VOTE VAULT- BLOCKCHAIN-BASED SYSTEM USING SHA-256 ALGORITHM AND SMART CONTRACT

A PROJECT REPORT

Submitted by

SAI KRISHNA R

SAKETH RAMAN R

YUTEESH S

in partial fulfillment for the award of the degree of

BACHELOR OF ENGINEERING

in

COMPUTER SCIENCE AND ENGINEERING

EASWARI ENGINEERING COLLEGE, CHENNAI

(Autonomous Institution)

affiliated to

ANNA UNIVERSITY :: CHENNAI - 600025

JUNE 2022

EASWARI ENGINEERING COLLEGE, CHENNAI

(AUTONOMOUS INSTITUTION) AFFILIATED TO

ANNA UNIVERSITY, CHENNAI 600025

BONAFIDE CERTIFICATE

Certified that this project report **"VOTE VAULT-BLOCKCHAIN-BASED SYSTEM USING SHA-256 ALGORITHM AND SMART CONTRACT"** is the bonafide work of "**SAI KRISHNA R (310618104079), SAKETH RAMAN R (310618104080), YUTEESH S (310618104120)"** who carried out the project work under my supervision.

SIGNATURE	**SIGNATURE**
Dr.G.S.ANANDHA MALA **HEAD OF THE DEPARTMENT**	**Mrs.SHYAMALA GOWRI.B** **SUPERVISOR** Assistant Professor
Department of Computer Science and Engineering Easwari Engineering College, Ramapuram, Chennai, 600 089.	Department of Computer Science and Engineering Easwari Engineering College, Ramapuram, Chennai, 600 089.

CERTIFICATE OF EVALUATION

College Name : Easwari Engineering College

Branch & Semester : Computer Science and Engineering & VIII

S. No	Name of the Students	Title of the Project	Name of the Supervisor with Designation
1.	SAI KRISHNA R (310618104079)	VOTE VAULT-BLOCKCHAIN-BASED SYSTEM USING SHA-256 ALGORITHM AND SMART CONTRACT	Mrs. SHYAMALA GOWRI B, ASSISTANT PROFESSOR
2.	SAKETH RAMAN R (310618104080)		
3.	YUTEESH S (310618104120)		

The report of the project work submitted by the above students in partial fulfillment for the award of Bachelor of Engineering Degree in Computer Science and Engineering of Anna University were evaluated and confirmed to be a report of the work done by the above students.

The viva voice examination of the project work was held on __.__.2022

INTERNAL EXAMINER **EXTERNAL EXAMINER**

ACKNOWLEDGEMENT

We hereby place our deep sense of gratitude to our beloved Founder Chairman of the institution, **Dr.T.R.Pachamuthu,B.Sc.,M.I.E.**, for providing us with the requisite infrastructure throughout the course. We would also like to express our gratitude towards our Chairman **Dr. R. Shivakumar, M.D., Ph.D.** for giving us the necessary facilities.

We convey our sincere thanks to **Dr.R.S.Kumar,M.Tech.,Ph.D.** Principal Easwari Engineering College, for his encouragement and support. We extend our hearty thanks to **Dr.V.Elango,M.E.,Ph.D.**, Vice-Principal (academics), and **Dr.S.Nagarajan,M.E,Ph.D.,** Vice-Principal (Admin), Easwari Engineering College, for their constant encouragement.

We take the privilege to thank **Dr.G.S.Anandha Mala,M.S.,M.E., Ph.D.**, Head of the Department, Computer Science and Engineering, Easwari Engineering College for her suggestions, support, and encouragement towards the completion of the project with perfection.

We would like to express our gratitude to our Project Coordinator, **Dr.Ahamed Ali,M.E.,(Ph.D).,** Associate Professor, Department of Computer Science and Engineering, Easwari Engineering College for her constant support and encouragement.

We would also like to express our gratitude to our guide, **Mrs.B.Shyamala Gowri,M.E.,(Ph.D).,** Assistant Professor, Department of Computer Science and Engineering, Easwari Engineering College for her constant support and encouragement.

Finally, we wholeheartedly thank all the faculty members of the Department of Computer Science and Engineering for their warm cooperation and encouragement.

ABSTRACT

One of the most important indications of how residents participate in their country's administration is voter turnout. Since the 1990s, the number of countries that hold national elections has increased substantially. However, the global average voter turnout has decreased significantly over the same period. We propose a Blockchain-based Voting System to solve the above-mentioned issue. We argue that our system is more secure, and reliable, and can protect voter privacy which will help boost the number of voters and their trust in the electoral system as well as reduce considerably the cost of national elections. Since it is an online voting system, it allows citizens to vote from the comfort of their homes. The system makes use of the SHA-256 Algorithm for voter authentication and Smart Contract for the voting process. The vote information that the user enters is sent to the Blockchain which holds the data. We implement the proposed protocol using the smart contract in such a way that the Ethereum Blockchain's consensus mechanism enforces the execution of the voting protocol. Blockchains are used since they can store and execute programs that are written as smart contracts. The implemented protocol is also decentralized compared to centralized mechanisms of the existing systems. Compared to other state-of-the-art Blockchain-based voting systems it respects voters' privacy and has user-friendly terminals, which will boost the confidence of people in the voting system and therefore increase the number of participants in the election.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF ABBREVIATIONS

DVS	Digital Voting System
SHA	Secure Hash Algorithm
ETH	Ethereum
NFT	Non-Fungible Token
ECC	Elliptic Curve Cryptography
HTML	Hyper Text Markup Language
CSS	Cascading Style Sheets

CHAPTER 1

INTRODUCTION

1.1 GENERAL

Voting systems have been used for over decades in various democracies helping in letting voters choose their leaders. But for the most part, the systems are unchanged for a long period of time. The majority of the systems are devoid of the security of the voter and the votes which can be tampered with. There has been a decline in the number of voters in recent elections in various parts of the world. For example, the Tamil Nadu State Legislative Assembly Elections held in 2021 recorded only a 72.8% voter turnout, which is 2% lesser than the preceding 2016 elections. As stated in the study conducted by IDEA on the recent Voter Turnout Trends, Institutional factors such as Electoral systems, Registration, and Voting Arrangements play an important role. We have proposed a Blockchain-based voting system that succeeds in securing the privacy of the voter and their votes. Blockchain was first developed as a cryptocurrency financial services company where an immutable ledger facilitates the process of recording transactions and tracking assets in a business network. But as time passed, Blockchain has been used in multiple domains and now we are proposing it to be used in the voting systems.

1.2 PROBLEM DESCRIPTION

To provide a digital voting system to make public electoral cheaper, faster, and more secure without the risk of tampering. The system also provides the Vote count by using Ganache from Blockchain.

1.3 OBJECTIVES

- To implement a voting System using the Ethereum Blockchain mechanism which enforces the execution of the voting protocol.
- To allow people to vote from the comfort of their homes and increase the voter turnout during elections.
- To enhance the security of the voting process by using a decentralized Blockchain mechanism.

1.4 SCOPE OF THE PROJECT

The scope of the project is to create a Digital Voting System (DVS) that aids to make Public electoral cheaper, easier, faster and more secure from hacking. It uses Ethereum Blockchain in such a way that each user is allowed to vote once in the system by logging in and every single vote is saved in a separate block with a unique address. Ganache is used to provide the number of votes each party acquired in the Public Electoral.

1.5 ORGANIZATION OF THE THESIS

The report consists of 6 chapters, the contents of which are described below: Chapter 1 is the introduction that explains the basic information of the system. Chapter 2 is the literature survey that elaborates on the research works on the existing systems. Chapter 3 describes the system design. Chapter 4 gives details regarding the system implementation. Chapter 5 describes different test scenarios for the modules described by the proposed system and the performance analysis of the proposed system. Chapter 6 provides the conclusion, which summarizes the efforts undertaken in the proposed system and states findings and shortcomings in the proposed system.

CHAPTER 2

LITERATURE SURVEY

2.1 GENERAL

A literature survey is the documentation of a comprehensive review of the publishers and unpublished work from the secondary sources data in the areas of specific interest to the researchers. It provides an overview of current knowledge, allowing you to identify relevant theories, methods, and gaps in the existing research. Conducting a literature review involves collecting, evaluating, and analyzing publications (such as books and journal articles) that relate to the research question.

Many research works are carried out to create a Digital Voting System with high security standards and invulnerable to security threats. Some of the works include the usage of Centralized Systems which provides the Voter to use the System with intermediates (Admin). The recently carried works have been explained in detail below.

2.2 EXISTING SYSTEM

In existing systems, Votes are stored on a Database which can be retrieved and the Total count for each party is shown. The existing system uses basic cryptography techniques which allows the User to directly vote into the system. It uses a centralized mechanism in which the admin or intermediate can manipulate the Vote of the user which is a major security threat in the System. This system which is fully centralized reduces flexibility, distributability and embraces dictatorship. The main issue is the usage of the centralized mechanism that makes the system vulnerable which can be overcome by using the decentralized mechanism which works only between the User and the System without any intermediate who handles the system.

2.2.1 Literature Survey on ‘Multi-Candidate Voting Model Based on Blockchain’

The authors of Dongliang Xu et al., (2021) propose a "k-out-of-m" multi-candidate voting model based on Blockchain technology that ensures: 1) the preservation of anonymity during the voting process through ECC encryption and a signature mechanism; and 2) the prevention of forgery of voting information by combining Blockchain technology, automatic statistics, and the display of voting results through a smart contract. In addition, this research introduces an anonymity-preserving voting (APV) algorithm to address the shortcomings of the Blockchain voting technique. They conduct experiments on the suggested paradigm using the Hyperledger Fabric technology. Hyperledger Caliper is used to evaluate the model's performance. Experiments are planned and thoroughly tested in terms of the model's security criteria as well as the proposed APV algorithm's efficacy and efficiency. They confirm that the model is especially well-suited for small-scale voting circumstances and that it solves the shortcomings of traditional electronic voting, such as lack of anonymity, excessive centralization, and ease of forging, through these trials.

2.2.2 Literature Survey on ‘The Design and Implementation of a Blockchain-Enabled E-Voting Application Within IoT- Oriented Smart Cities’

The authors of Geetanjali Rathee et al., (2021) introduced the concept of E-voting attacks, which occur during the polling mechanism in smart cities. The Privacy and security vulnerabilities are addressed by computing each entity's trust and then storing it in a Blockchain to compare their continuous conduct.

Furthermore, when compared to the baseline scheme, the proposed phenomena exhibit a substantial improvement because the provided technique ensured security by employing Blockchain and trust computation instead of checking certificates and using cryptographic algorithms. By comparing numerous security metrics, our suggested method is comprehensively evaluated against the baseline mechanism. Furthermore, by tracking the activity of each election process, the proposed mechanism outperformed the baseline mechanism. However, the system's drawback is that it is merely a simulation (theory) that can only be employed in smart cities that use IoT.

2.2.3 Literature Survey on ‘A Smart Contract System for Decentralized Borda Count Voting, IEEE Transactions on Engineering Management’

The authors of Somnath Panja et al., (2020) proposed a two-round self-tallying Borda count e-voting mechanism in their paper. The tally is not computed by any trustworthy party in this approach. Instead, the approach assures that anyone with access to the bulletin board can calculate the total using public data. The technique protects voter privacy, with voters only being able to discover the election tally and their individual inputs after the procedure is completed successfully. They presented security proofs to demonstrate the protocol's security. Furthermore, the scheme is publicly verifiable. Every voter generates NIZK proofs to demonstrate that they have followed the protocol specification to the letter while keeping their secret input hidden. For implementation, Blockchain is employed. The findings of both the theoretical and experimental analyses demonstrate that this technique can be employed in practice. However, this approach has the drawback of not being designed for a traditional voting process. Instead, it's designed for the Borda count method of voting.

2.2.4 Literature Survey on ‘Security Properties of Light Clientson the Etherium Blockchain’

The findings of Santeri Paavolainen et al., (2020) refute many conventional assumptions about light client security. While light clients may be regarded as secure against adversarial invalid block injections in "typical" settings, these are optimistic scenarios that include a brief attack window, an honest majority, and no network partitioning. However, if one gets away from these optimistic assumptions, reality changes dramatically. Under partial network isolation, an adversary has a statistically substantial chance of success even with modest hashing power. They also find that if an opponent can undermine an existing mining pool on the Ethereum network, their chances of succeeding skyrocket. When adversaries who can change the target node's operating environment are considered, light clients' security is further weakened. The attacker may have access to the device, and it could modify its network accessibility or power availability even if it did not meddle with the device itself. Jammers for GSM and Wi-Fi are reasonably easy to buy and deploy, allowing a hostile actor to manipulate the availability of the network to which a client is connected. Finally, their assumptions constraints the analysis. For example, Ethereum allows for gradual adjustments to the cryptographic puzzle's difficulty in reaction to changes in network hashing power. This indicates that a partitioned network will adjust over time to reach a block interval similar to the complete network. They don't take this into consideration and believe it has minimal impact; this is a significant departure from the Ethereum protocol. Similarly, they believe the adversary's strategy is likely to be ineffective.

2.2.5 Literature Survey on 'A Secure Verifiable Ranked Choice Online System Based on Homomorphic Encryption'

The authors of Xuechao Yang et al., (2018) present a secure voter-verifiable e-voting system that allows voters to cast ballots by allocating unlimited numbers of points to various candidates. This means that voters can give each candidate the same number of points, or they can give each contender a different number of points. A distributed ElGamal cryptosystem is used in the system. Before being submitted, each cast ballot is encrypted and stays encrypted at all times. The exponential ElGamal cryptosystem's additive homomorphic characteristic allows for efficient ciphertext processing throughout these procedures. Furthermore, anyone can check the legitimacy of voters and their submissions without revealing the contents of the ballots. The security and performance research not only verifies the online voting system's practicality for real-world elections but also reveals that it outperforms other systems previously studied. However, the system's shortcoming is that it must assume that at least one authority is trustworthy; otherwise, the system is insecure.

2.3 ISSUES IN THE EXISTING SYSTEM

Existing voting system uses cryptography techniques, which, if not implemented properly, makes the system highly vulnerable to attacks. Current systems are fully centralized which reduces flexibility, and distributability, and embraces dictatorship. Tallying authorities collide among themselves, thus voters' privacy will be lost. These systems depend on authorities to achieve voters' privacy. Many Blockchain systems are concepts and do not have real-world applications since Blockchain is new.

2.4 PROPOSED SYSTEM

In this proposed system, the voter logs into the system via their UserID and password which is encrypted using the SHA-256 algorithm and compared with the encrypted hash in the database for authentication. After they complete the authentication process, they enter the voting system where they are provided with multiple parties they wish to vote for. When they are done selecting the party and casting their vote, a Blockchain is created to store the vote data. Smart Contract is used during the Blockchain creation process and the created Blockchains are immutable. Thus, the votes cannot be changed and since it is an online voting system, people can vote from the comfort of their homes which will increase the number of voters in the elections.

2.5 SUMMARY

The literature survey has covered information regarding the existing systems that help the public in Digital Voting. The issues and challenges have been identified in related work. This chapter has highlighted the proposed model.

CHAPTER 3

SYSTEM DESIGN

3.1 GENERAL

The main purpose of the design phase is to plan solutions to the problems specified by the requirement document. The design phase takes as input the requirement in the Requirements Analysis stage. This phase is the step in moving from the problem domain to the solution domain. The design of the system is perhaps the most critical factor affecting the quality of the software, and has a major impact in the later phases, particularly in testing and maintenance. The output of this phase is the functional design document. The design documents created for the proposed system consist of the system architecture and the data flow diagram. This is also known to be the top-level design that identifies the modules in the system, and how the data transfer takes place between the modules.

This chapter deals with design documents created for the proposed system which consists of functional architecture and activity diagrams etc. During detailed design, the internal logic of all the modules specified in the system design is decided. The system architecture deals with the components to be used in the proposed system which explains its interaction with one another. The data flow diagram describes the graphical representation of how the data flows between the different modules in the system. The data flow diagram gives a very clear picture of the flow of data among the working modules. These design documents are used as a continuous reference point for further system development and coding.

3.2 SYSTEM ARCHITECTURE

The following diagram in Figure 3.2 is the system architecture of the vote vault. The input is acquired from the user who signs up into the system. The voting system's components include a PC on which the user votes in the system using data stored in a block, and Metamask is used to produce the session. Admins can access the system without having to use the Admin Credentials to create a private session.

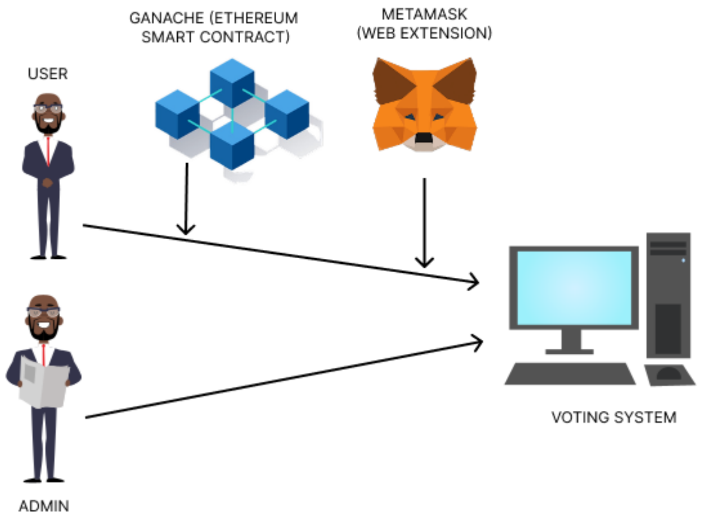


Figure 3.2 System Architecture

3.3 FUNCTIONAL ARCHITECTURE

The following diagram in figure 3.3 is the functional architecture of the system. It tells us about the complete flow of the system with the utmost details. It shows the working of the four modules of the system, first of which is, after generating a private key for a particular block for the user, A private session is created using Metamask with the provided credentials through which the user enters into the system. After generating a private session, the user can sign up in the voting system by entering values in the required fields and then can login into the system using the credentials which were entered while signing up. After logging into the system for the first time, the user can cast his vote from the parties available in the dropdown by clicking. After casting the vote, the voter can neither change his vote nor remove it.

The result of the election can be seen only by the admin after logging into the system but the admin has no privileges to manipulate any of the votes. The output of the system is the election result and the winner is the one who acquires the maximum number of votes in the election.

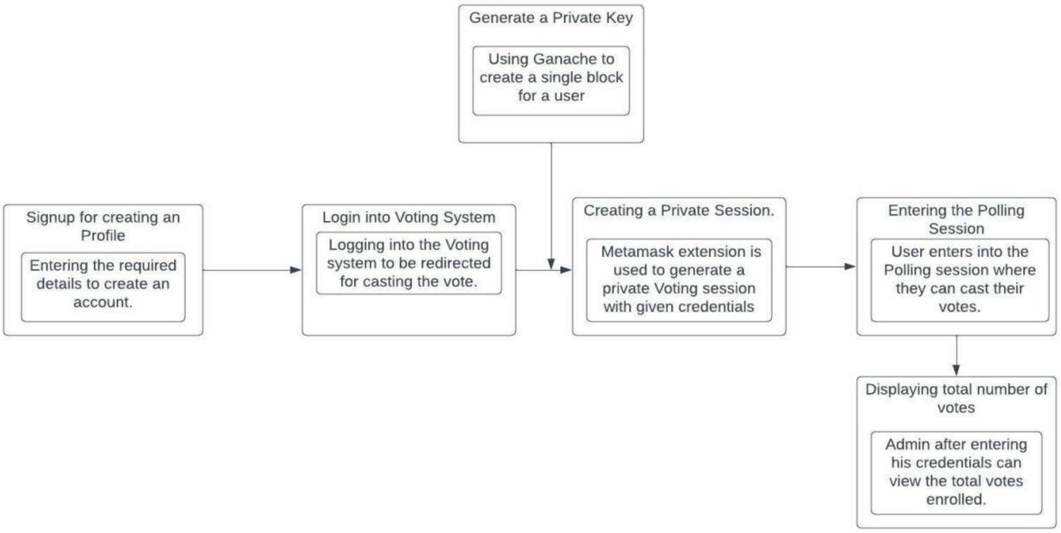

Generate a Private Key
Using Ganache to create a single block for a user
Signup for creating an Profile
Entering the required details to create an account.
Login into Voting System
Logging into the Voting system to be redirected for casting the vote.
Creating a Private Session.
Metamask extension is used to generate a private Voting session with given credentials
Entering the Polling Session
User enters into the Polling session where they can cast their votes.
Displaying total number of votes
Admin after entering his credentials can view the total votes enrolled.

Figure 3.3 Functional Architecture

3.4 MODULAR DESIGN

The modular structure defines the structure of the overall module. Modularity is a general concept typically defined as a degree to which a system's components may be separated and recombined.

The proposed system contains the following modules:

- Vote Session Creation using Ganache and Metamask
- Voter Login and Blockchain Creation
- Implementation of Voting into the System
- Generation of Vote Count (Admin only)

3.4.1 Vote Session Creation using Ganache and Metamask

The user who has to vote in the Voting System is provided with a separate block available in Ganache in which each block created has a private number referred to as a Private Key. Ganache is a personal Blockchain and is a high-end development tool for rapid Ethereum and Corda distributed application development. With this private key, after logging in to Metamask, the user is allowed to enter into the voting system by creating a separate session in which he is allowed to vote only once. After acquiring the Private key for the particular user, the user enters the Metamask credentials after logging in to Metamask which generates a private voting session allowing the user to vote once in the session using the given private key. Metamask allows users to keep and manage account keys, broadcast transactions, send and receive Ethereum-based money and tokens, and securely connect to decentralized applications using a suitable web browser or the mobile app's built-in browser. Metamask is a web browser extension. After connecting to Metamask and signing into the voting system, the user is allowed to log in and vote in the system.

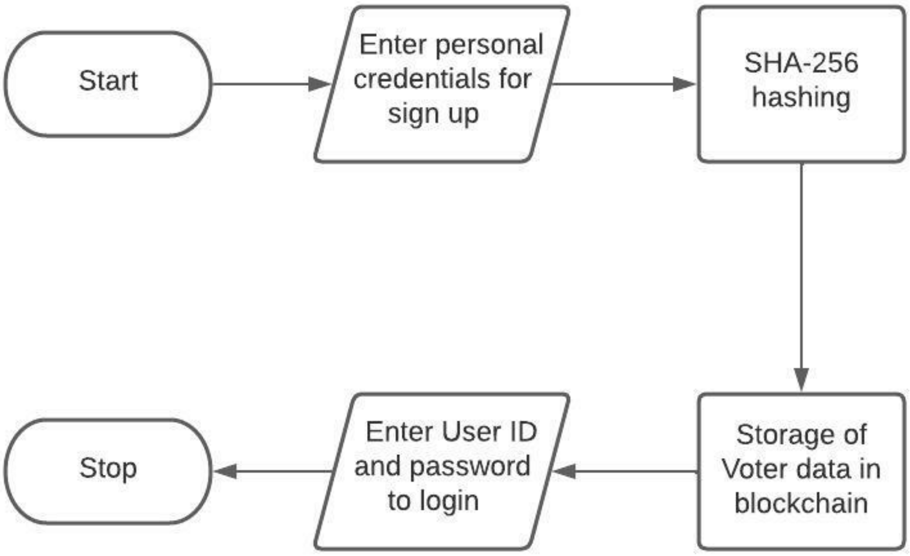

Start
Enter personal credentials for sign up
SHA-256 hashing
Storage of Voter data in blockchain
Enter User ID and password to login
Stop

Figure 3.4.1 Flow chart for Vote Session Creation

3.4.2 Voter Login and Blockchain Creation

After generating a private key using Ganache and creating a vote session using Metamask, the voter is then allowed to register with the voting system. Firstly, the voter must enter their personal credentials like name, date of birth, Aadhar details, etc. and the given voter data is hashed using the SHA-256 algorithm. Once the data is hashed a Blockchain is created which is specific to that particular voter. The User ID and password are then generated which the voter uses to log in to the system to cast their vote.

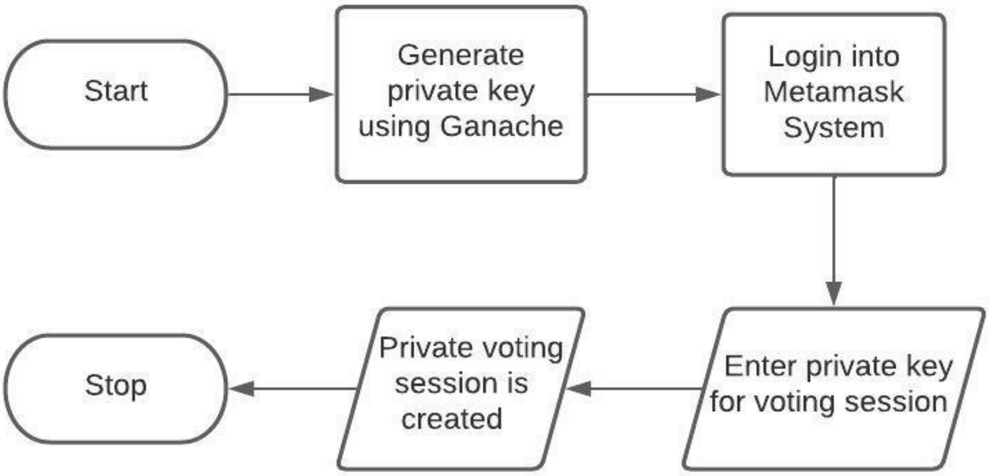

Start
Generate private key using Ganache
Login into Metamask System
Enter private key for voting session
Private voting session is created
Stop

Figure 3.4.2 Flow chart of Voter Login and Blockchain Creation

3.4.3 Implementation of Voting into the System

Users are allowed into the voting page after setting up Metamask and logging into the system using the credentials done in sign up. After logging in, the user has to select the party for which he/she wants to vote from the dropdown menu and click the party for whom he wants to select and submit the vote. Once the user submits a vote, there is no possibility of changing the vote or removing it as it gets stored in the block permanently and whenever the user logins again into the system, the vote button is disabled for the particular user.

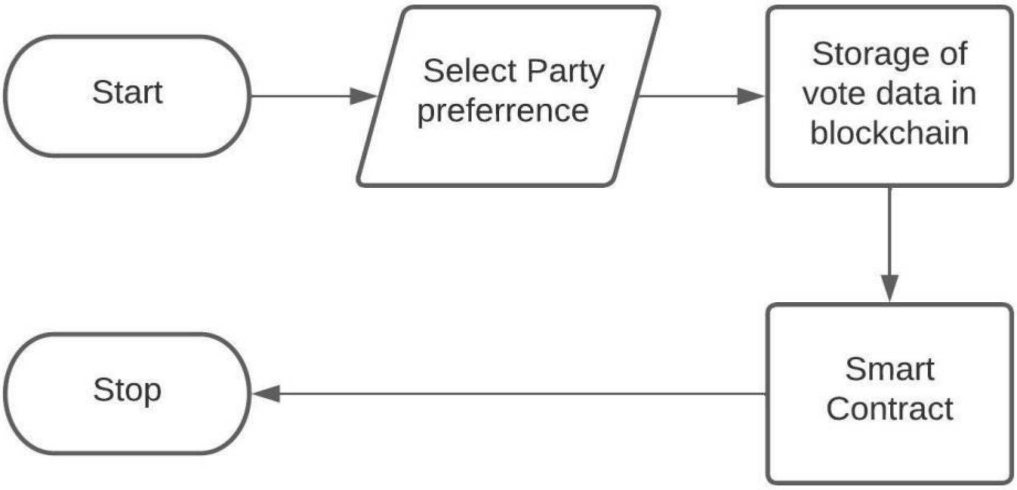


Figure 3.4.3 Flow chart for Voting into System

3.4.4 Generation of Vote Count (Admin only)

After the user votes in the system, there is no more accessibility for them in the system. The result of the election, which is the total number of votes acquired by each party can only be seen by the admin in the admin login where the admin has no access to manipulation of data. The total number of votes of each party is displayed in a table where the party with the maximum votes is considered the winner of the election.

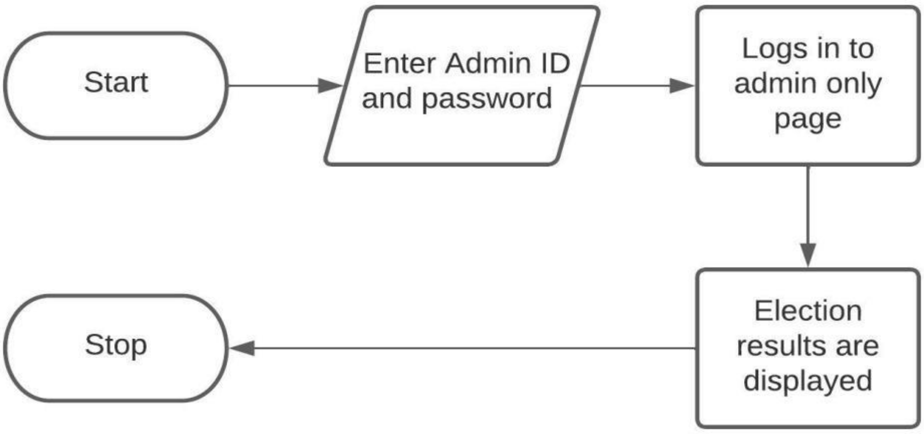

Start
Enter Admin ID and password
Logs in to admin only page
Election results are displayed
Stop

Figure 3.4.4 Flow chart for Generation of Vote Count (Admin only)

3.5. SYSTEM REQUIREMENT

3.5.1 Hardware Specification

Processor - Intel i3 AMD/ Ryzen or above

RAM - 4GB or above.

Internet Connectivity - Stable

3.5.2 Software Specification

Operating System - Windows 7 and above

Language - Solidity, HTML, CSS and Javascript (Node JS)

Web Extension- Metamask

3.6 SUMMARY

This chapter gives an overview of system design and its importance in the software life cycle. The system architecture provides what the system comprises. The functional architecture gives the entire functionality of the proposed system along with its modular structure and its interactions between the modules.

CHAPTER 4

SYSTEM IMPLEMENTATION

4.1 GENERAL

This describes the implementation of the project. It describes various modules of the proposed system, tools, and platforms, which were used. Implementation details are also mentioned and snapshots of output for each module are displayed. The following presents the input and output models used while implementing the proposed system.

4.2 OVERVIEW OF THE PLATFORM

4.2.1 Metamask

Metamask is a software cryptocurrency wallet used to interact with the Ethereum Blockchain. It allows users to access their Ethereum wallet through a browser extension or mobile app, which can then be used to interact with decentralized applications. Metamask allows users to store and manage account keys, broadcast transactions, send and receive Ethereum-based cryptocurrencies and tokens, and securely connect to decentralized applications through a compatible web browser or the mobile app's built-in browser. Metamask is a browser plugin. At its core, it serves as an Ethereum wallet: By installing it, you will get access to a unique Ethereum public address, with which you can start sending and receiving ether or tokens.

4.2.2 Ethereum

Ethereum is a decentralized, open-source Blockchain with smart contract functionality. Ether (ETH or Ξ) is the native cryptocurrency of the platform. Among cryptocurrencies, Ether is second only to Bitcoin in market capitalization. Ethereum allows anyone to deploy permanent and immutable decentralized applications onto it, with which users can interact. Ethereum also allows users to create and exchange NFTs, which are unique tokens representing ownership of an associated asset or privilege, as recognized by any number of institutions. Ethereum is a decentralized Blockchain platform that establishes a peer-to-peer network that securely executes and verifies application code, called smart contracts. Ethereum offers an extremely flexible platform on which to build decentralized applications using the native Solidity scripting language and Ethereum Virtual Machine. Decentralized application developers who deploy smart contracts on Ethereum benefit from the rich ecosystem of developer tooling and established best practices that have come with the maturity of the protocol.

4.2.3 Smart Contract

A smart contract is an application code that resides at a specific address on the Blockchain known as a contract address. Applications can call the smart contract functions, change their state, and initiate transactions. They typically are used to automate the execution of an agreement so that all participants can be immediately certain of the outcome, without an intermediary's involvement or time loss. Smart contracts allow participants to transact with each other without a trusted central authority. Transaction records are immutable, verifiable, and securely distributed across the network, giving participants full ownership and visibility into transaction data. Smart contracts are written in programming languages such as Solidity and Vyper and are compiled by the Ethereum Virtual Machine into bytecode and executed on the Blockchain.

4.2.4 Solidity

Solidity is an object-oriented, high-level language programming language for implementing smart contracts on various Blockchain platforms Solidity is a high-level programming language for implementing smart contracts on multiple Blockchain platforms that is object-oriented. It's used to establish smart contracts in the Blockchain system that apply business logic and generate a chain of transaction records. User-defined programming, libraries, and inheritance are all supported in Solidity. Solidity is the core programming language used by Blockchain systems. Solidity is a programming language that may be used to develop contracts such as voting, blind auctions, crowdsourcing, multi-signature wallets, and more. Solidity has been intended to target the Ethereum Virtual Machine and is heavily influenced by C++, Python, and JavaScript (EVM).

4.2.5 HTML, CSS

Hyper Text Markup Language (HTML) is the standard markup language for documents designed to be displayed in a web browser. HTML describes the structure of a web page semantically and originally included cues for the appearance of the document. HTML elements are the building blocks of HTML pages. With HTML constructs, images and other objects such as interactive forms may be embedded into the rendered page.

Cascading Style Sheets (CSS) is a style sheet language used for describing the presentation of a document written in a markup language such as HTML. It is a simple mechanism for adding style (e.g., fonts, colors, spacing) to web documents. It also supports content positioning, table layout, features for internationalization, and some properties related to the user interface.

4.2.6 Node JS

Node.js is an open-source, cross-platform, back-end JavaScript runtime environment that runs on the V8 engine and executes JavaScript code outside a web browser. Node.js represents a "JavaScript everywhere" paradigm, unifying web application development around a single programming language, rather than different languages for server-side and client-side scripts. A Node.js app runs in a single process, without creating a new thread for every request. Node.js provides a set of asynchronous I/O primitives in its standard library that prevent JavaScript code from blocking and generally, libraries in Node.js are written using non-blocking paradigms, making blocking behavior the exception rather than the norm. Node.js has a unique advantage because it is able to write the server-side code in addition to the client-side code without the need to learn a completely different language.

4.2.7 Ganache

Ganache is a personal Blockchain and is a high-end development tool for rapid Ethereum and Corda distributed application development. You can use Ganache across the entire development cycle; enabling you to develop, deploy, and test your Apps in a safe and deterministic environment. Ganache comes in two flavors: a UI and CLI. It is used for setting up a personal Ethereum Blockchain for testing your Solidity contracts. Ganache is helpful in all parts of the development process. The local chain allows you to develop, deploy and test your projects and smart contracts in a deterministic and safe environment. Ganache is effective and efficient because it saves a lot of money and time.

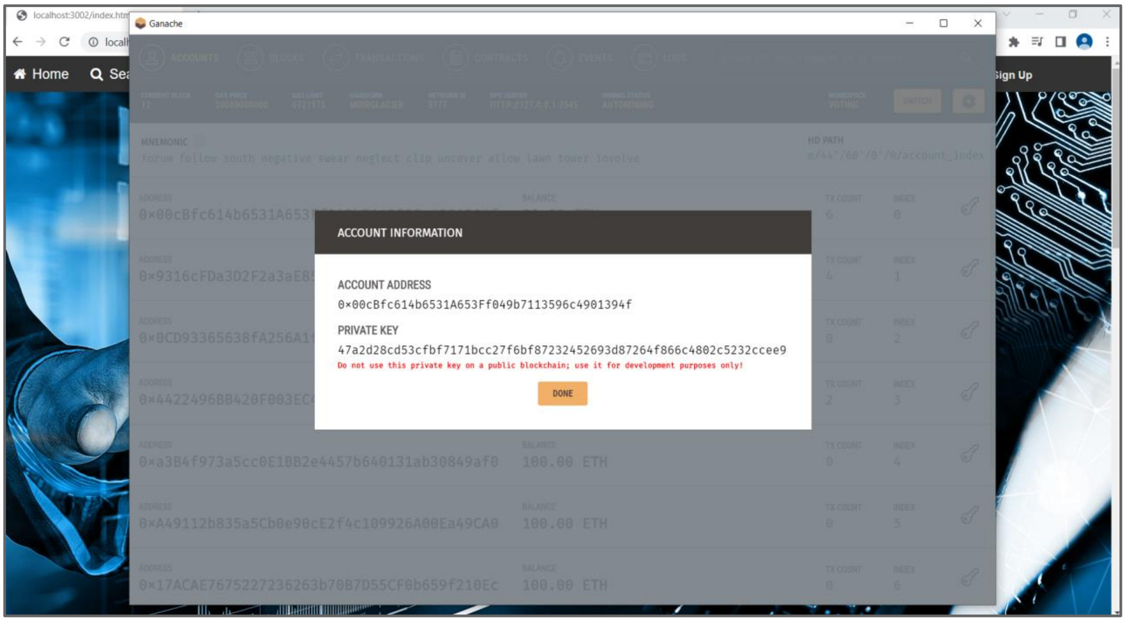
4.2.8 SHA-256

Secure Hashing Algorithm (SHA) -256 is the hash function and mining algorithm of the Bitcoin protocol, referring to the cryptographic hash function that outputs a 256-bit long value. It moderates the creation and management of addresses and is also used for transaction verification. It is a hashing algorithm used to convert the text of any length into a fixed-size string of 256 bits (32 bytes). It takes any input and produces an output (often called a hash) of fixed length. It doesn't matter if the input is a single word, a full sentence, a page from a book, or an entire book, the output of a hashing algorithm like SHA256 will always be the same length. Specifically, it will be 256 bits, which is 32 bytes, which is displayed as 64 alphanumeric characters.

4.3 MODULE IMPLEMENTATION

4.3.1 Implementation of generating a private key for a User

The user who has to vote in the Voting System is provided with a separate block available in Ganache in which each block created has a private number referred to as a Private Key. Ganache is a personal Blockchain and is a high-end development tool for rapid Ethereum and Corda distributed application development. With this private key, after logging in to Metamask, the user is allowed to enter into the voting system by creating a separate session in which he is allowed to vote only once.



ACCOUNT INFORMATION

ACCOUNT ADDRESS

0×00cBfc614b6531A653Ff049b7113596c4901394f

PRIVATE KEY

47a2d28cd53cfbf7171bcc27f6bf87232452693d87264f866c4802c5232ccee9

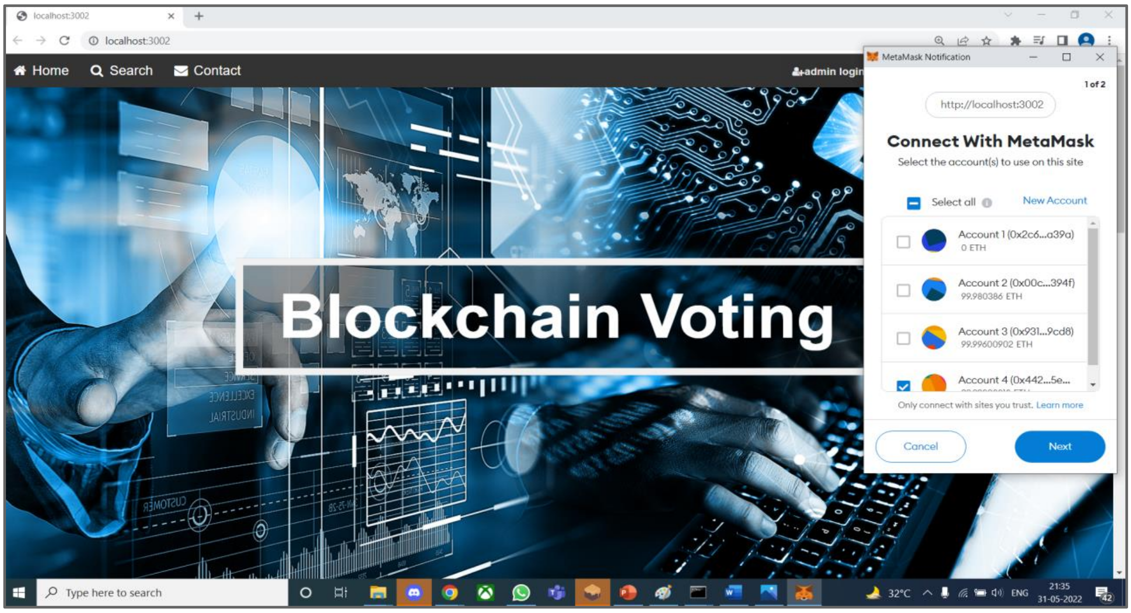
Do not use this private key on a public blockchain; use it for development purposes only!

DONE

Figure 4.3.1 Private Key Generation

4.3.2 Implementation of creating a Private voting session using Metamask

After acquiring the Private key for the particular user, the User enters the Metamask credentials after logging in to Metamask which generates a private voting session allowing the user to vote once in the session using the given private key. Metamask allows users to store and manage account keys, broadcast transactions, send and receive Ethereum-based cryptocurrencies and tokens, and securely connect to decentralized applications through a compatible web browser or the mobile app's built-in browser. Metamask is a browser plugin. Post connection of Metamask and signing into the voting system, the user is allowed to log in and vote in the system.



1 of 2

http://localhost:3002

Connect With MetaMask

Select the account(s) to use on this site

☐ Select all New Account

☐ Account 1 (0x2c6...a39a) 0 ETH

☐ Account 2 (0x00c...394f) 99.980386 ETH

☐ Account 3 (0x931...9cd8) 99.99600902 ETH

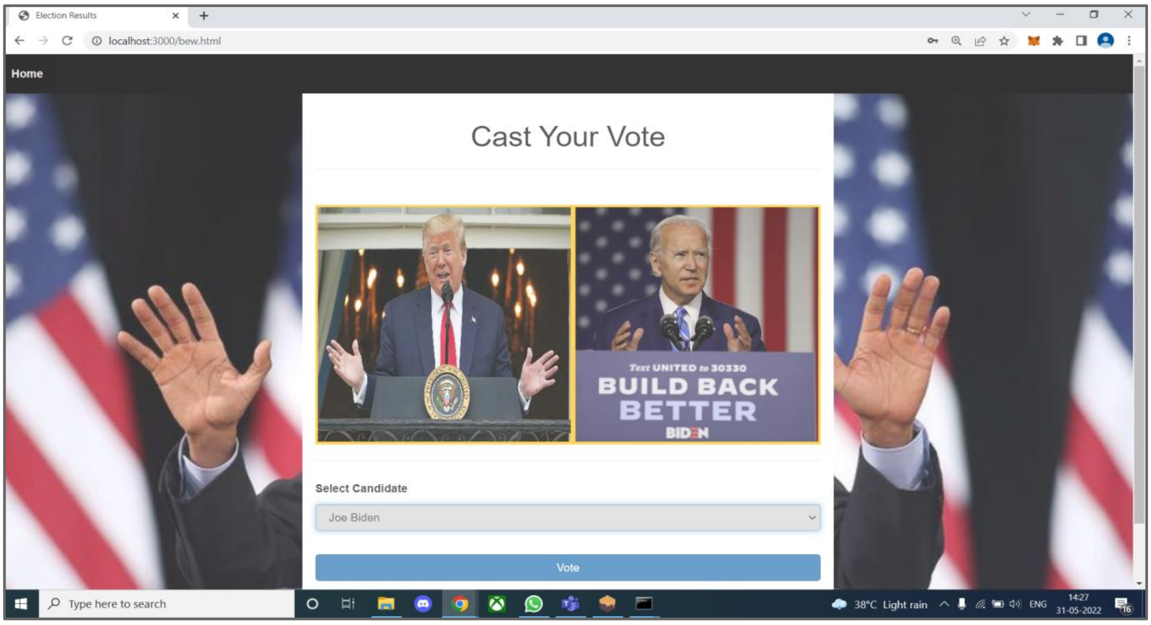
☑ Account 4 (0x442...5e...

Only connect with sites you trust. Learn more

Figure 4.3.2 Creation of Private Voting Session

4.3.3 Implementation of Voting into the System

Users are allowed into the voting page after setting up Metamask and logging into the system using the credentials done in sign up. After logging in, the user has to select the party for which he/she wants to vote from the dropdown menu and click the party for whom he wants to select and submit the vote. Once the user submits a vote, there is no possibility of changing the vote or removing it as it gets stored in the block permanently and whenever the user logins again into the system, the vote button is disabled for the particular user.



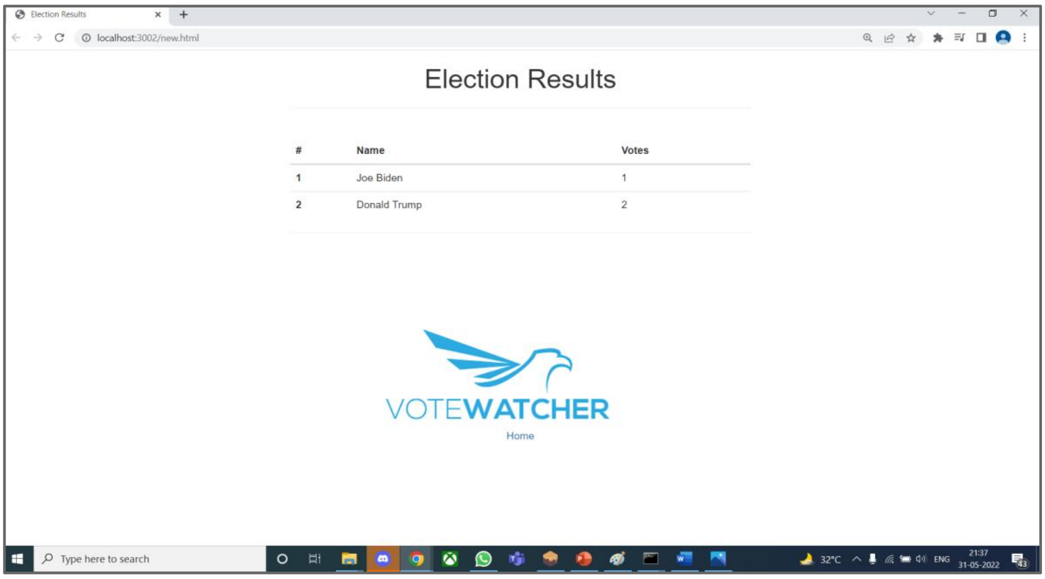
Cast Your Vote

Select Candidate: Joe Biden

Figure 4.3.3 Cast one’s vote in the system

4.3.4 Implementation of displaying Election Results

After the user votes in the system, there is no more accessibility for them in the system. The result of the election, which is the total number of votes acquired by each party can only be seen by the admin in the admin login where the admin has no access to manipulation of data. The total number of votes of each party is displayed in a table where the party with the maximum votes is considered as the winner of the Election.



Election Results

#	Name	Votes
1	Joe Biden	1
2	Donald Trump	2

Figure 4.3.4 Displaying Election Results

4.4 SUMMARY

This chapter brought out the analysis of each technology used to develop this project. This chapter also brought out the basic system implementations such as the platforms, languages, and tools used here. And the screenshots of different modules implemented using those platforms, languages, and tools.

CHAPTER 5

SYSTEM TESTING AND PERFORMANCE ANALYSIS

5.1 GENERAL

Testing is the process of trying to find out every conceivable fault or weakness in a work product. It provides a way to check the functionality of components, subassemblies, assemblies, and finished products. It is the process of exercising software with the intent of ensuring that the software system meets its requirements and user expectations.

It is an integral part of the entire development and maintenance process. The goal of the testing during this phase is to verify that the specification has been accurately and completely incorporated into the design, as well as to ensure the correctness of the design itself. For example, the design must not have any logic faults in the design and it must be detected before coding commences, otherwise, the cost of fixing the faults will be considerably higher as reflected. Detection of design faults can be achieved by means of inspection as well as a walkthrough.

Testing is one of the important steps in the software development phase. Testing checks for the errors, as a whole project testing involves the following test cases. Static analysis is used to investigate the structural properties of the Source code. Dynamic testing is used to investigate the behavior of the source code by executing the program on the test data.

5.2 TEST CASE

The system was tested with 5 participants to measure accuracy and recognition time. The users were made to speak pseudo-codes of five different programs and accuracy and time were noted for each user. The programs were in increasing levels of difficulty.

5.3 TEST CASE DESIGN

Testing is the process of running a program with the goal of detecting errors. An excellent test case is one that has a high chance of uncovering an as-yet-undiscovered error and has a high probability of detecting an as-yet-undiscovered error. System testing is a stage of implementation focused at ensuring that the system performs as planned before going live. System testing necessitates a test containing numerous key actions and processes for every program, string, and system, and is critical to the effective implementation of a new system. Before the system is installed for user acceptability testing, this is the last chance to find and rectify faults.

After the program has been constructed and the documentation and relevant data structures have been designed, the software testing process begins. Errors in software must be corrected, which necessitates the use of software testing. Software testing is an important part of software quality assurance since it provides the final check on specification design and code. Testing is the process of running a program with the goal of identifying errors. A good test case design is one that has a chance of uncovering an error that has yet to be detected. A successful test is one that uncovers a previously unknown flaw.

5.4 PERFORMANCE MEASURES

5.4.1 Accuracy:

Accuracy is the measure of the quality of the result.

5.4.2 Precision:

Precision refers to how close measurements of the same item are to each other.

5.5 SUMMARY

This chapter provides a general overview of the various testing techniques that are used throughout the project development process. It also takes into account the performance study of a system that has been shown to boost user satisfaction when compared to the previous system.

CHAPTER 6

CONCLUSION AND FUTURE WORK

6.1 CONCLUSION

This chapter concludes all the ideas that were discussed until now in the proposed system. In today's world, the idea of adapting digital voting techniques to make the public political process cheaper, faster, and easier is appealing. This paper is thus made to give the readers a survey of various papers on Blockchain-based voting systems. Making the voting process inexpensive and quick normalizes it in the eyes of voters, lowers a power barrier between the voter and the elected official, and puts pressure on the elected official. We presented a novel Blockchain-based electronic voting system in this study that provides secure and cost-effective elections. We introduced a unique, Blockchain-based electronic voting system that ensures secure and cost-efficient elections. We have used a database to store the data of the voter which is monitored by the admin console to safeguard the privacy of the voter. An Ethereum-based Blockchain mechanism is used to store the vote data of each voter and To protect the privacy of the voter, we employed the SHA-256 hashing mechanism. The vote data of each voter is stored using a Blockchain system, and the vote data is protected in it.

6.2 FUTURE WORK

The future work of this project can include an increase in the usability of the system and optimize the system using Aadhaar authentication and user verification using Real time data. We would like to add another Blockchain system to add the voter credentials instead of a database. We also want to get our voting system mainstream and use it in actual real-life elections.

APPENDICES

SAMPLE CODING

index.html:

```
<!DOCTYPE html>

<html>

<head>

  <meta charset="utf-8">

  <meta http-equiv="X-UA-Compatible" content="IE=edge">

<meta name="viewport" content="width=device-width, height=device-height,initial-scale=1">

<link rel="stylesheet" href="https://cdnjs.cloudflare.com/ajax/libs/font-awesome/4.7.0/css/font-awesome.min.css">

<link href="https://fonts.googleapis.com/icon?family=Material+Icons" rel="stylesheet">

<!--<link href="css/bootstrap.min.css" rel="stylesheet">-->

<style>

body, html {

  height: 100%;

  font-family: Arial, Helvetica, sans-serif;

  margin: 0px;

}

*{
  box-sizing: border-box;
}
/* Full-width input fields */
input[type=text], input[type=password], input[type=tel], input[type=date],
select {
  width: 100%;
  padding: 15px;
  margin: 5px 0 22px 0;
  display: inline-block;
  border: none;
  background: #f1f1f2;
}
/* Add a background color when the inputs get focus */
input[type=text]: focus, input[type=password]:
focus,input[type=tel]:focus,input[type=date]:focus,select:focus{
  background-color: #ddd;
  outline: none;
}
/* Set a style for all buttons */
button {
  background-color: #333;
  color: white;
  padding: 14px 20px;

  margin: 8px 0;

  border: none;

  cursor: pointer;

  width: 100%;

  opacity: 0.9;

}

button:hover {

  background-color: #4CAF50;

  opacity:1;

}

/* Extra styles for the cancel button */

.cancelbtn {

  padding: 14px 20px;

  background-color: #f44336;

}

/* Float cancel and signup buttons and add an equal width */

.cancelbtn, .signupbtn {

  float: left;

  width: 50%;

}

/* Add padding to container elements */
```

```
.container {
  padding: 16px;
}
/* The Modal (background) */
.modal {
  display: none; /* Hidden by default */
  position: fixed; /* Stay in place */
  z-index: 1; /* Sit on top */
  left: 0;
  top: 0;
  width: 100%; /* Full width */
  height: 100%; /* Full height */
  overflow: auto; /* Enable scroll if needed */
  /*background-color: #474e5d;*/
  padding-top: 50px;
}
/* Modal Content/Box */
.modal-content {
  background-color: #fefefe;
  margin: 5% auto 15% auto; /* 5% from the top, 15% from the bottom and
centered */
  border: 1px solid #888;
  width: 50%; /* Could be more or less, depending on screen size */
}
/* Style the horizontal ruler */
hr {
  border: 1px solid #f1f1f1;
  margin-bottom: 25px;
}
/* The Close Button (x) */
.close {
  position: absolute;
  right: 35px;
  top: 15px;
  font-size: 40px;
  font-weight: bold;
  color: #f1f1f1;
}
.close:hover,
.close:focus {
  color: #f44336;
  cursor: pointer;
}
/* Clear floats */
```

```
.clearfix::after {
  content: "";
  clear: both;
  display: table;
}
.navbar {
  position: fixed;
 width: 100%;
 background-color: #333;
 margin-top: 0px;
 margin-right: 0px;
 overflow: auto;
}
.navbar a {
 float: left;
 text-align: center;
 padding: 12px;
 color: white;
 text-decoration: none;
 font-size: 17px;
}
.navbar a:hover {
 background-color: #4CAF50;
}
/* Change styles for cancel button and signup button on extra small screens */
@media screen and (max-width: 300px) {
  .cancelbtn, .signupbtn {
     width: 100%;
  }
}
.bg-img {
  /* The image used */
  background-image: url("./Blockchain-blog.png");
  /*position: relative;*/
  height:100%;
  background-attachment: fixed;
  background-position: center;
  background-repeat: no-repeat;
  background-size:100% 100% ;
/*background: hsla(180,0%,50%,0.25)
background: rgba(250,250,250,250,0);*/
}
.bg-img2 {
  /* The image used */
  background-image: url("./Blockchain-in-Democracy.jpg");
  /*position: relative;*/
  height:100%;
  background-attachment: fixed;
  background-position: center;
  background-repeat: no-repeat;
  background-size:100% 100% ;
/*background: hsla(180,0%,50%,0.25)
background: rgba(250,250,250,250,0);*/
}
.bg-img1 {
  /* The image used */
background-image: url("./The-Importance-of-Cyber-Security-in-Modern-Internet-Age-1.jpg");
  /*position: relative;*/
  height:100%;
  background-attachment: fixed;
  background-position: center;
  background-repeat: no-repeat;
  background-size:cover ;
/*background: hsla(180,0%,50%,0.25)
background: rgba(250,250,250,250,0);*/
```

```
}
.but{
  margin-left:65%;
  padding:10px;
  margin-top:5px;
  position:fixed;
}
.but1{
  margin-left:58%;
  padding:10px;
  margin-top:5px;
  position:fixed;
}
.but2{
  padding:10px;
  margin-top:5px;
  margin-left:48%;
  position:fixed;
}
.bg-text {
  background-color: rgb(0,0,0); /* Fallback color */
  background-color: rgba(0,0,0, 0.4); /* Black w/opacity/see-through */
  color: white;
  font-weight: bold;
  font-size: 80px;
  border: 10px solid #f1f1f1;
  position: fixed;
  top: 50%;
  left: 50%;
  transform: translate(-50%, -50%);
  z-index: 2;
  width: 800px;
  padding: 20px;
  text-align: center;
}
</style>
<body >
  <div class="navbar">
  <a class="active" href="#"><i class="fa fa-fw fa-home"></i> Home</a>
  <a href="#"><i class="fa fa-fw fa-search"></i> Search</a>
  <a href="./bew.html"><i class="fa fa-fw fa-envelope"></i> Contact</a>
<button class="but1"
onclick="document.getElementById('id02').style.display='block',document.get
ElementById('tex').style.display='none'" style="width:auto;"><b class="fa fa-
fw fa-user"></b><b>Login</b></button>
```

```
<button class="but2"
onclick="document.getElementById('id03').style.display='block',document.get
ElementById('tex').style.display='none'" style="width:auto;"><b class="fa fa-
user-plus"></b><b>admin login</b></button>

<button class="but"
onclick="document.getElementById('id01').style.display='block',document.get
ElementById('tex').style.display='none'" style="width:auto;"><b class="fa fa-
user-plus"></b><b>Sign Up</b></button>

</div>

<div class="bg-img1"></div>

<div id="tex" class="bg-text">Blockchain Voting</div>

<div class="bg-img"></div>

<div class="bg-img2"></div>

<div id="id01" class="modal">

  <span onclick="document.getElementById('id01').style.display='none'"
class="close" title="Close Modal">&times;</span>

  <form class="modal-content"  onsubmit="App.reg(); return false;">

    <div class="container">

      <h1>Sign Up</h1>

      <p>Please fill in this form to create an account.</p>

      <hr>

<label for="name"><b>Name</b></label>  <i class="fa fa-user icon"></i>

<input id="name" type="text" placeholder="Enter Name" name="name"
required>
```

```
<label for="email"><b>Email</b></label> <i class="fa fa-envelope icon"></i>

<input id="email" type="text" placeholder="Enter Email" name="email" required>

<label for="psw"><b>Password</b></label> <b class="fa fa-key icon"></b>

<input id="psw" type="password" placeholder="Enter Password" name="psw" required>

<label for="psw-repeat"><b>Repeat Password</b></label> <b class="fa fa-key icon"></b>

<input type="password" placeholder="Repeat Password" name="psw-repeat" required>

<labelfor="birthday"><b>D.O.B:</b></label> <i class="material-icons">&#xe916;</i>

<input type="date" id="birthday" name="birthday">

<label for="gender"><b>Select Gender:</b></label>

<select id="gender" placeholder="enter">

 <option value="volvo">Male</option>

 <option value="saab">Female</option>

 <option value="opel">Others</option>

 <option value="audi">Not reveal</option>

</select>

  <label for="adhar-number"><b>ADHARNUBER</b></label> <b class="fas fa-address-book"></b>

    <input type="text" id="adhar-number" name="A.no" placeholder="ENTER
```

```
YOUR ADHAR NO.">
   <label for="voteid"><b>Voter Id:</b></label> <b class="fas fa-address-
book"></b>
     <input type="text" id="voteid" name="voteid" placeholder="ENTER
YOUR VOTER ID NO:.">
  <label for="number"></label>
  <label for="phone"><b>Enter your phone number:</b></label></label> <b
class="fas fa-phone"></b>
    <input type="tel" id="phone" name="phone"
    pattern="[0-9]{10}"
    required>
<small>Format: 9490513655</small>
<p>By creating an account you agree to our <a href="#"
style="color:dodgerblue">Terms & Privacy</a>.</p>
     <label>
       <input type="checkbox" checked="checked" name="remember"
style="margin-bottom:15px"> Remember me
     </label>
     <div class="clearfix">
<button type="button"
onclick="document.getElementById('id01').style.display='none'"
class="cancelbtn">Cancel</button>
       <button type="submit" class="signupbtn">Sign Up</button>
     </div>
   </div>
  </form>
</div>
<div id="id02" class="modal">
  <span onclick="document.getElementById('id02').style.display='none'" class="close" title="Close Modal">&times;</span>
  <form class="modal-content"  onsubmit="App.login(); return false;">
   <div class="container">
     <h1>Login</h1>
     <p>Please fill in this form to create an account.</p>
     <hr>
     <label for="username"><b>Username</b></label>
     <input id="username" type="text" placeholder="Enter Username" name="username" required>
     <label for="psw"><b>Password</b></label>
     <input id="psw" type="password" placeholder="Enter Password" name="psw" required>
       <!--<label for="psw-repeat"><b>Repeat Password</b></label>
     <input type="password" placeholder="Repeat Password" name="psw-repeat" required>-->
   <label>
       <input type="checkbox" checked="checked" name="remember" style="margin-bottom:15px"> Remember me
      </label>
<p>By creating an account you agree to our <a href="#"
style="color:dodgerblue">Terms & Privacy</a>.</p>
<div class="clearfix">
<button type="button"
onclick="document.getElementById('id02').style.display='none'"
class="cancelbtn">Cancel</button>
        <button type="submit" class="signupbtn">Login</button>
      </div>
    </div>
  </form>
</div>
<div id="id03" class="modal">
  <span onclick="document.getElementById('id03').style.display='none'"
class="close" title="Close Modal">&times;</span>
  <form class="modal-content"  onsubmit="App.admin(); return false;">
    <div class="container">
      <h1>Login</h1>
      <p>Please fill in this form to create an account.</p>
      <hr>
      <label for="username"><b>Username</b></label>
      <input id="username1" type="text" placeholder="Enter Username"
name="username" required>
```

```
      <label for="psw"><b>Password</b></label>
      <input id="psw1" type="password" placeholder="Enter Password"
name="psw" required>
        <!--<label for="psw-repeat"><b>Repeat Password</b></label>
      <input type="password" placeholder="Repeat Password" name="psw-
repeat" required>-->
<label>
<input type="checkbox" checked="checked" name="remember"
style="margin-bottom:15px"> Remember me
</label>
<p> By creating an account you agree to our <a href="#"
style="color:dodgerblue">Terms & Privacy</a>.</p>

      <div class="clearfix">
<button type="button"
onclick="document.getElementById('id03').style.display='none'"
class="cancelbtn">Cancel</button>
        <button type="submit" class="signupbtn">Login</button>
      </div>
    </div>
  </form>
</div>
<script>
```

```
// Get the modal
var modal = document.getElementById('id01');
var modal2 = document.getElementById('id02');
var modal3 = document.getElementById('id03');
// When the user clicks anywhere outside of the modal, close it
window.onclick = function(event) {
 if (event.target == modal) {
  modal.style.display = "none";
  document.getElementById('tex').style.display='block';
 }
 else if (event.target == modal2) {
   modal2.style.display = "none";
   document.getElementById('tex').style.display='block';
 }
 else if (event.target == modal3) {
   modal3.style.display = "none";
   document.getElementById('tex').style.display='block';
}
}
</script>
<!-- jQuery (necessary for Bootstrap's JavaScript plugins) -->
<script
```

```
src="https://ajax.googleapis.com/ajax/libs/jquery/1.12.4/jquery.min.js"></scrip
t>
  <!-- Include all compiled plugins (below), or include individual files as
needed -->
  <script src="js/bootstrap.min.js"></script>
  <script src="js/web3.min.js"></script>
  <script src="js/truffle-contract.js"></script>
  <script src="js/app.js"></script>
</body>
</html>
```

Election.contract

```
pragma solidity >= 0.4.21 < 0.7.0;
contract Election {
    // Model a Candidate
    struct Candidate {
        uint id;
        string name;
        uint voteCount;
    }
```

```
mapping(string => string) public reg;

// Store accounts that have voted

mapping(address => bool) public voters;

// Store Candidates

// Fetch Candidate

mapping(uint => Candidate) public candidates;

// Store Candidates Count

uint public candidatesCount;

// voted event

event votedEvent (

    uint indexed _candidateId

);

constructor () public {

  addCandidate("Joe Biden");

    addCandidate("Donald Trump");

}

function addCandidate (string memory _name) private {

    candidatesCount ++;

    candidates[candidatesCount] = Candidate(candidatesCount, _name, 0);

}

function vote (uint _candidateId) public {

    // require that they haven't voted before
```

```
        require(!voters[msg.sender]);
        // require a valid candidate
        require(_candidateId > 0 && _candidateId <= candidatesCount);
        // record that voter has voted
        voters[msg.sender] = true;
        // update candidate vote Count
        candidates[_candidateId].voteCount ++;
        // trigger voted event
        emit votedEvent(_candidateId);
    }
    function register(string memory _user, string memory _pass ) public{
      reg[_user]=_pass;
    }}
```

Migrations.contract

```
// SPDX-License-Identifier: MIT
pragma solidity >=0.4.21 <0.7.0;
contract Migrations {
  address public owner;
  uint public last_completed_migration;

  modifier restricted() {

    if (msg.sender == owner) _;

  }

  constructor() public {

    owner = msg.sender;

  }

  function setCompleted(uint completed) public restricted {

    last_completed_migration = completed;

  }

}
```

REFERENCES

1. Anusha Vangala, Anil Kumar Sutrala, Ashok Kumar Das, Minho Jo, (2021), ‘Smart Contract-Based Blockchain-Envisioned Authentication Scheme for Smart Farming’, IEEE Internet, Vol.6, Issue: 4, Page(s):20506 - 20519.

2. Basit Shahzad, Jon Crowcroft, (2019), ‘Trustworthy Electronic Voting Using Adjusted Blockchain Technology’, IEEE Access, Vol.7, Page(s): 24477 - 24488.

3. Dongliang Xu, Wei Shi, Wenshang Zhai, Zhihong Tian, (2021), ‘Multi-Candidate Voting Model Based on Blockchain’, IEEE/CAA Journal of Automatica Sinica, Vol.8, Issue: 12, Page(s): 1891 - 1900.

4. Geetanjali Rathee, Razi Iqbal, Omer Waqar, Ali Kashif Bashir, (2021), ‘On the Design and Implementation of a Blockchain-Enabled E-Voting Application Within IoT- Oriented SmartCities’, IEEE Access, Vol. 9, Issue-10 Page(s): 34165 -
34176.

5. Huilin Li, Yannan Li, Yong Yu, Baocang Wang, Kefei Chen, (2020), ‘A Blockchain-Based Traceable Self-Tallying E-Voting Protocol in AI Era’, IEEE Transactions on Network Science and Engineering, Vol.8, Issue: 2, Page(s): 1019-1032.

6. Santeri Paavolainen, Christopher Carr, (2020), ‘Security Properties of Light Clients on the Etherium Blockchain’, IEEE Access, Vol.8, Page(s): 124339 - 124358.

7. Saurabh Singh, A.S.M. Sanwar Hosen, Byungun Yoon, (2021), ‘Blockchain Security Attacks, Challenges, and Solutions for the Future Distributed IoT Network’, IEEE Access, Vol.9, Page(s): 13938 - 13959.

8. Somnath Panja, Samiran Bag, Feng Hao, Bimal Roy, (2020), ‘A Smart Contract System for Decentralized Borda Count Voting’, IEEE Transactions on Engineering Management, Vol.67, Issue: 4, Page(s): 1323 – 1339.

9. Xiwang Yung, Chao Liang, Miao Zhao, Hongwei Wang, Hao Ding, Yong Liu, Yang Li, Junlin Zhang, (2017), ‘Collaborative Filtering-Based Recommendation of Online Social Voting’, IEEE Transactions on Computational Social Systems, Vol.4, Issue: 1, Page(s): 1 - 13.

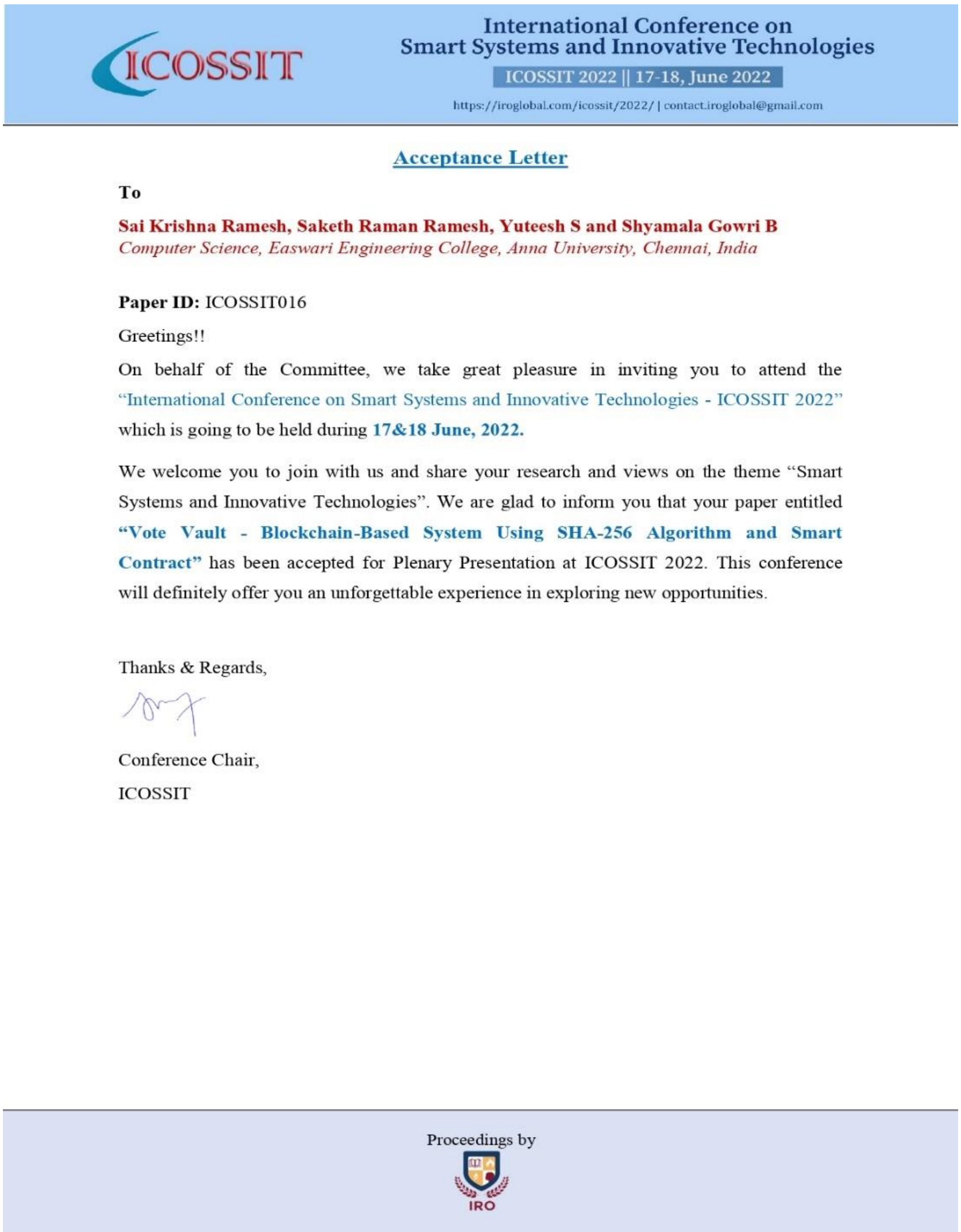
10. Xuechao Yang, Xun Yi, Surya Nepal, Andrei Kelarev, Fengling Han, (2018), ‘A Secure Verifiable Ranked Choice Online System Based on Homomorphic Encryption’, IEEE Access, Vol.6, Issue: 4, Page(s): 20506 - 20519.

LIST OF PUBLICATIONS

1. ICOSSIT 2022 (International Conference on Smart Systems and Innovative Technologies):

Authors: Sai Krishna Ramesh, Saketh Raman Ramesh, Yuteesh S, and Shyamala Gowri B

Title: Vote Vault - Blockchain-Based System Using SHA-256 Algorithm and Smart Contract



ICOSSIT

International Conference on
Smart Systems and Innovative Technologies
ICOSSIT 2022 || 17-18, June 2022
https://iroglobal.com/icossit/2022/ | contact.iroglobal@gmail.com

Acceptance Letter

To

Sai Krishna Ramesh, Saketh Raman Ramesh, Yuteesh S and Shyamala Gowri B
Computer Science, Easwari Engineering College, Anna University, Chennai, India

Paper ID: ICOSSIT016

Greetings!!

On behalf of the Committee, we take great pleasure in inviting you to attend the "International Conference on Smart Systems and Innovative Technologies - ICOSSIT 2022" which is going to be held during **17&18 June, 2022.**

We welcome you to join with us and share your research and views on the theme "Smart Systems and Innovative Technologies". We are glad to inform you that your paper entitled **"Vote Vault - Blockchain-Based System Using SHA-256 Algorithm and Smart Contract"** has been accepted for Plenary Presentation at ICOSSIT 2022. This conference will definitely offer you an unforgettable experience in exploring new opportunities.

Thanks & Regards,

Conference Chair,
ICOSSIT

Proceedings by
IRO