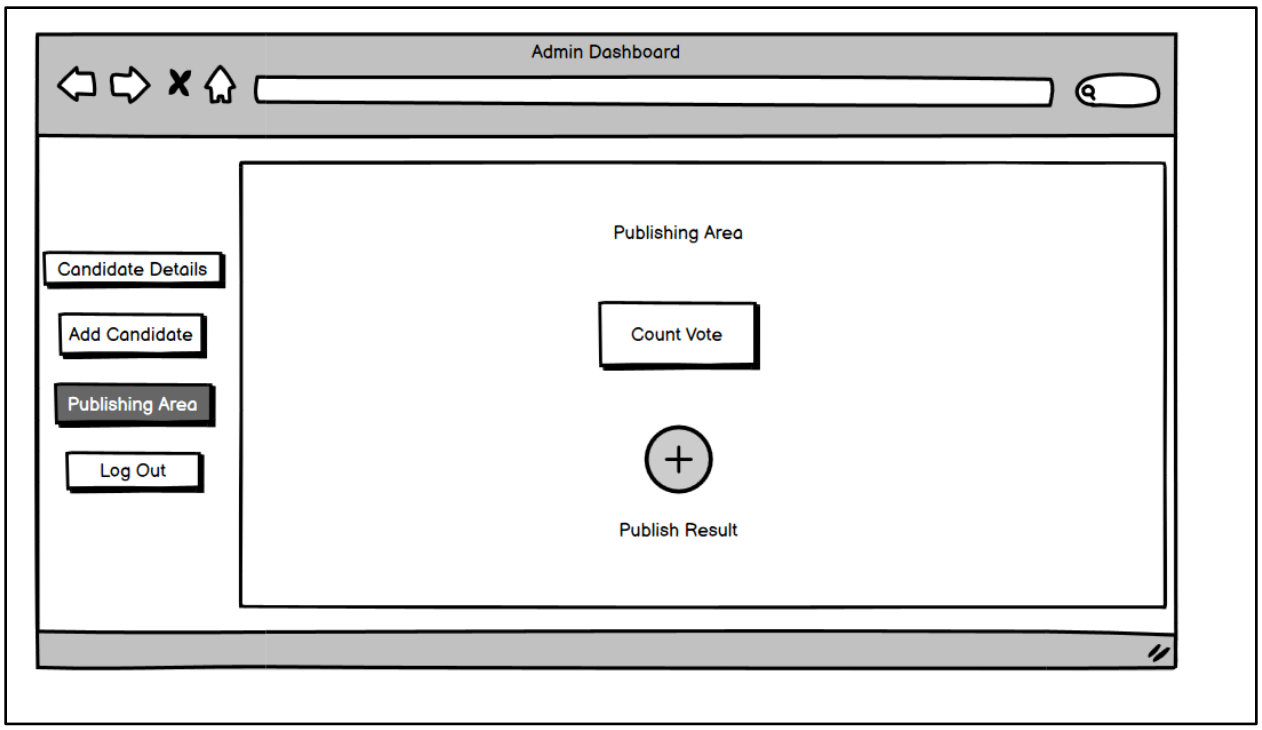


Figure 29: Count Vote

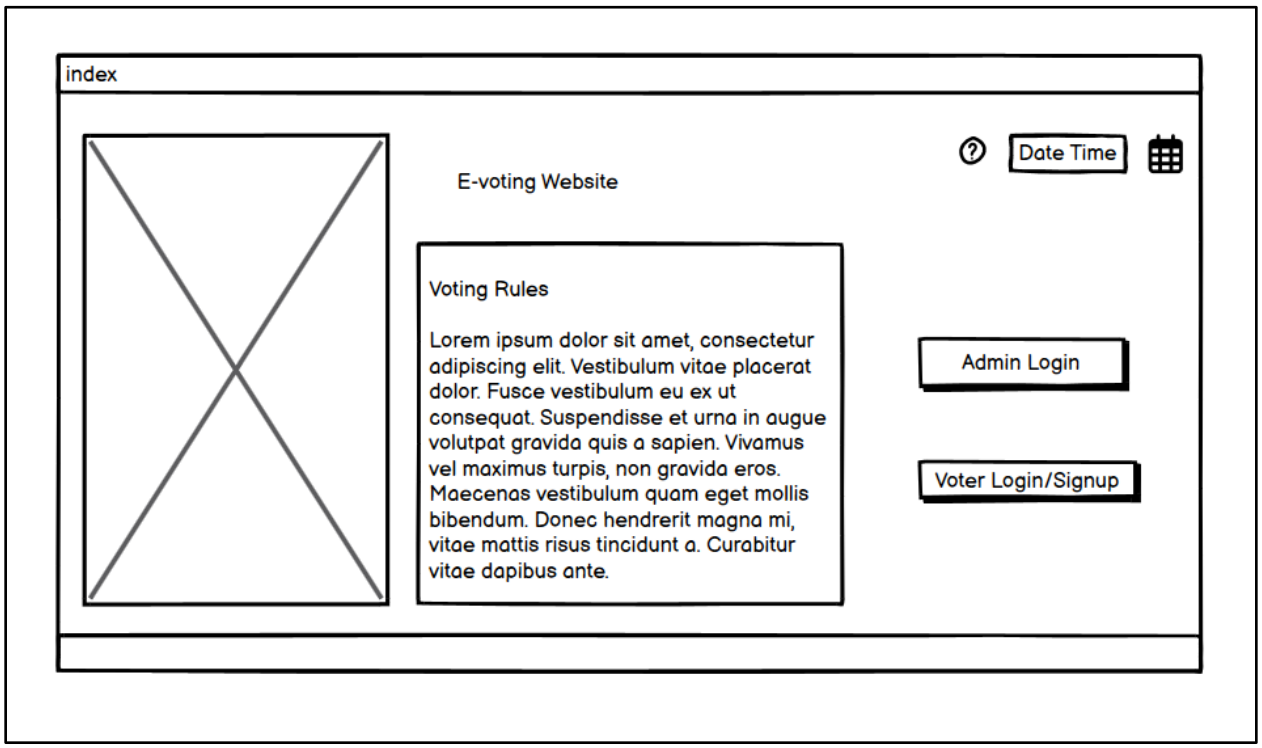


Figure 30:Wireframe: Index

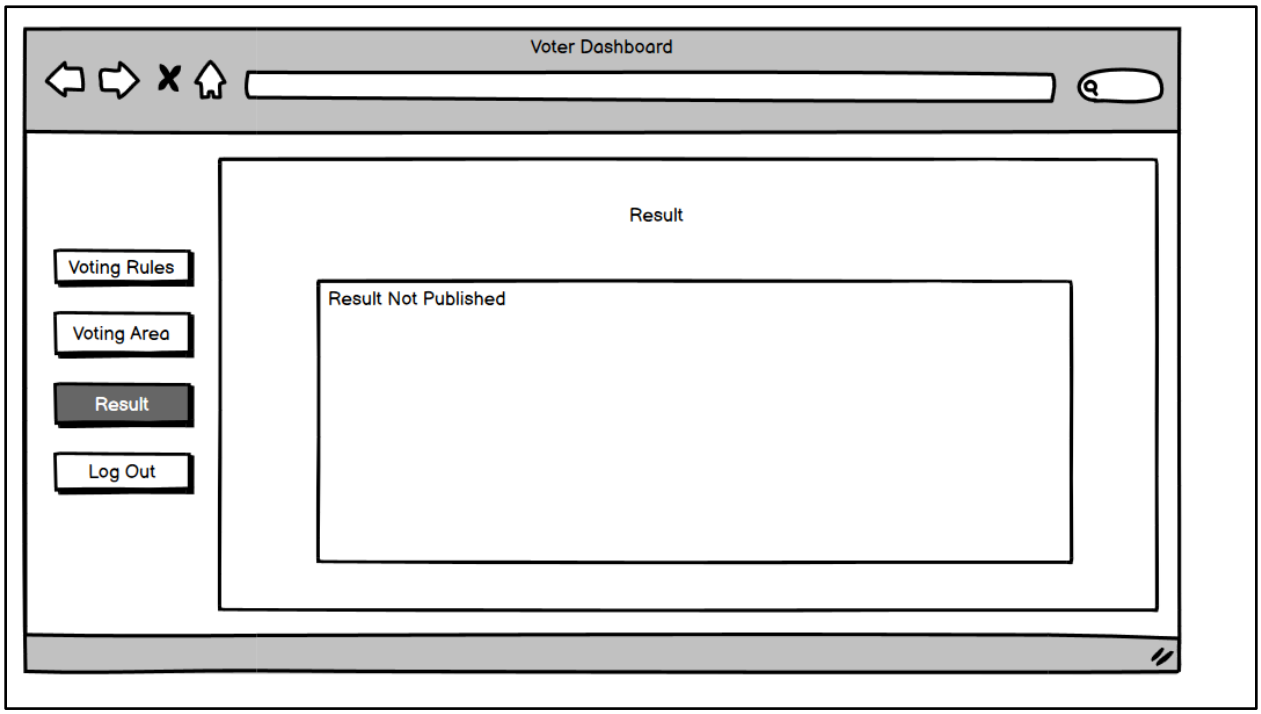


Figure 31:Wireframe: Result

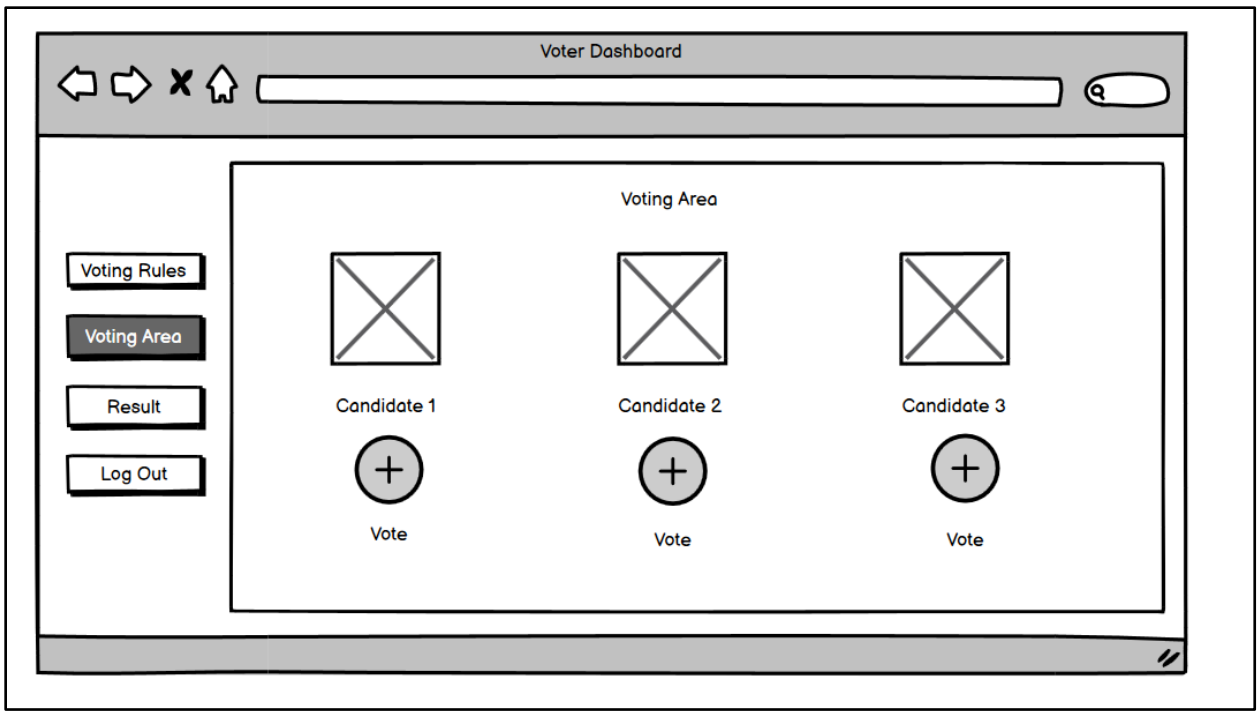


Figure 32: Wireframe: Voting Area

Webpages and Code

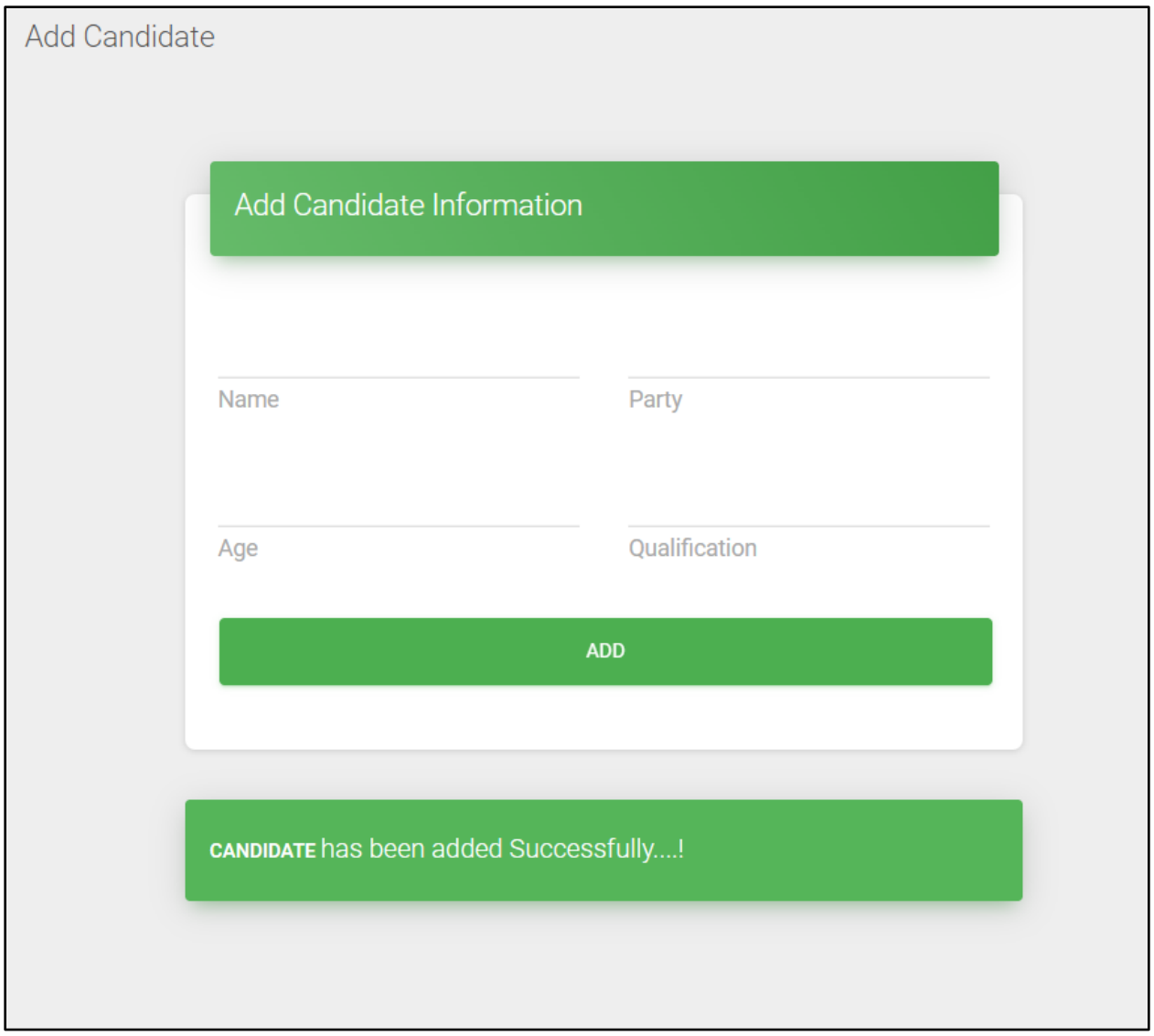


Figure 33: Add Candidate Info. Page

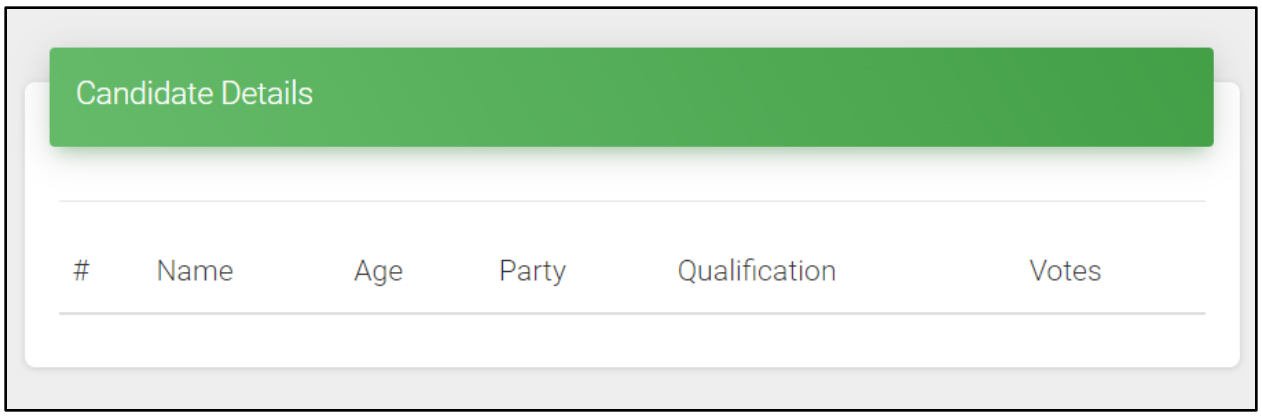


Figure 34: Candidate Details

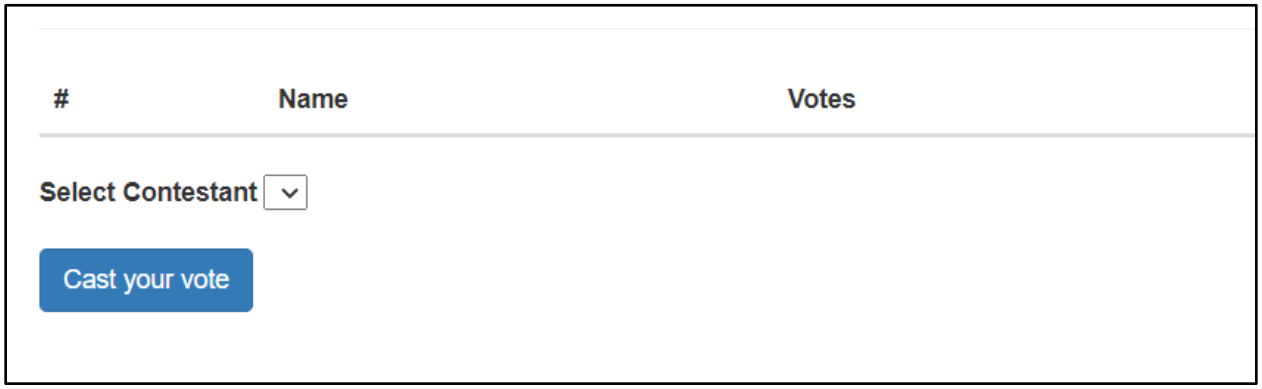


Figure 35: Voting Page

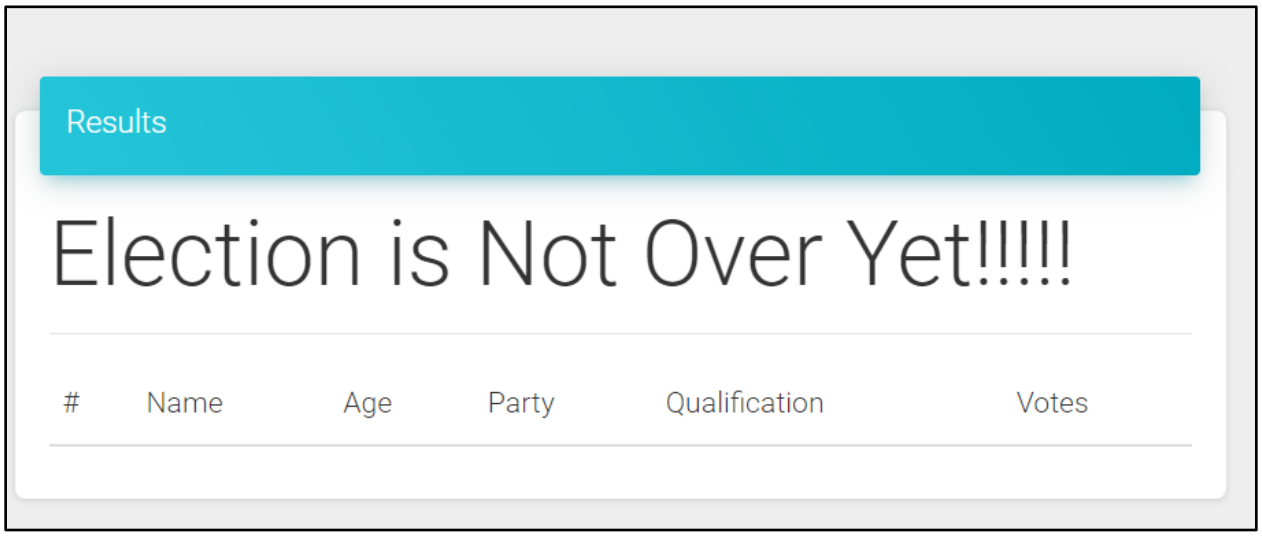


Figure 36: Results

```
<!DOCTYPE html>
<html>
<head>
    <title>Block-Vote</title>
    <link rel="stylesheet" type="text/css" href="css/index.css">
    <link href="https://fonts.googleapis.com/css?family=Montserrat" rel="stylesheet">
</head>
<body>
    <header class="site-header clearfix">
        <nav>
            <div class="logo">
                <h1>E-Vote</h1>

            </div>

        </nav>
        <section>
            <div class="leftside">
                <img src="img/homepage.jpg">
            </div>
            <div class="rightside">
                <h1>VOTING</h1>
                <p>Every Citizens Right</p>
                <a href="/register"><button>USER SIGN-UP/LOGIN</button></a>
                <a href="/adlogin"><button>ADMIN-LOGIN</button></a>

            </div>

        </section>

    </header>
</body>

</html>
```

Figure 37:Index Code

index.html # adminstyle.css ×

D: > College > 3rd Year > FYP > src > css > # adminstyle.css > ...

```
@import url('https://fonts.googleapis.com/css?family=Noto+Sans+TC&display=swap');
body{
    margin: 0;
    padding: 0;
    background: radial-gradient(#f3f1f3,#9fc7e2,#50afe6);
    height: 100vh;
    overflow: hidden;
    font-family: 'Noto Sans TC', sans-serif;
}
.center{
    width: 430px;
    top: 200px;
    left: 36%;
    position: relative;
}
.center .header{
    font-size: 28px;
    font-weight: bold;
    color: white;
    padding: 25px 0 30px 25px;
    background: #34d8eb;
    border-bottom: 1px solid  #f2f4f7;
    border-radius: 5px 5px 0 0;
}
.center form{
    position: absolute;
    background: white;
    width: 100%;
    padding: 50px 10px 0 30px;
    box-sizing: border-box;
    border: 1px solid #f5f8fa;
    border-radius: 0 0 5px 5px;
}
form input{
    height: 50px;
    width: 90%;
```

Figure 38: CSS

Smart Contract

```
pragma solidity ^0.6.0;

// We define a struct to represent a candidate in the election
struct Candidate {
    string name;
    uint votes;
}

// We define a mapping to store the candidates in the election
mapping(uint => Candidate) public candidates;

// We define a variable to store the total number of candidates in the election
uint public candidateCount;

// We define a variable to store the minimum age for voters
uint public minVotingAge;

// We define an event to signal when a vote has been cast
event VoteCast(uint indexed candidateId);

// We define a constructor to initialize the contract
constructor(uint _minVotingAge) public {
    minVotingAge = _minVotingAge;
}

// We define a function to add a candidate to the election
```

Figure 39: Smart Contract 1

```
// We define a function to add a candidate to the election
function addCandidate(string memory _name) public {
    candidates[candidateCount] = Candidate(_name, 0);
    candidateCount++;
}

// We define a function to cast a vote for a candidate
function vote(uint _candidateId) public {
    // We first check that the voter is of legal age
    require(msg.sender.age >= minVotingAge, "Voter is not of legal age");
    // We check that the candidate id is valid
    require(_candidateId < candidateCount, "Invalid candidate id");
    // We update the vote count for the candidate
    candidates[_candidateId].votes++;
    // We emit an event to signal that a vote has been cast
    emit VoteCast(_candidateId);
}

// We define a function to retrieve the total number of votes for a candidate
function getVoteCount(uint _candidateId) public view returns (uint) {
    return candidates[_candidateId].votes;
}

// We define a function to retrieve the winner of the election
function getWinner() public view returns (uint) {
```

Figure 40: Smart Contract 2

```
// We define a function to retrieve the winner of the election
function getWinner() public view returns (uint) {
    uint winningCandidate = 0;
    for (uint i = 0; i < candidateCount; i++) {
        if (candidates[i].votes > candidates[winningCandidate].votes) {
            winningCandidate = i;
        }
    }
    return winningCandidate;
}
```

Figure 41: Smart Contract 3

Appendix 3: Originality Report

12/27/22, 11:23 PM [Prabesh Hada] 20049173 Mihir Rauniyar - Interim

Originality report

COURSE NAME

CS6P05 - FYP - Subekshya Pradhan and Prabesh Hada

STUDENT NAME

MIHIR RAUNIYAR

FILE NAME

[Prabesh Hada] 20049173 Mihir Rauniyar - Interim

REPORT CREATED

27 Dec 2022

Summary

Flagged passages	1	1%
Cited/quoted passages	0	0%

Web matches e-unwto.org

1 1%

1 passage

Student passage FLAGGED

…support throughout the project. Without their help and understanding, **this report would not have been possible**.

Top web match

proposing this collaboration and providing his assistance throughout the process. In addition, **this report would not have been possible** without the valuable support of the cities, the.

'Overtourism'? – Understanding and Managing Urban Tourism ...
https://www.eunwto.org/doi/pdf/10.18111/9789284420070