

Election Portal Using Blockchain

Project report submitted in partial
fulfilment of the requirements of the degree of

Bachelor of Engineering

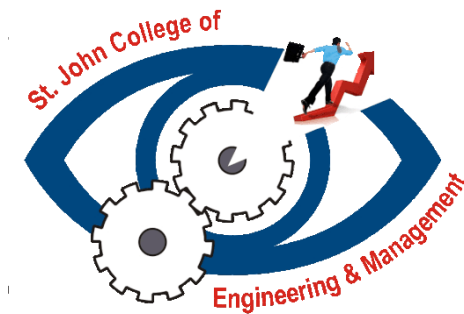
by

Shehan Shetty EU1162068

Vishal Thakur EU1162054

Adhij Vartak EU1162098

Under the guidance of
Ms. Shraddha Dabhade



Department of Computer Engineering
St. John College of Engineering and Management
University of Mumbai

2019-2020

CERTIFICATE

This is to certify that the project entitled **“Election Portal using Blockchain”** is a Project Phase I report of

“Shehan Shetty” (EU1162068)

“Vishal Thakur” (EU1162054)

“Adhij Vartak” (EU1162098)

submitted in partial fulfilment of **“Bachelor of Engineering”** in **“Computer Engineering”** as laid down by University of Mumbai during the academic year 2019-2020.

Ms. Shraddha Dabhade
Guide

Dr. Rahul Khokale
Head of Department

Dr. G. V. Mulgund
Principal

Declaration

We declare that this written submission represents our ideas in our own words and where others' ideas or words have been included, we have adequately cited and referenced the original sources. We also declare that we have adhered to all principles of academic honesty and integrity and have not misrepresented or fabricated or falsified any idea/data/fact/source in our submission. We understand that any violation of the above will be cause for disciplinary action by the Institute and can also evoke penal action from the sources which have thus not been properly cited or from whom proper permission has not been taken when needed.

.

Signature
Shetty Shehan (EU1162068)

Signature
Thakur Vishal (EU1162054)

Signature
Vartak Adhij (EU1162098)

Date:

Project Report Approval for B. E.

This project synopsis entitled ***Election Portal using Blockchain*** by ***Shetty Shehan, Thakur Vishal, Vartak Adhij*** is approved for the degree of ***Bachelor of Engineering*** in ***Computer Engineering*** from ***University of Mumbai***.

Examiners

1.-----

2.-----

Date:

Place:

Abstract

With blockchain technology steadily striving towards becoming the new system for decentralized payment schemes, it is easy to imagine why this technology can be considered an ethical liberator with regards to different application domains. Blockchain, although a relatively new concept, has gained enough popularity for applications to emerge like Bitcoin. The number of blockchain systems is steadily increasing, however the electronic voting system is very slow to adapt to changes in technology with a relatively low number of system devised so far, which introduce a fresh look on the electronic voting scene, based on our observation of the state of the art. However, these systems are impractical to use due to the limitations on the voter and candidate numbers supported, and their security framework, which highly depends on the underlying blockchain protocol and suffers from potential attacks. To deal with two aforementioned issues, the project proposes a practical platform-independent secure and verifiable voting system that can be deployed on any blockchain that supports an execution of a smart contract

Table of Contents

	Abstract	v
	List Of Figures	viii
Chapter 1	Introduction	1
	1.1 Introduction	1
	1.2 Motivation	1
	1.3 Problem Statement	2
	1.4 Scope	2
	1.5 Objectives	2
Chapter 2	Review of Literature	3
	2.1 E-Voting System using Blockchain Technology for Distributed Environments	3
	2.2 Blockchain-Based E-Voting System	5
	2.3 BroncoVote: Secure Voting System using Ethereum	7
Chapter 3	Requirement Analysis	9
	3.1 Functional Requirements	9
	3.2 Non-Functional Requirements	9
	3.3 Minimum Hardware/Software Requirements	10
	3.3.1 Hardware Requirements	10
	3.3.2 Software Requirements	10
Chapter 4	Design	11
	4.1 ER Diagram for Election Portal Using Blockchain	11
	4.2 DFD Diagram for Election Portal Using Blockchain	13
	4.3 Use Case Diagram for Election Portal Using Blockchain	14
	4.4 Class Diagram for Election Portal Using Blockchain	15
	4.5 Sequence Diagram for Election Portal Using Blockchain	16

	4.6 Activity Diagram for Election Portal Using Blockchain	17
	4.7 Collaboration Diagram for Election Portal Using Blockchain	18
Chapter 5	Report on the Present Investigation	19
	5.1 Methodology	19
	5.1.1 System Architecture for Election Portal Using Blockchain	20
	5.2 Implementation	21
Chapter 6	Results and Discussions	24
Chapter 7	Conclusion	26
	References	27
Appendix	Technologies Used	28
	Acknowledgement	31

List of Figures

Figure No.	Figure Name	Page No.
2.1	System Architecture for E-Voting System using Blockchain Technology for Distributed Environment	4
2.2	System Architecture for Blockchain-Based E-Voting System	6
2.3	System Architecture BroncoVote: Secure Voting System using Ethereum's Blockchain	8
4.1	ER Diagram for Election Portal Using Blockchain	12
4.2	DFD Diagram for Election Portal Using Blockchain	13
4.3	Use Case Diagram for Election Portal Using Blockchain	14
4.4	Class Diagram for Election Portal Using Blockchain	15
4.5	Sequence Diagram for Election Portal Using Blockchain	16
4.6	Activity Diagram for Election Portal Using Blockchain	17
4.7	Collaboration Diagram for Election Portal Using Blockchain	18
5.1	System Architecture for Election Portal Using Blockchain	20
5.2	Sample Screenshot of Voter Organisation Creation	21
5.3	Sample Screenshot of Associate Identity	21
5.4	Sample Screenshot of Register User	22
5.5	Sample Screenshot of Add Peer	22
5.6	Sample Screenshot of Add Certificate Authority	23
5.7	Sample Screenshot of Ordering Service	23
6.1	Sample Screenshot of Registration	24
6.2	Sample Screenshot of Voting Process	25
6.3	Sample Screenshot of Counting Polls	25

Chapter 1

Introduction

In today's world, widespread mistrust towards the government and interference in countries' processes by external actors have made the democratic process of voting more critical than ever. Democratic countries have been experiencing dictatorial regimes which have introduced widespread terror among their people. People have had their human rights violated and their fundamental freedoms provided by their constitution taken away. In such an atmosphere, having a fair and transparent election is something that is paramount for the freedom most people enjoy today. People are supposed to trust the result which is provided by an Election commission or a government body. This makes the process of counting, a major vulnerability in the current process. There are also other major electoral scams such as voter fraud, ballot stuffing and booth capturing. The system that is being proposed solves most of the issues mentioned above and can be implemented in the current world environment.

1.1 Motivation

The pitfalls of the current system of ballot voting are being taken advantage of by people or organizations looking to gain power. These instances of controversial elections could all have

been avoided if the counting process was fair, transparent and verifiable. The current ballot system does offer anonymity to the voter but the counting process is not transparent. In order to overcome all the problems this project is being implemented

1.2 Problem Statement

The main goal of this project is to overcome all the possible fraud problems present in current paper ballot as well as E-Voting system. In normal E-voting there are clients, servers and a separate database(Centralised System).But blockchain technology incorporates the client, server, database in itself (Decentralized System).With the help of blockchain technology the possibility of data tampering or data leakage is completely eliminated. The final objective is to hold a fair election in which each and every participant gets closure after the complete process

1.3 Objectives

The objectives are as follows:

- To make a complete decentralized system
- To generate unique ID for each person taking part in the elections
- To avoid double voting
- To perfectly count all votes and give a fair result

1.4 Scope

- To develop a system using Blockchain JS
- To implement high security aspects
- To invoke a login system that would help in enhancing the security of the system

Chapter 2

Review Of Literature

This chapter gives details about the literature papers based on the system statement. The chapter also contains advantages and disadvantages of the research which are considerably giving an overview of the research paper

2.1 E-Voting System using Blockchain Technology for Distributed Environment

The research paper studies about how an e-voting system must be secure, as it should not allow duplicated votes and be fully transparent, while protecting the privacy of the attendees. The disadvantages of traditional voting system are that there is no reliability of voting. No assurances that people gave the votes are not changed before they are counted on the system. There is no transparency between the voter and the system. So, to overcome all these issues the project proposes to use block chain technology as a medium in the voting system. The objective of such a scheme would be to provide a decentralized architecture to run and support a voting scheme that is open, fair, and independently verifiable.

In this, the paper propose a potential new e-voting protocol that utilizes the block chain as a transparent ballot box. Therefore, there would be more transparency between the user and the system.[2]

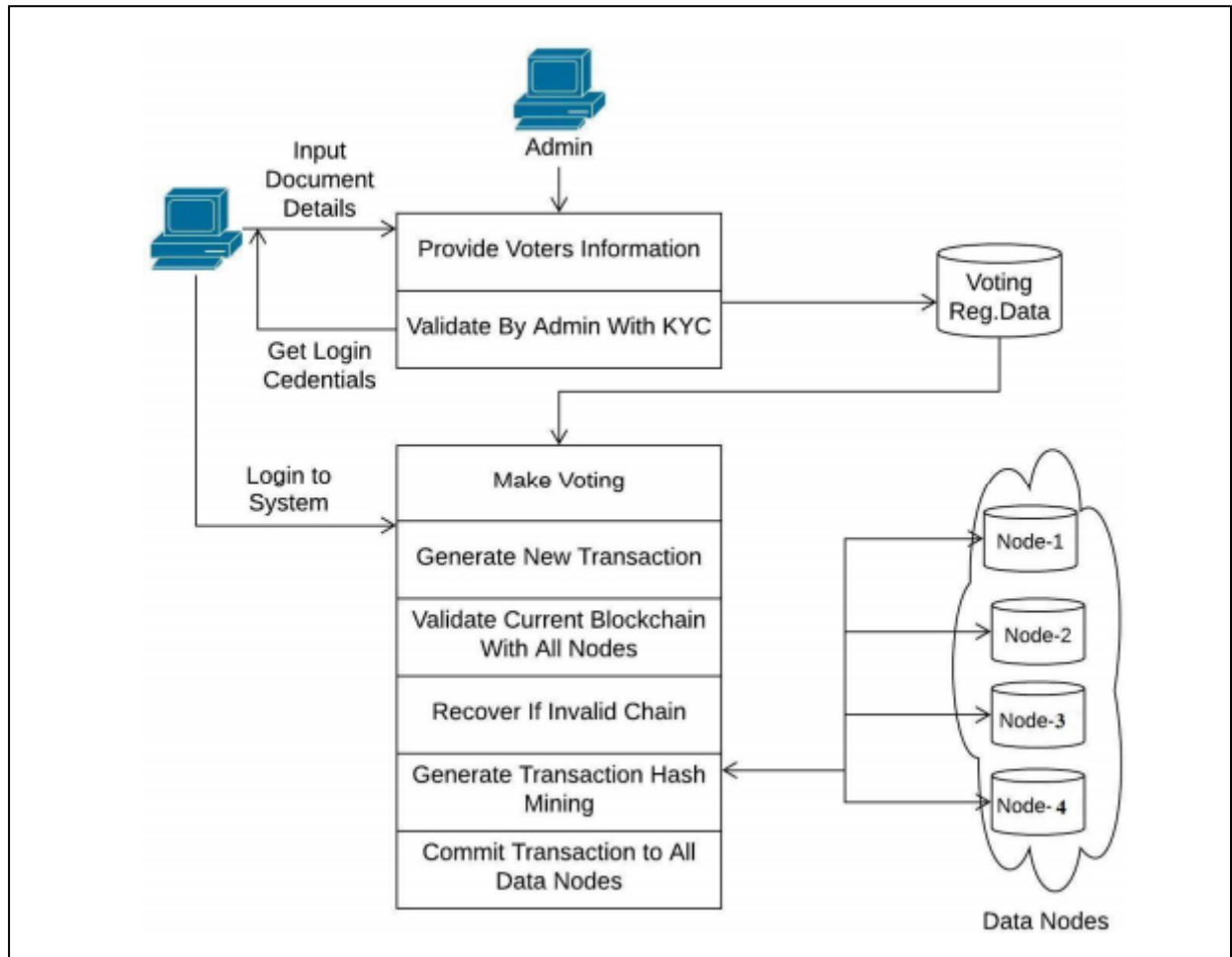


Fig 2.1 System Architecture for E-Voting System using Blockchain Technology for Distributed Environment [2]

There are many research directions in applying Blockchain technology to the voting industry due to the complexity of this domain and the need for more robust and effective information technology systems. An interoperable architecture would undoubtedly play a significant role throughout many voting use cases that face similar data sharing and communication challenges.

2.2 Blockchain-Based E-Voting System

The research paper studies about how building a secure electronic voting system that offers the fairness and privacy of current voting schemes, while providing the transparency and flexibility offered by electronic systems has been a challenge for a long time. In this work-in-progress paper, an application of blockchain as a service to implement distributed electronic voting systems. The paper proposes a novel electronic voting system based on blockchain that addresses some of the limitations in existing systems and evaluates some of the popular blockchain frameworks for the purpose of constructing a blockchain-based e-voting system. In particular, they evaluate the potential of distributed ledger technologies through the description of a case study; namely, the process of an election, and the implementation of a blockchain based application, which improves the security and decreases the cost of hosting a nationwide election. The blockchain is an append-only data structure, where data is stored in a distributed ledger that cannot be tampered with or deleted. This makes the ledger immutable. The blocks are chained in such a way that each block has a hash that is a function of the previous block, and thus by induction the complete prior chain, thereby providing assurance of immutability.[4]

Electronic voting systems have been the subject of active research for decades, with the goal to minimize the cost of running an election, while ensuring the election integrity by fulfilling the security, privacy and compliance requirements. Replacing the traditional pen and paper scheme with a new election system has the potential to limit fraud while making the voting process traceable and verifiable. This paper evaluates the use of blockchain as a service to implement an electronic voting (e-voting) system. The paper makes the following original contributions: it proposes a novel electronic voting system based on blockchain that addresses some of the limitations in existing systems and evaluates some of the popular blockchain frameworks for the purpose of constructing a blockchain-based e-voting system. In particular, user evaluates the potential of distributed ledger technologies through the description of a case study; namely, the process of an election, and the implementation of a blockchain-based application, which improves the security.

The permissioned blockchain”, and review of existing blockchain frameworks suited for constructing blockchain-based e-voting system. The authentication process will be very tough to bypass for potential hackers since every blockchain system makes use of strong ‘smart contracts’.

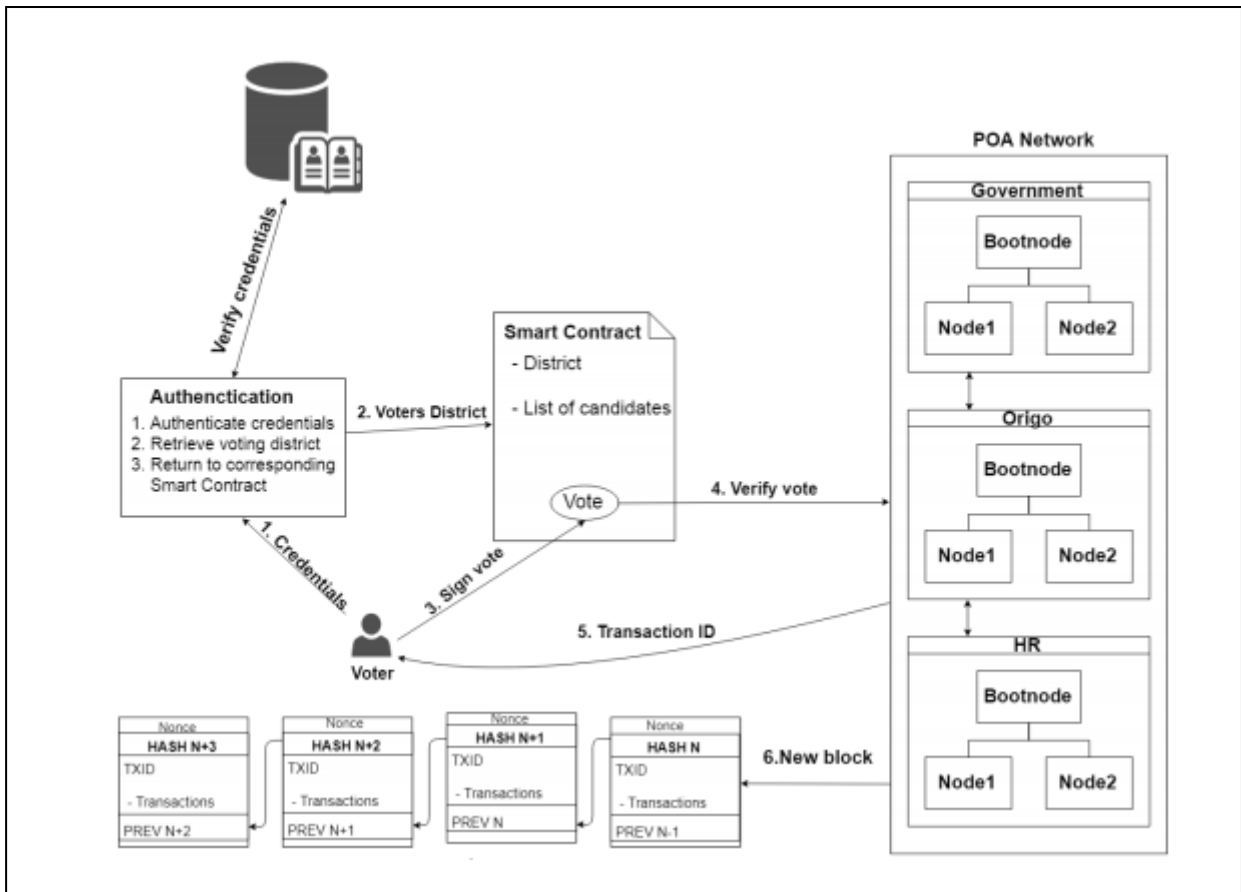


Fig 2.2 System Architecture of Blockchain-Based E-Voting System[4]

As explained at the beginning of this section, in order to satisfy the privacy, security and transparency requirements for e-voting and to ensure that the election system should not enable coerced voting, in the system, users are using a private (permissioned) blockchain for setting up blockchain infrastructure, where the smart contracts are deployed. In this paper, a blockchain-based electronic voting system that utilizes smart contracts to enable secure and cost-efficient election while guaranteeing voters privacy is introduced. It is shown that the blockchain technology offers a new possibility to overcome the limitations and adoption barriers of electronic voting systems which ensures the election security and integrity and lays the ground for transparency.

2.3 BroncoVote: Secure Voting System using Ethereum's Blockchain

The research paper studies about how expanding e-voting into blockchain technology could be the solution to alleviate the present concerns in e-voting. In this paper, the system proposes a blockchain-based voting system, named BroncoVote, that preserves voter privacy and increases accessibility, while keeping the voting system transparent, secure, and cost-effective. BroncoVote implements a university-scaled voting framework that utilizes Ethereum's blockchain and smart contracts to achieve voter administration and auditable voting records. In addition, BroncoVote utilizes a few cryptographic techniques, including homomorphic encryption, to promote voter privacy. The implementation was deployed on Ethereum's Testnet to demonstrate usability, scalability, and efficiency.

The implemented system, BroncoVote, provides a secure and private e-voting system that is also easily accessible. BroncoVote is a university scale voting system that utilizes smart contracts in Ethereum and Paillier Homomorphic Encryption[8] to achieve our goals. The system also allows for different types of ballots: users have the freedom to create polls or elections as well as have the option to choose who can vote on their ballot. BroncoVote provides voter privacy on all the ballots by encrypting every vote, homomorphically tallying, and revealing the vote count using Paillier cryptosystem decryption process. To maintain data integrity, all ballot and voting data is publicly available as part of the smart contracts or blockchain in our system.

Preceding the introduction to voting system, it merits mentioning that the Ethereum protocol utilized as part of system has not been modified in any way. The system, BroncoVote, uses existing functionality and features provided by Ethereum to provide the ability for creating and voting on ballots. The implementation consists of three smart contracts coded in Ethereum's Solidity language, two scripts written in JavaScript[5], one solidity[9] and one Html5[1] page. The implementation in the paper is such that it will not override other systems as it is very high functioning. The paper proposes a novel electronic voting system based on blockchain that addresses some of the limitations in existing systems and evaluates some of the popular blockchain frameworks for the purpose of constructing a blockchain-based networks.

BroncoVote is an open source project and the entirety of the code is available for public use .It has been implemented in various college and mini election campaigns. [6]

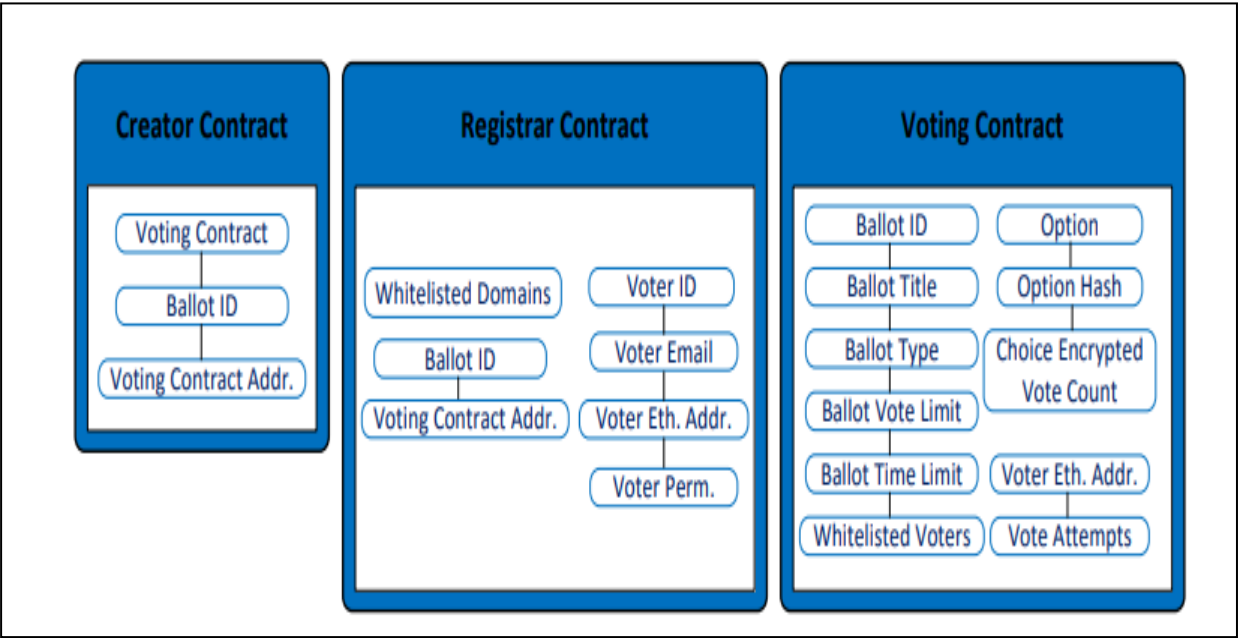


Fig 2.3 System Architecture of BroncoVote: Secure Voting System using Ethereum's Blockchain[6]

With the deployment of the system on the testnet for experiments it showed that the system can easily be deployed and setup to use as a voting system for universities or other similar settings. For some, blockchain promises a way to improve security and make voting more accessible. It is believed blockchain only adds vulnerabilities to an already weak system. Without a papertrail, electronic voting machines cast a cloud of doubt over whether the voting record could be manipulated. In future work, the authors will investigate the possibility of implementing Paillier cryptosystem as a library in Solidity. With the current system, moving the cryptography to a library in Solidity could largely improve individual ballot verifiability. This will help the system to achieve individual voter audit on different ballots without compromising the other ballots. To increase user accessibility, system will also look into integrating the Ethereum. Finally, to help with voter verification, system will try to integrate an API/process that will allow to check the validity of all e-mails used to register into the system.

Chapter 3

Requirement Analysis

3.1 Functional Requirements

In the development of this project, some of the functional requirements could be:

- The system should be able to display voting transactions of a particular block
- The system should generate report about unauthorised transaction and detection of same
- The system should be able to give general details of voter by a user in a particular block
- The system should generate digital signature about a particular block
- The system should be able to generate list of candidates and voters
- The system should be able to digitally verify ownership of a particular block

3.2 Non-Functional Requirements

In the development of this project, some of the non-functional requirements could be:

- The application should be easy to use by people above the age of 18
- The application should be accessible to all the people using it
- The application should allow several transactions to be made at the same time without downgrading performance
- To deny uneligible voters
- To use all cost of system

3.3 Minimum Hardware/Software Requirements

3.3.1 Hardware Requirements

- 4GB RAM
- i3 5th Generation and Above
- Keyboard
- Mouse
- Strong Internet Connection

3.3.2 Software Requirements

- HTML5
- PHP
- MySql
- Javascript
- JSON
- Mozilla Firefox/Google Chrome

Chapter 4

Design

4.1 ER Diagram for Election Portal Using Blockchain

The ER or (Entity Relational Model) is a high-level conceptual data model diagram. Entity-Relation model is based on the notion of real-world entities and the relationship between them. ER modelling helps to analyze data requirements systematically to produce a well-designed database. It is considered a best practice to complete ER modelling before implementing your database. Entity relationship diagram displays the relationships of entity set stored in a database. In other words, ER diagrams help the user to explain the logical structure of databases. At first look, an ER diagram looks very similar to the flowchart.

Facts about ER Diagram Model:

- ER model allows you to draw Database Design
- It is an easy to use graphical tool for modelling data
- Widely used in Database Design
- It is a GUI representation of the logical structure of a Database
- It helps you to identifies the entities which exist in a system and the relationships between those entities

An entity can be place, person, object, event or a concept, which stores data in the database. The characteristics of entities are must have an attribute, and a unique key. Every entity is made up of some 'attributes' which represent that entity.

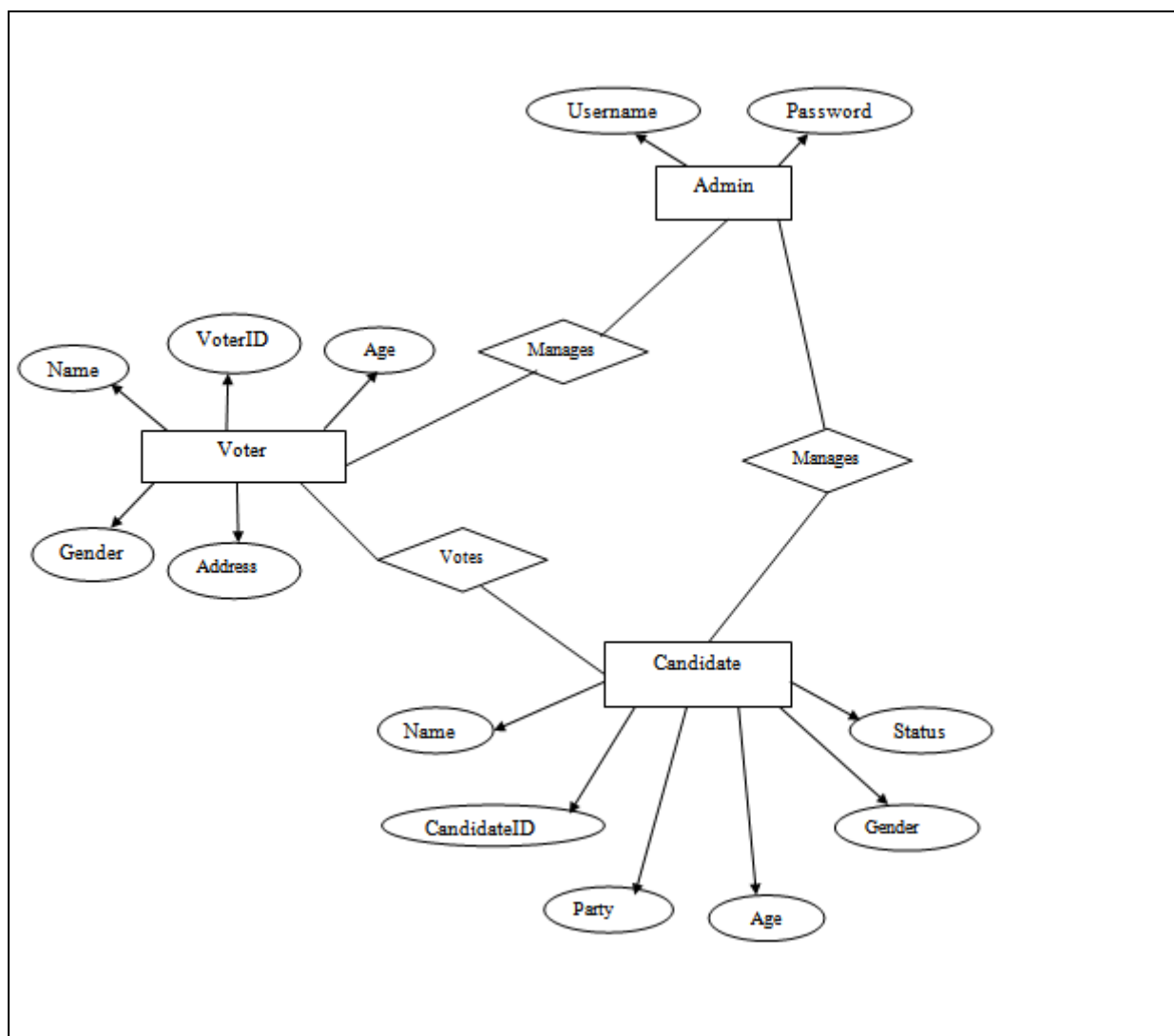


Fig 4.1 ER Diagram Of Election Portal Using Blockchain

ER Diagram depicts the relation between all the entities present in the system. Each entity in the system is related to one another but the relation between each entity is one to one. Admin is the main entity of the system and overlooks most of the process in the system. The relation can be one to many if there are multiple admins ,but this is an extremely rare case and doesn't follow the "ethics" of ethical voting

4.2 DFD Diagram for Election Portal Using Blockchain

A data flow diagram (DFD) maps out the flow of information for any process or system. It uses defined symbols like rectangles, circles and arrows, plus short text labels, to show data inputs, outputs, storage points and the routes between each destination. Data flowcharts can range from simple, even hand-drawn process overviews, to in-depth, multi-level DFDs that dig progressively deeper into how the data is handled. It can be used to analyze an existing system or model a new one. Like all the best diagrams and charts, a DFD can often visually “say” things that would be hard to explain in words, and it works for both technical and nontechnical audiences, from developer to CEO.

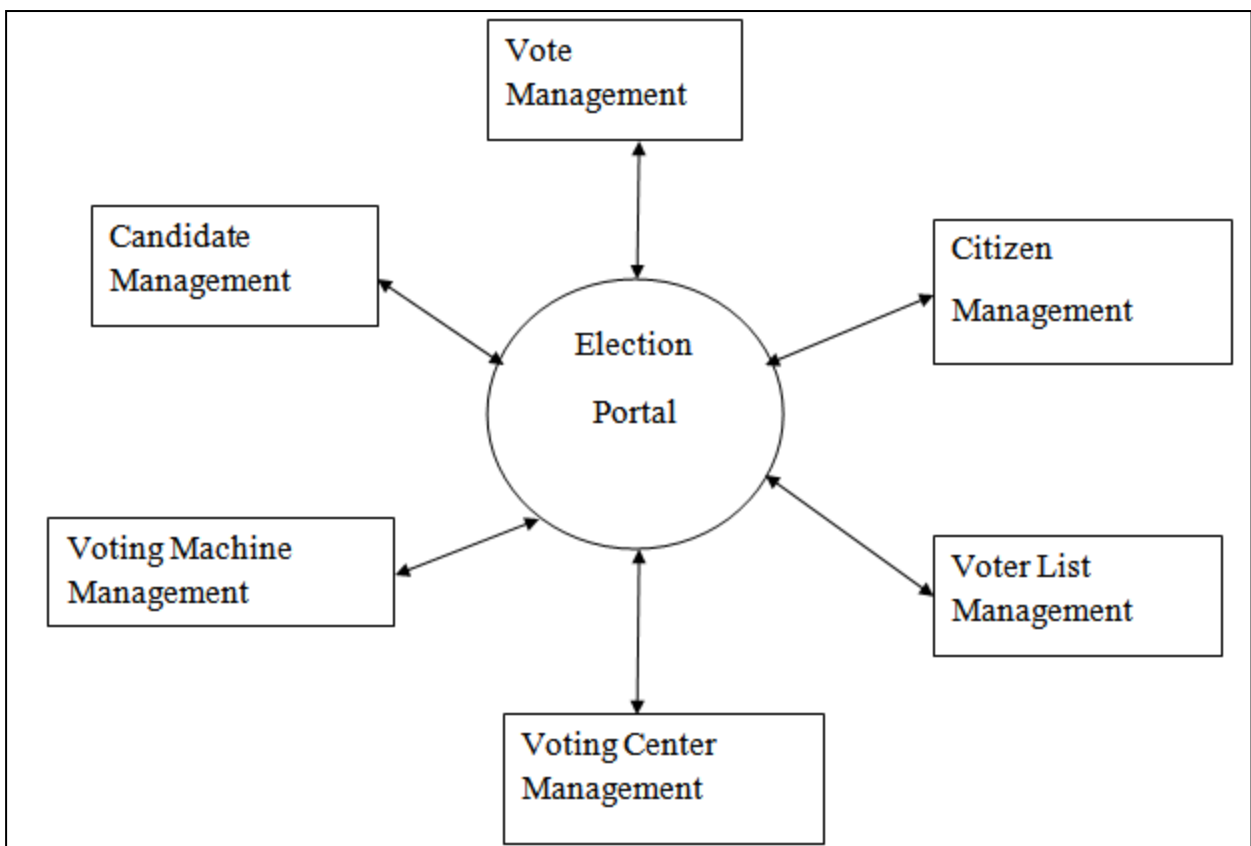


Fig 4.2 Data Flow Diagram for Election Portal Using Blockchain

The election portal is the ‘brain’ of the system and manages all the data flow within the system. The flow of data can be in any direction since data is updated on a regular basis. Any failure in updating data in one entity can lead to errors in subsequent entities

4.3 Use Case Diagram for Election Portal Using Blockchain

A use case diagram can summarize the details of your system's users (also known as actors) and their interactions with the system. To build one, you'll use a set of specialized symbols and connectors. An effective use case diagram can help your team discuss and represent:

- Scenarios in which your system or application interacts with people, organizations, or external systems
- Goals that your system or application helps those entities (known as actors) achieve
- The scope of your system

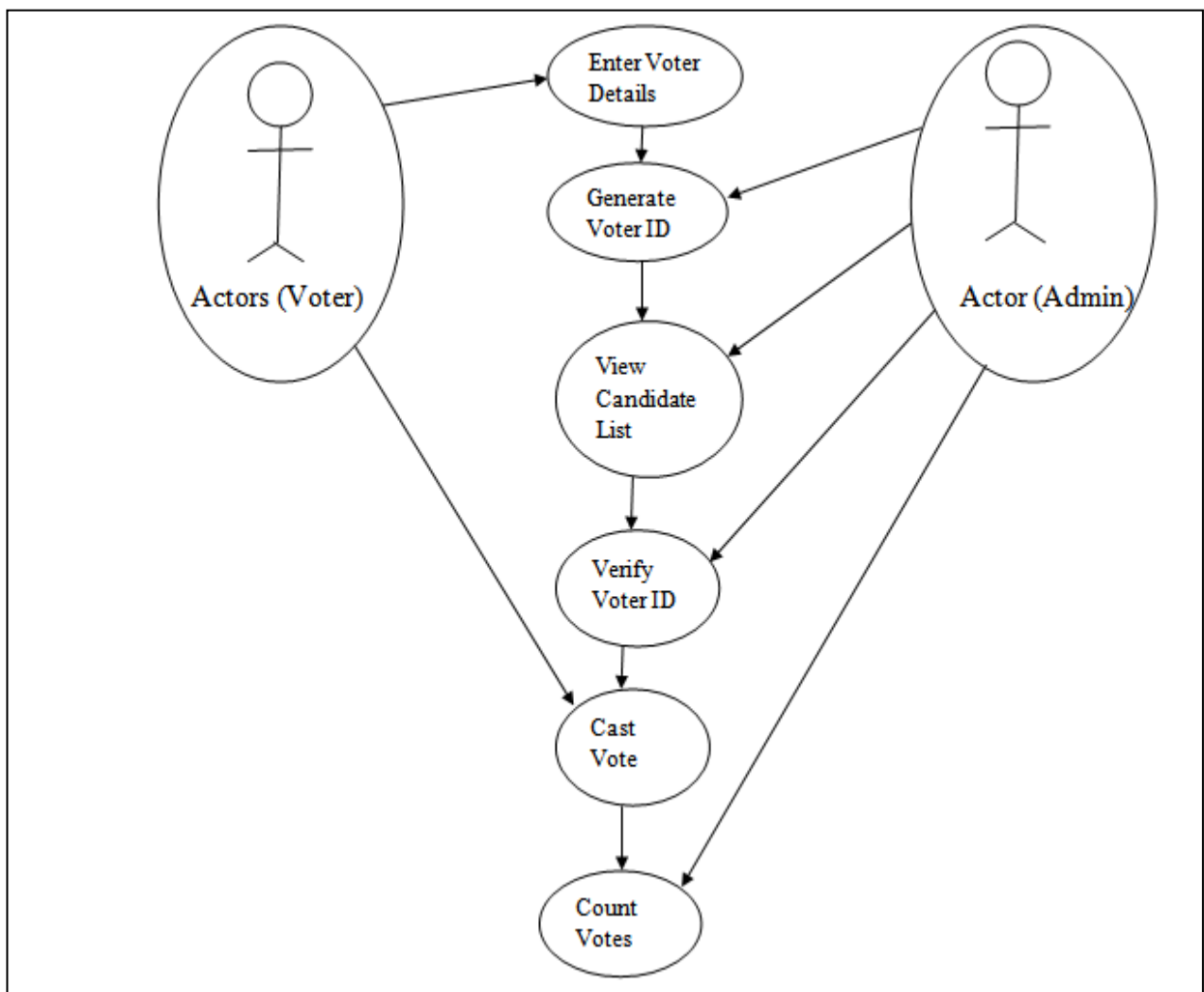


Fig 4.3 Use Case Diagram for Election Portal Using Blockchain

4.4 Class Diagram for Election Portal Using Blockchain

Class diagram is a static diagram. It represents the static view of an application. Class diagram is not only used for visualizing, describing, and documenting different aspects of a system but also for constructing executable code of the software application. Class diagram describes the attributes and operations of a class and also the constraints imposed on the system. The class diagrams are widely used in the modelling of object oriented systems because they are the only UML diagrams, which can be mapped directly with object-oriented languages.

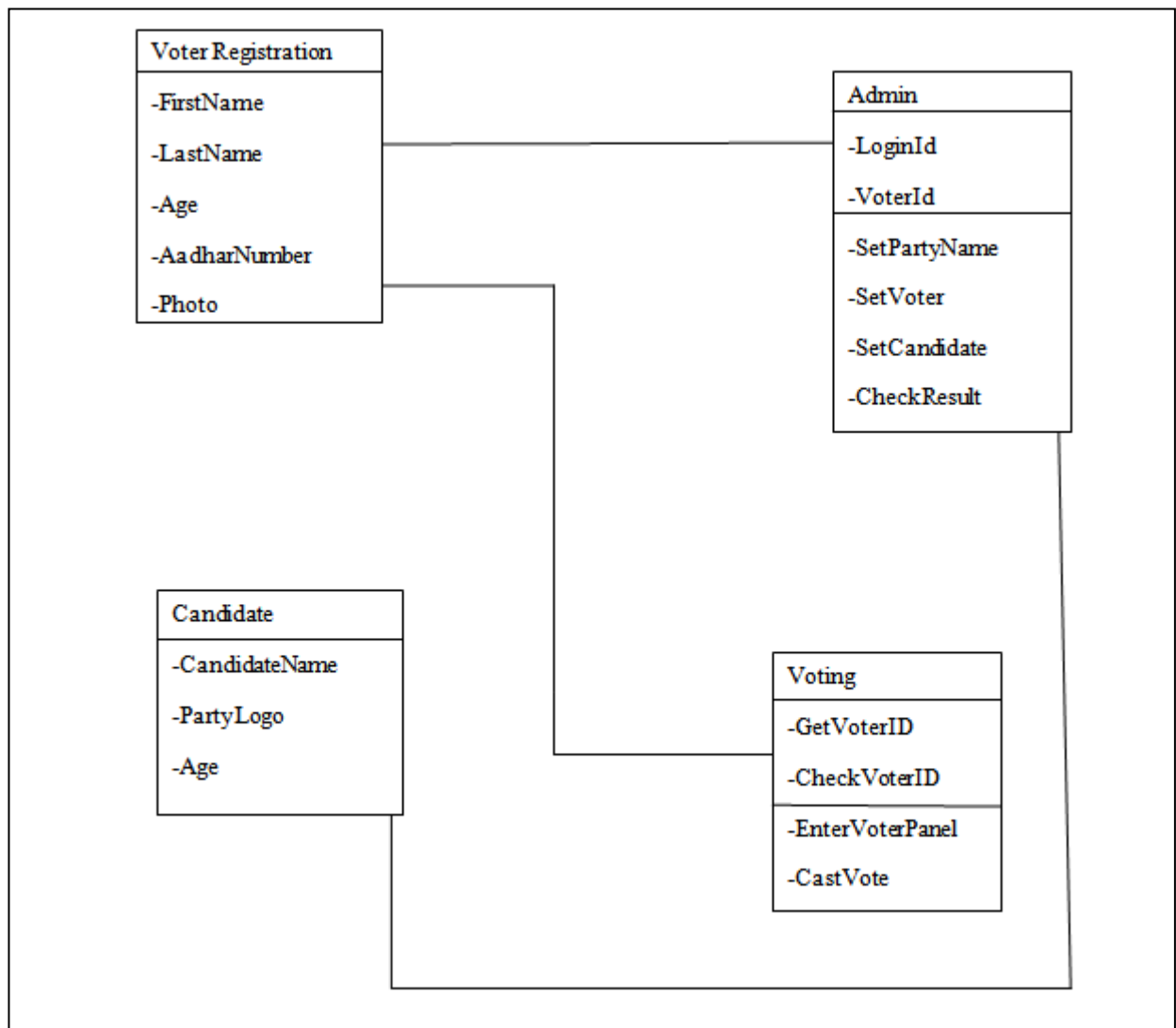


Fig 4.4 Class Diagram for Election Portal Using Blockchain

4.5 Sequence Diagram for Election Portal Using Blockchain

A sequence diagram simply depicts interaction between objects in a sequential order i.e. the order in which these interactions take place. User can also use the terms event diagrams or event scenarios to refer to a sequence diagram. Sequence diagrams describe how and in what order the objects in a system function. These diagrams are widely used by businessmen and software developers to document and understand requirements for new and existing systems.

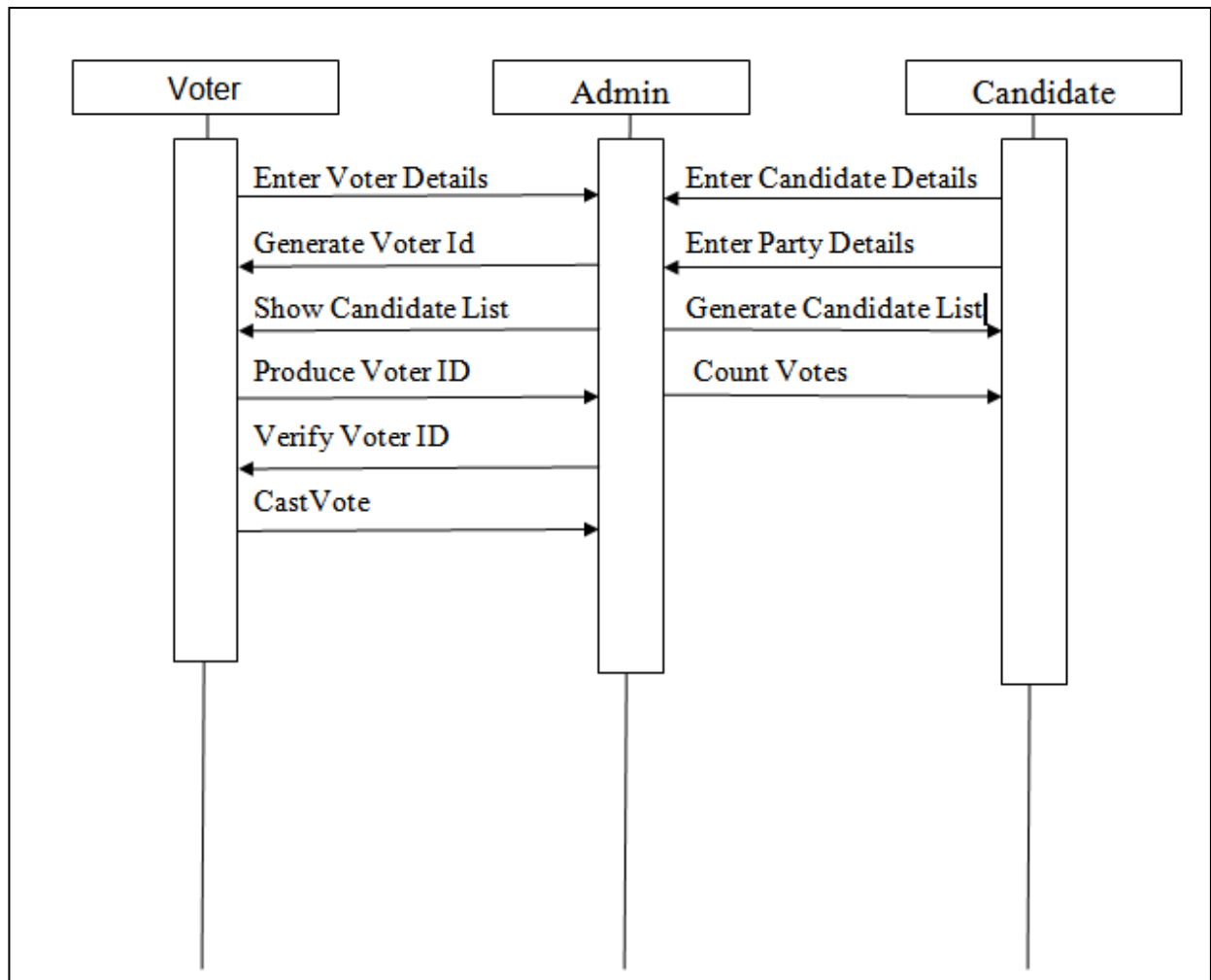


Fig 4.5 Sequence Diagram for Election Portal Using Blockchain

The basic process in which the system performs its activities are similar for all three functions. The addition of the third process is that system has to establish a two-way communication between voter and candidate.

4.6 Activity Diagram for Election Portal Using Blockchain

Activity diagram is another important diagram in UML to describe the dynamic aspects of the system. 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. Activity diagrams deal with all type of flow control by using different elements such as fork, join, etc

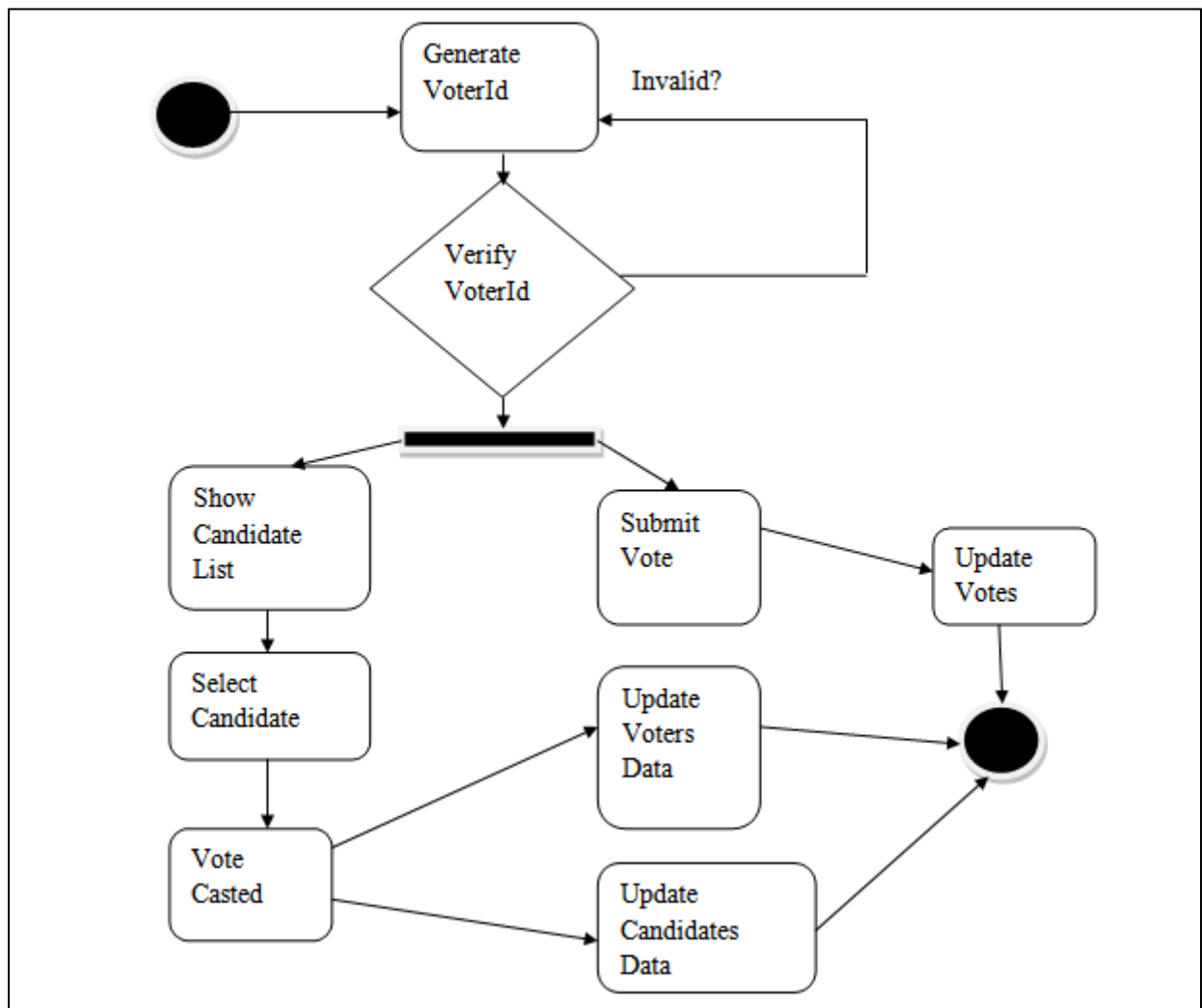


Fig 4.6 Activity Diagram for Election Portal Using Blockchain

Activity diagrams are mainly used as a flowchart that consists of activities performed by the system. Activity diagrams are not exactly flowcharts as they have some additional capabilities. These additional capabilities include branching, parallel flow, swimlane etc.

4.7 Collaboration Diagram for Election Portal Using Blockchain

A collaboration diagram, also known as a communication diagram, is an illustration of the relationships and interactions among software objects in the Unified Modelling Language. These diagrams can be used to portray the dynamic behaviour of a particular use case and define the role of each object. Collaboration diagrams are created by first identifying the structural elements required to carry out the functionality of an interaction. A model is then built using the relationships between those elements. Several vendors offer software for creating and editing collaboration diagrams.

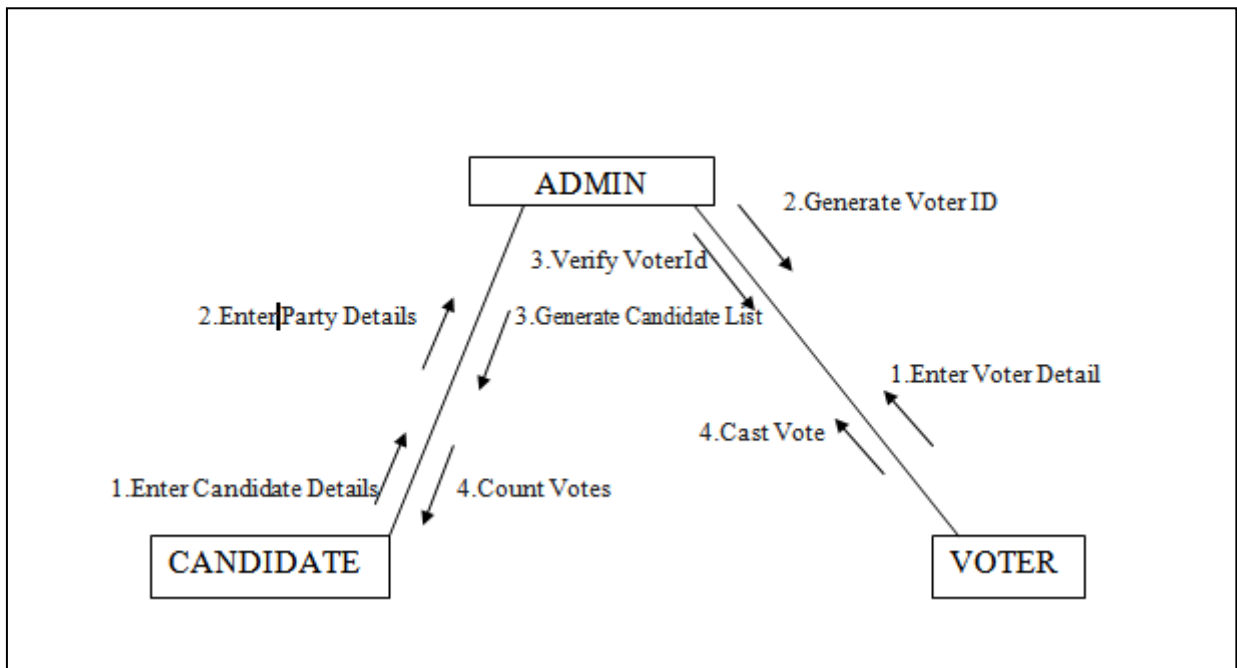


Fig 4.6 Collaboration Diagram For Election Portal Using Blockchain

A collaboration diagram resembles a flowchart that portrays the roles, functionality and behaviour of individual objects as well as the overall operation of the system in real time. The most important objects are placed in the center of the diagram, with all other participating objects branching off. After all objects are placed, links and messages should be added in between.

Chapter 5

Report on the Present Investigation

5.1 Methodology

At the start of the application, the user registers to vote by providing their drivers license number, registrar district, and first and last name. In this step, user can check to see if the drivers license is valid, and has not been registered previously. If all goes well, user create a private and public key for the voter with the certificate authority that is running on the cloud, and add those keys to the wallet. After that, drivers license number is used to submit the vote, during which the application checks if this drivers license number has voted before and tells the user they have already submitted a vote if so. If all goes well, the political party which the voter has chosen is given a vote, and the world state is updated. The application then updates the current standings of the election to show how many votes each political party currently has. Since each transaction that is submitted to the ordering service must have a signature from a valid public-private key pair, admin can trace back each transaction to a registered voter of the application, in the case of an audit. In conclusion, although this is a simple application, the developer can see how they can implement a Hyperledger Fabric web-app to decrease the chance of election-meddling, and enhance a voting application by using blockchain technology.

5.1.1 System Architecture for Election Portal Using Blockchain

The Fig 5.1 show the architecture of the system. The user needs an account with a wallet address with some Ether, Ethereum's cryptocurrency. Once connected to the network, user can cast their vote and pay a small transaction fee to write this transaction to the blockchain. The blockchain operator sets up the IBM Blockchain Platform[11] 2.0 service. The IBM Blockchain Platform 2.0 creates a Hyperledger Fabric network[10] on an IBM Kubernetes Service[12], and the operator installs and instantiates the smart contract on the network. The Node.js application server uses the Fabric SDK[14] to interact with the deployed network on IBM Blockchain Platform 2.0 and creates APIs[13] for a web client. The Vue.js client uses the Node.js application API to interact with the network. The user interacts with the Vue.js web interface to cast their ballot and query the world state to see current poll standings.

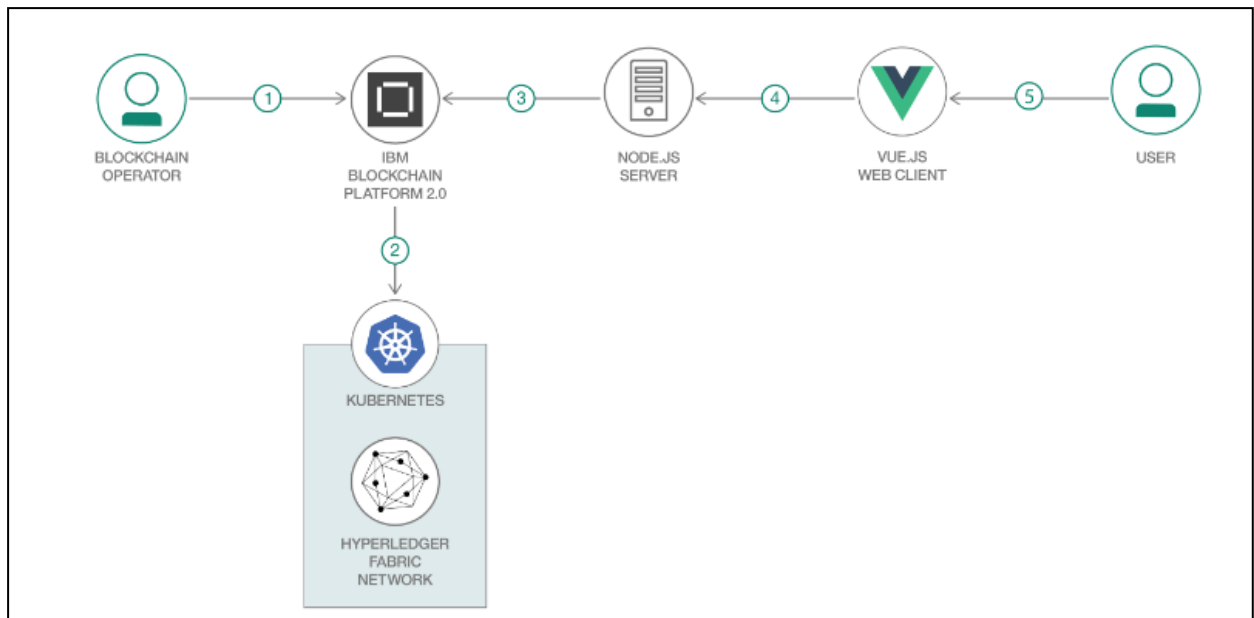


Fig 5.1 System Architecture For Election Portal Using Blockchain

IBM Blockchain Platform gives admin total control of your blockchain network with a user interface that can simplify and accelerate your journey to deploy and manage blockchain components on the IBM Cloud Kubernetes Service. Whereas the Kubernetes Service creates a cluster of compute hosts and deploys highly available containers. A Kubernetes cluster lets admin securely manage the resources that you need to quickly deploy, update, and scale applications

5.2 Implementation

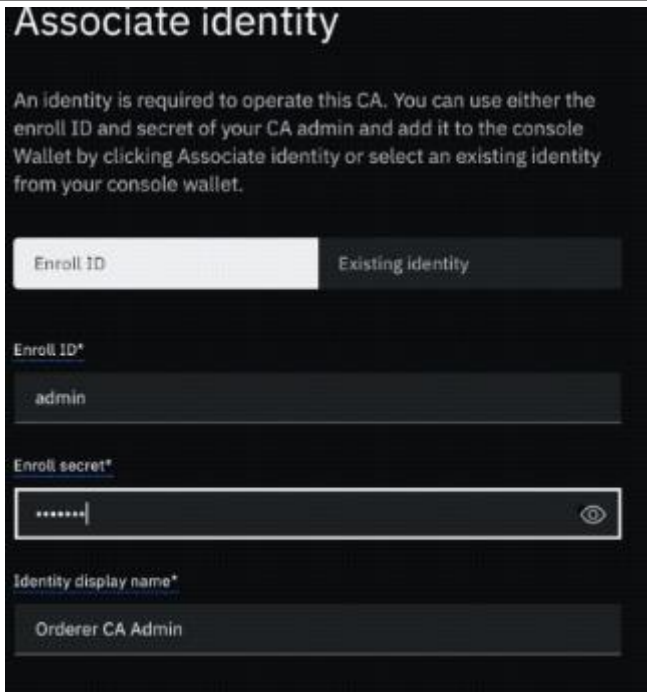
In Fig 5.2, a VoterOrganisation is created on the IBM Blockchain network. Admin privileges are given to the admin of the entire organisation



The screenshot shows a web form titled "Add Certificate Authority". It contains three input fields: "CA display name*" with the value "Org1 CA", "CA administrator enroll ID*" with the value "admin", and "CA administrator enroll secret*" with a masked password "*****". A toggle icon is visible on the right of the password field.

Fig 5.2 Sample Screenshot of Voter Organisation Creation

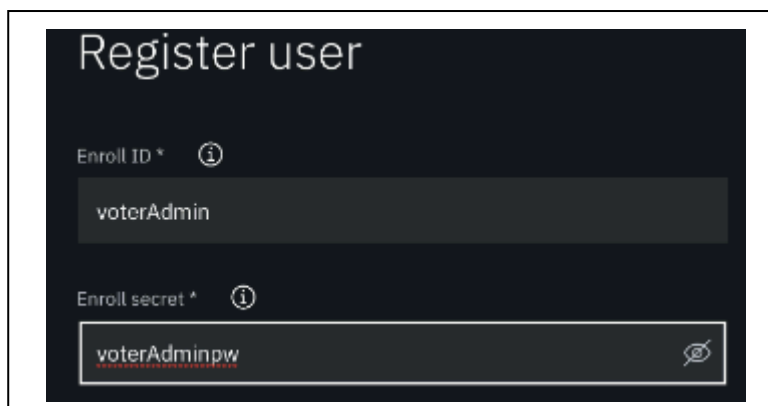
In Fig 5.3, an associate identity is created. It is required to operate the organisation. Admin adds the admin credentials to create a virtual 'console wallet'



The screenshot shows a web form titled "Associate identity". It includes a text block explaining that an identity is required to operate the CA and that users can either enroll a new identity or select an existing one. Below this, there are two tabs: "Enroll ID" (selected) and "Existing identity". The "Enroll ID" tab contains three input fields: "Enroll ID*" with the value "admin", "Enroll secret*" with a masked password "*****", and "Identity display name*" with the value "Orderer CA Admin". A toggle icon is visible on the right of the password field.

Fig 5.3 Sample Screenshot of Associate Identity

In Fig 5.4, an admin for the voting organisation is officially registered. Use the credentials created in Fig 5.2 to register the admin. Similarly, peers are also registered in the network.



Register user

Enroll ID * ⓘ

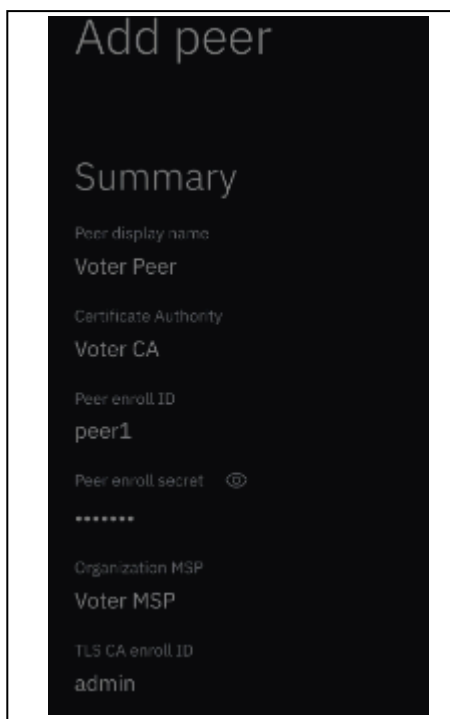
voterAdmin

Enroll secret ^ ⓘ

voterAdminpw

Fig 5.4 Sample Screenshot of Register User

In Fig 5.5, peers are added to the network. Peer are used to host smart contracts and store ledger. They allow the organization to transact on the network. Each organisation should deploy at least one peer.



Add peer

Summary

Peer display name
Voter Peer

Certificate Authority
Voter CA

Peer enroll ID
peer1

Peer enroll secret ⓘ
.....

Organization MSP
Voter MSP

TLS CA enroll ID
admin

Fig 5.5 Sample Screenshot of Add Peer

In Fig 5.6, certificate authority is created. The certificate authority creates identities for the nodes in the organisation and identities for the admin.

A screenshot of a web form titled "Add Certificate Authority". The form has three input fields. The first field is labeled "CA display name*" and contains the text "Orderer CA". The second field is labeled "CA administrator enroll ID*" and contains the text "admin". The third field is labeled "CA administrator enroll secret*" and contains the text "adminpw". There is a small eye icon to the right of the third field, indicating it is a password field.

Fig 5.6 Sample Screenshot of Add Certificate Authority

In Fig 5.7, an ordering service is created. It collects transactions, orders the transactions and bundles them into blocks. All channels have an ordering service associated with them.

A screenshot of a web page titled "Ordering Service". The page displays several configuration fields. The first field is "Ordering service display name" with the value "Orderer". The second field is "Certificate Authority" with the value "Orderer CA". The third field is "Ordering service enroll ID" with the value "orderer1". The fourth field is "Ordering service enroll secret" with the value "orderer1pw" and an eye icon to its right. The fifth field is "Organization MSP" with the value "OrdererMSP". Below these fields is a message: "Resource allocation is not available for free clusters." The last field is "Associated identity" with the value "Orderer Admin".

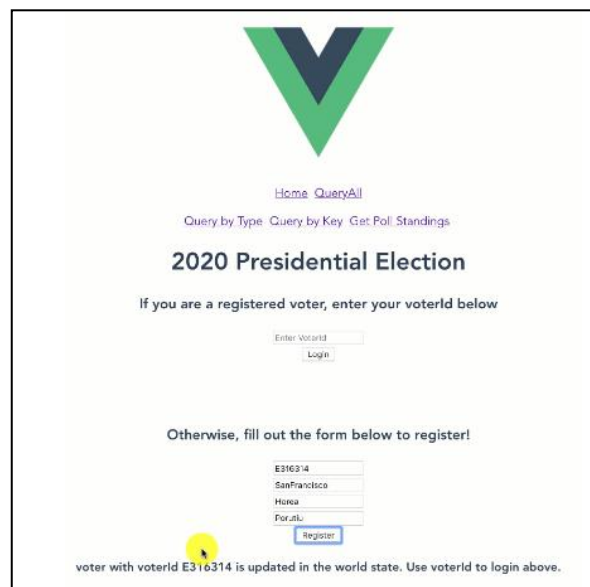
Fig 5.7 Sample Screenshot of Ordering Service

Finally a channel is created. A channel is the interface in which all the transactions take place. The ordering service is added to the channel and smart contracts are deployed in the network. Smart contracts are lines of code that are stored on a blockchain and automatically execute when predetermined terms and conditions are met.

Chapter 6

Results and Discussions

After deployment of smart contracts the backend is ready and the application is ready to be executed. After running the server, the user will be able to interact with the user. In Fig 6.1, the user needs to register as a voter, and create the digital identity with which he/she will submit the vote with. To do this, user will need to enter a uniqueId (drivers license) with a registrarId, and his/her first and last names. After user does that, and click ‘register’ the world state will be updated with voterId , name and registrarId.



The screenshot displays a web application interface for the 2020 Presidential Election. At the top, there is a large green and blue 'V' logo. Below the logo, navigation links include 'Home' and 'QueryAll'. A section titled 'Query by Type' lists 'Query by Key' and 'Get Poll Standings'. The main heading is '2020 Presidential Election'. Below this, a prompt states 'If you are a registered voter, enter your voterId below'. There is an input field labeled 'Enter VoterId' and a 'Login' button. Another prompt says 'Otherwise, fill out the form below to register!'. This is followed by a registration form with fields for 'E316314', 'San Francisco', 'Horea', and 'For:ku'. A 'Register' button is at the bottom of the form. A yellow cursor icon points to the 'Register' button. At the very bottom, a message reads 'voter with voterId E316314 is updated in the world state. Use voterId to login above.'

Fig 6.1 Sample Screenshot of Registration

Once login is done , user can cast vote. User will use voterId again to cast vote.In Fig 6.2, voting is for the presidential election for 2020, user can choose the party of his/her liking. Once user is done, he/she can choose submit, and then their vote is cast

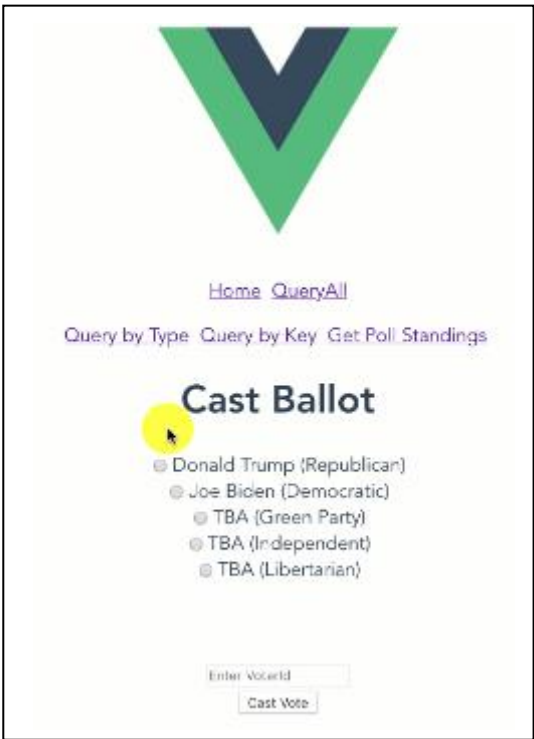


Fig 6.2 Sample Screenshot of Voting Process

Next, admin can view the poll standings by clicking ‘Get Poll Standings’ and clicking ‘Check Poll’. This will query the world state and get current poll standings

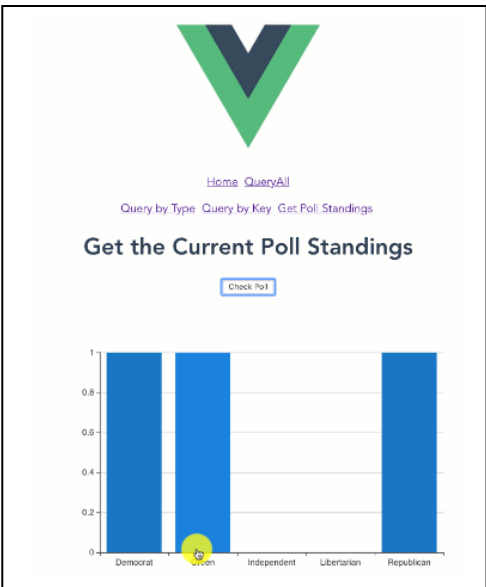


Fig 6.3 Sample Screenshot of Poll Standings

Chapter 7

Conclusion

The idea of adapting digital voting systems to make the public electoral process cheaper, faster and easier, is a compelling one in modern society. Making the electoral process cheap and quick, normalizes it in the eyes of the voters, removes a certain power barrier between the voter and the elected official and puts a certain amount of pressure on the elected official. It also opens the door for a more direct form of democracy, allowing voters to express their will on individual bills and propositions. In our project, we introduced a unique, blockchain-based electronic voting system that utilizes smart contracts to enable secure and cost efficient election while guaranteeing voters privacy. Our election scheme allows individual voters to vote at a voting district of their choosing while guaranteeing that each individual voters vote is counted from the correct district, which could potentially increase voter turnout.

References

- [1] Sagar Shah, Qaish Kanchwala, Huaiquin Mi, “Blockchain Voting System”, Available: <https://www.economist.com/sites/default/files/northeastern.pdf> (Accessed:8 August 2019)
- [2] Rahul Rakhe, Rahul Kale, Pritish Bisht, Prof K.S Balbudhe, “E-Voting System Using Blockchain Technology for Distributed Environment”, (IJIRSET) Volume 8, Issue 5, pp May 2019
- [3] Andrew Barnes, Christopher Brake, Thomas Perry, “Digital Voting with the use of Blockchain Technology”, Available: <https://www.economist.com/sites/default/files/plymouth.pdf> (Accessed:16 August 2019)
- [4] Friðrik Þ. Hjálmarsson, Gunnlaugur K. Hreiðarsson, Mohammad Hamdaqa, Gísli Hjálmtýsson, “Blockchain Based E-Voting System” (IEEE), 2018 IEEE 11th International Conference on Cloud Computing, ISSN: 2159-6190, pp 10 September 2018
- [5] Harsha V. Patil, Kanchan G. Rath, Malati V. Tribhuwan, “A Study on Decentralized E-Voting System Using Blockchain Technology”, (IRJET) Volume 5, Issue 11, pp November 2018
- [6] Gaby G Dagher, Praneeth Babu Marella, Matea Milojkovic, Jordan Mohler, “BroncoVote: Secure Voting System Using Ethereum’s Blockchain” ICISSP 2018: Proceedings of the 4th International Conference on Information Systems Security and Privacy, pp January 2018
- [7] Rifa Hanifatunnisa, Budi Rahardjo, “Blockchain based e-voting recording system design”, (IEEE), 2017 11th International Conference on Telecommunication Systems Services and Applications (TSSA), pp February 2018
- [8] The detailed Information of the UML diagrams [online], Available: https://www.tutorialspoint.com/uml/uml_standard_diagrams.htm (Accessed: 2nd October 2019)
- [9] Information on Blockchain [online], Available: <https://blockgeeks.com/guides/What-is-blockchain-technology/> (Accessed: 4 October 2019)

Appendix

Technologies Used

1. **HTML5**: It is a software solution stack that defines the properties and behaviors of webpage content by implementing a markup based pattern to it. HTML5 is the fifth and current major version of HTML, and subsumes XHTML.

2. **CSS3**: It is the latest version of the css specification. CSS3 adds several new stylish features and improvements to enhance the web presentation capabilities.

3. **jQuery**: The purpose of the jQuery is to make it much easier to use JavaScript on your website. jQuery takes a lot of common tasks that require many lines of JavaScript code to accomplish, and wraps them into methods that you can call with a single line of code.

4. **Bootstrap**: Bootstrap is a free and open-source CSS framework directed at responsive, mobile-first front-end web development. It contains CSS and JavaScript-based design templates for typography, forms, buttons, navigation and other interfaces.

5. **JavaScript**: It enables interactive web pages and is essential part of web applications. The vast major web browsers have a dedicated JavaScript engine to execute it.

6. **PHP**: PHP is a widely used open source general purpose scripting language that is especially suited for web development. It is used for server and client side scripting.

7. **Blockchain**: A Blockchain is used here to secure the users data and the transactions that is managed by a cluster of computers not owned by any single entity. The Blockchain is a democratized system.

8. **Paillier Homomorphic Encryption**: Paillier is a type of keypair-based cryptography. This means each user gets a public and a private key, and messages encrypted with their public key can only be decrypted with their private key.

9. **Solidity:** Solidity is an object-oriented programming language for writing smart contracts. It is used for implementing smart contracts on various blockchain platforms.

10. **Hyperledger Fabric Network:** A Fabric permissioned blockchain network is a technical infrastructure that provides ledger services to application consumers and administrators. In most cases, multiple organizations come together as a consortium to form the network and their permissions are determined by a set of policies that are agreed to by the consortium when the network is originally configured

11. **IBM Blockchain Platform:** It gives you total control of your blockchain network with a user interface that can simplify and accelerate your journey to deploy and manage blockchain components on the IBM Cloud Kubernetes Service.

12. **IBM Kubernetes Service:** It creates a cluster of compute hosts and deploys containers. A Kubernetes cluster lets you securely manage the resources that you need to quickly deploy, update, and scale applications.

13. **API:**An application programming interface is a computing interface to a software component or a system, that defines how other components or systems can use it.

14. **Fabric SDK:** The Hyperledger Fabric SDK allows applications to interact with a Fabric blockchain network. It provides a simple API to submit transactions to a ledger or query the contents of a ledger with minimal code.

Publications

Publications Paper Title is yet to be published in journal name

Acknowledgement

It is with sincerest courtesy that we thank everyone associated with the successful completion of our project. Our project guide, **Ms. Shraddha Dabhade**, proved quintessential throughout the length of the project-making, with her constructive criticism, valuable inputs, whole-hearted co-operation and relentless patience.

We deeply express our sincere thanks to our Head of Department, **Dr Prof. Rahul Khokale** for encouraging and allowing us to present the project on the topic "Election Portal using Blockchain" at our department premises for the partial fulfillment of the requirements leading to the award of BE degree.

We take this opportunity to thank all our lecturers who have directly or indirectly helped our project. We pay our respect and love to our parents and all friends for their love and encouragement through out our career. Last but not least we express our thanks to each other, we three team mates, who endured all the obstacles and learnt the true meaning of team spirit.

Shehan Shetty (EU1162068)

Vishal Thakur (EU1162054)

Adhij Vartak (EU1162098)

Bachelor of Computer Engineering

Final Year, Eight Semester

Session : 2020