A Minor Project Final Report on

Online Voting System

Submitted in Partial Fulfillment of the Requirements for

The Degree of **Bachelor of Engineering in Computer Engineering**

Under Pokhara University

Submitted by:

Aashma Adhikari, 181202

Anusha Thapa Magar, 181207

Bharosha Rai, 181210

Namuna Adhikari, 181220

Under the supervision of

Mr. Shivahari Acharya

Department of Computer Engineering

Nepal College Of Information Technology

Balkumari, Lalitpur

Acknowledgement

We would like to express our sincere gratitude to our trainer Roshan Chitrakar for giving us the golden opportunity to work on the project entitled “Online voting system”. The research from this project have helped us a lot to learn about blockchain, python, Fast API. The guidance that Shivahari Acharya sir laid down helped us complete this project with high efficiency within a short time.

We would also like to show our appreciation to our head of department Er. Resha Deo and our college principal Mr. Nirajan Khakurel for giving us chance to work on this project. We would also like to thank our college professors and friends for motivating us and also helping us on little things we made along the way of accomplishing this project.

The completion of this project within the limited time frame wouldn’t have been possible without the participation of these people. They are what keeps us going and we thank them from the bottom of our heart.

Abstract

Online voting system is web-based platform where voters can vote from wherever there is an Internet connection in secured manner. It has been used in different forms and in different field be it a democratic election, choosing best worker in offices or selecting any representative of schools/colleges union, club, or individual class. It has immense potential to decrease organizational costs and increase voter turnout. Our country Nepal is still following the traditional way of voting i.e. Using ballot paper and need of people to be present in the voting center/station. So, online voting system will cut the need to print ballot papers or open polling stations. However, there are challenges to achieve such system due to security threats. A single mistake can lead to large manipulation of votes and results. Blockchain technology came into the ground to overcome these issues and offers decentralized nodes for electronic voting, and it is used to produce electronic voting systems mainly because of their end-to-end verification advantages. It is a continuous and continuously growing ledger that holds, in a secure, chronological, and immutable way, a permanent record of all the transactions that have taken place. Value in a blockchain can be anything. In our case, the electorate is one party and the candidate who earns the vote is the other. Without having a controlling central authority body, the block chain can be implemented in a more stable manner in mass electoral voting practice. A voting system that uses a more stable, tamper-proof blockchain.

Keywords

Online voting system, Web-based, Blockchain, Decentralization, Distributed ledger

Table of Contents

List of Figures

List of Tables

1. Introduction

Voting is an important part of choosing a rightful candidate through the democratic process and hence is a crucial process for any country. Among different vote collection methods, a paper-based method is widely adopted. In this method, the voter selects the preferred candidate or party by marking the corresponding symbol or letter. In the context of Nepal too, the paper-based method is practiced for the process of election. With the advancement of technology all around the world, isn't it high time that we modernize the voting process? The use of machines at the polling centers or casting vote directly from home via the internet can digitize the voting process. It would be wonderful if the voting is carried out through a platform that is secure enough and provides transparency into the result of the election. The places like Estonia, Norway, and Australia have already conducted an online voting process. There are several challenges that we have to face to create this online platform. Digitizing the voting process may upsurge different security issues. The weakness in the architecture of the system can lead to manipulation of the election result. Vote fraud due to attacks from the intruders may manipulate the result or may result in the loss of the data which is totally unacceptable in a democratic process like the election and the election process cannot be rerun. One of the best potential solutions is to use blockchain technology.

1.1 Problem Statement

Electronic Voting is not a new concept, in fact, it was first-ever introduced in 1986. David Shagum was the first to introduce the first-ever electronic voting system which was based on public-key cryptography [1]. Keeping no connection between voters and ballots, a secure country election was conducted. Electronic voting machines have been viewed as vulnerable, based on physical security concerns. These machines can be sabotaged by anyone who has physical access to it, thus affecting all votes cast on the machine. The major vulnerability is the centralized database where election data are stored. This vulnerability could be a single point of failure. However, integrating blockchain technology on the aforementioned machine can take the entire voting system to a whole new level. The blockchain-based election was held for the first time in Sierra Leone, a country in Western Africa. Blockchain is a distributed, immutable public ledger introduced in 2008 by Satoshi Nakamoto in the form of the creation of the first cryptocurrency, called

Bitcoin. The Bitcoin blockchain uses a decentralized public ledger with POW (Proof-of-Work) consensus protocol [2]. The chain of the blocks in the blockchain is cloned, cryptographically signed, and verified publicly at every transaction so that no one can meddle with the data once written on the blockchain [3]. Due to its immutable property and decentralized architecture, it can carry out elections in a more transparent and secure way. Blockchain Online Voting System comprising of blockchain where votes cast is recorded, python scripts handling overall election processes, private nodes providing consensus, and the voter portal accessible to the voters can guarantee the security, integrity, and transparency of the casted votes including the anonymity of the voters avoiding the single point of failure as compared to the traditional centralized voting system. The script can be taken as an election, the transaction as a ballot paper, and the blockchain as a ballot box. This blockchain is accessible to the general public to make the election more transparent.

1.2 Objectives

The voting system that is hereby conceived must satisfy the following requirements:

1. To improve the existing voting system using Blockchain technology as a secure transaction database.
2. To digitize the overall voting system.
3. To ensure voting systems should be tamper-proof and decentralized.

1.3 Significance of the Study

Through the comprehensive exploration of Online Voting System based on blockchain, it introduces a different means of democratic changes to the previous system. The decentralized system will provide a proof of concept where a voter can vote by keeping their privacy i.e. a voted person or a system will have no knowledge of the voter's identity.

Previously Online Voting System systems were not taken into consideration as they possess a huge risk since they are centralized but introducing a concept of blockchain

technology will have high benefits as the chance of tampering with the vote count will be significantly zero.

Being a tamper-proof system even while keeping the privacy of the voters will be a huge turnover in this digital era where almost all of the systems are digital except for some like voting which doesn't earn public trust. Especially, this system will benefit the following:

Government: This system will allow the government to conduct the election without needing to spend a lot of resources on paper ballots or the security of the system.
Public: The process of Online Voting System will be easy and secure i.e. the public can vote from digital machines available at the election venue of their respective locations.

Future researchers: This system will open up a new way for future researchers, they can use this technology to further develop it or use it on different platforms.

1.4 Scope and Limitations

The scopes and limitations of our Online Voting System system based on blockchain are as follows:

Scope:

The scope of this project is to digitize the overall voting system with the help of blockchain technology for more accurate and convenient organization of election systems. Our system provides the following scope:

1. This system can be implemented on various platforms where the election needs to be held for a different purpose whether it be for government election or for choosing a representative.

2. Voters will have the opportunity to vote quickly and conveniently.

3. Votes can be counted in real time.

4. Voting can be completed in a very short period of time with better accuracy.

Limitations:

Although our project provides reliability by providing an alternative to the old existing paper based voting system, we have some limitations, which are to be taken into consideration. Some of the limitations of our projects are:

1. The system will act as a blueprint for the voting and election process. This will not cover all the systems that we can possibly use.

2. An admin with the highest level of privileges can see the details of the voting phase including who voted for whom.

3. Difficulties in implementation in rural areas.

2. Literature Review

2.1 Related Works

In 2008, an individual (or group) writing under the name of Satoshi Nakamoto published a paper entitled “Bitcoin: A Peer-To-Peer Electronic Cash System” (1). This paper described a peer-to-peer version of the electronic cash that would allow online payments to be sent directly from one party to another without going through a financial institution. Bitcoin was the first realization of this concept. Now “cryptocurrencies” is the label that is used to describe all networks and mediums of exchange that uses cryptography to secure transactions-as against those systems where the transactions are channeled through a centralized trusted entity. As in the case of bitcoin it was based on the concept of the blockchain, but it did not have any token in it. So, after its new cryptocurrency named Ethereum was developed who introduced new concept as Smart contract in the market. Idea of smart contract helps us to create a decentralized application with the help of blockchain. In the field of blockchain various companies and individual have developed Nakamoto. S.

Nir Kshetri (2) each voter is considered as a wallet, and the transactions between wallet is limited to one. As the candidates are considered as the receiver wallet. The vote is the transaction between all the candidates or receiver wallets. The method used in this paper is Blockchain enabled e-voting which is using an encrypted key along with the alteration-proof user IDs. The advantage is Blockchain enabled e-voting will help us to ensure the aspect of security as well as transparency which would help to reduce electoral violence and produce more mathematically precise voting results. The Disadvantage is They did not use a decentralized voting system (only meant for one single place). No consensus. The wallet-coin model can be amended to single wallet.

Crowcroft (3), to ensure data protection, such as block formation and sealing, it proposes useful hashing techniques. The method used are consensus based blockchain algorithm. The advantages are Used their own framework, better hashing algorithm. The disadvantage is No proof if the model will work or not, Untested with blockchain frameworks.

Fredrik (4), attempt 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 aims by using a Go-Ethereum Proof-of-Authority (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 most 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.

European commission research project entitled" Blockchain for digital government" (5). This report looks at the ongoing exploration of blockchain technology by governments. The analysis of a group of pioneering developments of public services shows that blockchain technology can reduce bureaucracy, increase the efficiency of administrative processes, and increase the level of trust in public recordkeeping.

Zhang (6) proposes a local voting mechanism conceptualized on block-chain to help decision making for its peers' networks. It protects the privacy and enables detection as well as correction against cheating. The method used are Distributed consensus based blockchain algorithm. The advantage is Elections can be used as a Blockchain part of Smart Contract, Peer to peer network, consensus, two phase validation (decryption private key, smart contract verification). The disadvantage is No proof if the model will work or not. Untested with many blockchain frameworks. Can use a better blockchain framework for increasing transactions per second. If evaluated with faster and premised blockchain frameworks it can set the standards.

2.2 Technologies Used

The tools and techniques that we have deployed in our system are explained below.

1. Blockchain

Blockchain is a system of recording information in a way that makes it difficult or impossible to change, hack, or cheat the system. A blockchain is essentially a digital ledger of transactions that is duplicated and distributed across the entire network of computer systems on the blockchain. Each block in the chain contains a number of transactions, and every time a new transaction occurs on the blockchain, a record of that transaction is added to every participant's ledger. The decentralized database managed by multiple participants is known as Distributed Ledger Technology (DLT).

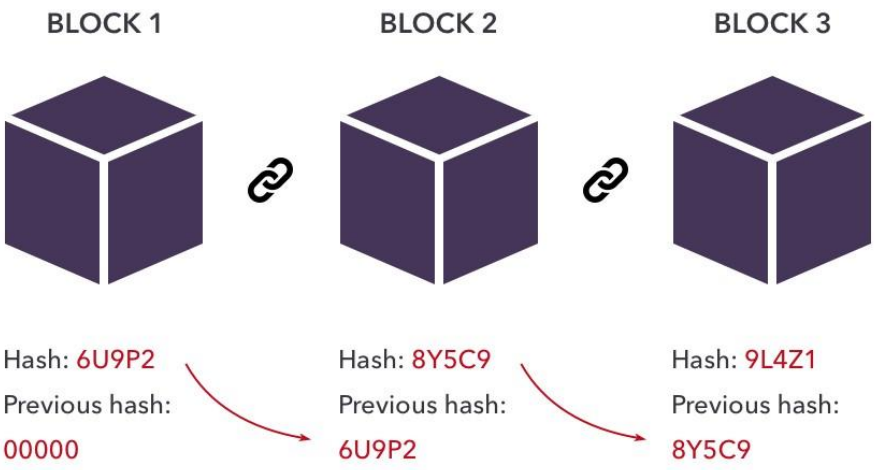

Figure i: Different blocks in Blockchain

Blockchain is a type of DLT in which transactions are recorded with an immutable cryptographic signature called a hash. This means if one block in one chain was changed, it would be immediately apparent it had been tampered with. If hackers wanted to corrupt a blockchain system, they would have to change every block in the chain, across all of the distributed versions of the chain.

2. Proof of Work

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 (a long string of characters) that

matches the target hash for the current block. Proof of work is a technique used by cryptocurrencies to verify the accuracy of new transactions that are added to a blockchain. The decentralized networks used by cryptocurrencies and other defi applications lack any central governing authority, so they employ proof of work to ensure the integrity of new data.

3. Face Recognition

For our facial verification process, we have used the face recognition python library. Face recognition algorithms can extract features from a face image, namely positions of forehead, eyes, nose, mouth, chin, jaws.

Face Landmarks: There are 68 specific points (called landmarks) that exist on every face.

Face Encodings: This is the 128-encoding feature vector from a pretrained network over millions of images.

Following are the basic steps which are used while creating this library.

1. Encode a picture using the HOG algorithm to create a simplified version of the image. Using this simplified image, find the part of the image that most looks like a generic HOG encoding of a face.

2. Figure out the pose of the face by finding the main landmarks in the face. Once we find those landmarks, use them to warp the image so that the eyes and mouth are centered.

3. Pass the centered face image through a neural network that knows how to measure features of the face. Save those 128 measurements.

4. Looking at all the faces we've measured in the past, see which person has the closest measurements to our face's measurements. That's our match!

The main reason behind using this face recognition feature is to provide additional security to our system so that only eligible and correct voters can participate in the voting process. Here, we simply input the image from our computer or get a direct image from our webcam. Then, the function we have created in our system will give us the output of the list of 128 encoding features which we mentioned earlier.

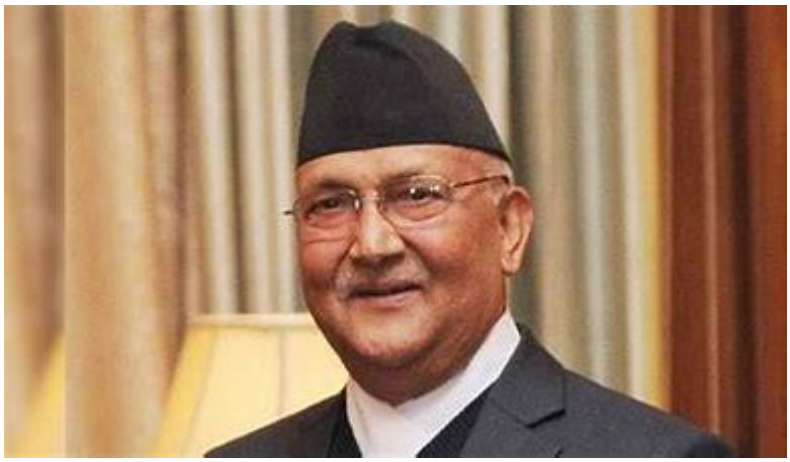

Figure ii: Sample picture of Kp Oli

```
[-3.22140977e-02  1.32757962e-01  1.03660554e-01 -1.54176261e-03
 -1.39290333e-01  4.59651649e-02 -6.04024976e-02 -1.88459799e-01
  1.57444119e-01  -4.63126823e-02   2.79171348e-01  -9.72657353e-02   -2.23405048e-01  -
2.79332399e-02 -7.91804120e-03  1.09720878e-01
 -1.88225403e-01 -1.01629436e-01  2.52081733e-03 -9.77773517e-02
  1.17015883e-01  6.97332993e-02  2.44141910e-02 -3.97718474e-02  -1.52565643e-01 -3.36885393e-
01 -9.99340191e-02 -9.76220816e-02
```

Figure iii: List of Encodings of Figure ii

We match these encodings to the already available encodings to us using the compare faces function. If the result obtained after comparing the encodings is more than the tolerance level of 0.6 units, we consider the input image is permitted to vote in the election.

4. Fast API

Fast API is a Web framework for developing RESTful APIs in Python. Fast API is based on Pydantic and type hints to validate, serialize, and deserialize data, and automatically autogenerate Open API documents.

It fully supports asynchronous programming and can run with Uvicorn and Gunicorn. To improve developer-friendliness, editor support was considered since the earliest days of the project.

5. Hashing

Hashing is the process of converting a given key into another value. A hash function is used to generate the new value according to a mathematical algorithm. The result of a hash function is known as a hash value or simply, a hash.

We have used the Python SHA512 hashing algorithm in our project for hashing the block's address such that a small change in one block will reflect the change in other blocks. SHA stands for Secure Hash Algorithms. These are a set of cryptographic hash functions. These functions can be used for various applications like passwords, etc. The hashlib module of Python is used to implement a common interface to many different secure hash and message digest algorithms.

6. SQLite

SQLite is a C-language library that implements a small, fast, self-contained, high-reliability, full-featured, SQL database engine. SQLite is the most used database engine in the world. Although we have used Blockchain for storing the transactions, we are using SQLite to manage the user's personal data and information.

7. OpenCV

OpenCV is the huge open-source library for computer vision, machine learning, and image processing and now it plays a major role in real-time operation which is very important in today's systems. By using it, one can process images and videos to identify objects, faces, or even handwriting of a human. We have used this library for a face recognition system.

8. Swagger UI

Swagger UI is a collection of HTML, JavaScript, and CSS assets that dynamically generate beautiful documentation from a Swagger-compliant API. Swagger UI allows the development team to visualize and interact with the API's resources without having any of the implementation logic in place.

3. Methodology

3.1 Software Development Life Cycle

Prototype model is a systems development model in which a prototype is built, evaluated, and reworked until an acceptable prototype is achieved. A prototype is a toy implementation of the system. A prototype usually turns out to be a very crude version of the actual system, exhibiting limited functional capabilities, low reliability, and in efficient performance as compared to actual software. In this model, we create the prototype of the actual system, update the requirements, and again rebuild the system until the final requirements are met.

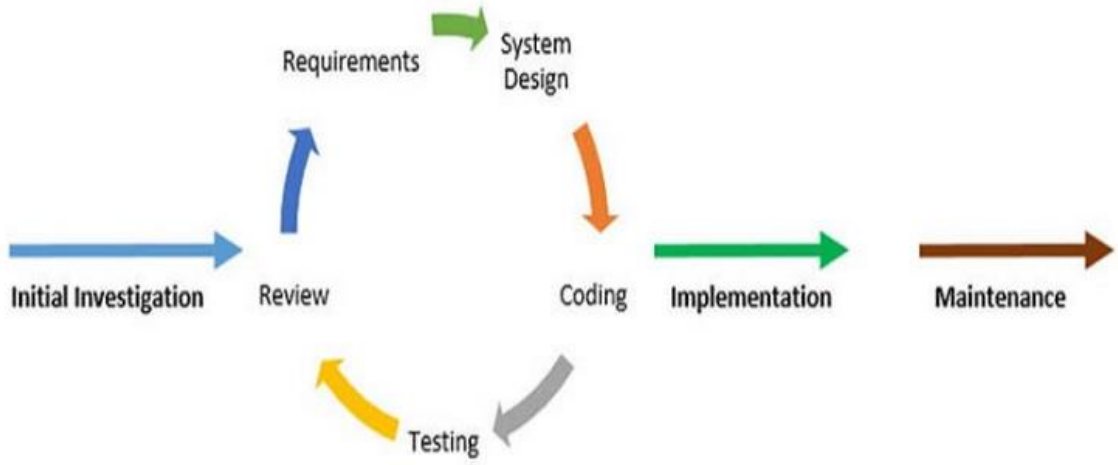

Figure iv: Prototype Model

Requirement Analysis: Complete analysis is performed on the requirement and how to make sure that this requirement will be compatible to previously developed.

Design: Once the requirement for this particular increment is understood and clear, then design will be drafted on how to implement and archive this requirement.

Code: Now the coding is performed in accordance to achieve the purpose of the requirements. All the coding standards will be followed without any defaults and unnecessary hard codes.

Test: This is the last in the incremental phase where aggressive testing is performed on the developed code and defects are reported and resolved.

3.2 Software Requirement Specifications

3.2.1 Generic Voting Principles

1. Only eligible persons can vote.
2. No person gets to vote more than once.
3. The vote is secret.
4. Every correct vote gets counted.
5. The voters trust that their vote is counted.

3.2.2 Voting System Design Criteria

Democratic: A system is considered to be "democratic" if only eligible voters are allowed to vote (eligibility) and if each eligible voter can only cast a single vote (reusability). An additional characteristic is that no one should be allowed to duplicate anyone else's vote.

Accuracy: Correctness of the system. Election systems should record the votes correctly. The announced result should match the actual outcome of the election.

Reliability: No reasonably sized coalition of voters or authorities (either benign or malicious) may disrupt the election. This includes allowing abstention of registered voters, without causing problems or allowing other entities to cast legitimate votes on their behalf, as well as preventing misbehavior of voters and authorities from invalidating the election outcome by claiming that some other actor of the system failed to properly execute its part.

Robustness implies that security should also be provided against external threats and attacks, e.g. denial of service attacks.

Integrity: Votes should not be able to be modified without detection.

Verifiability: Should be possible to verify that votes are correctly counted for in the final tally. Results can be found to agree on the election result by comparing election data with

other holders of election data or by checking whether an individual vote has been properly cast.

Auditability: There should be reliable and demonstrably authentic election records.

Secrecy: No one should be able to determine how or whom any individual voted.

Non-coercibility: Voters should not be able to prove how they voted. An incoercible scheme does not allow the voters to convince any other participant (e.g. a coercer) on what they have voted for.

Fairness: Should ensure that no one can learn the outcome of the election before the announcement of the tally. Therefore, acts like influencing the decision of late voters by announcing an estimate, or providing a significant but unequal advantage (being the first to know) to specific people or groups, are prevented.

Flexibility: Equipment should allow for a variety of ballot question formats.

Convenience: Voters should be able to cast votes with minimal equipment and skills.

Certifiability: Systems should be testable against essential criteria.

Transparency: Voters should be able to possess a general understanding of the whole process.

Cost-effectiveness: Systems should be affordable and efficient.

4. System Design

This field contains the detailed design and architecture of the system and the associated UML diagrams.

4.1 System Architecture

This system proposes a design of new prototype architecture on Online Voting System based on Blockchain technology.

The following figure illustrates the system architecture.

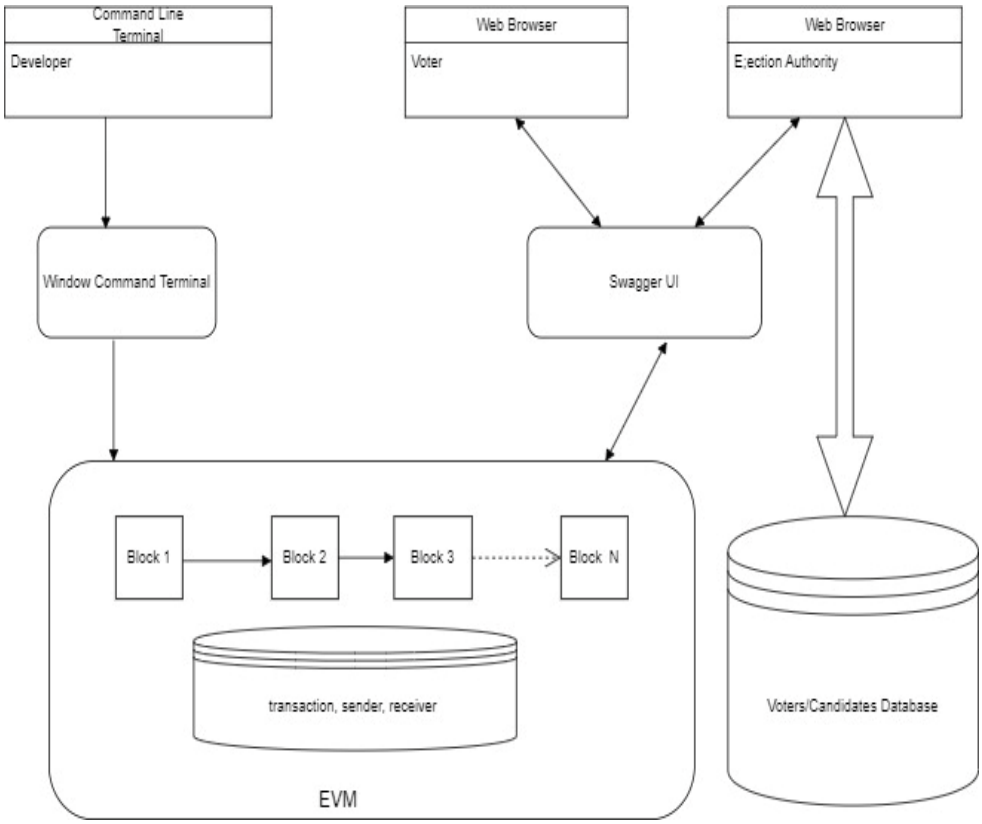

Figure v: System Architecture

The system architecture is the conceptual design of following important components:

1. Blockchain

The various blocks created during the voting transactions are the key components of our system which make our voting process secure and tamper-proof. Number of blocks

depends upon the number of transactions done in different nodes. These blocks are implemented in EVM which stores important details of sender, receiver and transaction.

2. Actors

The main actors of our system are Developer, Voter and Election Authority. Developer uses mostly the command line terminal for controlling the overall system. While Voters and Election Authorities interact with our Swagger UI from the supporting web browsers.

3. Terminal and Swagger UI

The developer mostly makes use of window terminal for operating overall workings of the system. But, the voters and election authorities, who are usually non-technical actors, interact with the Blockchain via a UI implemented using Swagger UI.

4. Database

Although the transactions are stored in Blockchain, the personal details of voters and candidates are stored in normal databases. This avoids the huge load on Blockchain to make it a little faster.

4.2 Use Case Diagram

Use case diagram shows the relation between actors, scenarios and system boundary to reach the user goal. There are basically four actors in our system: Voter, Candidate, Election Authority and Admin. The use case diagram for our project is as below.

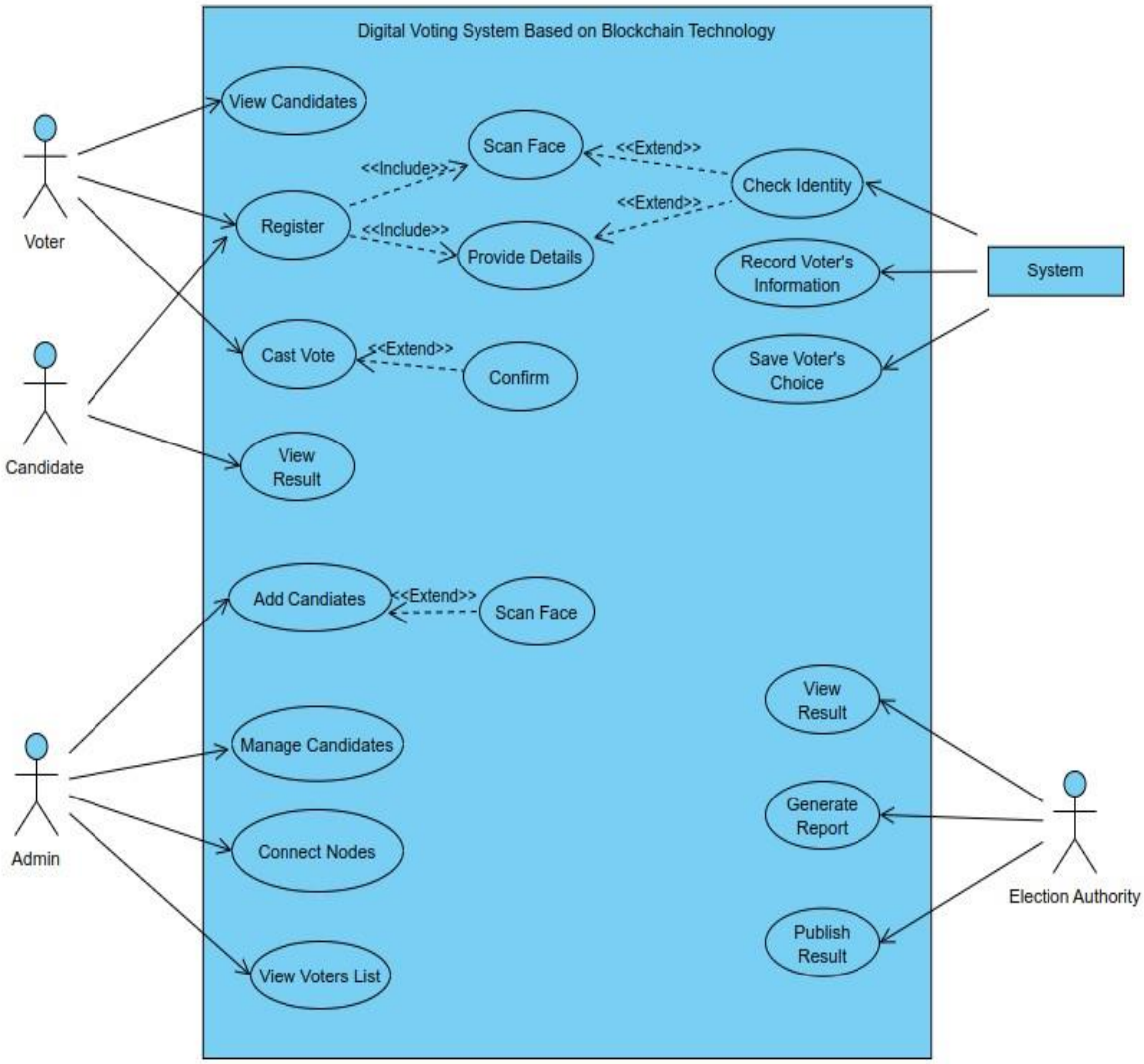

Figure vi: Use-Case Diagram of a System

4.3 Sequence Diagram

A sequence diagram shows object interactions arranged in time sequence. It depicts the objects involved in the scenario and the sequence of messages exchanged between the objects needed to conduct the functionality of the scenario.

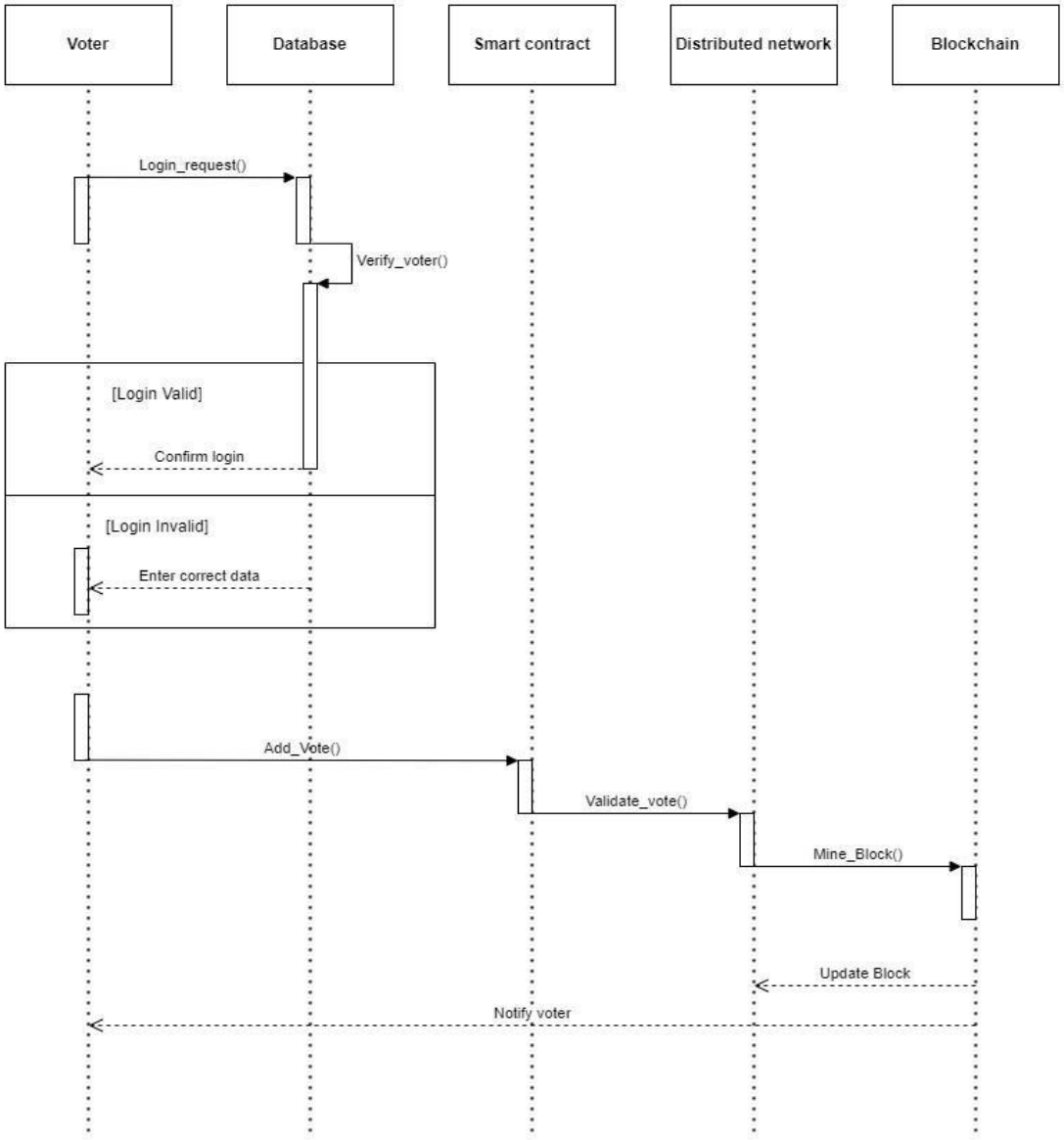

Figure v: Sequence diagram of a system

4.3 Activity Diagram

Activity diagram is basically a flowchart to represent the flow from one activity to another activity. The activity can be described as an operation of the system.

The control flow is drawn from one operation to another. This flow can be sequential, branched, or concurrent.

The activity diagrams for our project are as follows.

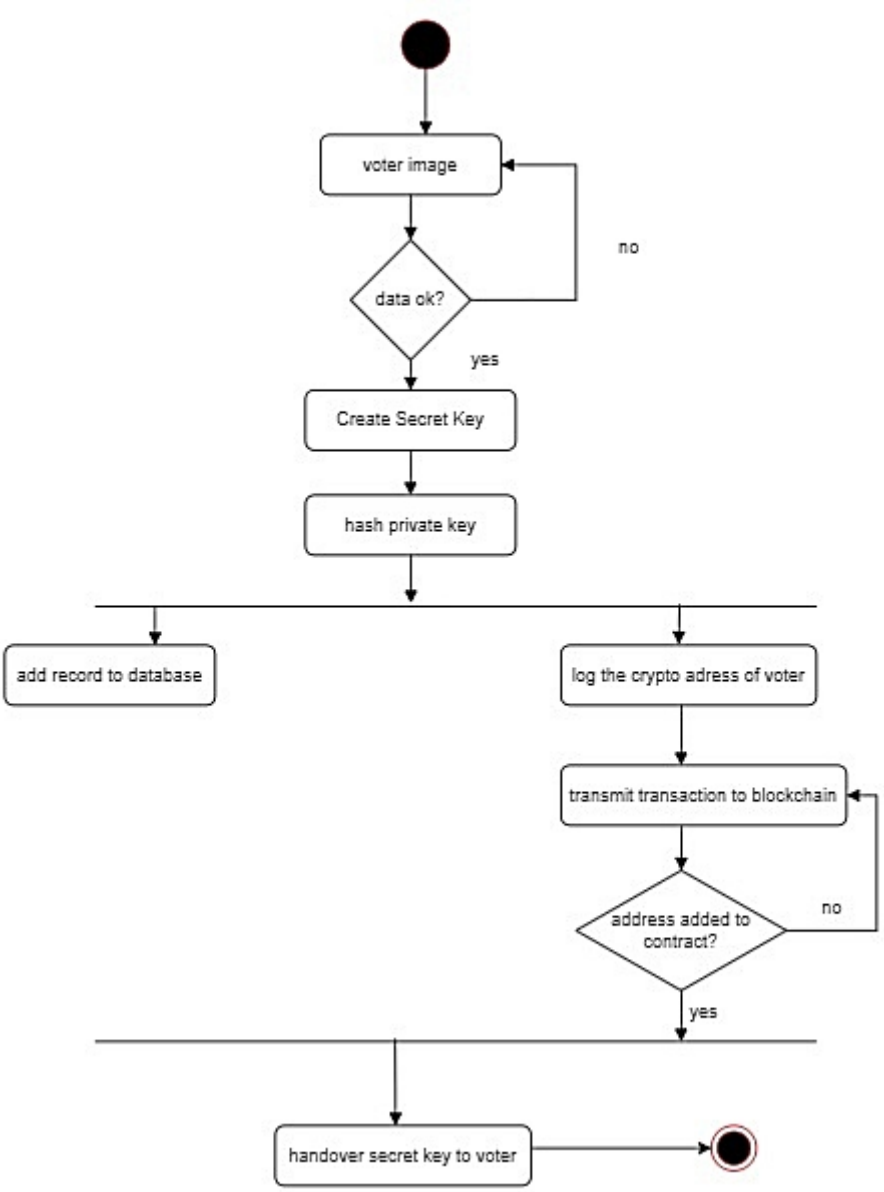

Figure vii: Activity diagram of voters' registration

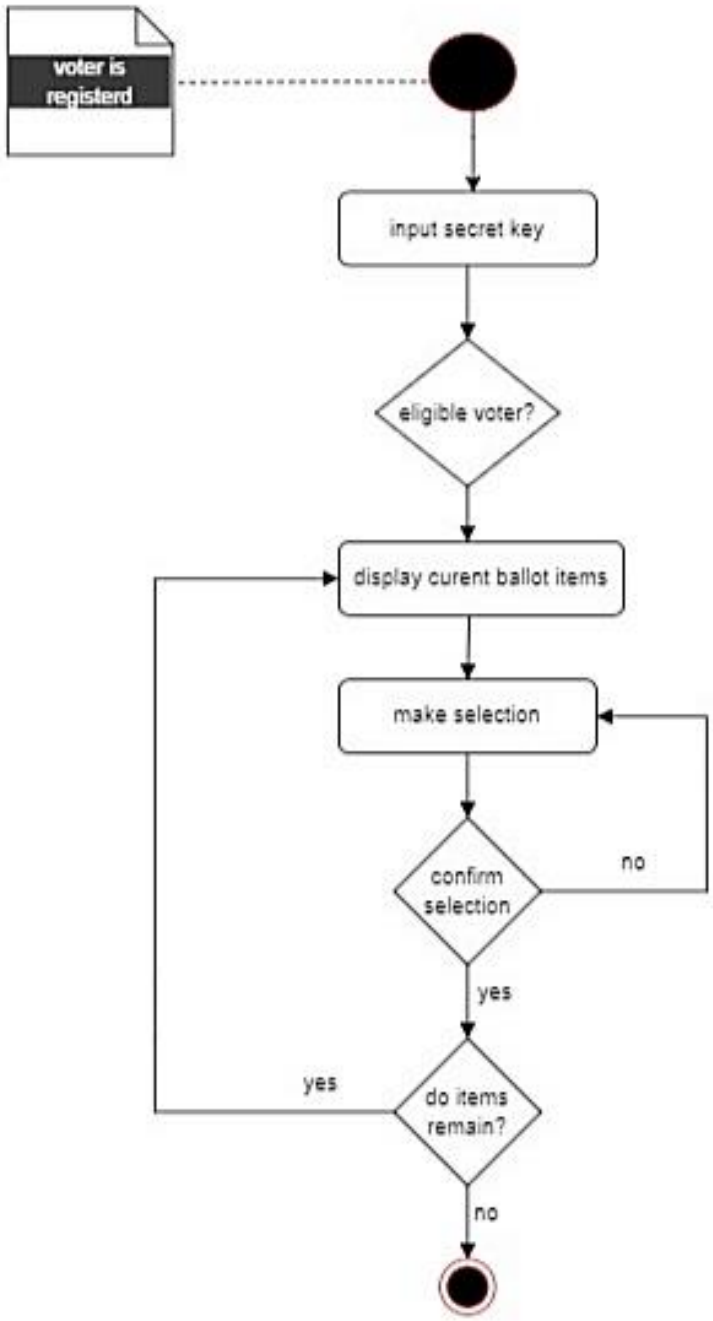

Figure viii: Activity diagram for voting process

5. Time Scheduling

The project has been followed as per requirements and time constraints involved in the table and chart below.

Task	Time Schedule					
	Week 1	**Week 2**	**Week 3**	**Week 4**	**Week 5**	**Week 6**
Requirement Analysis and Specification						
Analysis of the system						
System Design						
Procedure Requirement and Coding						
Testing and debugging						
Overall System Test						
Develop Document						

Table i: Time schedule

6. Testing

Testing is very important as it is necessary to determine whether our work is correct or not. So, we have created a test plan in which our system will be tested with various test cases. The system is tested for the normal condition.

6.1 List of Testing

All testing are given below.

1. Test 1

Purpose: To verify if a previously voted voter can vote or not.

Pre-Condition: Voters must be registered previously.

Test Data: receiver_id = 2, current_image = index.jpg, secret code = “b93556f7344914524s458668r415r6r665634e5162653v61311516”

Expected Output: This transaction will be added to Block 2.
Status: Pass

Figure x: Response of Test 1

2. Test 2

Purpose: To verify if a user can vote or not.

Pre-Condition: Voters must be registered previously. And the vote from the same account must be casted previously.

Test Data: receiver_id = 2, current_image = index.jpg, secret code = “b93556f7344914524s458668r415r6r665634e5162653v61311516”

Expected Output: Already Voted.

Status: Pass

Figure xi: Response of Test 2

3. Test 3

Purpose: Verify if a voter can vote for the candidate that is not registered.

Pre-Condition: Voters must be registered previously.

Test Data: receiver_id = 2, current_image = index.jpg, secret code = “b93556f7344914524s458668r415r6r665634e5162653v61311516”

Expected Output: Candidate not available.

Status: Pass

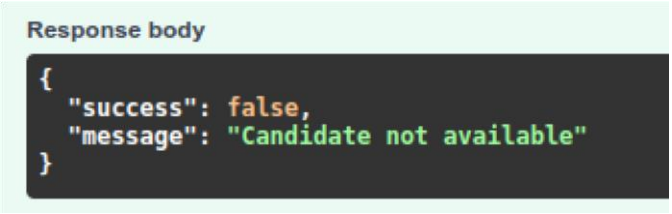

Figure xii: Response of Test 3

4. Test 4

Purpose: Verify if the voter can vote if the face doesn't match.

Pre-Condition: Voters must be registered previously.

Test Data: receiver_id = 2, current_image = index.jpg, secret code = “b93556f7344914524s458668r415r6r665634e5162653v61311516”

Expected Output: Invalid accounts.

Status: Pass

Figure xiii: Response of Test 4

7. Results and Discussion

The work on the described concept can be utilized in the development of a fully functional voting system over a blockchain network. With the immutability property and decentralized architecture of blockchain, properly implemented in the case of Online Voting System platforms, the wrangling around the voting process can be lessened tremendously. On the one hand, the inclusion of cryptography in the core architecture of blockchain the critical information of the voter and candidates results in the maintenance of anonymity while on the other hand, the public distributed ledger can be viewed by anyone on the system to verify the aftermath of the election.

A blockchain-based Online Voting System has been implemented that utilizes our python scripts to enable secure and cost-effective elections while ensuring voters' privacy. The system architecture and the design have been outlined. Compared to the naïve electronic voting system, it has been shown that blockchain technology has tremendous potential for democratic countries to advance from pen and paper schemes to a more cost-effective and time-efficient election scheme and offer new possibilities of transparency. The major stages of the system are Voter Registration, Voting, and Voting confirmation.

8. Conclusion

The concept of Online Voting System systems to make the democratic election process cheaper, faster, transparent and secured, seems more convenient. Each vote is important and should be registered as one vote can determine the fate of the election. Besides, the paper voting system results in multiple invalid votes. The Online Voting System scan also be one of the viable solutions to the problem of declining interest among the youth to participate in the election. Hence, to maintain fairness, privacy and verifiability in elections, blockchain has been implemented as the potential solution.

9. Future Enhancement/Recommendations

In this section, we discuss some possible further improvements when applying the Online Voting System protocol in special elections and scenarios.

- Real time face detection using webcam
- Biometric authentication
- As this is a design in web browser, we can further develop UI front-end which can be simulate in Voting Machine.

10. References

Nakamoto, S. Bitcoin: A Peer-to-Peer Electronic Cash System. Available online: https://bitcoin.org/bitcoin.pdf. (Accessed on 28 July 2020)

Blockchain-Enabled E-Voting Nir Kshetri and Jeffrey Voas https://ieeexplore.ieee.org/document/8405627

Trustworthy Electronic Voting Using Adjusted Blockchain Technology, BASIT SHAHZAD AND JON C ROWCROFT https://ieeexplore.ieee.org/document/8651451

Hjalmarsson, Friorik P., Gunnlaugur K. Hreioarsson, Mohammad Hamdaqa, and Gisli Hjalmtysson. "Blockchain-Based E-Voting System." In 2018 IEEE 11th International Conference on Cloud Computing (CLOUD), pp. 983-986. IEEE, 2018 https://ieeexplore.ieee.org/document/8457919/figures#figures

D. Allessie, M. Sobolewski, L. Vaccari, and F. Pignatelli, “Blockchain for digital government,” Luxembourg: Publications Office of the European Union, 2019.

A Blockchain-Based Network Security Mechanism for Voting Systems Hsin-Te Wu Department of Computer Science and Information Engineering, National Penghu University of Science and Technology https://ieeexplore.ieee.org/document/8567211/references#references

S. Haber and W. S. Stornetta, “How to time-stamp a digital document,” in Conference on the Theory and Application of Cryptography. Springer, 1990.

Blockchain technology https://www.investopedia.com/terms/b/blockchain.asp
Python learning https://docs.python.org/3/tutorial/index.html
PyCryptodome python-based library which contains basic encryption functions https://pycryptodome.readthedocs.io/en/latest/src/introduction.html

Fast API https://fastapi.tiangolo.com/