TRIBHUVAN UNIVERSITY

INSTITUTE OF SCIENCE AND TECHNOLOGY

Final Year Project Report on

E-voting using Blockchain

For the partial fulfillment of the requirements of the degree of

Bachelor of Science in Computer Science and Information Technology (B.Sc.CSIT)

awarded by Tribhuvan University

Submitted By:

Bijesh Shrestha (TU Roll No. 20327/075)

Milan Rawal(TU Roll No. 20338/075)

ST. XAVIER'S COLLEGE

Maitighar, Kathmandu, Nepal

Submitted To:

Office of the Dean

Institute of Science and Technology

Tribhuvan University

Kathmandu

12th March, 2023

CHAPTER 1:INTRODUCTION

1.1 INTRODUCTION

Voting is a method for a group, such as a meeting or an electorate, in order to make a collective decision or express an opinion usually following discussions, debates for election campaigns. In a democracy, the right to vote is the main way most citizens can influence the decisions about how their country is governed [1].

In a Democratic country, voting plays a huge role because it determines who will be the leaders of both the voters and nonvoters.There are different kinds of electronic voting systems used in several countries around the world like Punch card voting/tabulation systems, Optical scanning systems, DRE and Internet.E-voting is also used for voting in private institutions and organizations for voting purposes [2].

A blockchain is a distributed database or ledger that is shared among the nodes of a computer network. As a database, a blockchain stores information electronically in digital format. Blockchains are best known for their crucial role in cryptocurrency systems, such as Bitcoin, for maintaining a secure and decentralized record of transactions. The innovation with a blockchain is that it guarantees the fidelity and security of a record of data and generates trust without the need for a trusted third party [3].

The proposed system is a blockchain based E-voting system, a decentralized friendly interface and secured system such that the local public can cast their votes easily in the comfort of their home or office. Using blockchain prevents any possible fraud and increases the trust of voters to cast their votes as well as there is greater security.

1.2 PROBLEM STATEMENT

The existing voting system used in Nepal in the election process is done manually which has a lot of problems.So, through this project those problems can be minimized or removed totally. Some of the problems are:

- Cost of expenditure on elections is high.
- A lot of ballot papers are deemed invalid.
- Decreasing participation of voters in voting.

1.3 OBJECTIVES

The objectives of the proposed project is given below:

- To prevent fraud, manipulation, and unauthorized access to the voting process.
- To make voting more accessible to all eligible voters, regardless of their physical location.

1.4 PROJECT SCOPE

This project aims to develop a decentralized voting system that allows a voter to cast vote easily, securely and conveniently..If the technology is used correctly, the blockchain is a digital, decentralized, encrypted, transparent ledger that can withstand manipulation and fraud. Because of the distributed structure of the blockchain, a blockchain electronic voting system reduces the risks involved with electronic voting and allows for a tamper-proof for the voting system. Since, the votes can be cast online so the voters can cast their votes in the comfort of their own home.This helps in increasing the participation of voters which ultimately contributes to the appointment of the best candidates or the best option.

1.5 DEVELOPMENT METHODOLOGY

This project takes voter information in order to give votes to the candidates. The voter is required to register by providing voter information such as name, encrypted address and valid phone number. After the voter is registered. the admin will approve or disapprove the voter. After the voter is approved ,the voter is able to give votes to the candidates. Once the voter successfully gives votes to the candidate, the voting information is added to the blockchain.

Voters connect to the system using a metamask. The encrypted address used in this project is generated using ganache for testing purposes only. All this functionality is provided to the user throughout the web based interface.

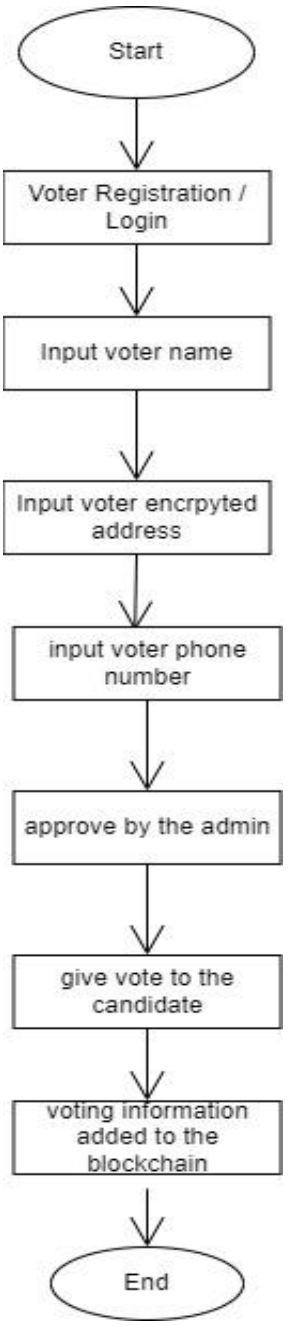

Figure 1:Development Methodology Steps

1.6 REPORT ORGANIZATION

The first chapter of this project consists of a short introduction to the project,problem statement,objectives,project scope and development methodology.Literature review and existing systems are presented in the following chapter.The third chapter contains the system analysis(requirement analysis and feasibility study) while the system design is presented in the fourth chapter.The fifth chapter comprises of the implementation and testing of the system and finally the report is concluded in the last chapter.

CHAPTER 2:BACKGROUND STUDY AND LITERATURE REVIEW

2.1 BACKGROUND STUDY

2.1.1 BLOCKCHAIN

Blockchain is a technology that is rapidly gaining momentum in the era of industry 4.0. With high security and transparency provisions, it is being widely used in supply chain management systems, healthcare, payments, business, IoT, voting systems, etc.As a technology, blockchain is quickly becoming unrivaled. Blockchain technology provides a platform for creating a highly secure, decentralized, anonymized, yet auditable chain of record, used presently in cryptocurrency systems. This same technology could also be used to record and report votes and prevent many types of voter fraud in elections [3].

2.1.2 DECENTRALIZED

Decentralization refers to the transfer of control and decision-making from a centralized entity to a distributed network. Decentralized networks strive to reduce the level of trust that participants must place in one another, and deter their ability to exert authority or control over one another in ways that degrade the functionality of the network. By decentralizing the management of and access to resources in an application, greater and fairer service can be achieved. decentralization provides a trustless environment, Improves data reconciliation, Reduces points of weakness, and Optimized resource distribution [4].

2.1.3 CONSENSUS MECHANISM

Consensus for blockchain is a procedure in which the peers of a Blockchain network reach agreement about the present state of the data in the network. Consensus is the process by which a group of peers or nodes on a network determine which blockchain transactions are valid and which are not. It’s these sets of rules that help to protect networks from malicious behavior and hacking attacks. There are many different types of consensus mechanisms, each with various benefits and drawbacks [5].

2.1.3.1 PROOF OF WORK

PoW is a form of adding new blocks of transactions to a cryptocurrency's blockchain. The work, in this case, is generating a hash that matches the target hash for the current block. The proof-of-work model is a consensus mechanism used to confirm and record cryptocurrency transactions.With proof-of-work cryptocurrencies, each block of transactions has a specific hash. For the block to be confirmed, a crypto miner must generate a target hash that's less than or equal to that of the block. The reason proof of work in cryptocurrency works well is because finding the target hash is difficult but verifying it isn't. The process is difficult enough to prevent the manipulation of transaction records. At the same time, once a target hash is found, it's easy for other miners to check it [6].

2.1.3.2 PROOF OF STAKE

Proof of stake is a type of consensus mechanism used to validate cryptocurrency transactions. With this system, owners of the cryptocurrency can stake their coins, which gives them the right to check new blocks of transactions and add them to the blockchain [7].

In a proof of stake system, staking serves a similar function to proof of work's mining, in that it's the process by which a network participant gets selected to add the latest batch of transactions to the blockchain and earn some crypto in exchange. The exact details vary by project, but in general proof of stake blockchains employ a network of "validators" who contribute or "stake" their own crypto in exchange for a chance of getting to validate a new transaction, update the blockchain, and earn a reward [8].

2.1.3.3 DELEGATED PROOF OF STAKE

Delegated Proof of Stake is a consensus algorithm that assigns voting rights for the approval of transactions to users based on the amount of the corresponding crypto assets held in their accounts. DPoS is an evolution of Proof of Stake , which grants the ability to approve transactions based on the amount of the corresponding currency held [9] .

The DPoS algorithm creates a voting system that is directly dependent on the delegates' reputation. If an elected node misbehaves or does not work efficiently, it will be quickly expelled and replaced by another one [10] .

2.1.3.4 PRACTICAL BYZANTINE FAULT TOLERANCE

Byzantine Fault Tolerance is the feature of a distributed network to reach consensus even when some of the nodes in the network fail to respond or respond with incorrect information. Byzantine fault tolerance can be achieved if the correctly working nodes in the network reach an agreement on their values. There can be a default vote value given to missing messages i.e., we can assume that the message from a particular node is 'faulty' if the message is not received within a certain time limit. Furthermore, we can also assign a default response if the majority of nodes respond with a correct value [11].

2.1.4 ENCRYPTION AND DECRYPTION

Encryption and decryption are among the most common uses of cryptography. Encrypted data is unintelligible; and without the correct decryption key, it cannot be recreated in its original form. For electoral purposes, encryption is often used to obscure the contents of a voter's ballot selections and the contents of a digital ballot box. The voter's encrypted ballot selections may be stored on a voting machine or sent over an insecure channel like the Internet or the telephone network. When casting an electronic vote, the value of the vote will be encrypted using an encryption key produced by the EMB and available at all electronic voting locations. However, only the EMB will have the key that is needed to decrypt encrypted data [12].

2.1.5 HASHING

Hashing refers to the process of generating a fixed-size output from an input of variable size using the mathematical formulas known as hash functions. This technique determines an index or location for the storage of an item in a data structure.The hash function creates a mapping between key and value, this is done through the use of mathematical formulas known as hash functions. The result of the hash function is referred to as a hash value or hash. The hash value is a representation of the original string of characters but usually smaller than the original. There are majorly three components of hashing:Key,Hash Function and Hash Table [13].

2.1.6 DIGITAL SIGNATURE

Digital signatures are mathematical procedures that allow recipients to verify the authenticity of digital messages, documents, or transactions almost with 100% certainty. Digital signatures are created every time a new document, email, or message is signed, so each signature is unique, resistant to tampering, and virtually impossible to counterfeit. A random integer is created and multiplied with a point known as the generator point to form the random component.. The generator point used to create a public key from a Bitcoin private key is the same generator point used for digital signatures [14].

2.1.7 SMART CONTRACT

A smart contract is a self-executing contract whose terms of the agreement between the contract's counterparties are embedded into lines of code. Essentially, a smart contract is a digital version of the standard paper contract that automatically verifies fulfillment and enforces and performs the terms of the contract [15].

Using a smart contract, certain predefined terms and conditions are pre-set in the contract. No voter can vote from a digital identity other than his or her own. The counting is foolproof. Every vote is registered on a blockchain network, and the counting is tallied automatically with no interference from a third party or dependency on a manual process. Each ID is attributed to just one vote. Validation is accomplished by the users on the blockchain network itself. Thus, the voting process can be in a public blockchain, or it could be in a decentralized autonomous organization-based blockchain setup [16].

2.1.8 ETHEREUM

Ethereum is a decentralized blockchain platform that establishes a peer-to-peer network that securely executes and verifies application code, called smart contracts. 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. Transactions are sent from and received by user-created Ethereum accounts. A sender must sign transactions and spend Ether, Ethereum's native cryptocurrency, as a cost of processing transactions on the network [17].

2.2 LITERATURE REVIEW

The paper attempts to use a case study to determine the potential of distributed ledger technologies; such as the election process and its implementation via a block-chain-based framework, which will boost security and reduce the cost of conducting national elections. The technique is to achieve these objectives by using a Go-Ethereum PoA blockchain authorization setup. They have used the algorithm through a process based on identity as a stake, which delivers faster transactions. They use district and boot. The voting data is checked by the majority of the district nodes when any individual elector casts a vote from their compliant smart contract, and any vote they agree on is appended to the blockchain. The advantage is Elections can be used as a Blockchain part of Smart Contract, using developer friendly Framework (Go-Ethereum), Centralized consensus. The disadvantage is limited to 5000 votes/second. Can use some better blockchain frameworks for increasing transactions per second [18].

The paper focuses on systematic mapping study of all peer-reviewed technology-oriented research in smart contracts. The interest is twofold, namely to provide a survey of the scientific literature and to identify academic research trends and uptake. The paper focuses on peer-reviewed scientific publications, in order to identify how academic researchers have taken up smart contract technologies and established scientific outputs. We obtained all research papers from the main scientific databases, and using the systematic mapping method arrived at 188 relevant papers. The papers are into six categories, namely, security, privacy, software engineering, application, performance & scalability and other smart contract related topics. It is found that the majority of the papers fall into the applications and software engineering (21%) categories. Compared to our 2017 survey, we observe that the number of relevant articles has increased about eightfold and shifted considerably towards applications of smart contracts [19].

This literature examines the IoT experiencing explosive growth and has gained extensive attention from academia and industry in recent years. However, most of the existing IoT infrastructures are centralized, which may cause the issues of unscalability and single-point-of-failure. Consequently, decentralized IoT has been proposed by taking advantage of the emerging technology called blockchain. Voting systems are widely adopted in IoT, for example a leader election in wireless sensor networks. Self-tallying voting systems are alternatives to unsuitable, traditional centralized voting systems in

decentralized IoT. Unfortunately, self-tallying voting systems inherently suffer from fairness issues, such as adaptive and abortive issues caused by malicious voters. To address these issues, in this paper, we introduce a framework of the self-tallying voting system in decentralized IoT based on blockchain. The paper proposes a concrete construction and proves that the proposed system satisfies all the security requirements, including fairness, dispute-freeness and maximal ballot secrecy. We simulate the algorithms on a laptop, an Android phone and a Raspberry Pi to test the time consumption and evaluate the gas cost of each algorithm in a private blockchain as well. The implementation results demonstrate the practicability of our system [20].

This paper suggests a framework by using effective hashing techniques to ensure the security of the data. The concept of block creation and block sealing is introduced in this paper. The introduction of a block sealing concept helps in making the blockchain adjustable to meet the need of the polling process. The use of consortium blockchain is suggested, which ensures that the blockchain is owned by a governing body (e.g., election commission), and no unauthorized access can be made from outside. The framework proposed in this paper discusses the effectiveness of the polling process, hashing algorithms' utility, block creation and sealing, data accumulation, and result declaration by using the adjustable blockchain method. This paper claims to apprehend the security and data management challenges in blockchain and provides an improved manifestation of the electronic voting process [21].

The paper proposes the first self-tallying decentralized e-voting protocol for a ranked-choice voting system based on Borda count. The protocol does not need any trusted setup or tallying authority to compute the tally. The voters interact through a publicly accessible bulletin board for executing the protocol in a way that is publicly verifiable. The main protocol consists of two rounds. In the first round, the voters publish their public keys, and in the second round they publish their randomized ballots. All voters provide Non-interactive Zero-Knowledge proofs to show that they have been following the protocol specification honestly without revealing their secret votes [22].

This literature proposes a blockchain-based voting system which achieves all the properties expected from secure elections without requiring too much from the voter. Coercion resistance and receipt-freeness are ensured by means of a randomizer token – a tamper-resistance source of randomness which acts as a black box in constructing the

ballot for the user. Universal verifiability is ensured by the append-only structure of the blockchain, thus minimizing the trust placed in election authorities. Additionally, tallying the votes has linear overhead, hence our system is scalable and practical for large scale elections [23].

This literature proposes an improved FOO e-voting protocol using blockchain, which tries to address the limitation or weakness in existing systems. The traditional trusted third party is replaced by smart contract; specifically, our scheme is deployed using hyperledger fabric. The implementation is enforced by the consensus mechanism, which ensures the security of the blockchain. Through the analysis, the proposed scheme is proved to satisfy the necessary requirements for an e-voting protocol; meanwhile the trust assumption is reduced significantly. Therefore, the proposed protocol is more versatile and practical [24].

The paper analyzes how decentralization affects consensus effectiveness, and how the quintessential features of blockchain reshape industrial organization and the landscape of competition. Smart contracts can mitigate informational asymmetry and improve welfare and consumer surplus through enhanced entry and competition, yet the irreducible distribution of information during consensus generation may encourage greater collusion. In general, blockchains can sustain market equilibria with a wider range of economic outcomes. It further discusses antitrust policy implications targeted to blockchain applications, such as separating consensus record-keepers from users [25].

This paper focuses on smart contract technology reshaping conventional industry and business processes. Being embedded in blockchains, smart contracts enable the contractual terms of an agreement to be enforced automatically without the intervention of a trusted third party. As a result, smart contracts can cut down administration and save services costs, improve the efficiency of business processes and reduce the risks. Although smart contracts are promising to drive the new wave of innovation in business processes, there are a number of challenges to be tackled. This paper presents a survey on smart contracts. It first introduced blockchains and smart contracts [26].

With the rapid development of Information Technology industries, data or information security has become one of the critical issues. Nowadays, Blockchain technology is widely used for improving data security. It is a tool for the individual and organization to interchange the digital asset without the intervention of a trusted third party i.e. a central

administrator. This technology has given the ability to create digital tokens for representing assets, innovation and likely reshaping the scenery of entrepreneurship. Blockchain has several key properties, such as decentralization, immutability and transparency without using a trusted third party. It can be used in several fields, such as healthcare, digital voting, Internet of Things and many more. This study aims to discuss the fundamentals of Blockchain. In this paper, the technology or working procedure of Blockchain including many applications in several fields are discussed. Finally, future work directions and open research challenges in the domain of Blockchain have been also discussed in detail [27].

Since the inception of Bitcoin, cryptocurrencies and the underlying blockchain technology have attracted an increasing interest from both academia and industry. Among various core components, consensus protocol is the defining technology behind the security and performance of blockchain. From incremental modifications of Nakamoto consensus protocol to innovative alternative consensus mechanisms, many consensus protocols have been proposed to improve the performance of the blockchain network itself or to accommodate other specific application needs. In this survey, we present a comprehensive review and analysis on the state-of-the-art blockchain consensus protocols. To facilitate the discussion of our analysis, we first introduce the key definitions and relevant results in the classic theory of fault tolerance which help to lay the foundation for further discussion. We identify five core components of a blockchain consensus protocol, namely, block proposal, block validation, information propagation, block finalization, and incentive mechanism. A wide spectrum of blockchain consensus protocols are then carefully reviewed accompanied by algorithmic abstractions and vulnerability analyses. The surveyed consensus protocols are analyzed using the five-component framework and compared with respect to different performance metrics. These analyses and comparisons provide us new insights in the fundamental differences of various proposals in terms of their suitable application scenarios, key assumptions, expected fault tolerance, scalability, drawbacks and trade-offs [28].

Blockchain technology brings innovation to various industries. Ethereum is currently the second blockchain platform by market capitalization, it’s also the largest smart contract blockchain platform. Smart contracts can simplify and accelerate the development of various applications, but they also bring some problems. For example, smart contracts are used to commit fraud, vulnerability contracts are deliberately developed to undermine

fairness, and there are numerous duplicative contracts that waste performance with no actual purpose. In this paper, we propose a transaction-based classification and detection approach for Ethereum smart contracts to address these issues.The extensive experimental results show that our approach can distinguish different types of contracts and can be applied to anomaly detection and malicious contract identification with satisfactory precision, recall, and f1-score [29].

Blockchain technology is claimed to be and perceived as one of the revolutionary technologies that will have an enormous impact on our lives in the forthcoming years and decades. The legal questions surrounding blockchain appear to be among the most controversial issues surrounding this novel technology, which create uncertainties as to the scope and speed of its eventual adoption. Is it legal to use blockchain technology? Does or should any governmental authority or court take a record stored in blockchain into consideration in their decisions? Is blockchain reliable? This paper will focus primarily on the possible opportunities that blockchain may offer with respect to the future of IP law and discuss its potential impact on the registration, management and enforcement of intellectual property rights. The paper concludes by providing some suggestions to pave the way for the advancement of blockchain technology and to increase the number of people that this technology reaches, as well as its successful integration into the various services and registration/transaction channels that we use today [30].

Blockchain, as a potentially revolutionary technology, has been used in cryptocurrency to record transactions chronologically among multiple parties. Due to the fast development of the blockchain and its de-centralization, blockchain technology has been applied in broader scenarios, such as smart factories, supply chains, and smart cities. Consensus protocol plays a vital role in the blockchain, which addresses the issue of reaching consensus on transaction results among involved participants. Nevertheless, with the complexity of the network environment and growing number of network users, the advance of blockchain is gradually restricted by the efficiency, security and reliability of consensus protocols. In this paper, we propose a delegated randomization Byzantine fault tolerance consensus protocol named DRBFT based on Practical Byzantine Fault Tolerance to enhance the efficiency and reliability of the consensus procedure. Specifically, a random selection algorithm called RS is developed to cooperate with the voting mechanism, which can effectively reduce the number of nodes participating in the

consensus process. Our proposed scheme is characterized by the unpredictability, randomicity and impartiality, which accelerate the system to reach consensus on the premise of ensuring the system activity. Furthermore, the feasibility of our proposed scheme is also proved by both theoretical analysis and experimental evaluations [31].

In recent decades, several countries have faced political tensions due to citizens' perceptions that their elections are fraudulent; some electors have even chosen not to vote because they believe that the results may be falsified. Thus, electoral fraud is a major issue. E-governance and e-voting are now being used in many countries, some of which are investigating blockchain solutions. The aim of this study is to investigate the potential contributions of blockchain technology to peace on a worldwide level by securing voting systems. Unfortunately, this technology is complex and could potentially generate conflict between actors in elections. Taking an exploratory approach, the authors chose a qualitative method to address this specific topic. Election observers and blockchain experts were interviewed to identify the technology's strengths and weaknesses [32].

The hopes that e-voting would increase voter turnout have not really materialized; any turnout increases in those countries where Internet voting has been introduced have been negligible. But the effect of one mechanism by which this mode of voting might influence turnout is still largely unknown, namely its potential to keep voters voting at higher rates than paper voting, and thus if not reversing, then at least putting a halt to further decline in turnout. This article examines the degree to which people who vote on the Internet once carry on doing so, thereby testing the hypothesis that e-voting is more habit forming than paper voting.Survey data are analyzed from five consecutive and e-enabled nationwide elections in Estonia between 2009 and 2015. The results suggest e-voting to be very “sticky”; a first time e-voter is very likely to stay e-voting in subsequent elections at consistently higher rates than a typical paper voter is to stay paper voting, or a nonvoter to remain a nonvoter. The results call for a re-examination of the association between Internet voting and turnout [33].

The articles indicate that blockchain-supported voting systems may provide different solutions than traditional e-voting.The main prevailing issues were classified into the five following categories: general, integrity, coin-based, privacy and consensus. As a result of this research, it was determined that blockchain systems can provide solutions to certain problems that prevail in current election systems. On the other hand, privacy protection

and transaction speed are most frequently emphasized problems in blockchain applications. Security of remote participation and scalability should be improved for sustainable blockchain based e-voting. It was concluded that frameworks needed enhancements in order to be used in voting systems due to these reservations [34].

Electronic voting systems face many challenges, including authentication, privacy, data integrity, transparency and verifiability. However, developed over 10 years ago blockchain technology provides an out-of-the-box solution for many of those challenges. Despite that, some issues are still to be solved, like remote authentication, anonymity and end-to-end verifiability. The main goal of this study is to highlight the current trends in this research and its eventual shortcomings. This was accomplished by conducting a SLR which resulted in selecting 35 publications. The performed SLR allowed to define trends in utilized blockchain technologies, intended scenarios, testing methods, main benefits and challenges faced by various systems and the most used cryptographic solutions [35].

The blockchain is a technology which accumulates and compiles data into a chain of multiple blocks. Many blockchain researchers are adopting it in multiple areas. However, there are still lacking bibliometric reports exhibiting the exploration of an in-depth research pattern in blockchain. This paper aims to address that gap by analyzing the widespread blockchain research activities conducted thus far. This study analyzed the Scopus database by using bibliometric analysis in a pool of more than 1000 articles that were published between 2013 and 2018. In particular, this paper discusses various aspects of blockchain research conducted by researchers globally. This study also focuses on the utilization of blockchain and its consensus algorithms.This study also highlighted the utilization and consensus of the algorithm in blockchain research [36].

CHAPTER 3: SYSTEM ANALYSIS

3.1 SYSTEM ANALYSIS

3.1.1. REQUIREMENT ANALYSIS

i) Functional Requirements

Functional requirements are product features or functions that developers must implement to enable users to accomplish their tasks. So, it's important to make them clear both for the development team and the stakeholders. [37].

Some of the functional requirements for the proposed system are:

- Perform voting transactions and validate it. Display the real time votes of different candidates.
- Allow the voting transaction to be completed through the decentralized system.
- Allow for the votes to be audited and provide a way for voters to verify that their vote was counted correctly.

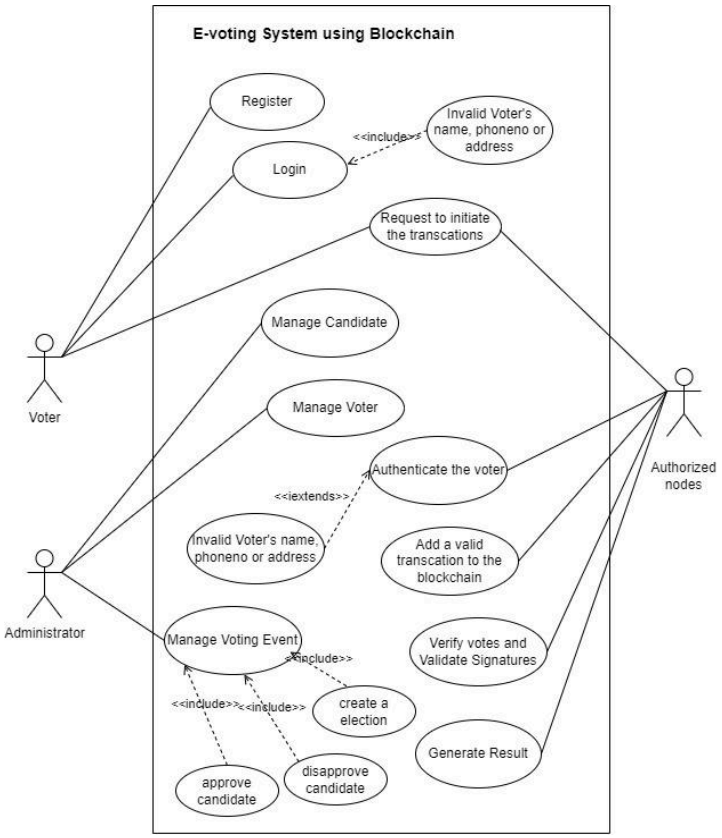

Figure 2: Use Case Diagram of Proposed Model

A use case diagram is a visual representation of the interactions between the users (actors) and the system (use cases) in a specific scenario. Figure above shows the Use Case diagram of the system. The system consists of two types of actors: primary and secondary.

Voter: The voter is the primary actor whereas the authorized nodes are the secondary actors. The vote register and then login to the system and initiate the transaction by giving a vote to the candidate.

Administrator: The administrator is responsible for managing the candidate, voter and overall voting event

Validator: The authorized node must be able to validate transactions and add transactions to their blockchain. The authorized nodes also authenticate the voter and verify votes by validating signatures.

ii) NON-FUNCTIONAL REQUIREMENTS

Non-functional requirements are a set of specifications that describe the system's operation capabilities and constraints and attempt to improve its functionality. These are basically the requirements that outline how well it will operate including things like speed, security, privacy,performance,transparency,availability,accessibility,maintainability etc [38].

Some of them are explained below:

- **Security:** The system should be secure and protect against potential attacks, such as hacking or tampering, to ensure the integrity of the voting process.
- **Privacy:** The system should protect the privacy of voters and keep their personal information and voting choices confidential.
- **Transparency:** It is a key aspect of a decentralized voting system, as it ensures that the voting process is secure, fair and accurate.
- **Availability:** The app should be available to users at all times to ensure that they can vote when they want to.
- **Performance:** The app should have a fast and responsive user interface, and be able to handle a large number of users and transactions without experiencing delays or downtime.

3.1.2. FEASIBILITY ANALYSIS

A feasibility study is an assessment that determines the likelihood of a proposed project being successful, such as a new product line or technical system. The study analyzes the project's relevant factors, such as technical, economic and legal considerations, to assess whether the project is worth an investment [39].

Below are the findings of the feasibility study for this project. The proposed system is a system demonstrating through which voters can vote their candidates. When conducting a feasibility analysis for a decentralized voting app, some of the key considerations include:

i. Technical feasibility

The project is technically feasible, since the necessary technology, such as blockchain, smart contracts, and cryptography, are available and can be implemented easily.

ii. Economic feasibility

The project is economically feasible, since the costs of development, testing, maintenance, and operation are done with the help of free and open source software and tools that are easily available on the internet.

iii. Operational feasibility

The project is operationally feasible, since it can be integrated into existing systems and infrastructure, and that the necessary resources required for this project are low and can be created with minimum manpower.

iv. Schedule feasibility

The project is feasible with regards to the schedule as well. The Gantt chart depicting timeline for the project is given below:

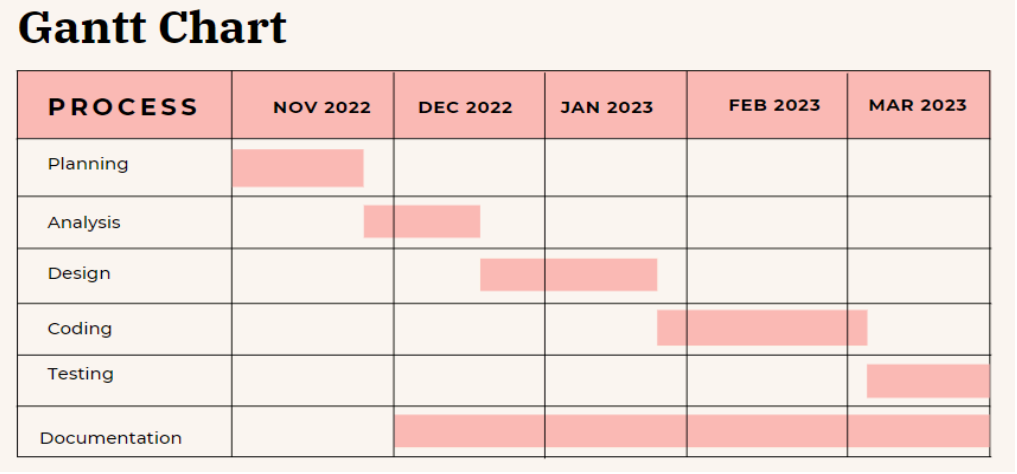

Figure 3: Gantt chart

3.1.3 ANALYSIS

3.1.3.1 OBJECT MODELING

The object Model sees an information system as a set of objects and classes. The reader will get an in-depth understanding of the object model and its elements. The basic factors of an object model are classes and objects.An object is a physical component in the object-oriented domain which may be tangible such as a person, car, or intangible such as a project. A class is a representation of objects. It represents a group of objects that have similar properties and exhibit an expected behavior [40].

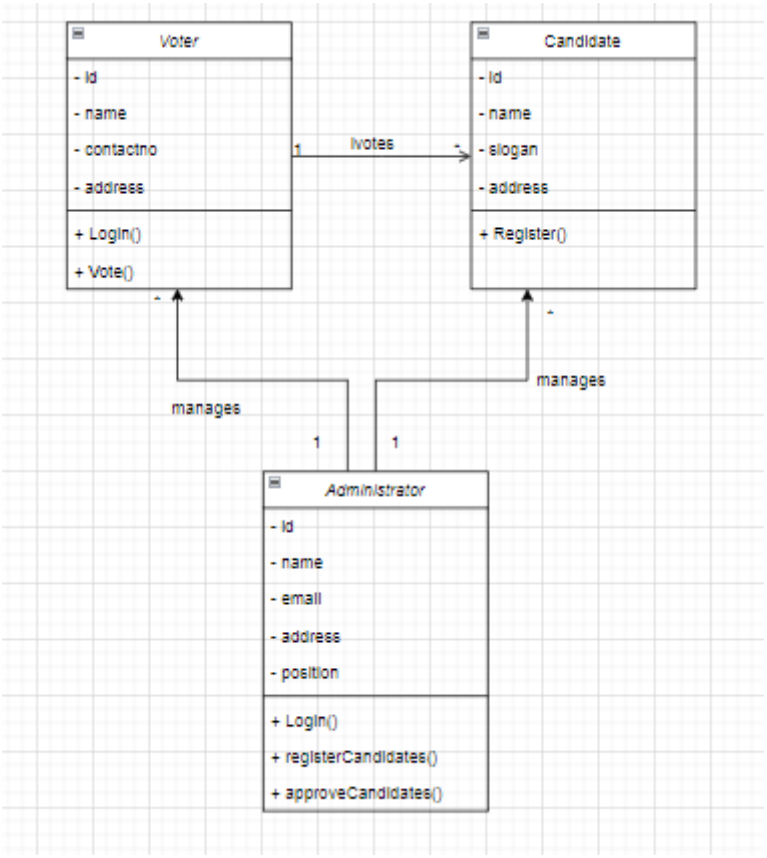

Figure 4: Object modeling using Class Diagram

3.1.3.2 DYNAMIC MODELING

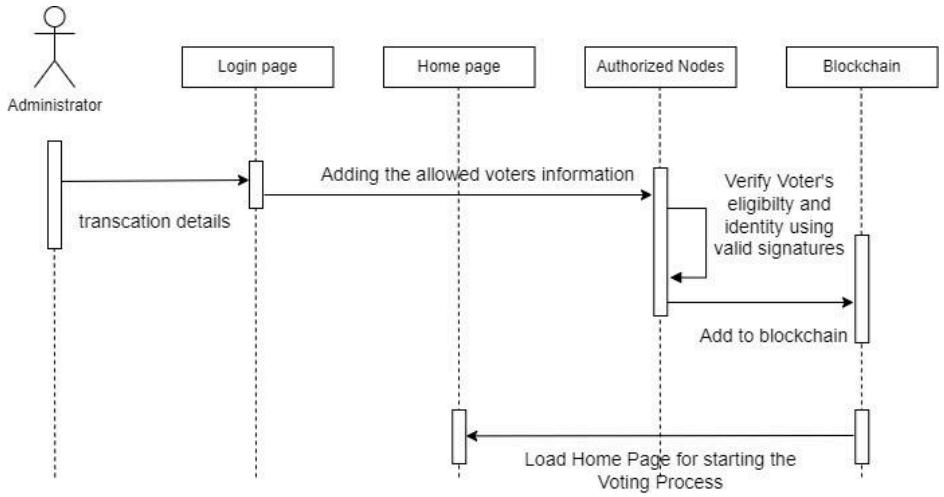

Figure 5: Dynamic modeling of voter registration using sequence diagram

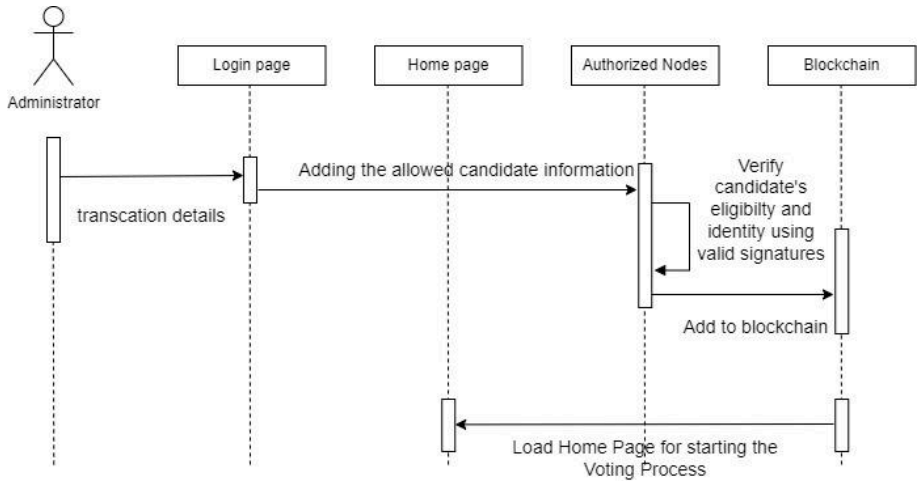

Figure 6: Dynamic modeling of candidate registration using sequence diagram

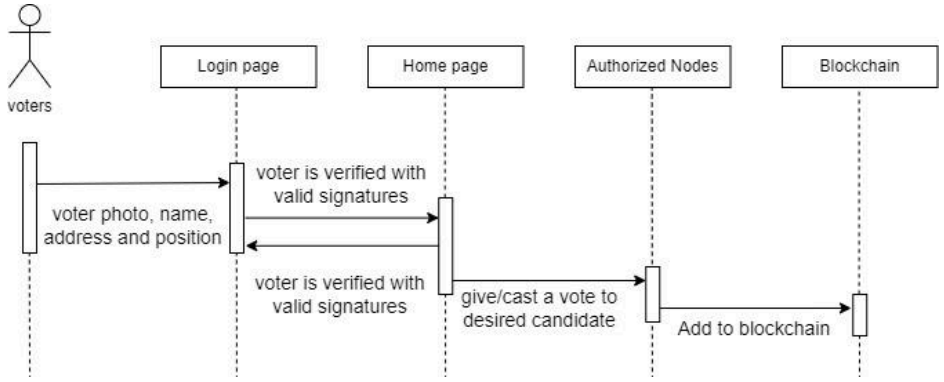

Figure 7: Dynamic modeling of giving vote using sequence diagram

3.1.3.3 PROCESS MODELING

Process modeling is the graphical representation of business processes or workflows. A process model allows visualization of business processes so organizations can better understand their internal business procedures so that they can be managed and made more efficient [41].

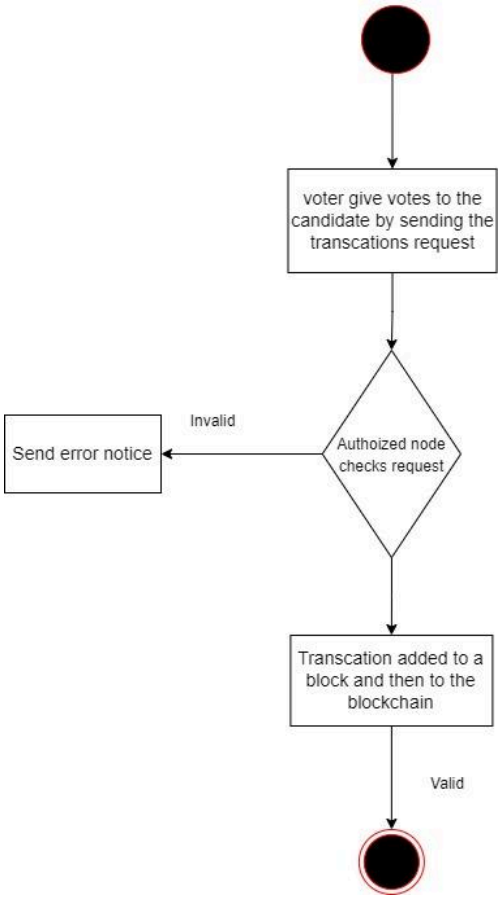

Figure 8: Process modeling of giving vote using Activity Diagram

A voter casts a vote for a preferred candidate, then they log in to their account and use their private keys to sign the transaction. This is sent to the authorized nodes for verification. The signatures are verified and the transaction is validated by the authorized nodes using a consensus algorithm. If it is deemed valid, the transaction is added to the blockchain, otherwise it is discarded.

CHAPTER 4: SYSTEM DESIGN

4.1 DESIGN

4.1.1 COMPONENT DIAGRAM

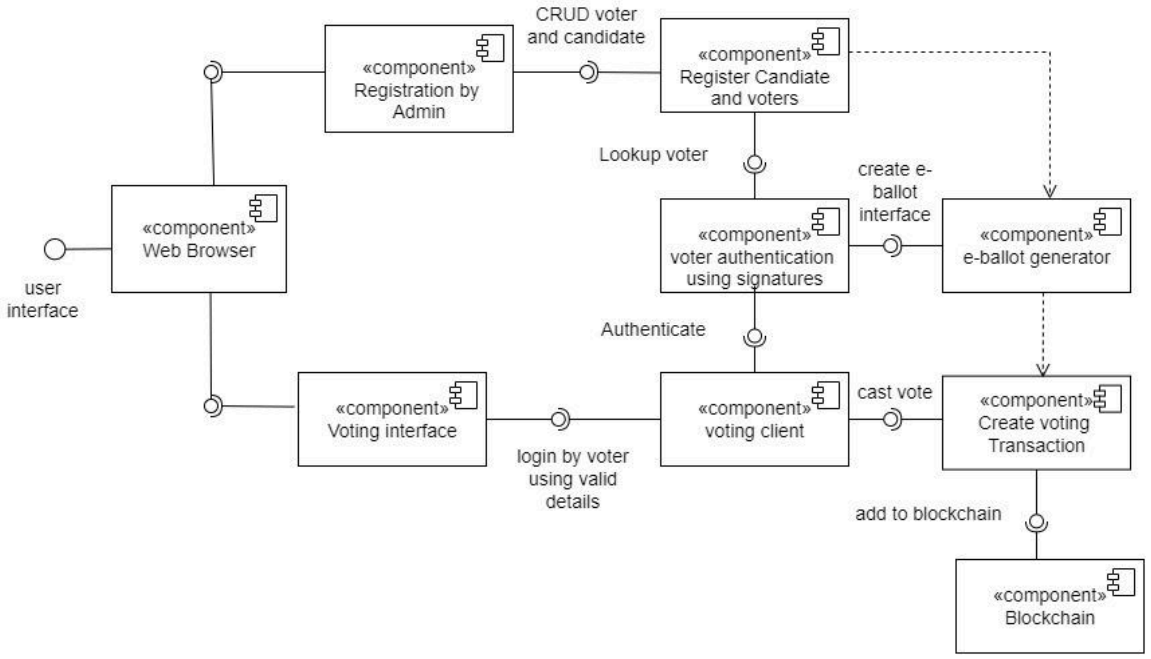

Figure 9: Component Diagram

The web browser will contain the user interface where admin and voter can interact with the browser. The admin can register the candidate and voters are stored on the blockchain. The voter interacts with the browser and login by using valid details like name, address(public key). Authentication of voters is done using valid signatures. After successful login, the voter casts a vote to the candidate from the e-ballot interface and the vote is added to the blockchain. The administrator and voter have different privileges.

4.1.2 DEPLOYMENT DIAGRAM

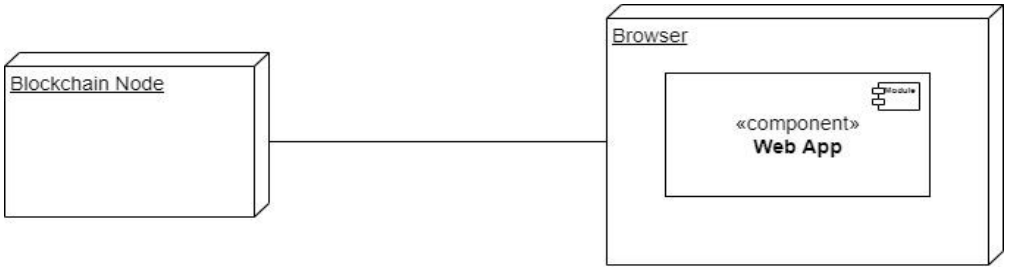

Figure 10: Deployment Diagram

4.2 ALGORITHM DETAILS

4.2.1 PRACTICAL BYZANTINE FAULT TOLERANCE

Consensus models are a primary component of distributed blockchain systems and definitely one of the most important to their functionality. They are the backbone for users to be able to interact with each other in a trustless manner, and their correct implementation into cryptocurrency platforms has created a novel variety of networks with extraordinary potential.In the context of distributed systems, Byzantine Fault Tolerance is the ability of a distributed computer network to function as desired and correctly reach a sufficient consensus despite malicious components (nodes) of the system failing or propagating incorrect information to other peers.The objective is to defend against catastrophic system failures by mitigating the influence these malicious nodes have on the correct function of the network and the right consensus that is reached by the honest nodes in the system PBFT is one of these optimizations and was introduced by Miguel Castro and Barbara Liskov in an academic paper in 1999 titled “Practical Byzantine Fault Tolerance”.It aims to improve upon original BFT consensus mechanisms and has been implemented and enhanced in several modern distributed computer system [42].

4.2.1.1 ADVANTAGES OF PRACTICAL BYZANTINE FAULT TOLERANCE

PBFT offers a few merits compared to other consensus algorithms, particularly PoW. These advantages include the following:

- Transaction finality: The PBFT model provides transaction finality without requiring confirmations. If the nodes agree on the validity of a proposed block, then the transactions in that block are final. This differs from PoW, where each node individually verifies a transaction before a mining node adds it to the chain. Bitcoin confirmations, for example, take around 10 to 60 minutes, depending on the number of nodes confirming the block.
- Low energy use: Unlike PoW, PBFT isn’t energy intensive since it doesn’t require nodes to solve complex mathematical problems. Bitcoin miners require electricity to expend proof-of-work, which can result in high electricity consumption.
- Even payouts: In PBFT, all nodes implement the client request, which means they all receive a reward [43].

4.2.1.2 LIMITATIONS OF PRACTICAL BYZANTINE FAULT TOLERANCE

The PBFT consensus model works efficiently only when the number of nodes in the distributed network is small due to the high communication overhead that increases exponentially with every extra node in the network.

Sybil attacks: The PBFT mechanisms are susceptible to Sybil attacks, where one entity (party) controls many identities. As the number of nodes in the network increases, Sybil attacks become increasingly difficult to carry out. But as PBFT mechanisms have scalability issues, too, the PBFT mechanism is used in combination with another mechanism (s).

Scaling: PBFT does not scale well because of its communication (with all the other nodes at every step) overhead. As the number of nodes in the network increases (increases as O(n^k), where n is the messages and k is the number of nodes), so does the time taken to respond to the request [44].

4.2.2 PUBLIC KEY INFRASTRUCTURE

Public key infrastructure is a catch-all term for everything used to establish and manage public key encryption, one of the most common forms of internet encryption. It is baked into every web browser in use today to secure traffic across the public internet, but organizations can also deploy it to secure their internal communications and access to connected devices.The most crucial concept involved in PKI is, as its name implies, the public cryptographic keys that are at its core. These keys not only are part of the encryption process, but they help authenticate the identity of the communicating parties or devices. The most important concepts to understand to grasp how PKI works are keys and certificates. A key, as already noted, is a long string of bits,a number, in other words,that's used to encrypt data. For instance, if you used the ancient and simple Caesar cipher with a cryptographic key of 3, that would mean that every letter in your message is replaced by one three letters later in the alphabet,A becomes D, B becomes E, and so forth. To decode its message, your recipient would need to know not only that you were using the Caesar cipher but that your key was 3. Obviously the mathematics behind modern encryption is much more complicated than this. One of the ways it's different gets around a somewhat obvious problem with the Caesar cipher: you have to somehow let your recipient know the key used to encode the encrypted message. PKI gets its name

because each participant in a secured communications channel has two keys. There's a public key, which you can tell to anyone who asks and is used to encode a message sent to you, and a private key, which you keep secret and use to decrypt the message when you receive it. The two keys are related by a complex mathematical formula that would be difficult to derive from brute force. If you want to get into the weeds on this form of encryption, known as asymmetrical [45].

Application design has changed dramatically since PKI emerged. With cloud and mobility, employees are no longer tied to their desks when they access computer services. They are in a remote office, at home, traveling, or visiting clients. In addition, cloud computing moves information processing out of the enterprise data center and into vendor’s premises. Rather than one homogenous block of code processed in sequential fashion on central systems, information is now divided up and sent to numerous servers residing in multiple locations.Blockchain has an open, transparent, secure architecture. Anyone on a blockchain can read all of its contents. This feature eliminates the potential problems stemming from relying on a third party CA’s actions. Companies no longer need to put their trust in CAs that may be duplicitous or error-prone in creating public and private keys. Everything that happens on a blockchain is available to anyone using it. So if a CA issues keys in someone else’s name, that information is seen by everyone on the chain. Information is time stamped, and a record is created each time an update occurs. Consequently, it is clear who did what when. Altering the source code becomes impossible. A hacker needs to change every item in the blockchain rather than just one record. Also, the metadata in its database is read only, which means that it is impossible to manipulate independently. The solution protects information in a secure distributed fashion and is more in tune with current needs than traditional PKI systems [46].

CHAPTER 5: IMPLEMENTATION AND TESTING

5.1 IMPLEMENTATION

5.1.1 TOOLS USED

The tools used to create this we based visual demonstration of the proposed system are as follows:

5.1.1.1 PROGRAMMING LANGUAGE

Javascript and ReactJS

JavaScript is a text-based programming language used both on the client-side and server-side that allows you to make web pages interactive Incorporating JavaScript improves the user experience of the web page by converting it from a static page into an interactive one. To recap, JavaScript adds behavior to web pages [47].

ReactJS is an open-source JavaScript library created by Facebook's Jordan Walke to make user interfaces for both web and mobile systems. ReactJS's primary goal is to make user interfaces that make apps load faster. It also uses virtual DOM (JavaScript object), which also helps the ReactJS app run faster [48].

ReactJS was the programming language used in our project in order to build the frontend user interface of the application. It was used to build various components of the user interface.

5.1.1.2 Truffle and Ganache

Truffle is a world-class development environment, testing framework and asset pipeline for blockchains using the Ethereum Virtual Machine, aiming to make life as a developer easier.Truffle is widely considered the most popular tool for blockchain application development with over 1.5 million lifetime downloads. Automated contract testing for rapid development.Scriptable, extensible deployment & migrations framework.Network management for deploying to any number of public & private networks.Package management with EthPM & NPM, using the ERC190 standard.Interactive console for direct contract communication.Configurable build pipeline with support for tight integration.External script runner that executes scripts within a Truffle environment [49].

Ganache is the future of blockchain. Ganache is a JavaScript library for building, testing, and deploying blockchain applications. It makes it easy to create a blockchain application by providing a set of APIs and tools for building smart contracts, issuing and managing transactions, and more. It provides a platform for developing decentralized applications, which can be used to power a wide range of applications, including but not limited to financial services, supply chains, and more. It is a client-side platform that allows you to create and manage your own custom blockchain applications. Ganache is a blockchain platform that allows organizations to create their own private blockchains. Ganache provides a secure environment for developers to build, test and deploy decentralized applications [50].

Truffle was used in our project to write and test smart contracts in a local environment. Ganache is a part of Truffle Suite which was used in our project to test the smart contracts, execute commands and inspect state while controlling how the chain operates in a local environment before deploying to the main ethereum network.

5.1.1.3 VISUAL STUDIO CODE

Visual Studio Code is a free, lightweight but powerful source code editor that runs on your desktop and on the web and is available for Windows, macOS, Linux, and Raspberry Pi OS. It comes with built-in support for JavaScript, TypeScript, and Node.js and has a rich ecosystem of extensions for other programming languages. Aside from the whole idea of being lightweight and starting quickly, Visual Studio Code has IntelliSense code completion for variables, methods, and imported modules; graphical debugging; linting, multi-cursor editing, parameter hints, and other powerful editing features; snazzy code navigation and refactoring; and built-in source code control including Git support. Much of this was adapted from Visual Studio technology [51].

Visual studio code was extensively used in our project. It was the main code editor used in our project. It was used to write smart contracts using solidity programming language. It was also used to write ReactJS code for developing user interfaces. Truffle and ganache were integrated with vscode to write and test smart contracts directly in the code editor. VScode was also for debugging our smart contracts and reactjs code.

5.1.1.4 GIT AND GITHUB

Git is the most commonly used version control system. Git tracks the changes you make to files, so you have a record of what has been done, and you can revert to specific versions should you ever need to. Git also makes collaboration easier, allowing changes by multiple people to all be merged into one source. Your files and their history are stored on your computer [52].

GitHub is a web-based interface that uses Git, the open source version control software that lets multiple people make separate changes to web pages at the same time. GitHub allows multiple developers to work on a single project at the same time, reduces the risk of duplicative or conflicting work, and can help decrease production time [53].

Git was used to manage code changes in our application. It was used to track changes to our code and collaborate with team members. Github was used to create pull requests and review code changes between team members.

5.1.1.5 MICROSOFT WORD AND EXCEL

MS Word is a processing software which is used for writing letters, essays, notes, etc. Whereas, MS Excel is a spreadsheet software where a large amount of data or information can be saved in a systematic tabular manner in numerical and alphabetical values[54] .

Microsoft Word was utilized throughout the entire project to document the needs, procedures, and workflow. Microsoft Excel was also utilized to make the Gantt chart and keep track of the project's deadlines.

5.1.1.6 GOOGLE DRIVE

Google Drive was extensively used in our project. It was used to collaborate on documents, spreadsheets and presentations. It was used for sharing the files and folders between the team members. With its cloud-based infrastructure, we were able to securely access and collaborate on our files from any location with an internet connection, which was particularly advantageous for team members who worked remotely or were frequently on the move.

5.1.1.7 Draw.io

Draw.io was a critical tool in our project because it provided an easy-to-use and versatile platform for creating diagrams, flowcharts, and other visual aids. Its cloud-based platform enabled us to work on diagrams in real time and easily share them with other team members, making it an invaluable tool for communication and project planning.

5.1.2 IMPLEMENTATION DETAILS OF MODULES

5.1.2.1 Code Snippet for Creating Election

```
contract Election
    {
    address public admin;
    uint256 candidateCount;
    uint256 voterCount;
    bool start;
    bool end;
    constructor() public
    {
// Initializing default values
        admin = msg.sender;
        candidateCount = 0;
        voterCount = 0;
        start = false;
        end = false;}
```

5.1.2.2 Code Snippet for Creating Candidate Structure

```
struct Candidate
    {
        uint256 candidateId;
        string header;
        string slogan;
        uint256 voteCount;
    }
```

5.1.2.3 Code snippet for adding new candidate function

```
function addCandidate(string memory _header, string memory _slogan)
        public
        // Only admin can add
        onlyAdmin
    {Candidate memory newCandidate =Candidate({
                candidateId: candidateCount,
                header: _header,
                slogan: _slogan,
                voteCount: 0
            });
        candidateDetails[candidateCount] = newCandidate;
        candidateCount += 1;
      }
```

5.1.2.4 Code Snippet for Creating Voter Structure

```
struct Voter {
        address voterAddress;
        string name;
        string phone;
        bool isVerified;
        bool hasVoted;
        bool isRegistered;
    }
```

5.1.2.5 Code Snippet for Creating Voter function

```
function registerAsVoter(string memory _name, string memory _phone)
public {Voter memory newVoter =Voter({
                voterAddress: msg.sender,
                name: _name,
                phone: _phone,
                hasVoted: false,
                isVerified: false,
```

```
                isRegistered: true
            });
        voterDetails[msg.sender] = newVoter;
        voters.push(msg.sender);
        voterCount += 1;}
```

5.1.2.6 Code Snippet for Modeling a election details

```
Struct ElectionDetails {
        string adminName;
        string adminEmail;
        string adminTitle;
        string electionTitle;
        string organizationTitle;
    }
    ElectionDetails electionDetails;
    function setElectionDetails(
        string memory _adminName,
        string memory _adminEmail,
        string memory _adminTitle,
        string memory _electionTitle,
        string memory _organizationTitle
    )public
        // Only admin can add
        onlyAdmin
    {
        electionDetails = ElectionDetails(
            _adminName,
            _adminEmail,
            _adminTitle,
            _electionTitle,
            _organizationTitle
        );
        start = true;
```

```
        end = false;
    }
```

5.1.2.7 Code Snippet for Connecting a Metmask wallet

```
const getWeb3 = () =>
  new Promise((resolve, reject) =>
    window.addEventListener('load', async () => {
      if (window.ethereum) {
        const web3 = new Web3(window.ethereum);
        try {
          // Request account access if needed
          // await window.ethereum.enable();
             await window.ethereum.request({ method: 'eth_requestAccounts'
});
          resolve(web3);
        } catch (error) {
          reject(error);}
      }
      else if (window.web3) {
        // Use MetaMask's provider.
        const web3 = window.web3;
        console.log('Injected web3 detected.');
        resolve(web3);
      }
      else {
        const provider = new Web3.providers.HttpProvider(
          'http://127.0.0.1:8545'
        );
        const web3 = new Web3(provider);
        console.log('No web3 instance injected, using Local web3.');
        resolve(web3);
      }});
  });
```

5.1.2.8 Code Snippet for Truffle Configuration

```
const path = require('path');
module.exports = {
 contracts_build_directory:path.join(__dirname, 'client/src/contracts'),
  networks: {
    development: {
      network_id: '*',host: '127.0.0.1',
      // port: 7545, // for ganache gui
      port: 8545, // for ganache-cli
      gas: 6721975,gasPrice: 20000000000,
    },
  },
};
```

5.2 TESTING

5.2.1 Unit Testing

Unit testing is a software testing technique in which individual units or components of a software application are tested in isolation from the rest of the application to ensure that they function as intended. In unit testing, each unit is tested separately to verify that it behaves as expected and produces the correct output for a given input.

5.2.1.1 Test Case 1

Test Objectives: Test for valid login(Admin Side)

Test Objectives:

Test Data:

Password:admin11aa

Expected Output:

Successful login and redirect to homepage

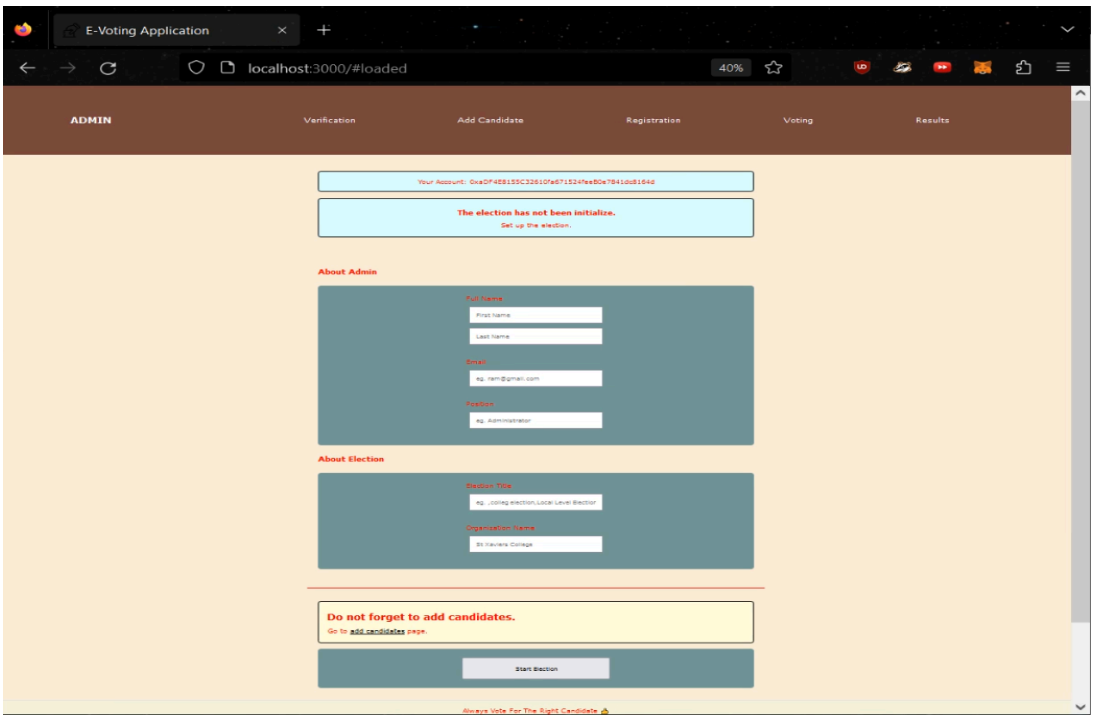

Figure 11: Test Case 1

5.2.1.2 Test Case 2

Test Objectives: Test for invalid login (Admin Side)

Test Data:

Password:efaeaweaa

Expected Output:

Unsuccessful login and popup box with incorrect password message from metamask

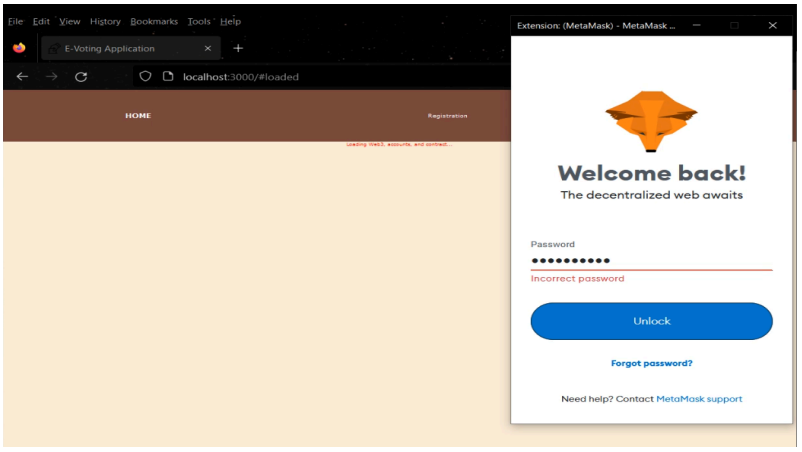

Figure 12: Test Case 2

5.2.1.3 Test Case 3

Test Objectives: Test for adding candidates

Expected Output:

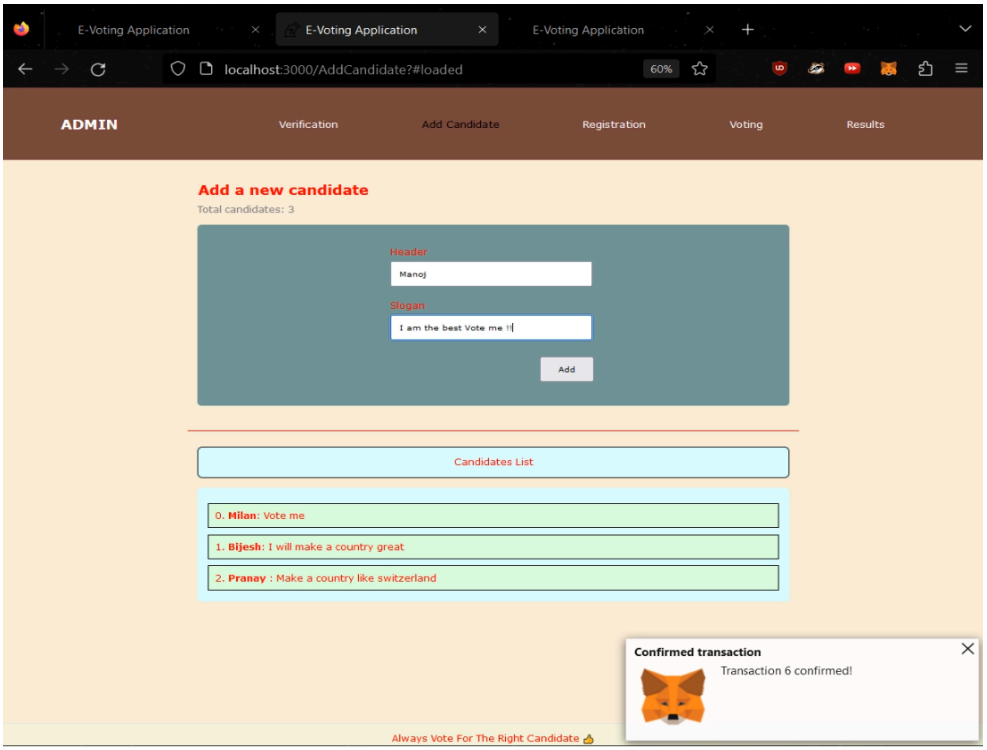

Figure 13: Test Case 3

5.2.1.4 Test Case 4

Test Objective: Test case for registration for voting

Expected Output:

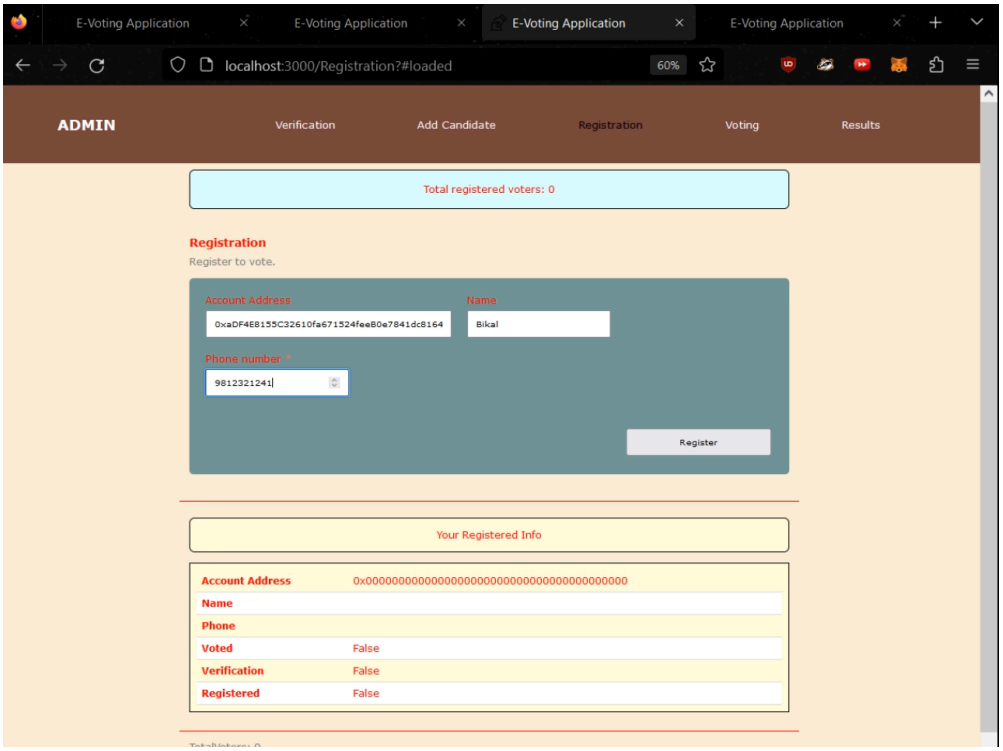

Figure 14: Test Case 4

5.2.1.5 Test Case 5

Test Objective: Test case for approval of voters(Admin Side)

Expected Output:

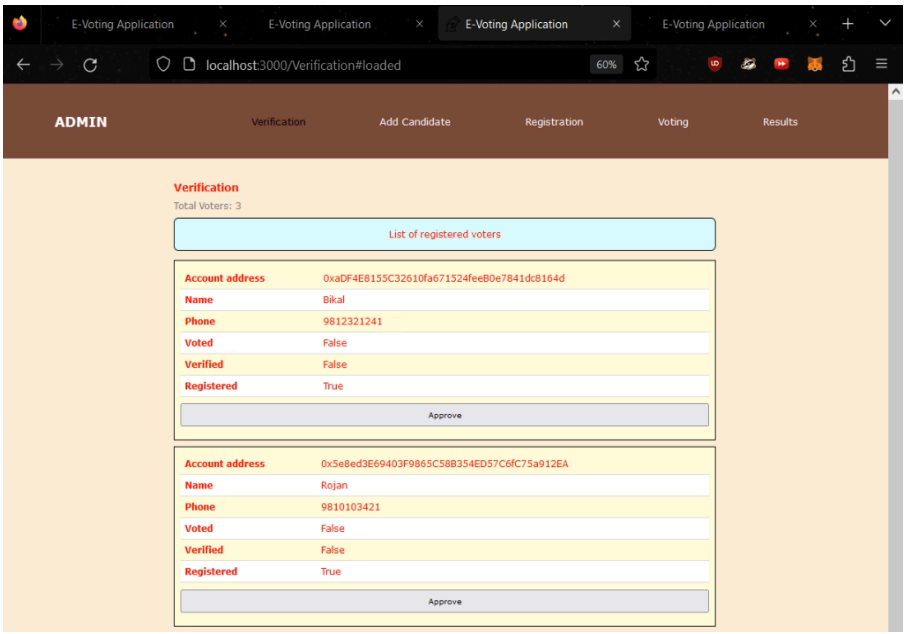

Figure 15: Test Case 5

5.2.1.6 Test Case 6

Test Objective: Test case for casting a vote by voter

Expected Output:

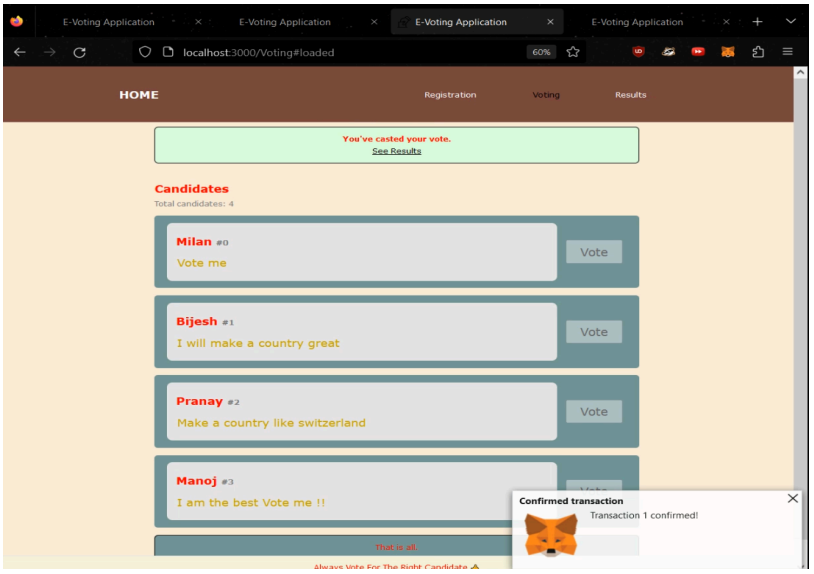

Figure 16: Test Case 6

5.2.1.7 Test case 7

Test Objective: Test case for selecting winner (Admin Side)

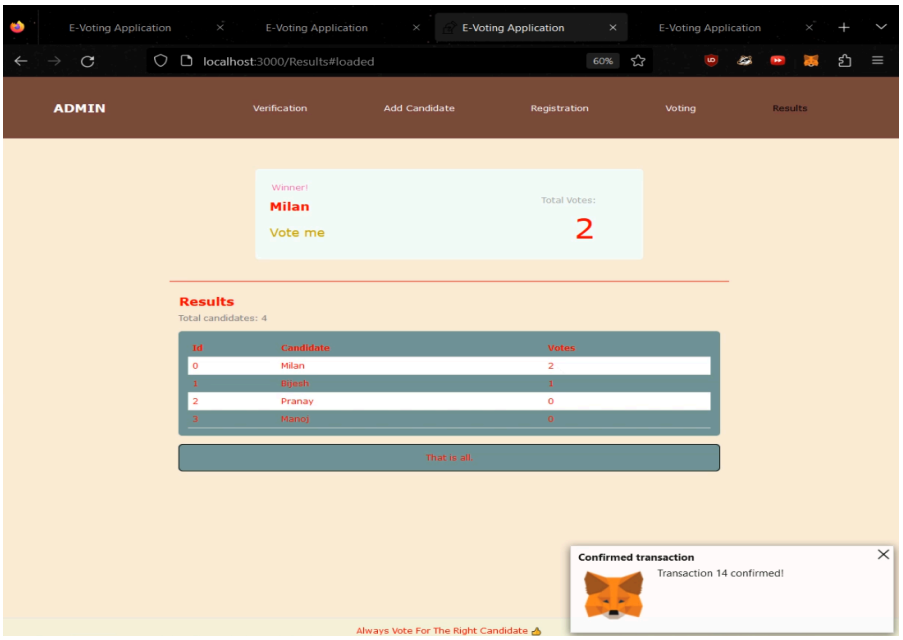

Figure 17: Test Case 7

5.2.2 System Testing

System testing is a software testing technique used to evaluate the behavior of an entire software system or application, typically by testing the system as a whole rather than individual components or modules. The goal of system testing is to ensure that the software system works as intended, meets the specified requirements, and is free from defects and bugs that could negatively affect its functionality or usability

5.2.1.7 Test case 8

Input: After each transaction related to the voting is completed via the system, a block should be generated in the blockchain. The transaction can be one of the following:

- Verification (Admin)
- Add candidate (Admin)
- Registration
- Voting

Expected Output:

If the transaction is successful, there is a contract call. The contract call generates new blocks into the blockchain. It can be viewed through Ganache Cli.

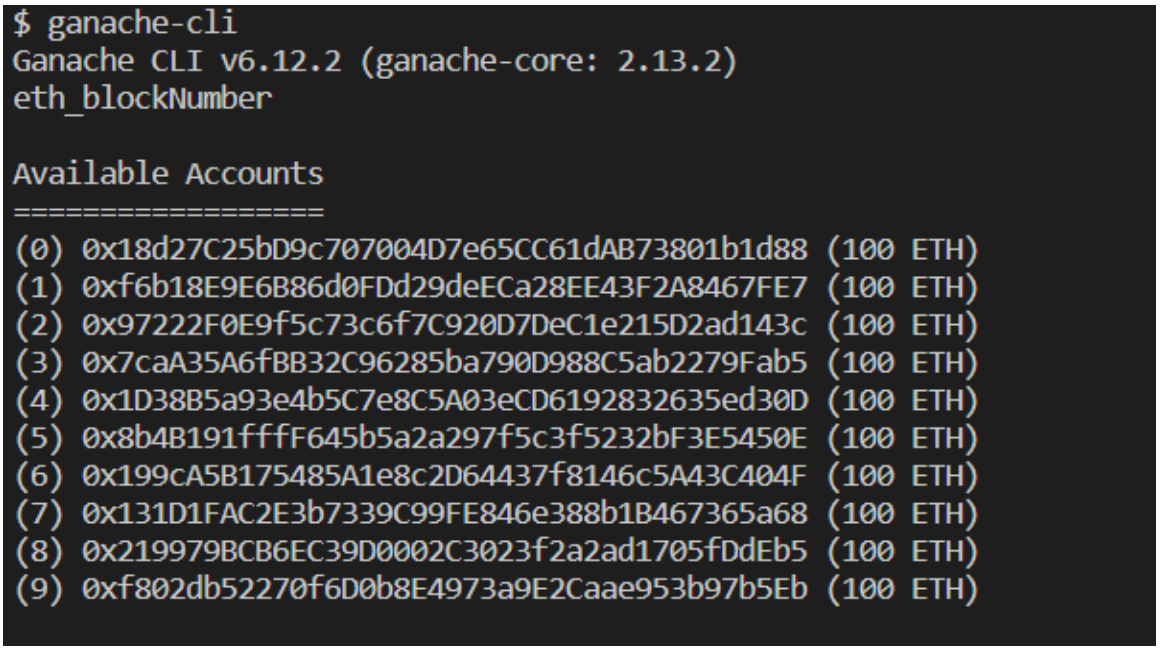

Figure 18: Ganache accounts

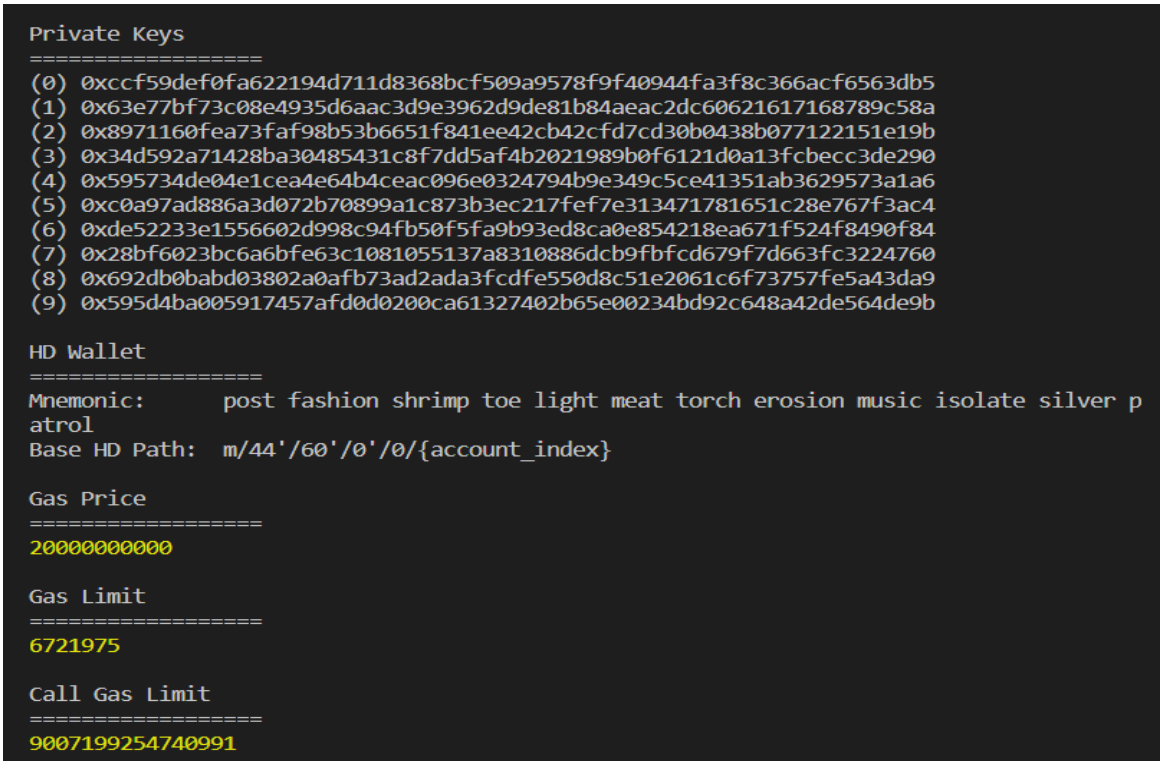

Figure 19: Ganache address and gas information

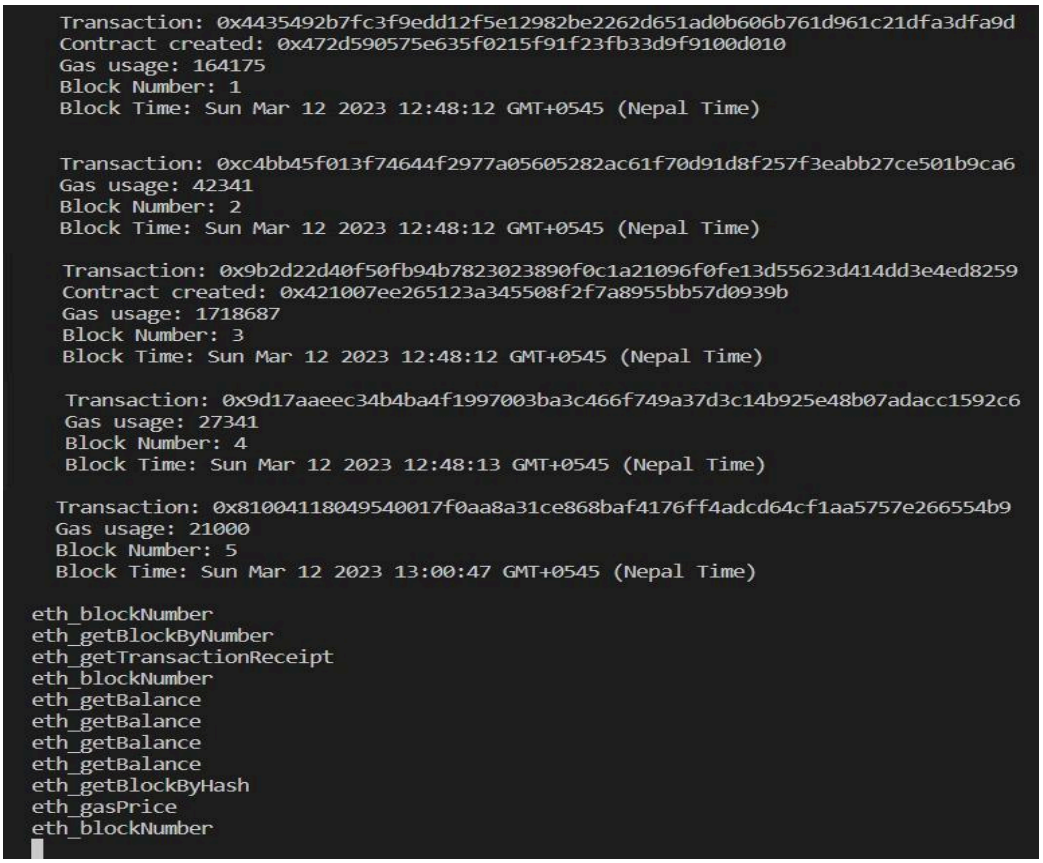

Figure 20: Transactions of each block

5.3 RESULT ANALYSIS

5.3.1 PROPOSED MODEL

The proposed model for e-voting using blockchain is a secure and transparent voting system that leverages the benefits of blockchain technology. The system is designed to ensure that every vote is counted and recorded correctly, without the risk of fraud or manipulation. The system uses a decentralized architecture, meaning that there is no central authority controlling the voting process. Instead, the voting process is managed by a network of nodes, with each node responsible for verifying and validating votes. To cast their vote, a voter first needs to authenticate their identity using a secure identity verification process using a private blockchain ethereum address. This process uses advanced cryptographic algorithms to verify the identity of the voter and ensure that they are authorized to cast a vote. Once authenticated, the voter can cast their vote using a secure and user-friendly interface. The vote is encrypted and transmitted to the blockchain network, where it is validated by the nodes. The nodes use a consensus algorithm to validate the vote, ensuring that it is valid and that it meets the rules of the voting system. Once the vote is validated, it is recorded on the blockchain ledger, which is a secure and tamper-proof record of all votes cast.In Nepal, the proposed e-voting system could increase voter turnout and promote democratic participation.Additionally, the use of blockchain technology could increase transparency and trust in the voting process, which is crucial for a thriving democracy.The blockchain ledger ensures that every vote is recorded correctly and that no votes are lost or manipulated. The system also includes features such as voter privacy protection and fraud detection to ensure the integrity and availability of the system. Overall, the proposed model for e-voting using blockchain is a secure, transparent, and user-friendly voting system that can help to increase trust in the voting process and promote democratic participation. However, implementing an e-voting system using blockchain technology in Nepal would require overcoming several challenges, such as ensuring secure and reliable internet connectivity and building trust among citizens in the new system. Therefore, careful planning and collaboration among stakeholders, including government agencies, civil society, and technology experts, would be essential to successfully implementing the proposed e-voting system in Nepal.

5.3.2 ADVANTAGES OF PROPOSED MODEL

The proposed e-voting system using blockchain technology offers several advantages over traditional voting systems. The proposed system is decentralized and is of tamper-proof nature . It ensures that votes are recorded securely and accurately. The use of cryptographic algorithms and digital signatures makes it virtually impossible for anyone to manipulate or alter the voting results. The use of digital identities and a user-friendly interface makes it easier for voters to participate in the voting process. Voters can cast their votes from anywhere in the world, making the process more accessible and convenient. The system can reduce the costs associated with traditional voting systems. The need for physical ballot boxes, paper ballots, and manual vote counting is eliminated, reducing the overall cost of the voting process.

5.3.3 DISADVANTAGES OF PROPOSED MODEL

While there are many advantages of the proposed blockchain system using e-voting, there are also few disadvantages to consider. The technical complexity is one of the huge disadvantages of the proposed system. Implementing a system requires a significant amount of technical expertise and resources. Developing, testing, and maintaining the system can be complex and time-consuming. There is currently no standard framework for the proposed system, which means that different systems may have different security features and voting processes. Although the proposed blockchain e voting system is considered secure, it is not invulnerable to cyber attacks. Hackers may be able to breach the system, alter voting results, or steal personal information, which could compromise the integrity of the voting process. Another disadvantage of the proposed e-voting system is that the address of the user is shown when the user login to the system. This may be vulnerable to attacks if any other person gets access to the address.

5.3.4 RELEVANT SCREENSHOT

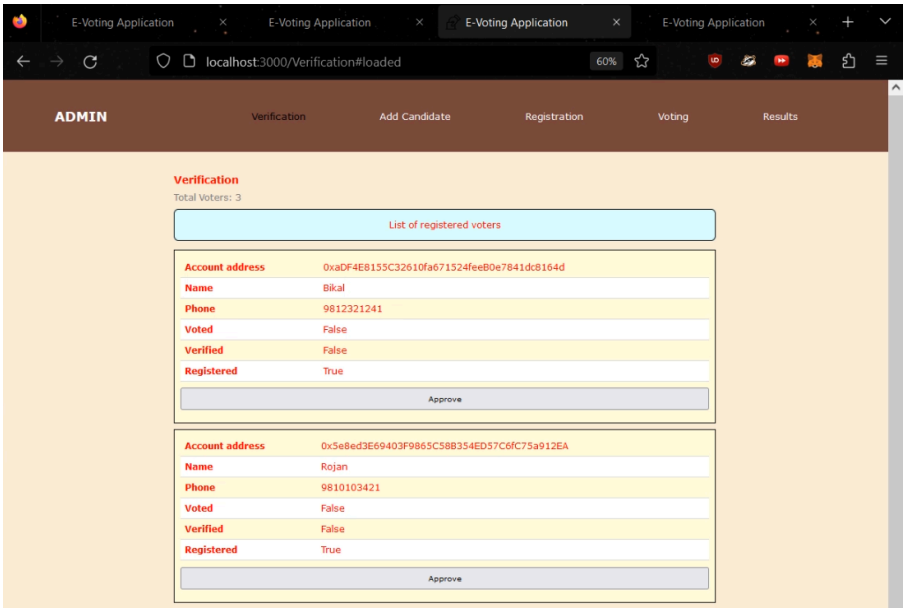

Figure 21: Blocks in blockchain

The blocks in blockchain containing hashes will help to identify if the records have been tampered or not.

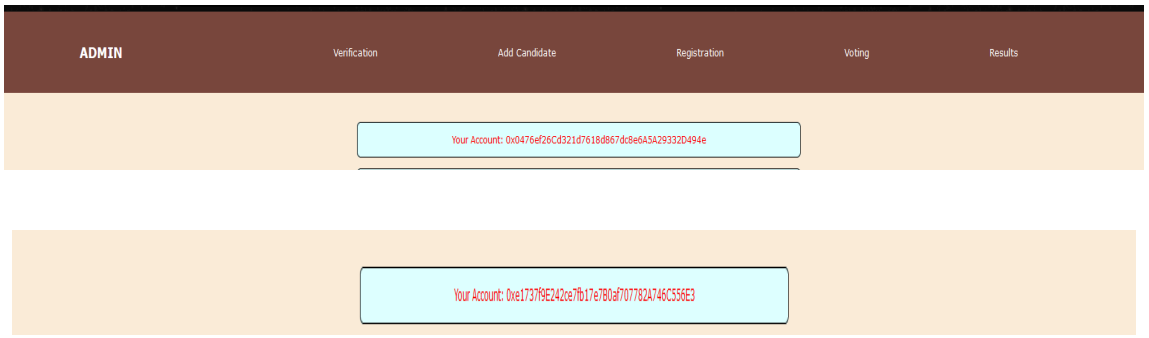

Figure 22: User address shown in the homepage

CHAPTER 6: CONCLUSION AND FUTURE RECOMMENDATIONS

6.1 CONCLUSION

Thus, the suggested system is e-voting that uses blockchain technology. It has emerged as a potential solution to many of the challenges associated with traditional voting systems, including security concerns, administrative inefficiencies, and lack of accessibility. Blockchain technology, with its decentralized and tamper-proof ledger system, offers a promising platform for e-voting that can address many of the limitations of traditional voting systems. Every vote is recorded on a distributed ledger that can be traced and verified by all participants, ensuring that the results are accurate and free from manipulation. The use of cryptographic algorithms provides additional layers of security, making it difficult for hackers or other bad actors to compromise the system. Another advantage of e-voting using blockchain is its potential to increase accessibility and participation in the voting process.

However, e-voting using blockchain also faces several challenges that need to be addressed before it can be widely adopted. Technical challenges, such as scalability and interoperability, need to be overcome to ensure that the system can handle large-scale elections and work seamlessly with other systems. Standardization is also an important issue, as the lack of standardization may result in variations in the design, development, and deployment of voting systems, affecting their reliability and accuracy. In addition, regulatory concerns and legal frameworks need to be addressed to ensure that e-voting using blockchain meets the necessary legal and ethical standards.

Despite these challenges, ongoing research and development efforts are focused on addressing these issues and advancing the adoption of blockchain e-voting worldwide. As the technology continues to evolve and mature, it is expected that e-voting using blockchain will play an increasingly important role in shaping the future of democracy and electoral processes. By providing a transparent, secure, and efficient voting system, e-voting using blockchain can help to promote democratic values and ensure the integrity of election results.

6.2 FUTURE RECOMMENDATIONS

Despite all the work put into this project, there are still many issues that require further research and development. Future improvements could include the following to address current issues and research gaps. The system can be integrated with other government systems, such as identity verification systems to increase the security and efficiency of the voting process. Including email/phone verification (OTP, etc.) when registering voters through email can be done. The system can include an automated verification for the registered users rather than manually authorizing by the admin. The system may include a feature to generate an overall report at the end of the election. Reports can include a variety of information, such as the number of people eligible to vote, the number of people who participated in elections, bar/pie charts showing election statistics, and more. Making the system accessible for people with disabilities can help to promote inclusivity and ensure that all eligible voters can participate in the voting process.Thus, the above recommendations can be employed in future works.as

REFERENCES

[1] Fun Facts for Kids on animals, Earth, history and more! (no date) DK Find Out! Available at: https://www.dkfindout.com/us/more-find-out/what-does-politician-do/what-is-an-election (Accessed: December 9, 2022).

[2] Electronic Voting Systems — (n.d.). Electronic Voting Systems &Mdash; https://aceproject.org/ace-en/topics/et/eth/eth02/eth02b/default(Accessed:December 11, 2022).

[3] Hayes, A. (2022) Blockchain facts: What is it, how it works, and how it can be used, Investopedia. Investopedia. Available at: https://www.investopedia.com/terms/b/blockchain.asp (Accessed: December 13, 2022).

[4] Decentralized voting system using blockchain (2022) GeeksforGeeks.GeeksforGeeks. Available at: https://www.geeksforgeeks.org/decentralized-voting-system-using-blockchain/ (Accessed: December 18, 2023).

[5] Consensus mechanisms in Blockchain: A beginner’s guide - crypto.com . Available at: https://crypto.com/university/consensus-mechanisms-in-blockchain/ (Accessed:Dec 25, 2023).

[6] Daly, L. What is proof of work (POW)?, The Motley Fool. Available at: https://www.fool.com/investing/stock-market/market-sectors/financials/cryptocurrency-stocks/proof-of-work/ (Accessed:January 1, 2023).

[7] Person (2022) A beginner's guide to proof-of-stake, Worldcoin. Worldcoin. Available at: https://worldcoin.org/articles/what-is-proof-of-stake (Accessed: January 9, 2023).

[8] What is "proof of work" or "Proof of stake" Coinbase. Coinbase. Available at: https://www.coinbase.com/learn/crypto-basics/what-is-proof-of-work-or-proof-of-stake (Accessed:January 14, 2023).

[9] Delegated proof of stake (DPoS) bitFlyer. Available at: https://bitflyer.com//glossary/delegated-proof-of-stake-dpos (Accessed: Jan 19, 2023).

[10] Binance Academy (2022) Delegated proof of stake explained, Binance Academy. Binance Academy. Available at: https://academy.binance.com/en/articles/delegated-proof-of-stake-explained (Accessed: January 22, 2023).

[11] Practical byzantine fault tolerance(pbft) (2022) GeeksforGeeks. GeeksforGeeks. Available at:https://www.geeksforgeeks.org/practical-byzantine-fault-tolerancepbft/ (Accessed: January 24, 2023).

[12] Apleasant (2013) The important uses of cryptography in electronic voting and counting, apleasant. Available at: https://www.ndi.org/e-voting-guide/examples/cryptography-in-e-voting (Accessed: February 2, 2023).

[13] Introduction to hashing - data structure and algorithm tutorials (2023) GeeksforGeeks. GeeksforGeeks. Available at: https://www.geeksforgeeks.org/introduction-to-hashing-data-structure-and-algorithm-tutorials/ (Accessed: February 7, 2023).

[14] Digital Signature and cryptography in cryptocurrency (no date) Sign.cc. Available at: https://sign.cc/digital-signature-in-cryptocurrency (Accessed: February 9, 2023).

[15] Smart contract (2023) Corporate Finance Institute. Available at: https://corporatefinanceinstitute.com/resources/valuation/smart-contract (Accessed: February 10, 2023).

[16] Arora, S. (2023) What is a smart contract in blockchain and how does it work [2022 edition]: Simplilearn, Simplilearn.com. Simplilearn. Available at: https://www.simplilearn.com/tutorials/blockchain-tutorial/what-is-smart-contract

[17] Laurence, T. (2019) Blockchain, Amazon. John Wiley & Sons, Inc. Available at: https://aws.amazon.com/blockchain/what-is-ethereum/ (Accessed:February 15 , 2023).

[18] Kshetri, N. and Voas, J. (2018) “Blockchain-enabled e-voting,” IEEE Software, 35(4), pp. 95–99. Available at: https://doi.org/10.1109/ms.2018.2801546.

[19] Macrinici, D., Cartofeanu, C. and Gao, S. (2018) “Smart contract applications within

Blockchain Technology: A systematic mapping study," *Telematics and Informatics*, 35(8), pp. 2337–2354. Available at: https://doi.org/10.1016/j.tele.2018.10.004.

[20] Li, Y. et al. (2022) "A blockchain-based self-tallying voting protocol in decentralized IOT," IEEE Transactions on Dependable and Secure Computing, 19(1), pp. 119–130. Available at: https://doi.org/10.1109/tdsc.2020.2979856.

[21] Shahzad, B. and Crowcroft, J. (2019) "Trustworthy electronic voting using adjusted blockchain technology," IEEE Access, 7, pp. 24477–24488. Available at: https://doi.org/10.1109/access.2019.2895670.

[22] Panja, S. et al. (2020) "A smart contract system for decentralized borda count voting," IEEE Transactions on Engineering Management, 67(4), pp. 1323–1339. Available at: https://doi.org/10.1109/tem.2020.2986371.

[23] Dimitriou, T. (2020) "Efficient, coercion-free and universally verifiable blockchain-based voting," Computer Networks, 174, p. 107234. Available at: https://doi.org/10.1016/j.comnet.2020.107234.Zhou, Y. et al. (2019) "An improved foo voting scheme using blockchain," International Journal of Information Security, 19(3), pp. 303–310. Available at: https://doi.org/10.1007/s10207-019-00457-8.

[24] Cong, L.W. and He, Z. (2018) "Blockchain disruption and smart contracts," National bureau of economic research [Preprint]. Available at: https://doi.org/10.3386/w24399.

[25] Zheng, Z. et al. (2020) "An overview on smart contracts: Challenges, advances and platforms," Future Generation Computer Systems, 105, pp. 475–491. Available at: https://doi.org/10.1016/j.future.2019.12.019.

[26] Namasudra, S. *et al.* (2020) "The revolution of blockchain: State-of-the-art and research challenges," *Archives of Computational Methods in Engineering*, 28(3), pp. 1497–1515. Available at: https://doi.org/10.1007/s11831-020-09426-0.

[27] Bamakan, S.Motavalli, A. and Babaei Bondarti,(2020) "A survey of Blockchain Consensus Algorithms Performance Evaluation," *Expert Systems with Applications*, 154, p. 113385. Available at: https://doi.org/10.1016/j.eswa.2020.113385.

[28] Hu, T. et al. (2021) "Transaction-based classification and Detection Approach for ethereum smart contract," Information Processing & Management, 58(2), p.

102462. Available at: https://doi.org/10.1016/j.ipm.2020.102462.

[29] Gürkaynak, G. et al. (2018) “Intellectual property law and practice in the blockchain realm,” Computer Law & Security Review, 34(4), pp. 847–862. Available at: https://doi.org/10.1016/j.clsr.2018.05.027.

[30] Zhan, Y. et al. (2021) “Drbft: Delegated randomization Byzantine Fault Tolerance Consensus Protocol for Blockchains,” Information Sciences, 559, pp. 8–21. Available at: https://doi.org/10.1016/j.ins.2020.12.077.

[31] Baudier, P. *et al.* (2021) “Peace engineering: The contribution of Blockchain Systems to the e-voting process,” *Technological Forecasting and Social Change*, 162, p. 120397. Available at: https://doi.org/10.1016/j.techfore.2020.120397.

[32] Slovak, M. and Vassil, K. (2017) “Could internet voting halt declining electoral turnout? new evidence that e-voting is habit forming,” Policy & Internet, 10(1), pp. 4–21. Available at: https://doi.org/10.1002/poi3.160.

[33] Taş, R. and Tanrıöver, Ö.Ö. (2020) “A systematic review of challenges and opportunities of blockchain for E-voting,” Symmetry, 12(8), p. 1328. Available at: https://doi.org/10.3390/sym12081328.

[34] Huang, J. et al. (2021) “The application of the blockchain technology in Voting Systems,” ACM Computing Surveys, 54(3), pp. 1–28. Available at: https://doi.org/10.1145/3439725.

[35] Pawlak, M. and Poniszewska-Marańda, A. (2021) “Trends in blockchain-based electronic voting systems,” *Information Processing & Management*, 58(4), p. 102595. Available at: https://doi.org/10.1016/j.ipm.2021.102595.

[36] Firdaus, A. *et al.* (2019) “The rise of ‘Blockchain’: Bibliometric Analysis of blockchain study,” *Scientometrics*, 120(3), pp. 1289–1331. Available at: https://doi.org/10.1007/s11192-019-03170-4.

[37] Editor (2019) Functional and non-functional requirements: Specification and types, AltexSoft. AltexSoft. Available at: https://www.altexsoft.com/blog/business/functional-and-non-functional-requirements-specification-and-types/ (Accessed: February 16, 2023).

[38] Editor (2020) Non-functional requirements: Examples, typesAltexSoft. AltexSoft. Available at: https://www.altexsoft.com/blog/non-functional-requirements/ (Accessed: February 16, 2023).

[39] What is a feasibility study? definition, benefits and types (no date). Available at: https://www.indeed.com/career-advice/career-development/feasibility-studies (Accessed: February 18, 2023).

[40] The basics of the Object Model (no date) Section. Available at: https://www.section.io/engineering-education/basics-of-the-object-model/ (Accessed: February 18, 2023).

[41] Vanner, C. (ed.) What is process modeling?What is process modeling? 6 essential questions answered. Available at: https://www.bizagi.com/en/blog/process-modeling-and-mapping/what-is-process-modeling-6-essential-questions-answered (Accessed: February 20, 2023).

[42] writer, A.B.C.B., Curran, A.B. and writer, B. (2022) What is practical Byzantine fault tolerance? complete beginner's guide, Blockonomi. Available at: https://blockonomi.com/practical-byzantine-fault-tolerance/ (Accessed: February 21, 2023).

[43] BeInNews Academy (2023) Byzantine fault tolerance (BFT) explained, BeInCrypto. Available at: https://beincrypto.com/learn/byzantine-fault-tolerance/#h-advantages-of-practical-byzantine-fault-tolerance (Accessed: February 23, 2023).

[44] The Practical Byzantine Fault Tolerance (pbft) - javatpoint -www.javatpoint.com. Available at: https://www.javatpoint.com/practical-byzantine-fault-tolerance (Accessed: February 23, 2023).

[45] Fruhlinger, J. (2020) What is PKI? and how it secures just about everything online, CSO Online. CSO. Available at: https://www.csoonline.com/article/3400836/what-is-pki-and-how-it-secures-just-about-everything-online.html (Accessed: February 25, 2023).

[46] How blockchain addresses public key infrastructure shortcomings (2020) Remme Blog. Available at:

https://remme.io/blog/how-blockchain-addresses-public-key-infrastructure-shortcomings (Accessed: February 28, 2023).

[47] Hack Reactor (2021) What is JavaScript used for?, Hack Reactor. Available at: https://www.hackreactor.com/blog/what-is-javascript-used-for (Accessed: February 28, 2023).

[48] Camus, A. (2022) Introduction to reactjs: A guide for beginners, Microverse. Microverse. Available at: https://www.microverse.org/blog/introduction-to-reactjs-a-guide-for-beginners (Accessed: March 1, 2023).

[49] Truffle: Blockchain and smart contract tools Kaleido. Truffle Developer Documentation. Available at: https://www.kaleido.io/blockchain-platform/truffle (Accessed: March 2, 2023).

[50] What is the ganache blockchain? (no date) University of Colorado blockchain alliance. Kaleido. Available at: https://www.cublockchainalliance.com/what-is-ganache-blockchain (Accessed: March 2, 2023).

[51] Heller, M. (2022) What is visual studio code? Microsoft's extensible code editor, InfoWorld. InfoWorld. Available at: https://www.infoworld.com/article/3666488/what-is-visual-studio-code-microsofts-extensible-code-editor.html (Accessed: March 4, 2023).

[52] What is Git and why should you use it? (2022) What is Git and Why Should You Use It? Free Intro to Git Guide. Available at: https://www.nobledesktop.com/learn/git/what-is-git (Accessed: March 5, 2023).

[53] An introduction to github (2020) Digitalgov. Available at: https://digital.gov/resources/an-introduction-github/ (Accessed: March 6, 2023).

[54] Byju's Exam Prep (2021) *Difference between MS word and MS excel - key differences*, *BYJUS*. BYJU'S. Available at: https://byjus.com/free-ias-prep/difference-between-ms-word-and-ms-excel/ (Accessed: March 8, 2023).

APPENDICES