UNIVERSITY of RWANDA

COLLEGE OF SCIENCE AND TECHNOLOGY

Thesis's title

Development of Blockchain technology-based e-voting system: A case study of High Institute of Computer and Management of Goma

By

Isamuna Nkembo

Ref: **221025542**

A dissertation submitted in partial fulfilment of the requirements for the degree of

MASTER IN SOFTWARE ENGINEERING

Supervisors: **Dr. Rwigema James**

Dr. Alexander Ngenzi

March 2023

Declaration

This is to certify that the project work entitled "**Development of Blockchain Technology based e-voting system: A case study of High Institute of Computer and Management of Goma**" contains my own work except where specifically acknowledged, and it has been passed through an anti-plagiarism system and found to be compliant and this the approved final version of the thesis.

Name: **Isamuna Nkembo**

Reg: **221025542**

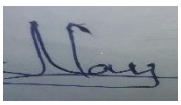
Signature: 

Date: **27th March 2023**

Supervisors

Dr. James RWIGEMA

Dr. Alexander NGENZI



Acknowledgments

I know that without the grace of God, I will not be able to reach this stage of my master's program, I give so many thanks to Him.

I also would like to thank the Postgraduate Programs Coordinator of CST at the University of Rwanda, **Dr. Frederic Nzanywayingoma**, for providing us with strong knowledge through the planning of courses during our master's study.

I can't forget the very useful advice of my supervisors, **Dr. Rwigema James** and **Dr. Alexander Ngenzi** to overcome some issues during the writing of this final work and let the hand of God be on them. I also want to express my gratitude to all the lecturers for the master's program in Software Engineering for their support and directives throughout the entire program of study.

Many thanks to my beloved wife, **Ombeni Materanya Vanessa** to support me during these two and a half difficult years, be God bless her. I am grateful to my kids **Isamuna Ngemba Enaelle**, **Isamuna Ngangu Blessed**, **Isamuna Masonga Miguel**, and **Isamuna Luzolo Great** for their solicitude during my study.

I thank all my family, my friends, and each people who contributed to the success of this work, let God be your reward.

Isamuna Nkembo

Abstract

Nowadays, computer systems allow to build more powerful and more efficient systems, focused on the enhancement of some manuals operations by using digital technologies.

Voting process is very challenging in most of Africa's countries, because is characterized by many contestations after election.

General elections still use a centralized system through one organization that manages it. Some of the problems that can occur in traditional electoral systems is with an organization that has full control over the database and system, it is possible to tamper with the database in considerable opportunities and, implies the failed of overall voting process.

Blockchain technology is one of solutions, because it embraces a decentralized system, and the entire database are owned by many users. Blockchain itself has been used in the Bitcoin system known as the decentralized Bank system. By adopting blockchain in the distribution of databases on e-voting systems can reduce one of the cheating sources of database manipulation.

The contribution of this work is to give a way for implementing a Blockchain-based e-voting system in the DRC's context. Precisely, it allows building an e-voting system based on the Ethereum Blockchain to be used on a narrow scope of an election process in a University (The target case study is the High Institute of Computer and Management of Goma). This e-voting system significantly improves users' (Voters) anonymity, guards against fraud in voting, and allows voters to cast ballots from anywhere (At home, at the polls, abroad, etc.). The benefit of this work in comparison with the actual voting system is the improvement of the entire voting system, and the possibility to use the same voting system for a whole province / region or country.

Keywords: Blockchain, e-voting, Vote, Database, Security, Peace engineering, Election.

Table of Content

List of Acronyms

ABI : Application Binary Interface

AI : Artificial Intelligence

CA : Contract Account

CST : College of Science and Technology

DApps : Decentralized Applications

DeFi : Decentralized Finance

DLT : Distributed Ledger Technology

DOM : Document Object Model

DRC : Democratic Republic of the Congo

EOA : External Owned Account

EVM : Electronic Voting Machine

EVM : Ethereum Virtual Machine

ICO : Initial Coin Offering

IDE : Integrated Development Environment

IoT : Internet of Things

IPFS : InterPlanetary File System

JSON : JavaScript Object Notation

JSON-RPC : JavaScript Object Notation-Remote Procedure Call

POS : Proof Of Stack

POW : Proof Of Work

SHA : Secure Hash Algorithm

TPS : Transaction Per Second

UML : Unified Modeling Language

UP : Unified Process

UR : University of Rwanda

List of Figures

List of Tables

Chapter 1: Introduction

This chapter gives us a brief overview of our study by understanding the problem statement, the objectives, and the relevance of this work. It also gives a breakdown of the entire work.

1.1 Context of the study

The truth of the overall election process become a great issue in most of Africa's countries [1], [2], including the Democratic Republic of the Congo [3], [4] however, it is an essential aspect for countries that can be really called democratic. The election process is mostly contested [5] by some stakeholders involved, because of probable cheating and, it is difficult to know exactly what happened during the voting process when there is a single organization that holds everything in a centralized way, even if witnesses could be present. In that way, the 2011 and 2019 DRC elections were characterized by irregularities and contestation [6]–[10] without any way to be convinced by the truth of the election process.

Many countries in Africa carry out the voting process using a ballot-box voting system, where voters are pre-registered before the day of the election, and each of them they have to use only one ballot paper on the day of the election to cast their vote by selecting their preferred candidate and then put the ballot paper into an enclosed or sealed ballot-box [11], [12]. The winner is who gains the highest ballot paper is known after the end of elapsed time of the electoral process, after that each sealed ballot box is opened by the electoral commission staff to perform counting of each ballot paper.

Figure 1.1: A woman voting in Sierra Leone in 2018 [12]

The results of this kind of voting process are mostly contested and, cheating in the ballot paper is most of the time reported because there is one organization that managed everything and can have a certain preference for a specific candidate. Recent elections in the Democratic Republic of the Congo in 2019 used Electronic Voting Machines [13] but the same contestations reported in the ballot box voting system were also reported because elections were far from free and fair [6], [14].

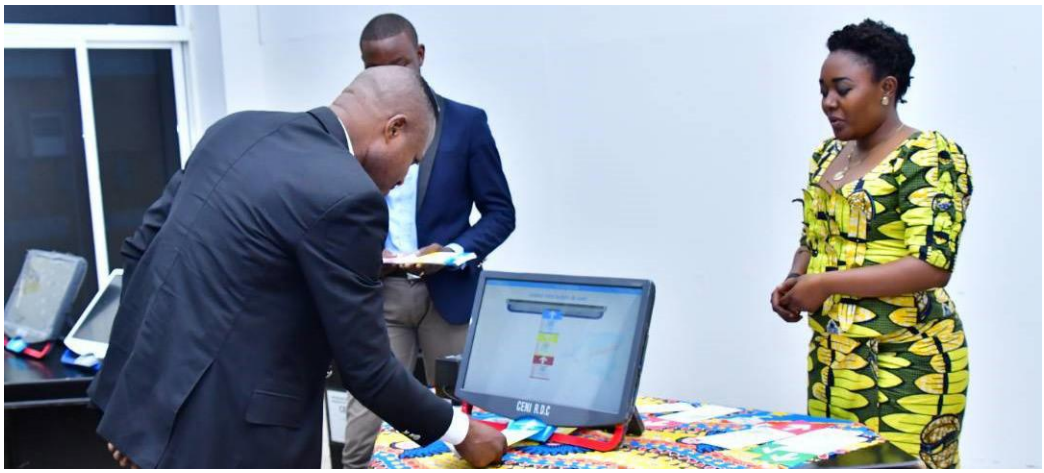


Figure 1.2: Professors from the University of Kinshasa try out the voting machines on [13]

Other authors have also proposed some kinds of web-based e-voting systems [15]–[17] without using Blockchain technology, and the main issue was the possibility to cheat with a centralized database used by each of their implemented systems. Also, privacy and anonymity were not guarantees.

By searching for possible solutions to avoid some issues involved in the traditional voting process, some researchers have conducted research to explore the way Blockchain technology [11], [18]–[22] could be useful.

In this work, we are exploring the way Blockchain technology could be useful to conduct a credible voting process by implementing a Decentralized web Application and improving the current voting system.

1.2 Background and Motivation

The election process becomes a more and more sensitive task for African countries to reach a consensus on the overall process. Most of the time, the winner of the elections is contested even if witnesses were present in each polling station. Also, cheating has often been reported during the voting process. In this case, the voting process needs a safer way to be conducted

to ensure all stakeholders by the truth of the overall voting process: That is the way Blockchain technology can help by using a decentralized system with a real-time update of voting records to each participant.

Lately, electronic voting systems have begun being used in many countries. Estonia was the first in the world to adopt an electronic voting system for its national elections, since 2005 and 2007, it has been used e-voting to conduct online voting [23]. After that, electronic voting was adopted by many other countries in the world [22], [24]–[26].

Most of these e-voting systems were used with a centralized system and with possible issues on security, transparency, and privacy for voters. Also, it was possible to tamper with the centralized database, and the anonymity of voters was not guaranteed. Certain potential hackers' attacks were also involved by these e-voting systems [27].

In the Democratic Republique of the Congo, voting was mainly conducted with the use of a ballot-box system in 2006, 2011, and 2019 with issues of fraud reported [7]–[9], [28], but last voting process in 2019 has also used Electronic Voting Machine which has not convinced stakeholders by the truth of the voting process, because same issues have also reported [14].

According to these issues, many researchers have conducted various studies in order to propose the best way for an e-voting system using Blockchain technology [22], [23], [25]–[27], [29], [30], thus, this work focuses- on the same way but with a different approach and in the DRC's context through a narrow scope, based on our case study.

1.3 Problem statement

The main problem in this work is focused on the challenge of the voting process in most African countries in general; and, particularly in DRC, because is often characterized by many contestations after the election [3], [4].

That aspect implies conflicts between peoples of the same country and freezes the integral development of those countries. The consequence is that the take-off of them is slow to be seen.

Regarding EVM (Electronic Voting Machine), the article by Kris Berwouts and Filip Reyntjens [7] said that the controversies of the voting machine were the most important instrument for fabricating the required fake results in the voting process in the DRC election in December 2018, and this process was affected by fraud. The main issue was the

incredibility of the voting machine. This aspect of fraud in the voting process was also confirmed in Horn policy's paper [6] which said that December 2018 elections were far from free and fair, even if Electronic Voting Machines were used in this process.

The voting machine also represents a great challenge for Congolese voters living abroad because it is very difficult to install them in a foreign country and allow those voters to fulfill their civic duty.

This work implements a Decentralized e-voting system on top of the Ethereum Blockchain to have a proper voting system that can contribute to peace by avoiding conflicts and post-elections violence, where innocent people are killed and improving the current voting system (By preventing fraud, cheating, and contestations of the voting process). This system is made to be tested in a narrow scope (On a university) before being used in a wide area.

1.4 Study Objectives

1.4.1 General Objective

The voting process in many African countries, and particularly in DRC are faces great issues, like contestation, post-election violence, election fraud, and falsification of the compilation results, and it difficult to convince all the parties (Voters, candidates, witnesses, and any other organ involved in the voting process) implied in the voting process by the truth of the overall voting process. Daxecker, Amicarelli, and Jung define two kinds of electoral contention: Publicly observable (Arrests, arson, attacks, bombings, boycotts, clashes, killings, intimidation, protests, rioting, shootings, or strikes) and linked to an electoral process, which can be violent (Include the threat or use of force intended to inflict harm on people: Armed groups, treat voters with violence if they participate in elections) or nonviolent in nature.

The DRC currently uses a centralized voting system, which means that all voting-related data are stored on a single server and are only managed by a single organization. Blockchain technology, however, is a decentralized system, meaning that data are stored on each user's node in the network rather than on a single server, improving the truth of the overall voting process and preventing fraud by making it possible to replicate voter records in real-time. We are confident to improve the voting system through the implementation of a Blockchain technology-based e-voting system and, to see its impact compared to the current one.

1.4.2 Specific objectives

We will accomplish our aim by meeting the following specific objectives:

- To build an e-voting system prototype based on Blockchain technology in the DRC's context and into a single University as a use case.
- To test this prototype and compare it to the actual voting process to see its benefits and its limits.
- To produce real-time voting results.

At the end of this work, we are convinced that we are going to improve the current voting system by building a Blockchain-based e-voting system that could serve as a model not only for the DRC and other countries in Africa but also for the entire world. The security of the system is guaranteed by the immutability of the Blockchain, where any changes can't be made once voters' votes are recorded on the ledger (Blockchain plays the role of Distributed Ledger Technology).

1.5 Hypotheses

We assume that a Blockchain-based e-voting system cannot only improve the voting system but can also contribute to peace by avoiding post-electoral crises. The expected system would also avoid issues like contestation, election fraud, or falsification of the compilation results, and build an ecosystem where all stakeholders (Voters, candidates, witnesses, and any other organ involved in the voting process) involved in the voting process should be convinced by the truth of election.

If we reach the truth of the voting process by all parts involved in it, the goal of this research will be achieved, and this work try to build a prototype able to give us this expected result. Our approach tries building an e-voting system on top of Ethereum's Blockchain using a Smart Contract inside a DApp.

1.6 Study Scope

We use only one type of Blockchain (Ethereum) with a limited use case (Voting system in one University) to create a working e-voting prototype because there are many other types of Blockchains that could be used. This work had taken a few months to be finalized corresponding to our master's program. However, the implementation phase had not covered

all aspects of the voter registration process (Including biometric identification and facial recognition) but had instead concentrated on the later stages of the counting process and real-time results viewing for all voters.

We tested our DApp (Decentralized Application) in one of the Ethereum Test nets called the Goerli test network [31] before being able to be recorded on the real Ethereum Blockchain (Ethereum Mainnet [32]), which involves paying fees with real cryptocurrency when transactions are validated on the Blockchain as trusted records. This DApp's output is divided into two parts: Frontend for user interface interactions and Backend for server-side processes, including the Blockchain's interactions.

1.7 Significance of the Study

The truth of the ballot box is important during the voting process because, without it, peace cannot be assured. Furthermore, if there is a chance of tampering with the database or cheating with voting results, the truth of the overall process would not be guaranteed. That is the case when a centralized database is used during the voting process and witnesses do not have the ability to control everything. (Because that is the case).

With a decentralized system such as the Blockchain, each voter will act as a witness because each voter's choice will be recorded on the Blockchain and visible to all voters in real-time. It will be impossible to falsify these records, and the overall voting process will be assured with high accuracy. Additionally, voter identity and privacy will be safeguarded.

In only a few words, the importance of this study or its potential effect can be summarized as ensuring the truth of the voting process and the ballot box and avoiding cheating during voters' votes, using Blockchain technology. Furthermore, our prototype is playing an important role in promoting peace and avoiding the negative effects of candidate contestations, which entail conflict and death.

1.8 Organization of the Study

Apart from the abstract, list of acronyms, list of figures, list of tables and reference list, this work is divided into six parts:

- The **first chapter** (***Context of the study***) contextualizes our study to understand the problem statement, the goal of the study, and relevance.

- The **second chapter** (***Literature Review***) explains the existing literature to gain a better understanding of the subject, the gap involved, and the solution.
- The **third chapter** (***Research Methodology***) explains methods to be used during project implementation.
- The **fourth chapter** (***System Analysis and Design***) presents models and explains how the system will function internally.
- The **fifth chapter** (***Results and Analysis***) presents the final system's outputs and demonstrates how our prototype functions.
- The **sixth chapter** (***Conclusion and Recommendations***) concludes the work and suggests further improvements.

1.9 Conclusion

This chapter served as a base for understanding our research's problem statement, the study's objective and motivation, and the work's relevance. This chapter has also provided us with an overview of the major keywords used in the work, which helps us comprehend what we need to accomplish. His organization was provided before the end of this chapter to guide each of his readers and serves as a brief recap of the entire work.

Chapter 2: Literature Review

2.1 Blockchain overview

The Blockchain is defined as the chain of digital blocks connected and associated with each other as an open distributed ledger [33], is also a technology that allows users to validate, keep and synchronize the content of a transaction ledger that is replicated across multiple users [34]. The concept of a chain of blocks was presented by Stuart Haber et al. in 1991 to digitally timestamp electronic documents to protect against tempering [33], [35]. The first version of Blockchain was used to promote cryptocurrencies which Bitcoin [34], [36] is the main, the second version includes a trust management feature using smart contracts that manages themselves with no involvement of any third parties, and the third version which is the present and future of the technology includes various application areas such as DeFi (Decentralized Finance), IoT (Internet of Things), education, identity management, big data, AI (Artificial Intelligence) and healthcare, etc.

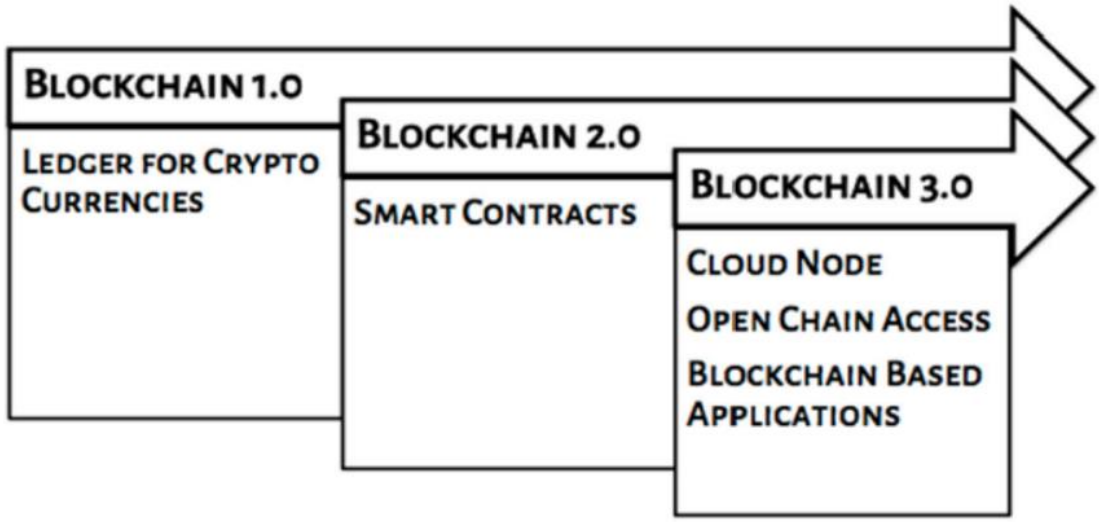


Figure 2.1: Stages of Blockchain Development [34]

The fundamental shift that blockchain technology represents is a method for moving away from an attempt to have a central trusted authority in a massively distributed network. But instead, to have multiple sources of trust that must all agree, based on an algorithm that this transaction can be trusted as valid [37].

Furthermore, most blockchain solutions offer an immutable and enduring record of a transaction as it is hard for any trusted or untrusted source to change or modify. This presents

a completely new level of security, privacy, and trust to in our online world [37]. When Blockchain technology began to exist, the first application that was tested on the platform was Bitcoin, because Bitcoin was the first application on the Blockchain technology [38].

Blockchain offers a secure, distributed database that can operate without a central authority or administrator. Blockchain uses a distributed, peer-to-peer network to make a continuous, growing list of ordered records called blocks to form a digital ledger. Each transaction, represented in a cryptographically signed block, is then automatically validated by the network itself [39].

Distributed Ledger Technology (DLT) [40], [41] is the foundation of blockchain. DLT offers a consensus validation mechanism through a network of computers that facilitates peer-to-peer transactions without the need for an intermediary or a centralized authority to update and maintain the information generated by the transactions. Each transaction is validated and, along with a group of validated transactions, is added as a new "block" to an already existing chain of transactions, giving rise to the name "*blockchain*". Once a transaction has been added to the chain, it generally cannot be altered or removed [42]. Below, is a representation of the Blockchain.

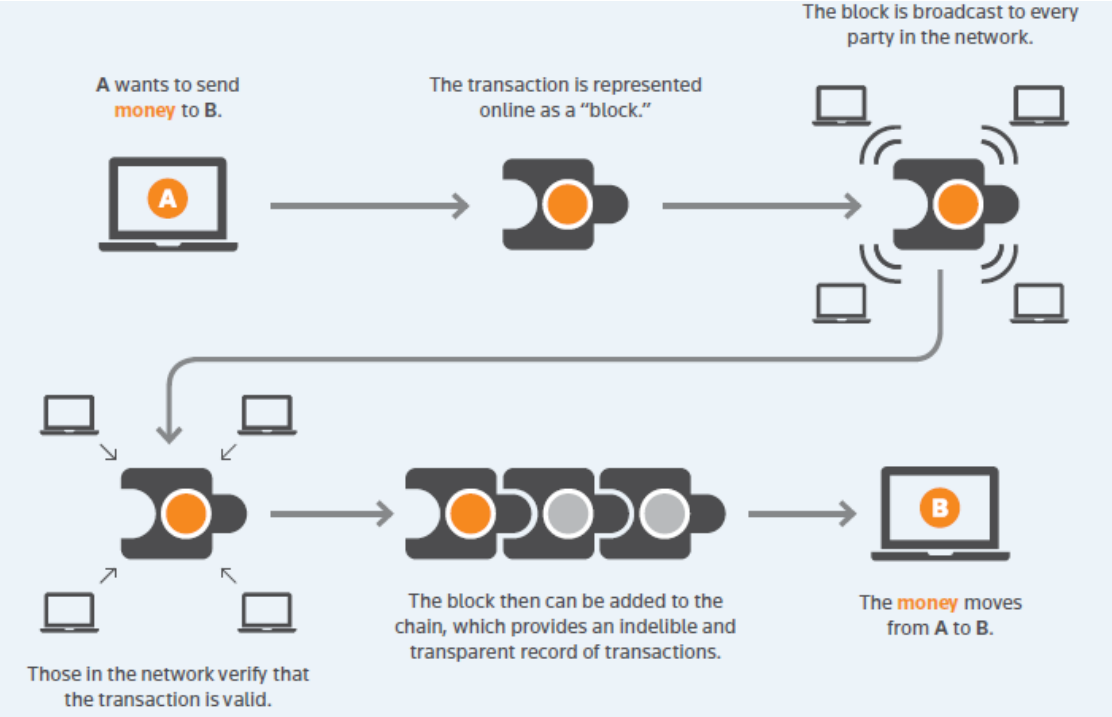


Figure 2.2: Visual representation of Blockchain [42]

There are two types of blockchain networks: Permissioned Blockchain or Private Blockchain, and Permissionless or Public Blockchain [42].

- **Permissioned Blockchains or Private Blockchain**: These networks are proprietary networks that specific individuals or entities use to conduct transactions (Such as a group of banks processing financial transactions).

  Prior authorization is required before you can access a private blockchain. Private blockchains are almost always owned by a single organization or a small group. The blockchain owner requires that each blockchain user request authorization to interact with the blockchain data and provide access credentials with each access request. Private blockchains provide organizations with the features of blockchain applications without having to expose all their data to the public.

  Permissioned blockchains are becoming an institutional-driven solution to the conduct of business with transactional efficiency, cost-cutting, and the management of the provenance and traceability of goods in global supply chains such as the wine industry [41].
- **Permissionless Blockchain or Public Blockchains**: These are open-source networks that anyone can access and use (Such as Bitcoin users who transact with each other using Bitcoin for payment).

  All you need to interact with it is a valid address, and you can read the blockchain and even submit transactions. This is the most popular type of blockchain, and one that most people think of when associating blockchain with cryptocurrency to be an alternative to trading currencies [41]. Public blockchains ensure that no one organization controls the blockchain because any computer can become a node and each computer maintains a full copy of the blockchain.

  Not all nodes store full copies of blockchain blocks. Full nodes do maintain complete copies of the blockchain, but lightweight nodes store just some blocks of the blockchain. Lightweight nodes often store recent blocks and provide transaction validation services for clients.

Unlike the bitcoin blockchain and other public networks, permissioned blockchain networks are typically developed by companies for their own private commercial use. Organizations may develop their own network or customize a basic network previously developed by a vendor. In some cases, a group of companies in an industry may collaborate to develop and

share a proprietary network to facilitate transactions among them, such as the R3 Blockchain Consortium[1], which offers a blockchain system for financial institutions [42], [43].

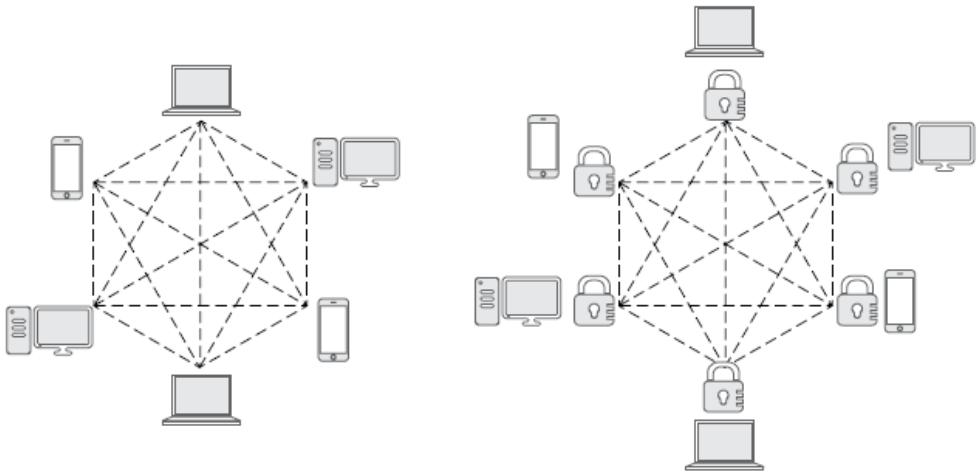


Figure 2.3: Public and Private versus Blockchains [43]

All blocks in the main chain are numbered, starting with the number 0, then 1, 2, 3, 4, 5, and so on. The green block is the first block that was created, and it is also known as a *genesis block*, and it has a block number *zero*.

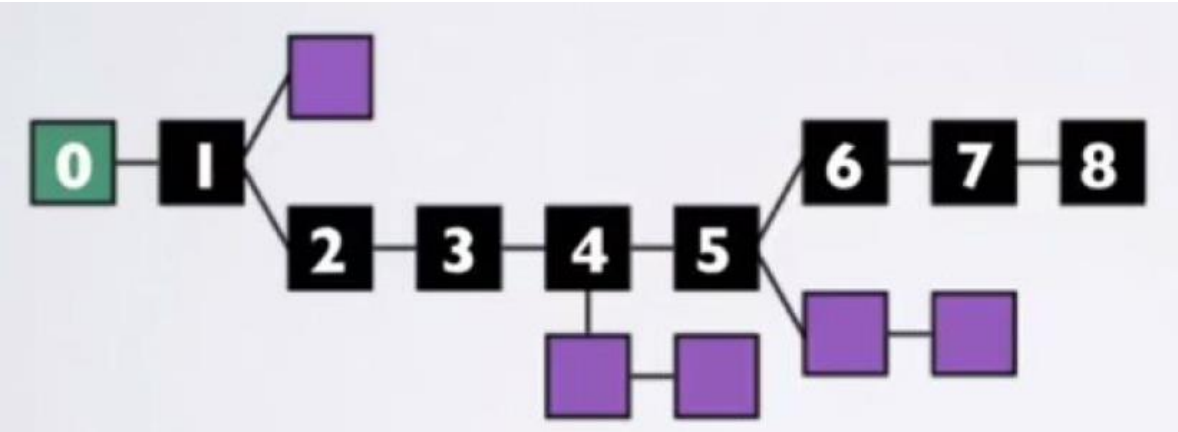


Figure 2.4: Block numbered in the Blockchain [38]

The purple blocks are the ones that are forming short and invalid chains, they are called *blockchain forks*. Blockchain forks do occur very often, additionally, these side forks are also known as *orphaned forks* [38].

1. https://www.r3.com

A Bitcoin block is created every ten minutes on average (For 10 TPS or 10 Transaction Per Second); however, Ethereum blocks are set up in every 15 seconds on average (For 15 TPS) [44]. Nodes on the peer-to-peer network responsible for creating these blocks are called miners. All the miners are collecting every transaction that people are sending to each other over the network, and only valid transactions are relayed to the other nodes. Each miner takes a number of these collected operations and puts them in a newly formed block. These lists of transactions are numbered tx0, tx1, tx2, … and so on. The first transaction (tx0) in a block is also known as the Coinbase transaction, where the miner assigns a block reward to his address: This is how Bitcoins are created.

Each transaction requires a small transaction fee [45]. This fee will continue to increase as an incentive for the miners to create new blocks because the block reward will continue to be lowered. This growth is attributed to the so-called “bitcoin halving” events, through which the amount of bitcoin that can be attained by mining an additional block is automatically halved after every 210,000 blocks mined (approximately every four years) [46]. The last halving event occurred on May 11, 2020, and it reduced the reward for mining a block from 12.5 to 6.25 bitcoin [46].

The Blockchain technology become a best way to increase security, privacy and data management [47], because all blocks of transactions are hashed (By cryptographic hashing function using SHA256) and linked together by forming an immutable chain of block in a decentralized way. The hashing mechanism is the key to the security aspect of the bitcoin Blockchain by using a non-reversible mathematical function. This security is increased by another mechanism using a pair of keys or public key and private key. This append because the private key is used for signing the transaction, and the public key is used for verification of the transaction. The public key is kept in the wallet (like MetaMask), which can be implemented in *software*, *hardware*, or *online* [48]: This is shown in the bellow Figure 2.5.

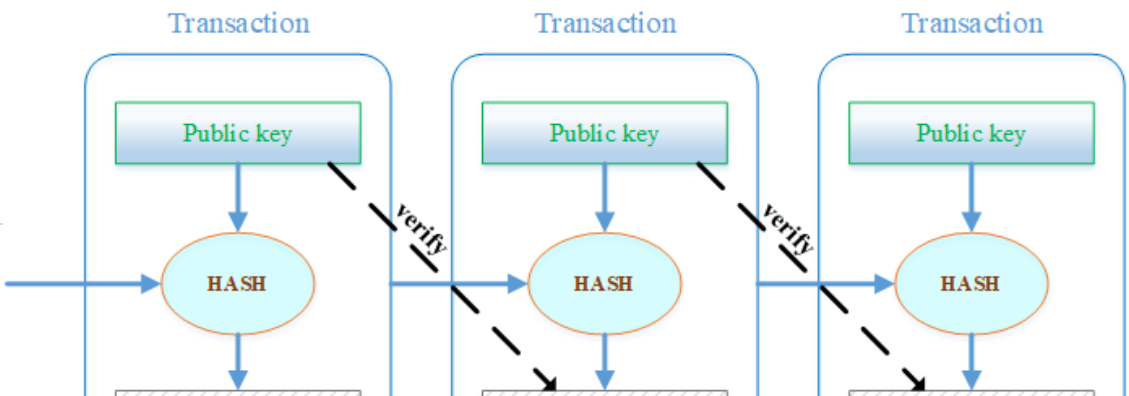


Figure 2.5: The structure of transaction in a Bitcoin blockchain [48]

The way transactions are organized within a block adds an additional layer of security by utilizing the so-called "Merkle Tree" technique, in which all transactions are hashed into a single hash using a binary tree of every single transaction [48]. The final output hash is called "Merkle root" or "Root hash", which will become the final hash of the block and the previous hash of the next block.

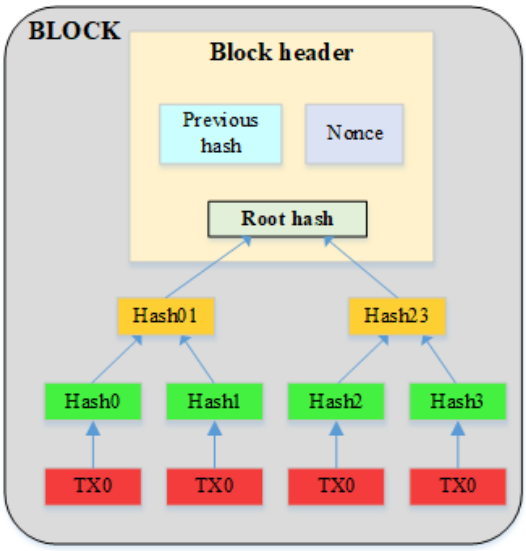


Figure 2.6: A Bitcoin block with hashed transactions into a Merkle tree [48]

2.2 Ethereum network

In the Ethereum universe, there is a single, canonical computer (Called the Ethereum Virtual Machine, or EVM) whose state everyone on the Ethereum network agrees on. Everyone who

participates in the Ethereum network (Every Ethereum node) keeps a copy of the state of this computer. Additionally, any participant can broadcast a request for this computer to perform arbitrary computation. Whenever such a request is broadcast, other participants on the network verify, validate, and carry out ("execute") the computation. This causes a state change in the EVM, which is committed and propagated throughout the entire network [49]. EVM is also known as the runtime environment for Smart Contracts in Ethereum [50].

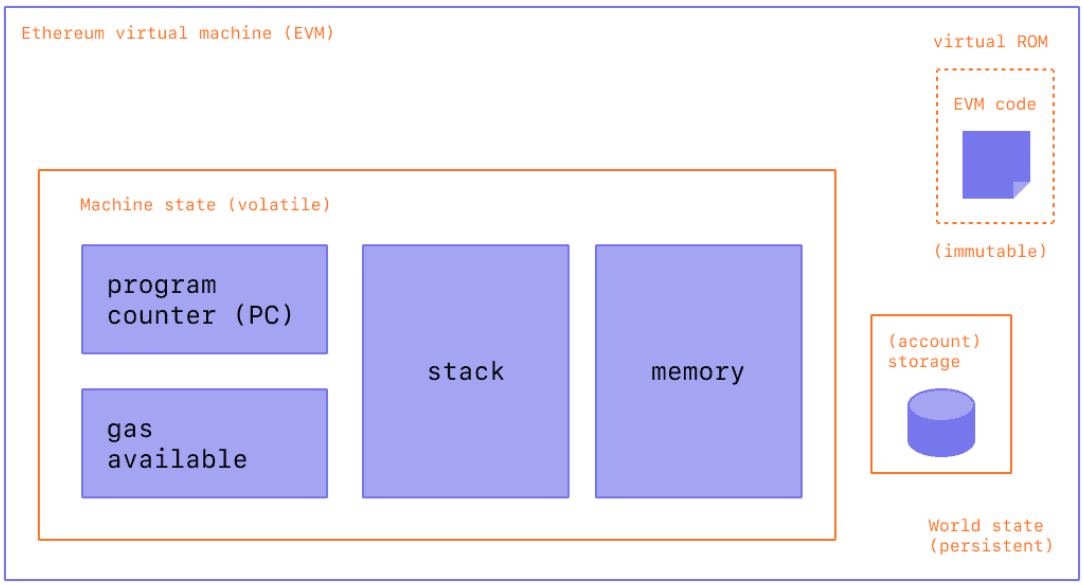


Figure 2.7: Ethereum Virtual Machine or EVM [51]

Requests for computation are called transaction requests; the record of all transactions as well as the EVM's present state is stored in the blockchain, which in turn is stored and agreed upon by all nodes.

Cryptographic mechanisms ensure that once transactions are verified as valid and added to the blockchain, they cannot be tampered with later; the same mechanisms also ensure that all transactions are signed and executed with appropriate "permissions" (No one should be able to send digital assets from Alice's account, except for Alice herself).

The primary difference between Bitcoin and Ethereum is that Bitcoin is mainly used as a distributed ledger for financial transactions, but Ethereum is designed to be used as a distributed computing platform for running applications. In addition, from Sept 6, 2022; Ethereum had upgraded to Ethereum 2.0 by changing the consensus mechanism from Proof Of Work (Still using for Bitcoin) to Proof Of Stack [52]: The major impact of this migration

is the drastically reduction of power consumption or electricity (More than 99 percent) require to maintain the Ethereum Network [53].

Proof Of Work (POW) use specifics nodes in the Blockchain network called "Miner" to validate transactions or mine block (To ensure their validity) by solving an increasingly difficult problem (A puzzle) and get back reward for the computation done: This computation consumes a huge amount of energy or electricity (Wastage of resources); but, the Proof Of Stack works by providing the user who having the highest stack in the network the opportunity to validate transactions or create new blocks, without the need to waste resources [54].

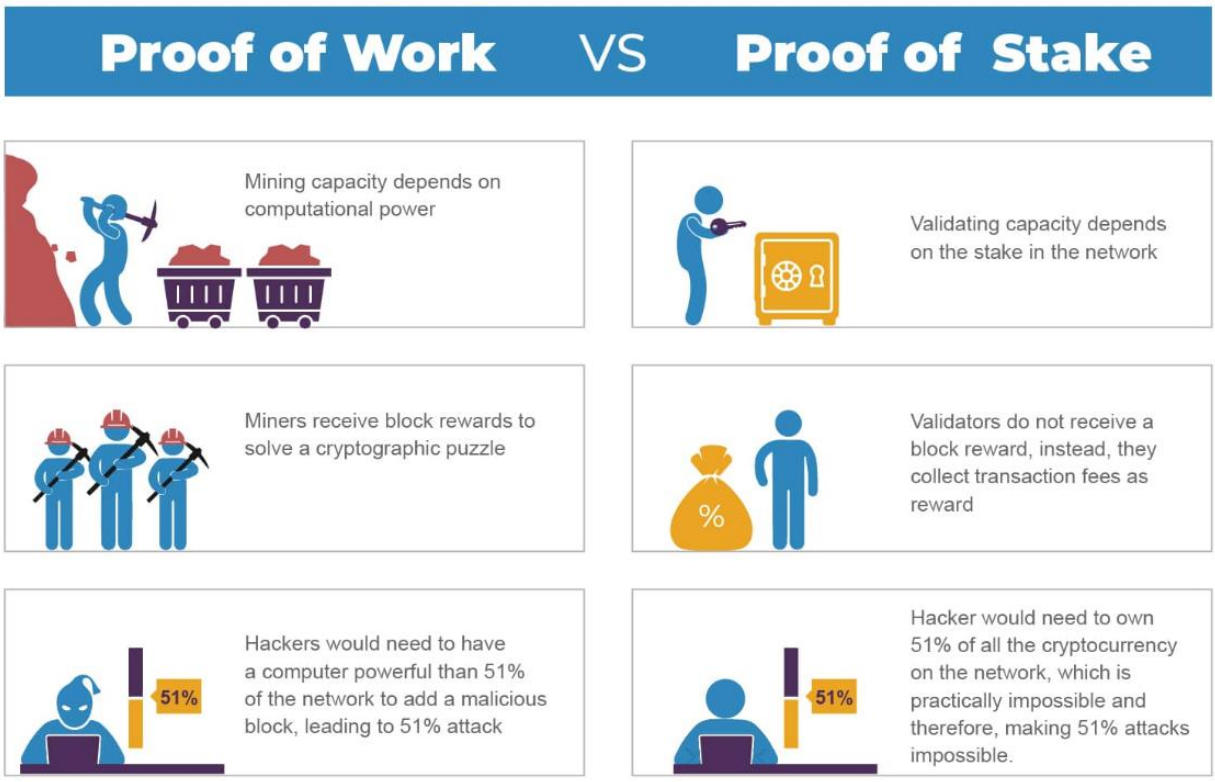


Figure 2.8: Ethereum POW vs POS [53]

Proof Of Stake underlies Ethereum's consensus mechanism. In 2022, Ethereum has switched from POW to POS because it is more secure, less energy-intensive, and better for implementing new scaling solutions compared to the previous architecture [49]. This upgrade has changed the Ethereum ecosystem and its version from 1.0 to 2.0. This new architecture is

based on the main chain or Coordinator Layer called “Beacon Chain”, held by a sub-chain or Data Layer called “Shard Chains”, which work directly with the EVM or Execution Layer.

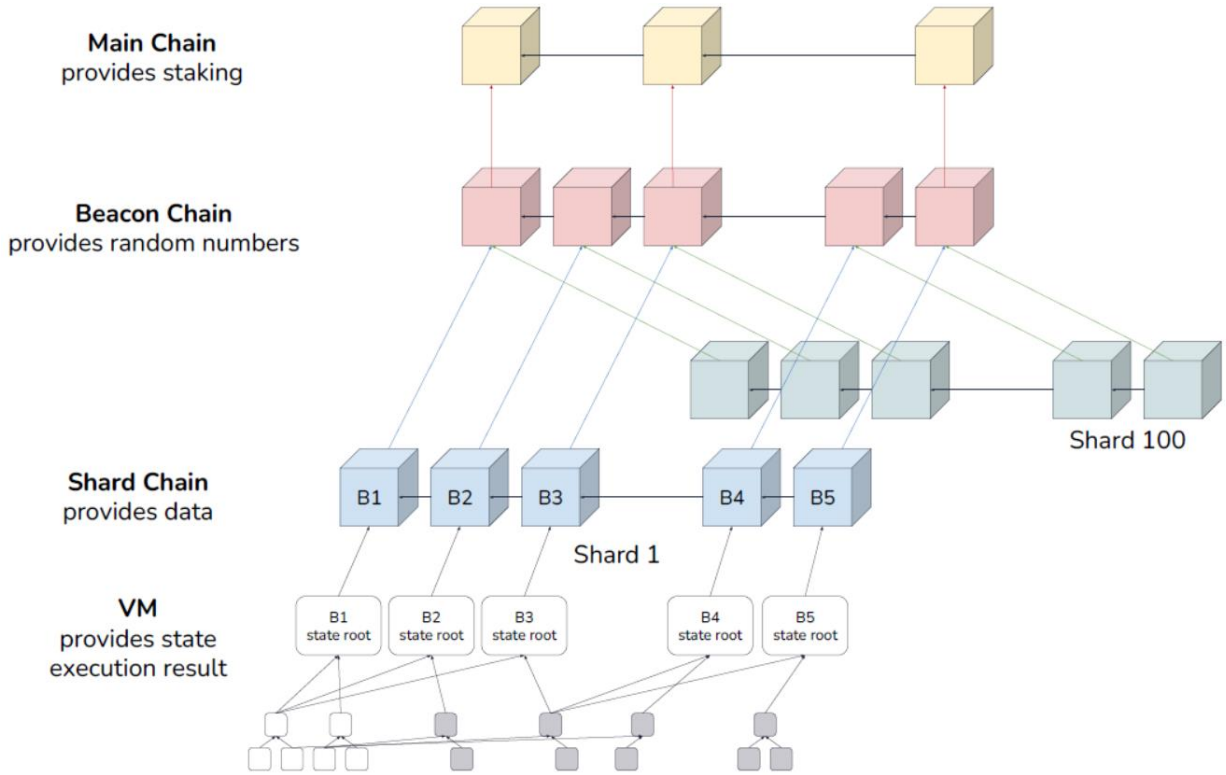


Figure 2.9: Ethereum 2.0 overall architecture [55]

Bitcoin can be used to pay for goods and services anywhere they are accepted, the currency of the Ethereum network “Ether” is designed to be used by developers to pay for computing power on the network when running decentralized applications.

Bitcoin and Ethereum both have digital currencies but from an overall point of view, they differ in purpose. The point of Ether was not to establish itself as a payment alternative but to encourage developers to create and run applications within Ethereum. The currency is one small part of the network as Ethereum, but it also has an entire computing platform on top of the Blockchain.

2.3 Decentralized Applications

Decentralized Applications (DApps) are applications that are open source, not controlled by one person or entity, and run across a distributed Blockchain or network of computers. DApps are not central servers. Instead, the users connect to each other through peer-to-peer connections [56]. They are programs with a combination of front-end and back-end technologies and run on a decentralized network. Some of them can be semi-centralized, a major part of activities in the truly decentralized application should happen out of a central party's control [57].

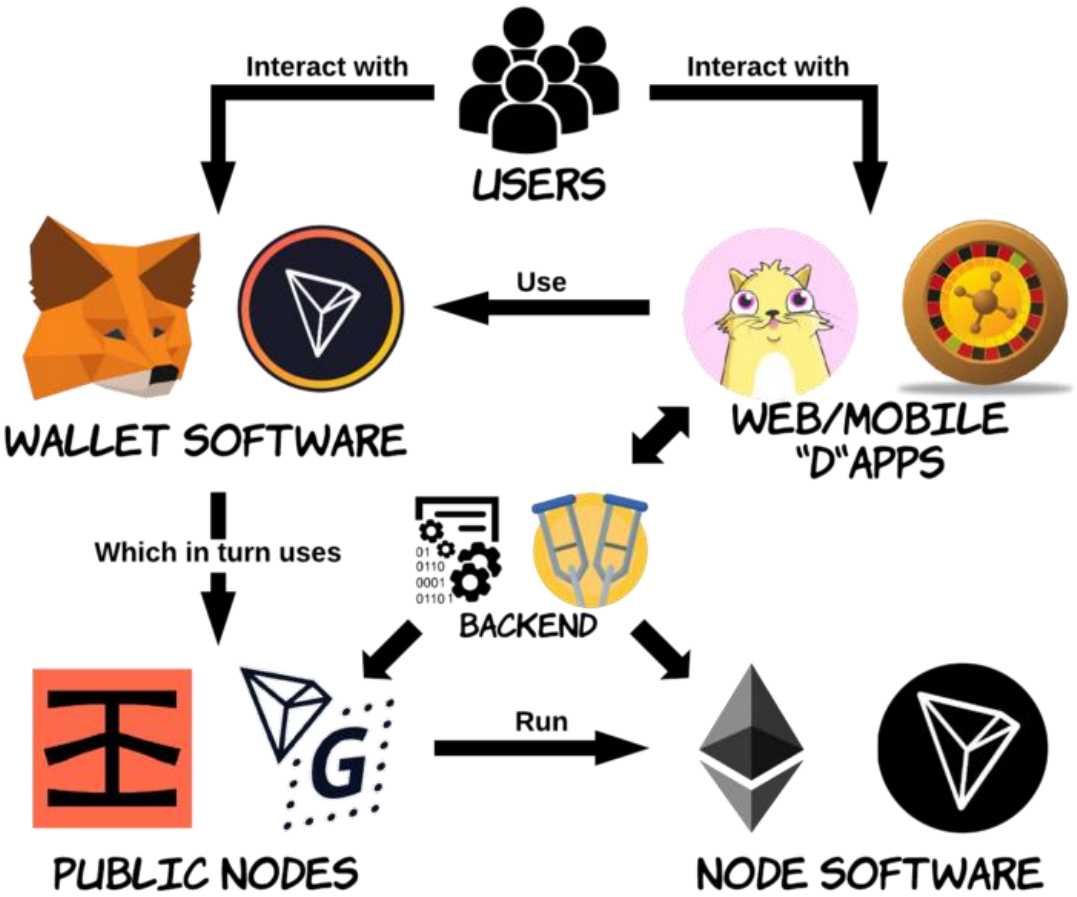


Figure 2.10: An example of Decentralized Application [57]

With standard applications, controlled by one entity and run on a centralized server, with possible hacking or downtime due to the server going offline; a decentralized Application has

not a single server or entity control. It runs across a network of computers and changes are decided by the users.

There is no central point that the server could crash or be hacked. If one computer on the network goes offline, the application is unaffected as there are thousands of other computers where the application is running from at the same time. Even if one computer on the network is hacked, it cannot make unauthorized changes to the application as the majority of the network must agree to the changes.

2.4 Smart contract

When you exchange items of value, generally rules govern how the transaction takes place. In many cases, the rules are simple. For example, *you give me $1.89, and I give you a soft drink*. Each party can see and validate the other party's contribution to the transaction. If you try to give me fake money, you will not get your soft drink. Even though this transaction seems simple, there is more to it than meets the eye [43]. In most cases, if a soft drink costs $1.89, you will have to tender more than that for it. You will have to pay taxes as well. So, there is another participant in the transaction: the government. Instead of keeping all the money, I must send some of it to the government for taxes.

Moving even simple transactions like the soft drink example into the digital world takes some careful thought. You cannot just send money to people and trust that they will do their part. You need some way to enforce rules and compliance to make sure that all parties are treated fairly.

Smart contracts help you enforce rules when you exchange anything of value in Ethereum. The simplest way to describe smart contracts is that they are programs that execute when certain transactions occur. For example, if you create a softdrink-purchase smart contract, that software code will run every time someone buys a soft drink. The smart contract code is stored in the blockchain, so all nodes have a copy of it. Also, it does not matter where the software runs: All nodes are guaranteed to run it the same and get the same results as every other node.

Smart contracts allow trusted transactions and agreements to be carried out among different anonymous parties [58]. Smart contracts are contracts that are written in computer code and operate on a blockchain or distributed ledger. They automatically verify, execute, and enforce

the contract based on the terms written in the code. Smart contracts can be partially or fully self-executing and self-enforcing [56].

Smart contracts can be used to exchange anything of value, according to the potential use of the Blockchain. When a Smart contract is run on the Blockchain, it operates automatically. If the conditions of a contract are met, payment or values are exchanged based on the term of the contract. Likewise, if conditions in the contract are not met, payment may be withheld if written in the Smart Contract.

Smart contracts run as they are programmed on a decentralized network of computers in the Blockchain removing risks around unauthorized changes, fraud, server failure or non-compliance with the terms of the contract. The contracts execute automatically, exchanging value and payments between people without the need of lawyers or courts to enforce them [59]. This could happen for example when someone sells his house to another, a third party (Lawyers, insurance, or brokers) may be necessary to act on behalf of each party [60]. With Smart Contract, the task can be completely automated, and once the buyer paid the total amount written on the contract, then the ownership property will be automatically transferred from seller to buyer without the involvement of a third party (Which implies supplementary fees for charges). The bellow figure illustrates this scenario.

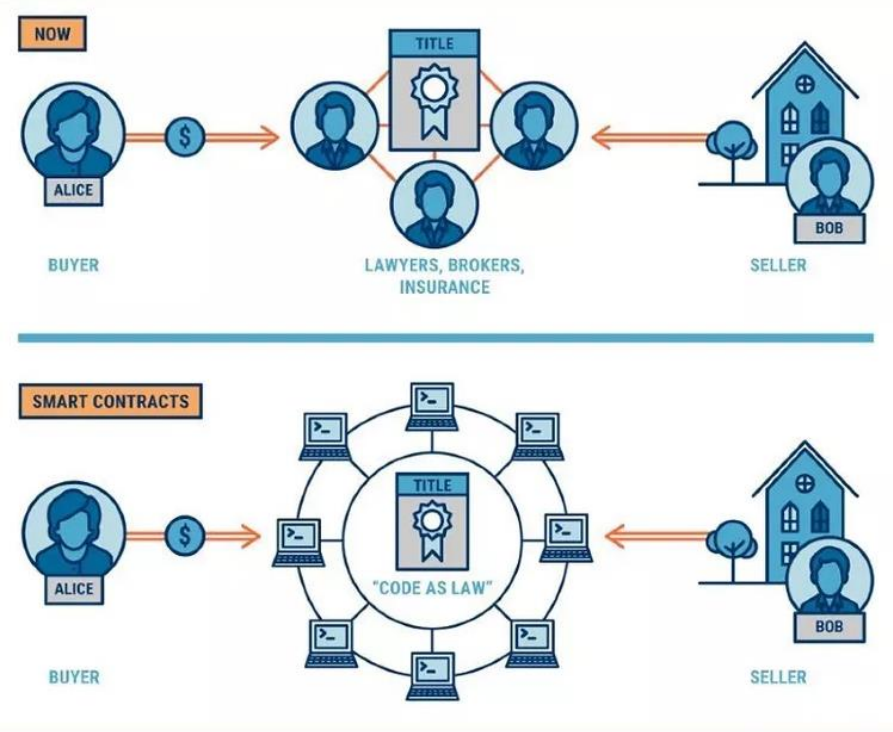


Figure 2.11: Buying a house on Ethereum [60]

Smart contracts can be used in various applications. One of the main applications of a smart contract is on crowdfunding. A crowdfunding platform is used when a group of people (e.g., a startup company) has a project and needs to raise funds from multiple investors. These investors typically set terms of conditions for their payments to be triggered. A conventional case of crowdfunding requires a reliable third-party intermediary who controls the fund flow based on the conditions set in the contract. A smart contract is an inexpensive, accurate, and secure alternative to this intermediary. Raising funds for a startup or a new project by issuing a new cryptocurrency is called an Initial Coin Offering (ICO) [59]. ICOs can help startups raise funds without dealing with financial intermediaries and regulatory compliance.

2.5 Data storage

As opposed to a centrally located server operated by a single company or organization, decentralized storage systems consist of a peer-to-peer network of user-operators who hold a portion of the overall data, creating a resilient system of file storage and sharing. IPFS (InterPlanetary File System) is a decentralized storage and file referencing system for Ethereum.

IPFS is a distributed system for storing and accessing files, websites, applications, and data. Instead of being location-based, IPFS addresses a file by what is in it, or by its content. The content identifier is a cryptographic hash of the content at that address. The hash is unique to the content that it came from, even though it may look short compared to the original content. It also allows to verify that you got what you asked [49].

We talk about “content” instead of “files” or “web pages” here, because a content identifier can point to many different types of data, such as a single small file, a piece of a larger file, or metadata. In this way, an individual IPFS address can refer to the metadata of just a single piece of a file, a whole file, a directory, a whole website, or any other kind of content.

While there is lots of complex technology in IPFS, the fundamental ideas are about changing how networks of people and computers communicate. Today's World Wide Web is structured on ownership and access, meaning that you get files from whoever owns them, if they choose to grant you access. IPFS is based on the ideas of possession and participation, where many people possess each other’s files and participate in making them available.

A desktop IPFS client, compatible with other IPFS systems worldwide, is shown in the figure below.

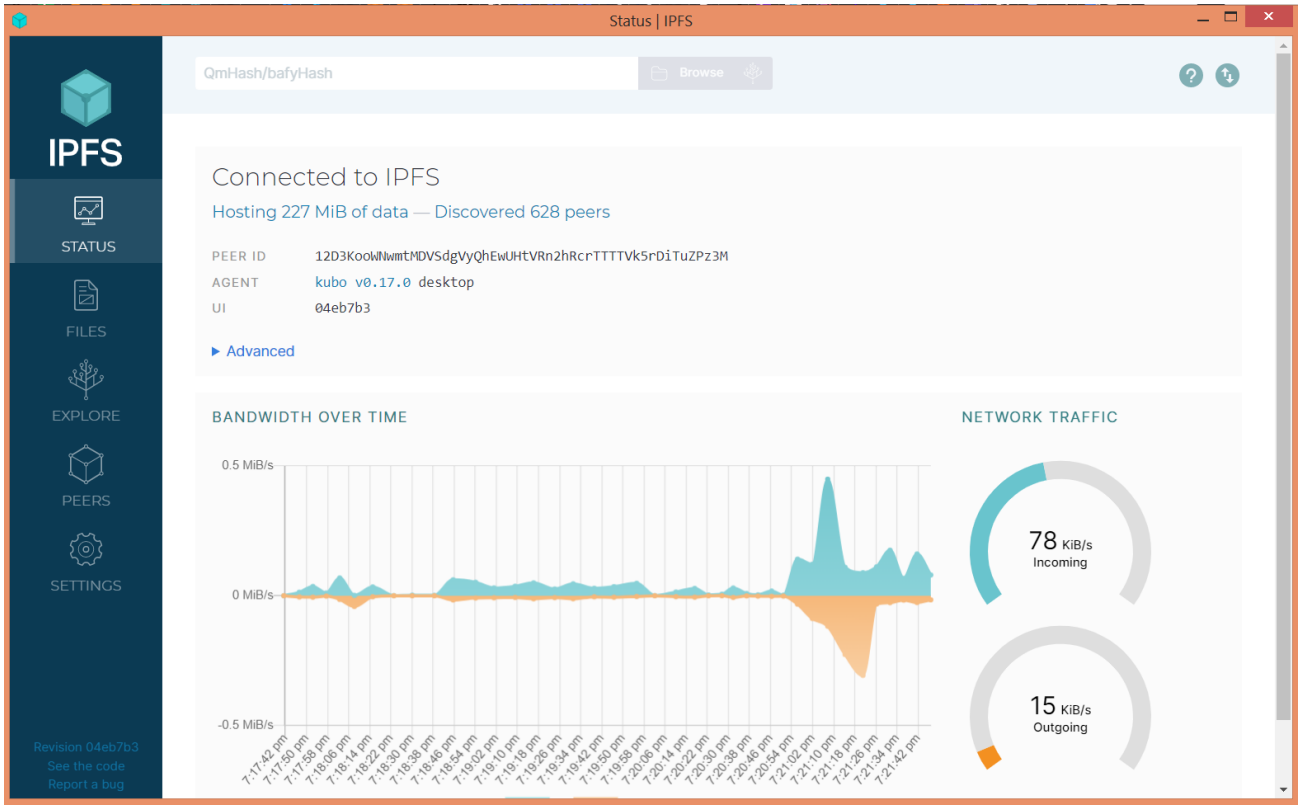


Figure 2.12: A Desktop IPFS Client

2.6 Earlier work in the Blockchain field

An interesting paper written by Baudier, Kondrateva, and Ammi [24] was very useful for his concise relevant contribution to remote voting through an analysis of the e-voting Blockchain-based concept in France and Russia.

Their paper was focused on the descriptive way, to demonstrate the benefits of technology when it is used to tie political, technical, and social actors to contribute to peace engineering (Application of science, engineering, and technology to promote and support peace.), by using e-voting Blockchain-based solutions [24].

Its strengths have based on the three key factors required for an efficient voting process: *Authenticity*, *transparency*, and *personal privacy protection of the individual*, which are promoted by Blockchain technology.

The main weakness of Baudier's paper et al. [24] is that it does not provide a concrete prototype to implement that kind of e-voting system to make a significant contribution to the engineering-driven peace concept.

In our work, we use both *descriptive* and *practical* approaches to implement a concrete e-voting prototype in the DRC's context to improve peace during the voting process.

For future works, there are many opportunities in the African context, due to the challenges of e-voting in Africa (Underdevelopment, low use of ICT technologies, the inclusion of rural Citizens, low standard of living, etc.) [13], [21], [26], [61].

Another interesting paper written by Shahzad and Crowcroft [26] focused on the practical aspects of the Blockchain such as hashing algorithm, securing data, block creation concept, block sealing, and the use of consortium blockchain to ensure the owner and avoid unauthorized access from outside. This paper focused on the way to improve the e-voting process through the security and data management in the Blockchain, by proposing a framework based on an adjustable Blockchain method. This work has given us more guidance to choose the right way to build our own prototype.

Cooley, Wolf, and Borowczak's paper [29] was also relevant and interesting because it was based on the benefits and flaws of two earlier Blockchain-based voting systems (Designed by Students at the University of Wyoming) to make them more efficient and extended to mobile voting and smart contract with more security and privacy. For us, the difference will be visible by implementing a prototype that could be deployed on any peripheral (Such as a computer, smartphone, or tablet) and, based on Blockchain technology.

The paper written by Patil, Kanchan, and Malati [27] has remarked that implementing a national e-voting system is really challenging. Also, this work has proposed a suitable e-voting model that can solve certain issues in the voting process; like vote-rigging, hacking of EVM (Electronic Voting Machine), election manipulations, and polling booth capturing.

The main goal of their work was focused on investigating the problems of the election voting system and proposing an appropriate e-voting system that solves the issues involved during the voting process.

Once again, this paper has not proposed a proper prototype for an e-voting system, and that was probably his big gap.

For our work, this paper would help us as a piece of advice, to be careful when implementing our prototype for an e-voting system, because that kind of system has very challenges when using Blockchain technology.

In a summary, we have seen a convergence of all the authors, who were convinced that the use of Blockchain technology as a service can handle 90% of some traditional voting systems issues, ensure anonymity for voters and facilitate public inspection (Like witnesses) because the violability of that kind of system is very reduced.

Chapter 3: Research methodology

During our research, we will be focusing on the alternate way of using Blockchain technology rather than its primary use in cryptocurrencies [59], [62], [63]. Thus, we use this technology in the voting process to ensure that voters' votes have not been tampered with and are safe and truly transparent. There are lots of other ways that blockchain can be used, but for this research, we have only chosen one way to make our contribution useful to society through the improvement of the e-voting process.

In this chapter, we are going to explain our architectural design, data collection technique, System Analysis, and design methods, but to understand the way we are going through, we will also explain some technologies and tools to be used during the building step of the e-voting prototype.

3.1 Architecture design

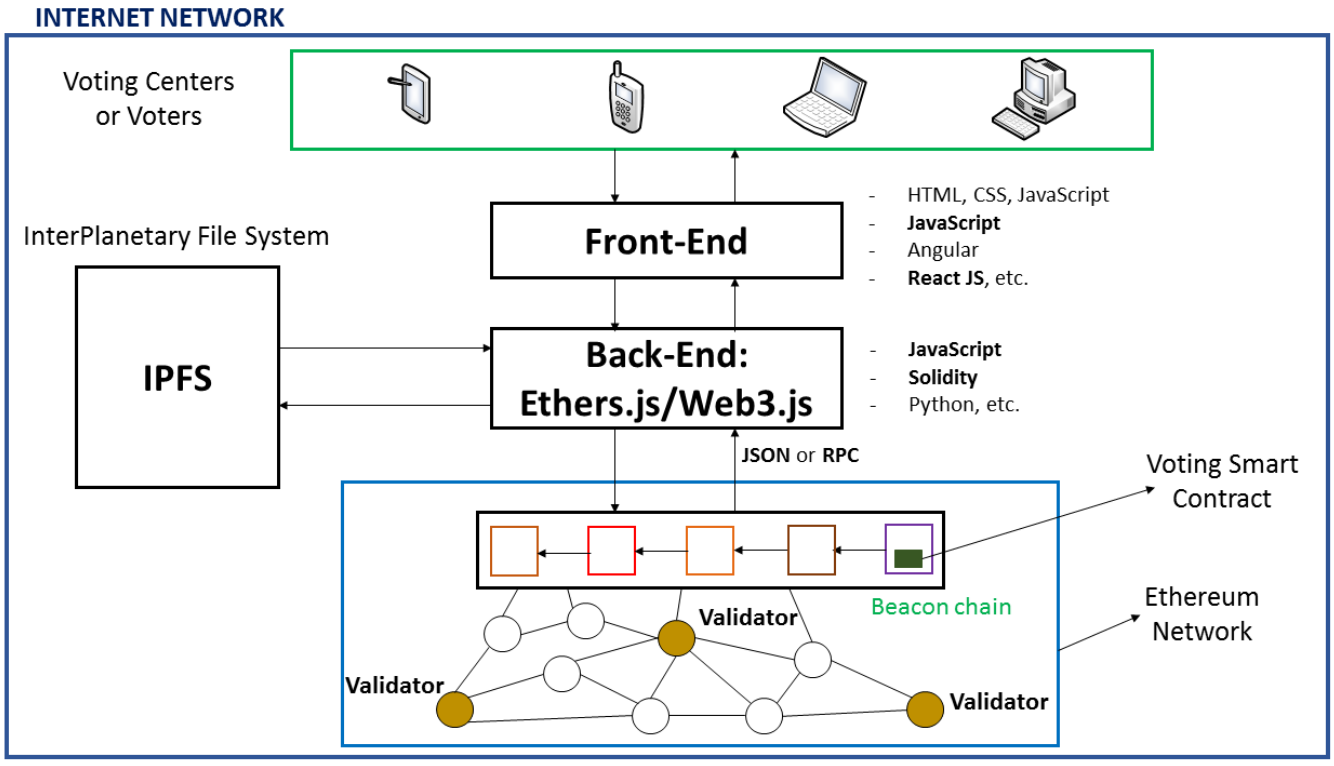


Figure 3.1: System Architecture Design

3.2 Technologies and tools

To be able to make our prototype functional (A DApp application Stands for Decentralized Application), some technologies and tools will be useful to reach our goal.

Firstly, we use a Web-based Blockchain development (With Frontend to handle user interactions on a User Interface, and the Backend to manage core interactions, data processing, and data flow); secondly, we use the following tools:

- **Frontend tools**: *ReactJs* or *React* in short (For better client-side rendering) or *NextJs* (For better server-side rendering), *HTML*, *CSS*, and *JavaScript*.
- **Backend tools**: *Ethereum Blockchain Network* or *Ethereum Network* (Which works as a decentralized immutable and public storage), *Smart contract* (To be running on the Ethereum Network and written with *Solidity* programming language) using Remix IDE (A web-based Integrated Development Environment for Solidity).

In this section, we give an overview of these ones in detail.

3.2.1 Front-End tools

ReactJS is a JavaScript framework trying to handle properly the View layer part. It can very well be defined and used as the V in any of the MVC frameworks. It is not opinionated about how it should be used. It creates abstract representations of views. It breaks down parts of the view in the Components. These components encompass both the logic to handle the display of view and the view itself. It can contain data that it uses to render the state of the app [64].

React is a JavaScript framework library for building user interfaces. It has three main characteristics: It is declarative, Component-Based, and respect the principle “Learn Once, Write Anywhere” [65].

- Declarative: React makes code painless to create interactive UIs. Makes able to design simple views for each state in your application and it will efficiently update and render just the right components when your data changes. Declarative views make your code more predictable and easier to debug.
- Component-Based: React build encapsulated components that manage their own state, then composes them to make complex UIs. Since component logic is written in JavaScript instead of templates, you can easily pass rich data through your app and keep the state out of the DOM (Document Object Model).

- Learn Once, Write Anywhere: We do not make assumptions about the rest of your technology stack, so you can develop new features in React without rewriting existing code. React can also render on the server using Node and power mobile apps using React Native.

3.2.2 Back-End tools

In Ethereum, there are three main languages available to write Smart contracts: Solidity, Vyper, and LLL. From all of them, Solidity is the official and most widely used language in the Ethereum Network. Using it, Smart contracts are written that are agreed on between two parties. It may seem like JavaScript, but it is more like Java for its statically typed feature. These contracts can be validated using Remix [66], a browser-based IDE with an integrated compiler. Solidity comes with its own compiler that generates machine-level bytecode that can be run on the Ethereum Virtual Machin or EVM [67].

Every contract has an ABI (Application Binary Interface) described by the JSON (JavaScript Object Notation) Interface, which is pretty much like an API that works as an interface between the high-level language and the lower-level binary code that gets processed by dumb computers. The ABI consists of the following:

- All function names.
- Input and output types of functions.
- All event names and their parameters.

The web3.js library is a collection of modules that contain functionality for the Ethereum ecosystem [68]. It makes us able to interact with Smart contracts through a web front-end.

Web3.js is a collection of libraries that allow interactions with a local or remote Ethereum node, using an HTTP (HypertText Transfer Protocol) connection. Simply speaking, it provides JavaScript APIs to communicate with Geth in production or the ganache-cli test network, and the Ethereum Mainnet or the Ethereum Testnet. It uses JSON-RPC (JavaScript Object Notation-Remote Procedure Call) internally to communicate with Geth/ganache-cli, which is a lightweight Remote Procedure Call protocol [68]. Most of the time, Ethers.js [69] and Hardhat [70] are preferred because they provide more customization instead of using Web3.js with Ganache [71] and Truffle [72].

For this work, our choices are Ether.js and Hardhat. Hardhat plays the role of a local Ethereum Blockchain on a PC with some features customizable directly using NodeJs and,

making it easier to perform Unit Tests of Smart Contracts in a single place (Through a Hardhat Project) with Visual Studio Code IDE [73]. Ether.js is a JavaScript library that allows us to interact with the Ethereum Blockchain and its ecosystem directly in a Hardhat project.

3.3 Prototyping

When we first started, we were willing to build a functional system to test a blockchain-based e-voting system. In order to make this possible, we have focused our software development on prototyping, which is a strategy that promotes the creation of applications by breaking them down into smaller parts to be simple to construct and comprehend [74]. In other words, prototyping may be seen as a software development technique that allows evaluating whether the final system can be feasible with core features functionalities or not [75].

The direct implication is the building of a functional prototype that can work within on production environment for testing purposes and enable further improvements while the system is working. That was the case for our Blockchain technology-based e-voting system which has tested on a narrow scope as we mention at the beginning of this research.

3.4 Data collection

To conduct a research thesis in a proper way, it is important to base it on a certain amount of data to have a good understanding of the final output of the work to be done. During this phase, data must be collected to gain pertinent information for the study.

Data collection is the process of gathering data, further measuring, processing, assessing, and analyzing for research purposes. It's conducted with the help of established, validated techniques, which make it possible to answer research questions, test hypotheses, and evaluate results [76]. The main goal of data collection is to get access to reliable sources of information that will provide data for further analysis and make data-driven decisions possible.

Before starting data collection, it is crucial to understand what type of data should be collected (What), what timeframe should be allocated for this task (When), how this should be done (How), and with which methods or techniques. For this research, the 'what' relates to voting systems in the African continent, and in DRC to definitely be focused on a narrow

scope only, then the ‘when’ represents the few months given to us to be able to finalize the final system, and the ‘how’ implements a working system (Called in our work a prototype, or functional prototype) which will be able to answer to the main research question asked at the beginning of the work.

For this work, the main data collections techniques are the following:

- **Conceptual Modeling**: This will allow us to draw some diagrams to show important parts and interactions of the system. It is mainly held using the UP (Unified Process) method with UML (Unified Modeling Language) language.
- **Existing data**: By exploring existing data from other researchers in the field of electronic voting (Documentation and observation techniques), we have to explore what they have already done, and what is missing to add our contribution.
- **Experiment**: This will be linked to the prototype's testing to determine whether aspects of a Blockchain-based e-voting system would be more relevant than others.

Our work's output won't be exhaustive since we want to be clear in showing how Blockchain technology may make apps safer and more helpful, even if they are hard to develop.

3.5 Software Development Life Cycle Model

According to Iqbal [77], Software Development Life Cycle (SDLC) Models are frameworks used to *design*, *develop,* and *test the software project*. It specifies the methods to be followed during the software development process.

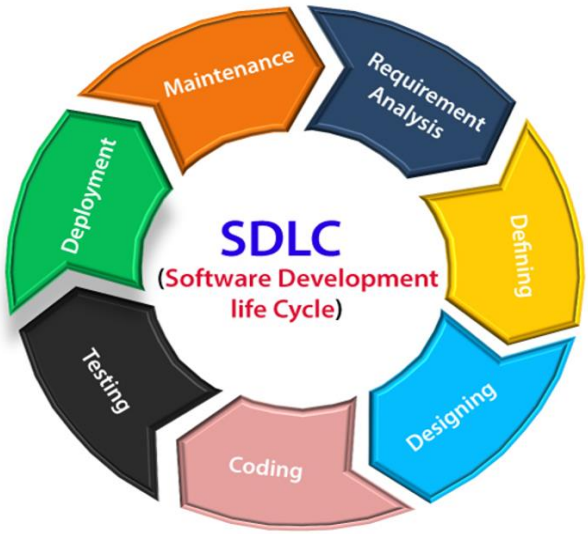


Figure 3.2: Stages of Software Development Life Cycle [78]

These stages are explained as follow:

- **Planning and requirement analysis**: Helps to understand what is needed for the final system. For the voting system, this stage was useful to understand what the system should do to get a proper voting platform.
- **Defining requirements**: The main effort here was to analyze the requirements' implications to form the initial software system model. That has also held to have a deep understand of how the final Decentralized App should works.
- **Designing the software**: This stage involves the detailed definition of the outputs, inputs, and processing procedures, including data structures and databases (Blockchain), software structure, etc. At this point, we've created various diagrams to describe the essential aspects of the system (All diagrams are given on Chapter 4).
- **Developing the project**: We put the idea into code in this step using the frontend and the backend programming languages.
- **Testing**: Following the completion of the coding process, system tests are performed. The basic purpose of testing is to find as many software defects as possible to reach an acceptable level of software quality after remedies are made. Backend Unit test was carried out using the Mocha and Chai libraries.
- **Deployment**: We tested our system on localhost (which means local Blockchain) and on Ethereum Testnet (Rinkeby Testnet and Polygon as scalable solutions), however, the final solution should be recorded on Mainet to operate with the production environment.

  The frontend was tested both locally and remotely (through the Netlify platform) to get a sense of how things perform in production.
- **Maintenance**: Regular software operation begins once installation and conversion have been completed. Throughout the regular operation period, which usually lasts for several years or until a new software generation appears on the scene, maintenance is needed. Maintenance helps to improve software product.

Among the numerous SDLC Models, we are going to use the Waterfall Model for this project. The waterfall model is a continuous software development model in which development is seen as flowing steadily downwards (like a waterfall) through the steps of

requirements, *analysis*, *design*, *implementation/development*, *testing* (validation), *integration*, and *maintenance or deployment*. This means that any phase in the development process begins only if the previous phase is complete.

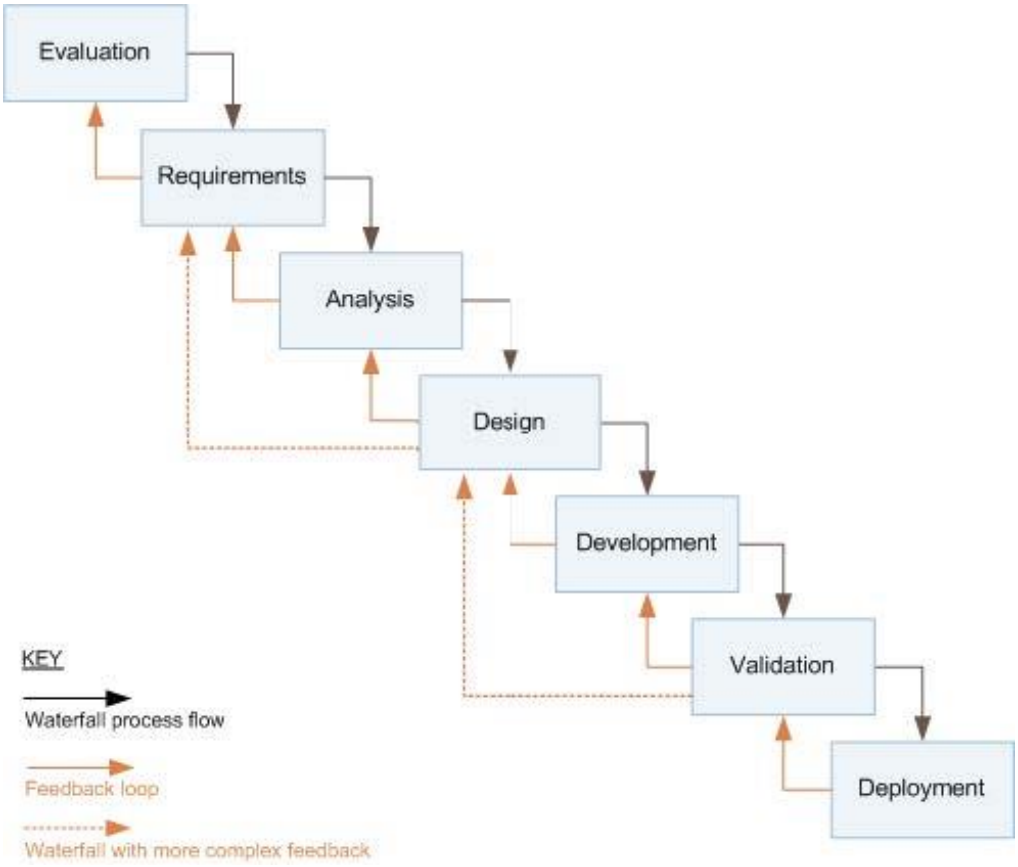


Figure 3.3: Waterfall Model [79]

Chapter 4: System analysis and design

The system analysis and design define the most relevant models of diagrams to be used during the implementation of the system.

Thus, we define the *Use Case diagram* (Which gives us different scenarios involving both the User and the System), the *Activity diagram* (Which provides steps involved when the User interacts with the System), the Class diagram (Which shows the main components of our application's design) and finally the Deployment diagram (Which shows components involved in the entire system).

4.1 Use Case diagram

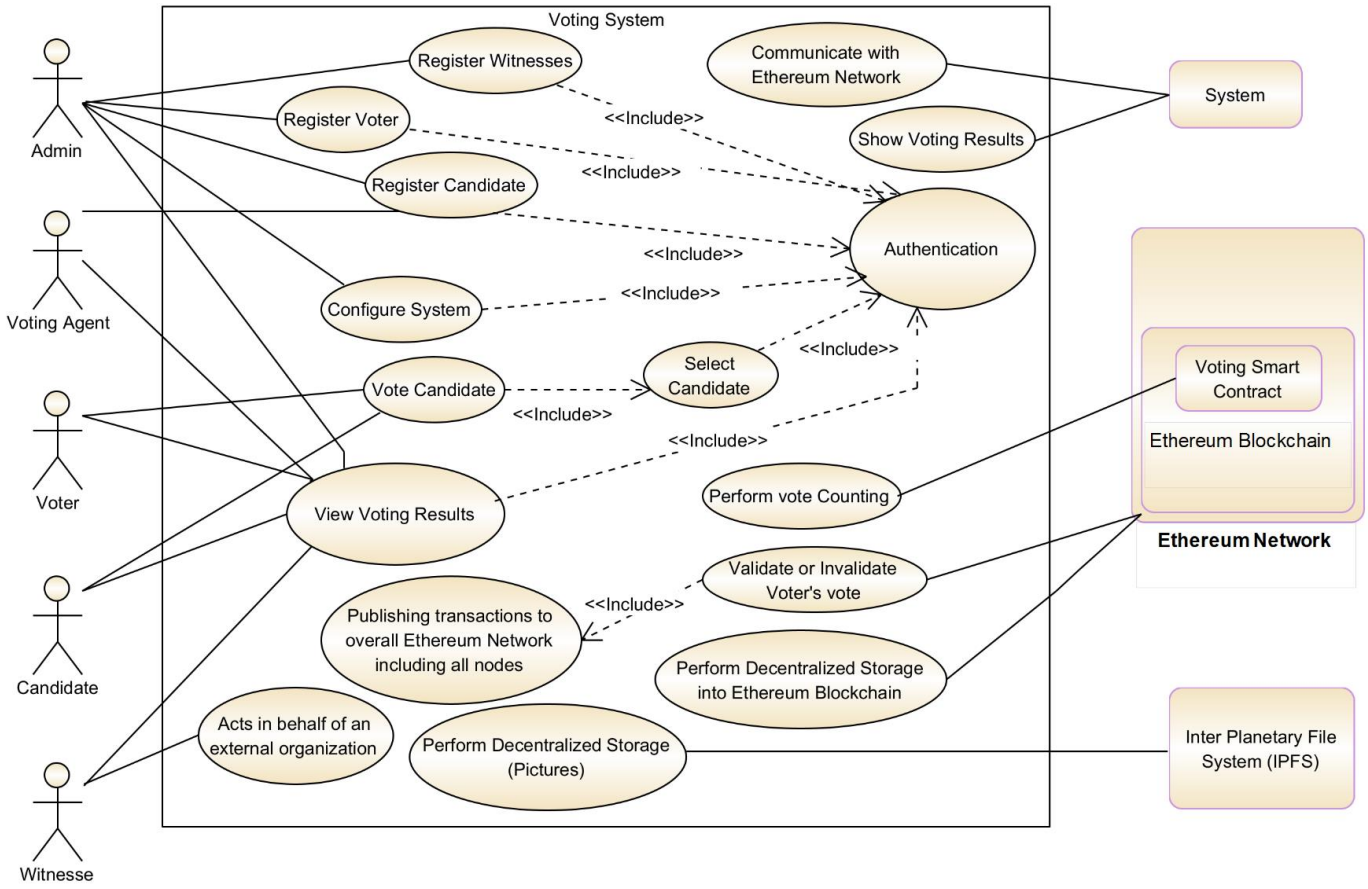


Figure 4.1: Use Case diagram

4.2 Activity diagram

4.2.1 User Authentication

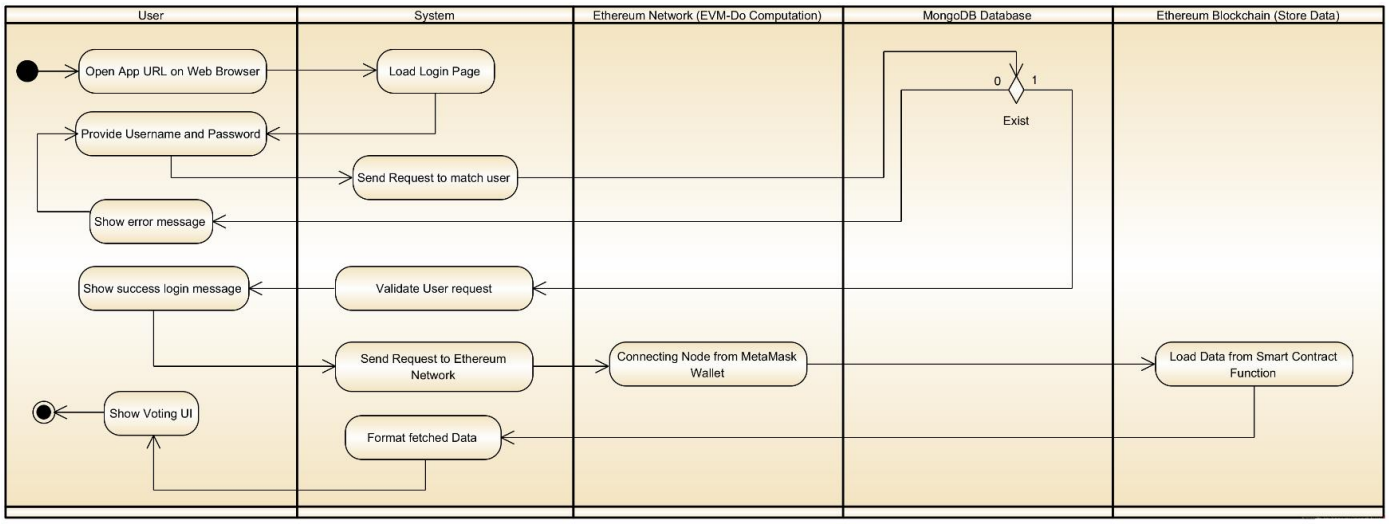


Figure 4.2: User Authentication

4.2.2 Voting Process

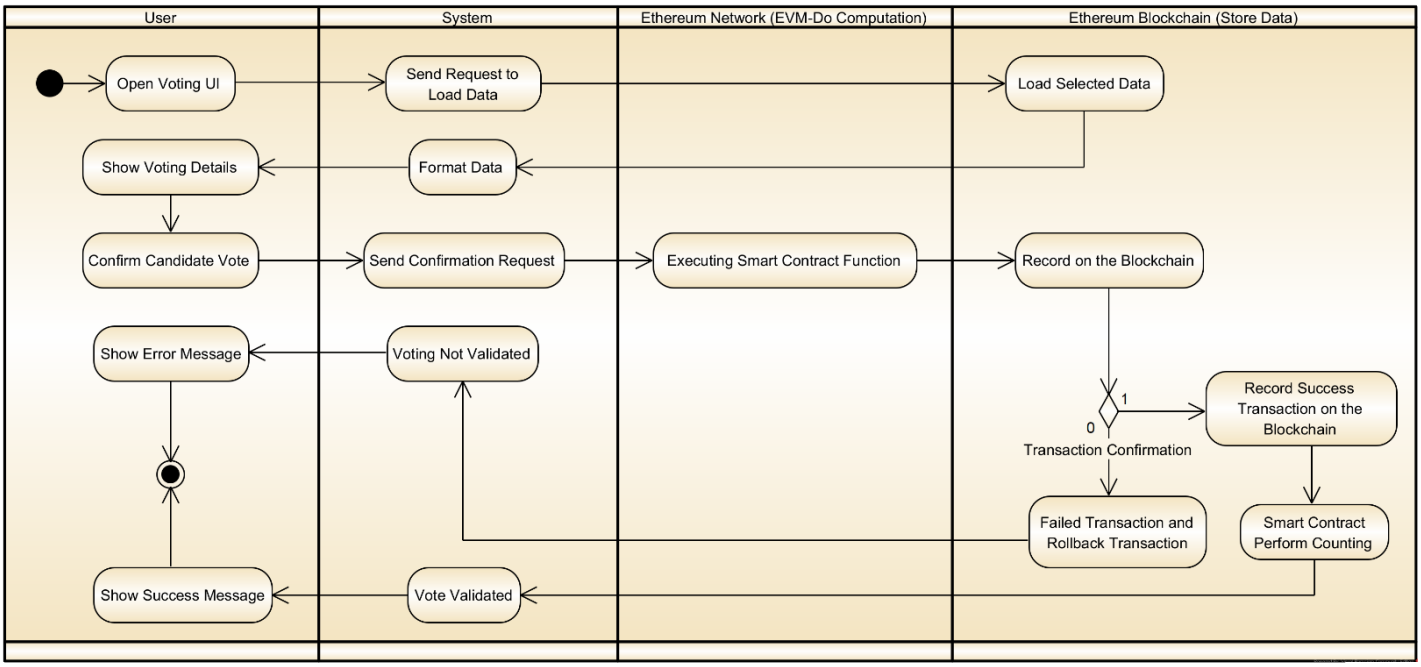


Figure 4.3: Voting Process

4.2.3 Show Voting Results

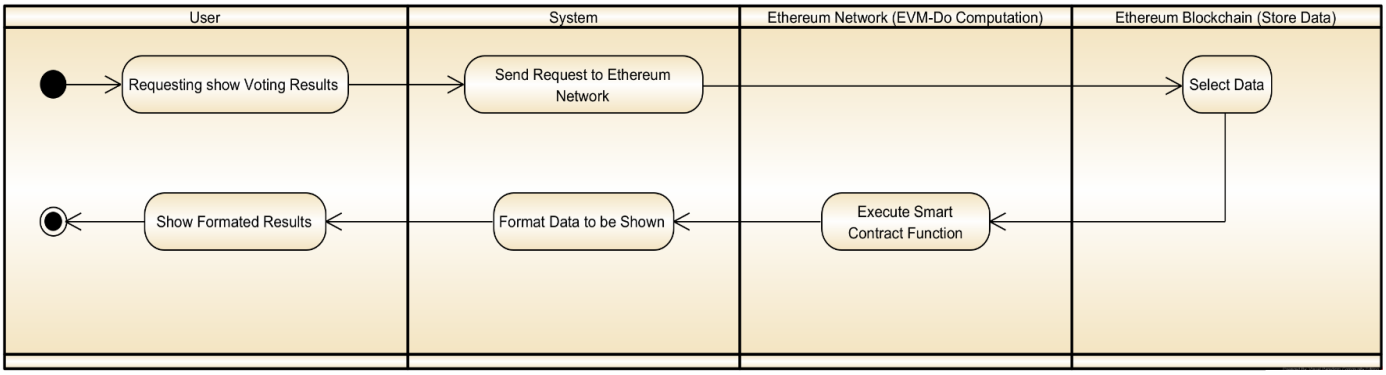


Figure 4.4: Show Voting Results

4.3 Class diagram

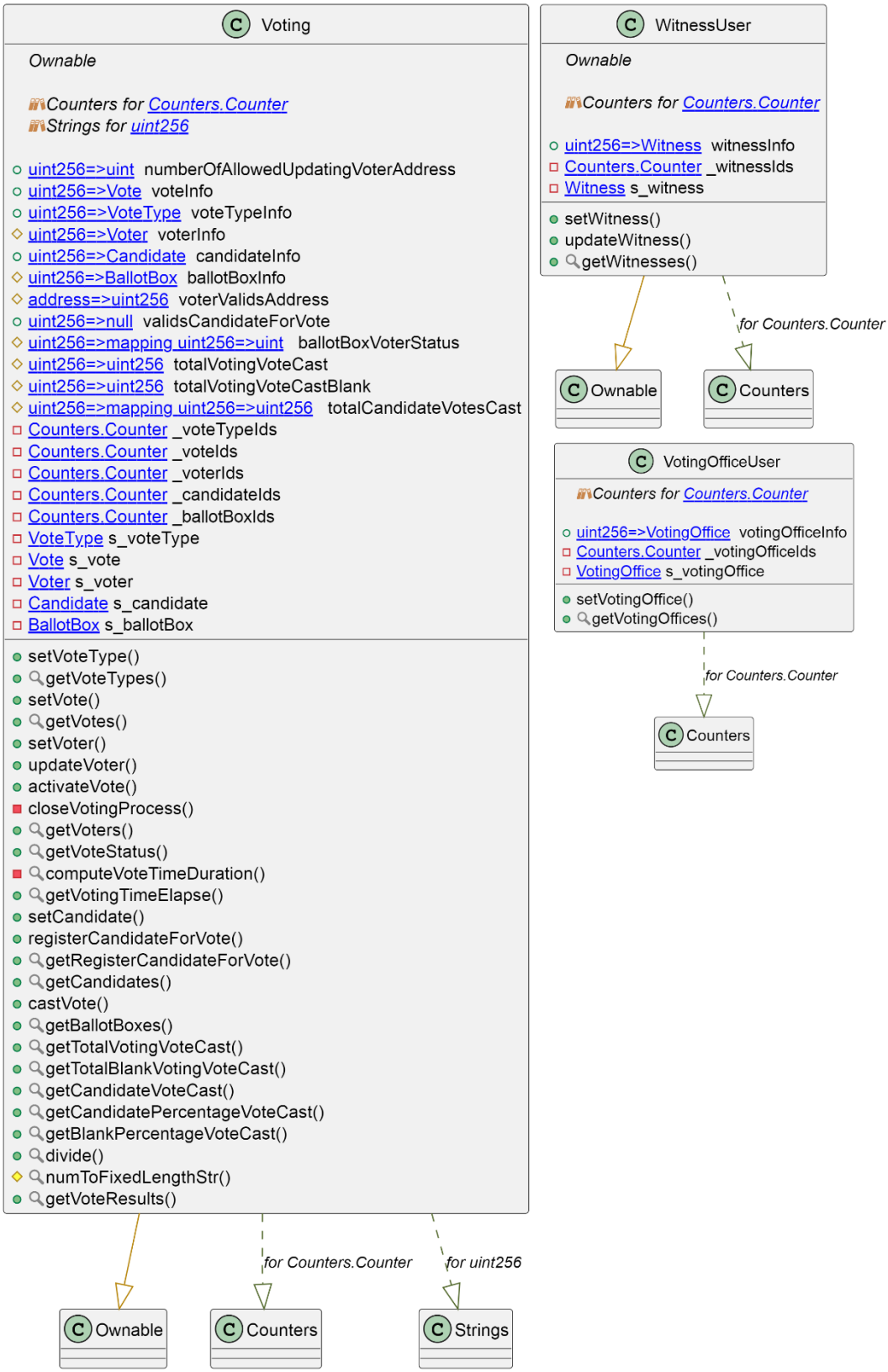


Figure 4.5: Class diagram

4.4 Deployment diagram

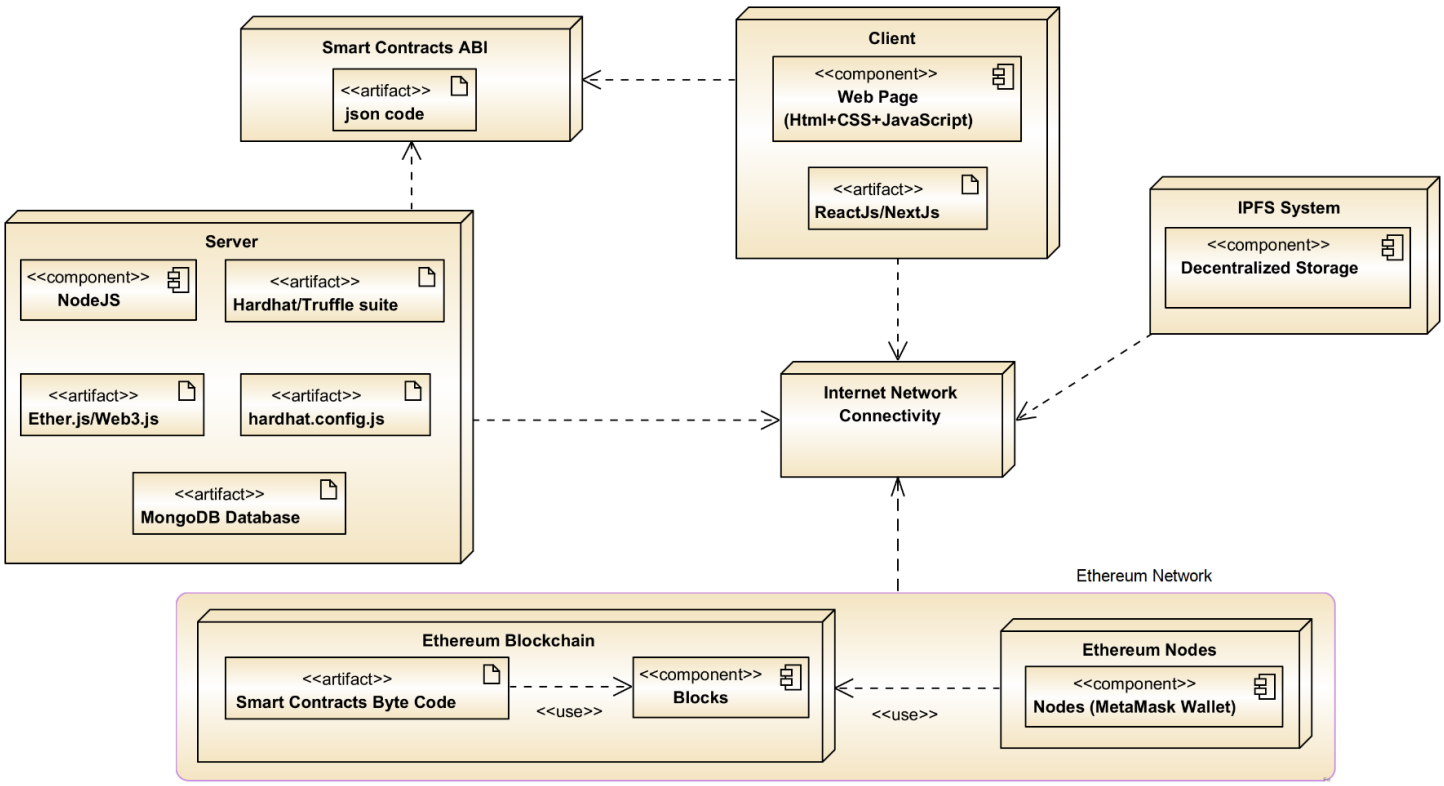


Figure 4.6: Deployment diagram

Chapter 5: Results and Analysis

The implementation of the e-voting DApp prototype is handled in this section of our work. It clarifies the general concept of the project.

To understand how this system can enhance the current voting system in terms of credibility, consistency, and tamper-proof characteristics, the application output will be supplied and explained in this chapter.

The UI, or user interface, help our readers fully comprehend how our Decentralized Voting program works and how it achieves the desired outcome.

5.1 Login form

The application's entry point is represented via the login form. Only the username (email address) and password are used. To prevent the establishment of false accounts, the Administrator has to create these two details all at once.

5.2 Register Vote type

This section aids in adding a Vote type for a certain vote. Every new record needs a gas fee, and then MetaMask displays a confirmation message. To be able to vote, a new vote type should be added, and his ID number will be used during the voting process. For each new record, we should pay fees to validate the transaction on the Blockchain, which is why a confirmation popup will appear to inform us of the cost of the transaction: This price may fluctuate periodically based on network congestion (More congestion will increase the gas, and the cost, but less congestion will decrease them). These kinds of actions will be followed for each screenshot of our application.

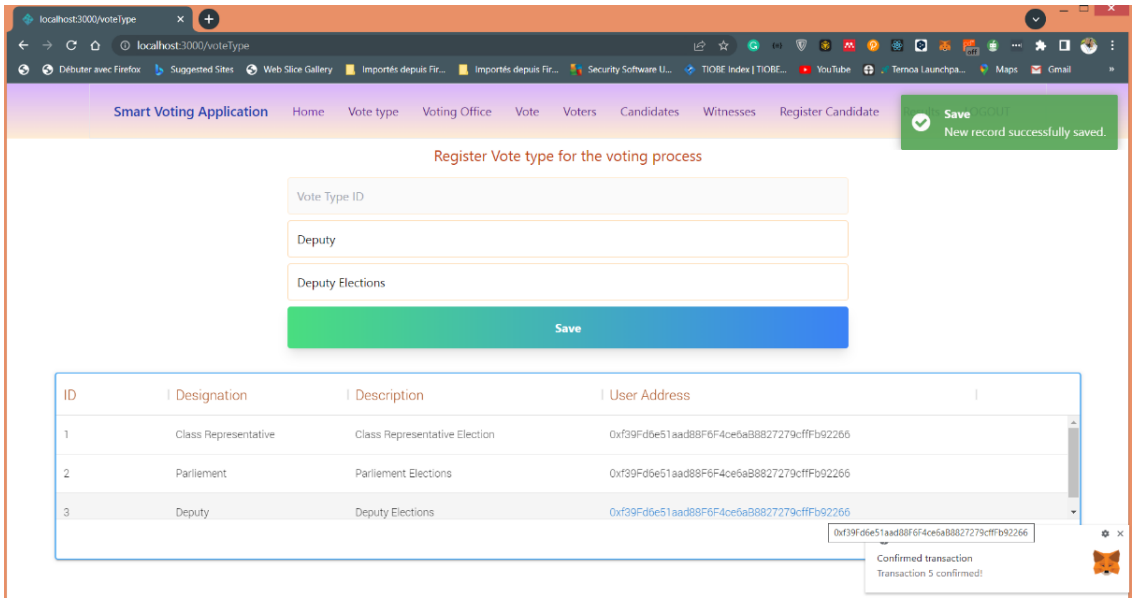


Figure 5.1: Register Vote type for the voting process

5.3 Register Voting office

Voting Offices are where the voter will cast their vote and in the case of this DApp, voters are able to cast their vote even at home or wherever they need to do so. (**That is an added value compared to the current voting system**).

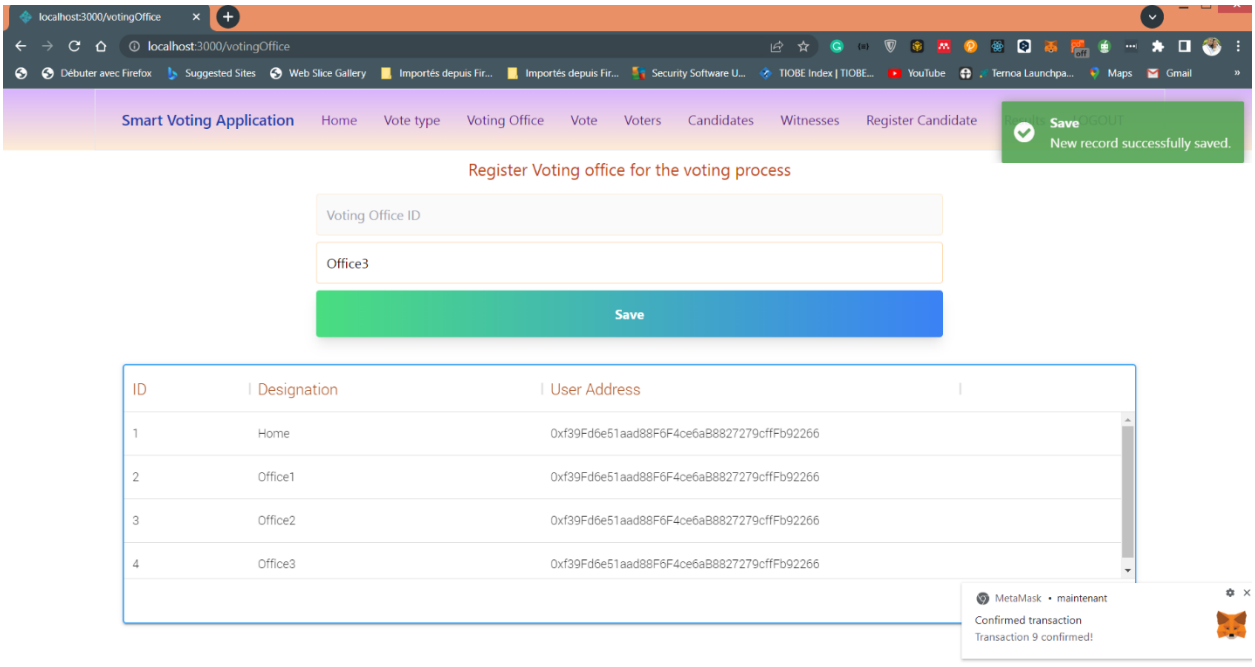


Figure 5.2: Register Voting office for the voting process

5.4 Register and activate Vote

Once a vote has been registered and activated, the voting process will be accessible. When a vote is registered, the time allotted for the voting procedure is always specified. In this project, minutes are employed as the unit of time for simplicity (e.g., 60 minutes, 10 minutes, etc.). Each vote should be linked to a Voting Type.

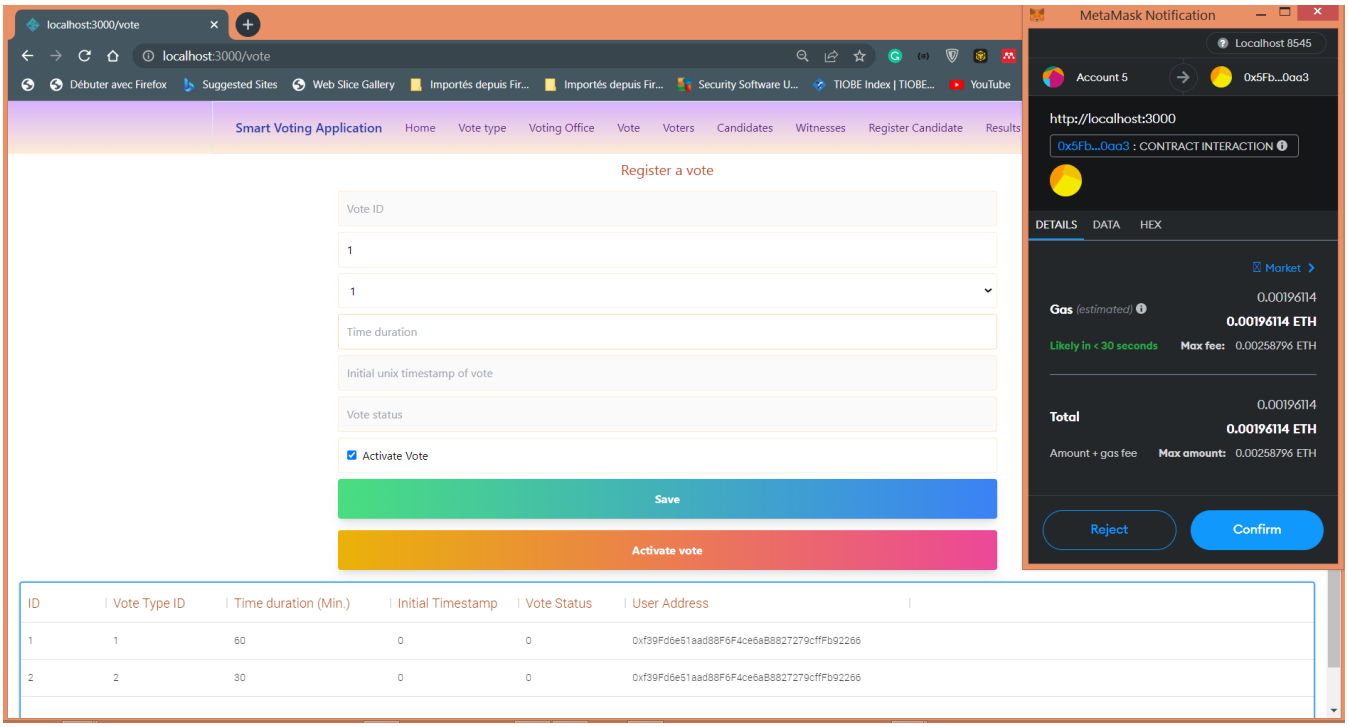


Figure 5.3: Register and activate Vote

The voter should first register before casting his vote (For that, he will receive a unique ID), otherwise, he can't do that. Before the voting process begins, the electoral commission will oversee registering every voter.

5.5 Register Candidates

Each candidate is registered with their full name, their photo (which is displayed when voters select their candidate ID), and their order number. Instead of utilizing his order number, the unique ID (Which is also used by voters to cast their vote) was used, to ensure the unicity of the candidate. A candidate should be first registered for the anticipated voting process before being eligible for it. Finally, once voting has begun, no more candidates could be registered

to the same vote, but one candidate could be eligible for more than one vote (Depending on circumstances). In this stage, the physical link of the candidate's photo is stored on an IPFS storage system, and this picture could be retrieved later easily through this link (This will be handled directly by our DApp).

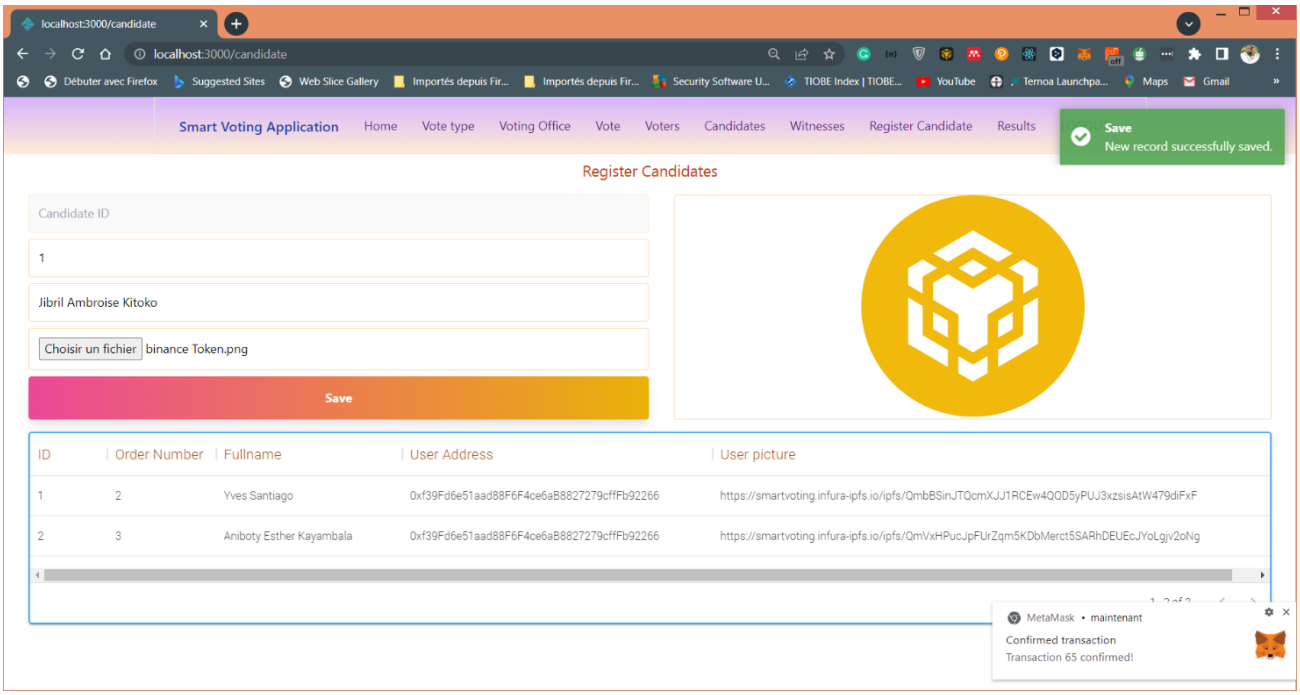


Figure 5.4: Register Candidates

5.6 Register Witnesses

Witnesses play a crucial role in ensuring that elections are free of fraud and that the entire voting process is conducted correctly. The Blockchain's immutability (Temper proof mechanism) also serves as a witness for the full voting process in this project (**One more benefit of our system compared to the actual one, and according to our objectives**).

5.7 Cast new vote

In this DApp, it is very easy for voters to cast their vote anywhere because the application is to be available on the Blockchain via the internet. Also, by using any smart device (Smartphone, tablet, or Computer), the user can vote (**This is again a benefit of our application compared with the current voting system**).

First, the voter chooses the candidate, then specifies his ID number, then the Voting Office ID, then the Vote Type for the corresponding vote, then the Vote ID, and finally the choice for his vote (The choice 1: represents a positive vote for the selected candidate, and 2: represent a blank vote even if a random candidate is selected). For security purposes and to enforce anonymity, the voter's address has hidden.

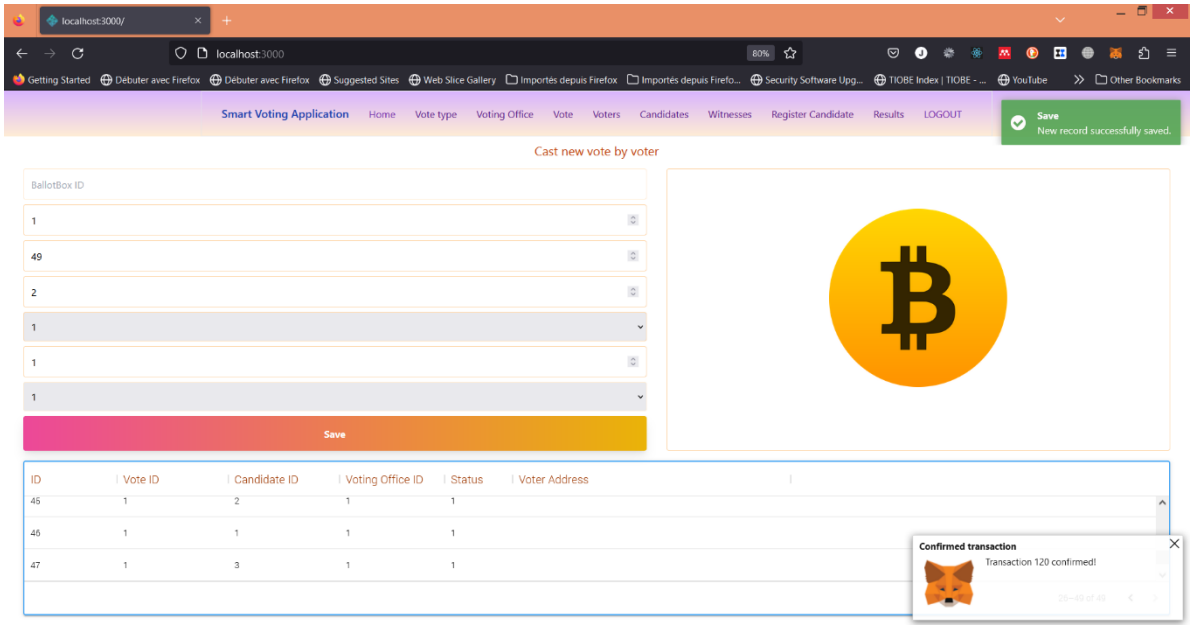


Figure 5.5: Cast new vote by voter

Following the next voter's vote, after the voting period has expired, the voting mechanism is closed automatically. The below screenshot shows how the system handles internally the end of the voting process without any human intervention (**One more benefit of our system compared to the actual one, and according to our objectives**).

5.8 Voting Results

A major part of this system according to our objectives is the real-time calculation of the voting result anytime and anywhere for all voters: Our DApp makes them available while the vote is going on, and each party involved in the voting process could see them without any cheating.

Before showing how it looks like in our application, we are going to show first the way we are conducting this vote: We use **49 voters** (To have the decimal part) and **03 candidates**

(With respectively following order numbers: 2, 3, and 1). The bellow table shows us how the vote will be going on.

Vote for candidate **1** is highlighted in **red**, candidate **2** in **green** and candidate **3** in **blue** to help the lector to distinguish them easily. Blank vote (-) will be in **black**.

Candidate ID (Vote for)	Voter ID	Voting Office ID	Vote ID
1	1	2	1
3	2	2	1
-	3	2	1
2	4	1	1
2	5	1	1
1	6	3	1
3	7	3	1
1	8	2	1
2	9	2	1
2	10	1	1
1	11	1	1
-	12	1	1
-	13	1	1
3	14	1	1
-	15	1	1
1	16	1	1
2	17	1	1
1	18	1	1
1	19	1	1
3	20	3	1
2	21	3	1
-	22	3	1
1	23	2	1
-	24	2	1
3	25	2	1
2	26	2	1

2	27	2	1
-	28	2	1
1	29	2	1
2	30	2	1
3	31	2	1
1	32	1	1
1	33	1	1
2	34	1	1
-	35	2	1
3	36	2	1
2	37	2	1
2	38	2	1
1	39	2	1
3	40	2	1
3	41	2	1
1	42	3	1
2	43	3	1
-	44	1	1
2	45	2	1
1	46	1	1
3	47	1	1
-	48	2	1
1	49	2	1

Table 5.1: The way vote will be going on

The percentage and the number of votes cast for each candidate are given in the below table: The winner is highlighted in **green** (***Is the candidate 1 with 30.61 percent***).

	Candidate			**Blank Vote**
Order Number	1	2	3	
Vote	15	14	10	10
Percentage	30.61%	28.57%	20.41%	20.41%

Table 5.2: Manual vote Results Calculation

Finally, we do the same calculation with our voting DApp through the User Interface (By specifying the Vote ID) to see how the same result will be found by using our application. To see the results, the voter should only specify the Vote ID and then load the results.

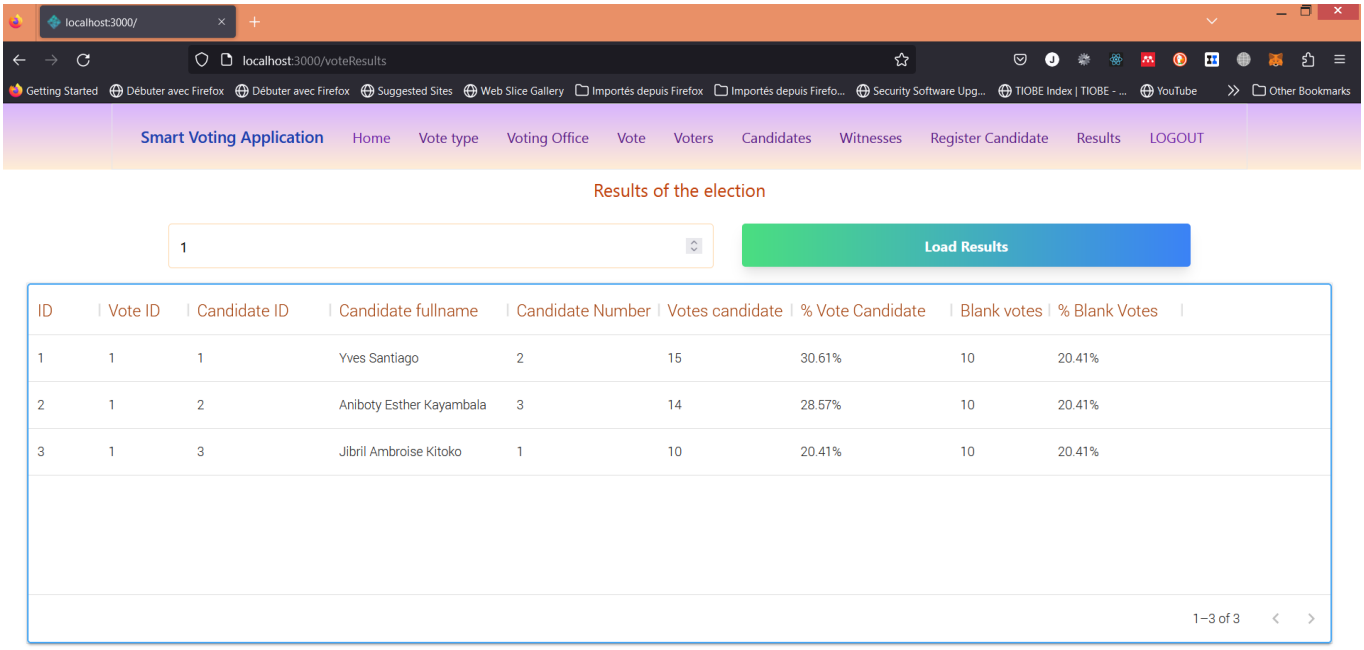


Figure 5.6: Results of the election

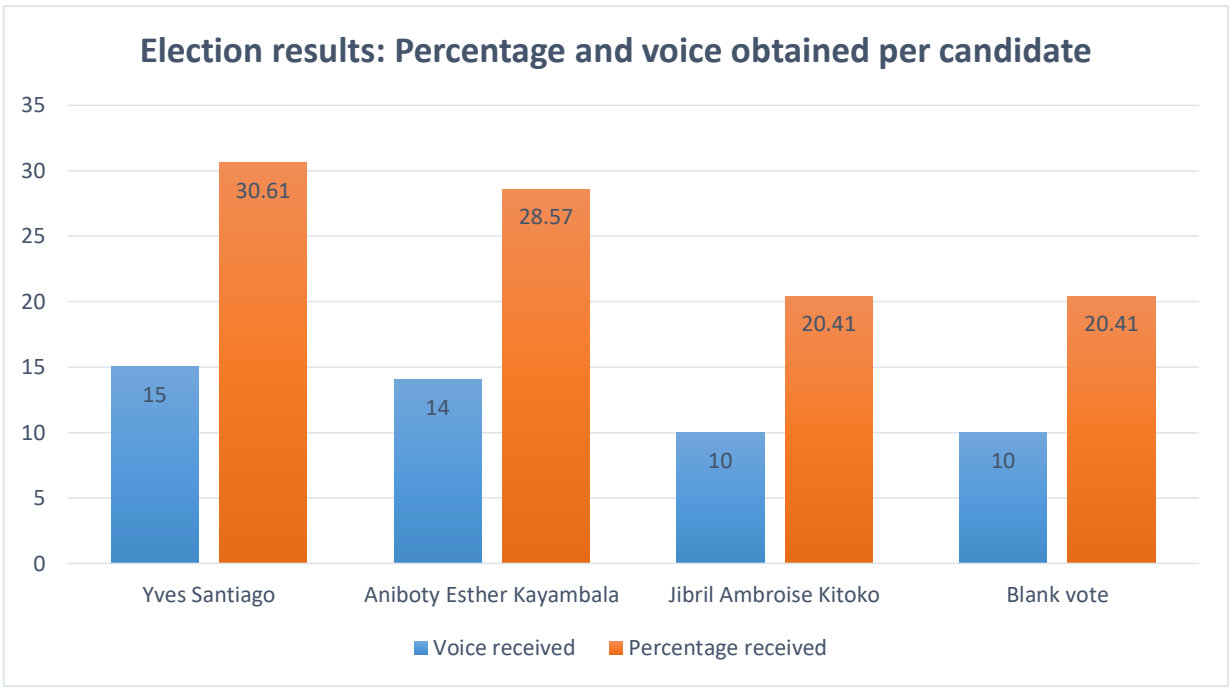


Figure 5.7: Election results - Percentage and voice received per candidate

5.9 Result Discussions

As we state at the beginning of the research, our investigation was only narrowly targeted because it would be challenging to cover a large area in a short period of time.

Researchers have discovered difficulties with the voting system on the African continent, specifically in the Democratic Republic of the Congo, as we begin to examine the literature review. These challenges implied looking for the best solutions to conduct democratic elections which would be made stakeholders involved in the voting process confident by the entire process, and mainly by its truth. Electronic voting was the main of the suggested methods to accomplish this, among the other alternatives they provided.

Because of examples of cheating and data tampering employing centralized systems and databases, most of these systems were incredibly helpful but had no meaningful results. Once more, this was a novel approach for researchers to go further and suggest a more efficient system to add an additional layer of confidence to the entire voting process.

It was in this way that some researchers have tried to explore Blockchain technology. Most of them have found it more relevant, because of its character of immutability: Once a record is stored on the Blockchain, no one can alter it without rebuilding the entire chain (Which involves a very huge amount of computation impossible to reach), and this aspect was a key feature for the future of e-voting systems in the world. Researchers have explored the security and privacy aspects involved when the voting system is coupled with Blockchain technology, but a working prototype was not provided to demonstrate how this kind of system may function and be helpful to the electoral process: *This was the key answer by this work*.

Our project was created with the goal of fully developing an electronic voting system that can produce voting results in real-time for each stakeholder involved in the voting process (Voter, Witnesses, and Candidates) while guaranteeing the accuracy of the entire process: Because no one could be able to change voting results, and everything is transparently shown for them. In addition, each voter is able to vote even using a smart device or a laptop at home, at a voting office, or abroad, without any way to alter the result of the vote. After, our experimentation through the web decentralized application we have built, we have seen that the result is updated dynamically by a Smart Contract deployed on the Ethereum Blockchain, and *the entire system has become a safe way to conduct elections anywhere, with a high level of confidence* avoiding contestations, election post-violence, and death.

The charge we must pay for each transaction conducted on the Blockchain can be interpreted as a constraint of this system, but this challenge can be overcome by employing Layer 2 solutions or scaling Ethereum alternatives. (Like Polygon Blockchain for example, etc.) which reduces the cost involved.

Chapter 6: Conclusion and Recommendations

6.1 Conclusion

The goal of this work was to demonstrate the usefulness of Blockchain technology, mainly on e-voting by building a full flesh Decentralized Application running on it, and with a temper-proof behavior on its structure.

Our motivation in this work was to avoid as much possible fraud and tempering involved in a centralized system and, protect voter identity and privacy by introducing a decentralized system, where data are distributed and owned by each node inside the network system: The advantage of this system is to convince stakeholders (Voters, Candidates, and Witnesses) involved in the voting process by its truth and introduce a high level of transparency.

To be able to reach our goal, we have used a lot of tools and technologies for both backend (Solidity, IPFS, Infura API, Coinmarketkap API, Ether.js, etc.) and frontend (ReactJs, HTML, Tailwind CSS framework, CSS) to build working application on the Ethereum Blockchain.

For this purpose, our work was breaking down in five chapters: The first one was focused on Context of the study (To understand the purpose of our work), the second one was focused on Literature review (To understand what others researchers have found), chapter three was Research methodology (Where tools to be used was explained in brief), chapter four, System analysis and design was given to us main design and architecture of our system, chapter five, Results and Analysis was focused on the output of our system and final outcome. Finally, chapter six was focused on conclusion and recommendation to "close the door opened at the beginning of this work": (Using Blockchain technology to sustain peace).

6.2 Recommendations

During this research, we have not able to do the earlier part of the voting system, but only the last stage of counting, because the identification would involve a lot of things to take care and it will be difficult to complete this work on time to obtain our final grade. For example, we will need to include biometrical identification, facial recognition, and even iris identification to be sure that each possible elector's information is safely saved. Finally, this would make

our web app more challenging because even for this project, it was a great challenge to finalize with both frontend and backend.

We strongly encourage future researchers to investigate this area and incorporate it into their work to have a fully functional electronic voting system from voter registration to the ballot box stage and receiving the final vote results. This will make Blockchain more useful for the entire world.

The field of Blockchain is very large and they are a lot of useful areas to explore. A potential enhancement to this project would be the addition of a national biometric card. Rather than providing his ID number, a voter might insert a biometric card, and the system would automatically load his information and authenticate him after inputting a secret password. Furthermore, we can also explore other fields like Patient Management Systems, or Academic Certificate management Systems using Blockchain technology. This is not an exhaustive list.

List of References

[1] T. Vircoulon, "Unfair and Dangerous Elections," *Peace Rev.*, vol. 23, no. 2, pp. 199–204, Apr. 2011, doi: 10.1080/10402659.2011.571610.

[2] U. Daxecker, E. Amicarelli, and A. Jung, "Electoral contention and violence (ECAV): A new dataset," *J. Peace Res.*, vol. 56, no. 5, pp. 714–723, Sep. 2019, doi: 10.1177/0022343318823870.

[3] R. J. V. Cole, "Power-sharing, post-electoral contestations and the dismemberment of the right to democracy in Africa," *Int. J. Hum. Rights*, vol. 17, no. 2, pp. 256–274, Feb. 2013, doi: 10.1080/13642987.2013.752946.

[4] K. Hausken and M. Ncube, "Determinants of Election Outcomes: New Evidence from Africa," *African Dev. Rev.*, vol. 26, no. 4, pp. 610–630, Dec. 2014, doi: 10.1111/j.1467-8268.2014.12117.x.

[5] U. E. Daxecker, "The cost of exposing cheating," *J. Peace Res.*, vol. 49, no. 4, pp. 503–516, Jul. 2012, doi: 10.1177/0022343312445649.

[6] H. POLICY, "Elections in the Democratic Republic of the Congo," *Int. Inst. Strateg. Stud.*, no. 18, p. 4, Feb. 2019, [Online]. Available: https://horninstitute.org/wp-content/uploads/2019/08/No.-18.-Elections-in-the-DRC-1.pdf

[7] K. Berwouts and F. Reyntjens, "The Democratic Republic of Congo : The Great Electoral Robbery (and how and why Kabila got away with it)," *Africa Policy Br.*, vol. 25, no. 25, pp. 1–6, 2019, [Online]. Available: https://www.jstor.com/stable/resrep21375

[8] E. Polls, "Situation report ELECTION POLLS," no. JuNe, pp. 1–14, 2012.

[9] K. Vlassenroot, A. Nyenyezi, E. M. Mudinga, and G. Muzalia, "Producing democracy in armed violence settings: Elections and citizenship in Eastern DRC," *J. Civ. Soc.*, no. May, pp. 1–18, May 2022, doi: 10.1080/17448689.2022.2068626.

[10] P. P. I. Oslo Prio, "Replication Datasets - Journal of Peace Research - PRIO." Feb. 03, 2021. [Online]. Available: https://www.prio.org/jpr/datasets/

[11] M. Alam, M. O. Yusuf, and N. A. Sani, "Blockchain technology for electoral process in Africa: a short review," *Int. J. Inf. Technol.*, vol. 12, no. 3, pp. 861–867, Sep. 2020, doi: 10.1007/s41870-020-00440-w.

[12] J. Siegle and C. Cook, "Taking Stock of Africa's 2021 Elections," *Africa Center for Strategic Studies*, 2021. https://africacenter.org/spotlight/2021-elections/ (accessed Aug. 28, 2022).

[13] Congo-Research-Group, "The electronic voting process," *Elections ACT*, no. April 2018, pp. 1–5, 2018, [Online]. Available: http://www.elections.act.gov.au/elections_and_voting/electronic_voting_https://www.congoresearchgroup.org/wp-content/uploads/2018/04/Electronic-Voting-Controversy-1.pdf

[14] T. Ali-Diabacté, "Operational and procedural integrity of elections in the Democratic Republic of Congo," *J. African Elections*, pp. 52–65, Jun. 2020, doi: 10.20940/JAE/2020/v19i1a3.

[15] Zinah J. Mohammed Ameen, "Application Voting System of Web based in Iraq," *Iraqi*

J. Sci., vol. 58, no. 1A, pp. 192–200, 2017.

[16] E. Aljarrah, H. Elrehail, and B. Aababneh, "E-voting in Jordan: Assessing readiness and developing a system," *Comput. Human Behav.*, vol. 63, pp. 860–867, 2016, doi: 10.1016/j.chb.2016.05.076.

[17] R. Khatun, T. Bandopadhyay, and A. Roy, "Data Modeling for E-Voting System Using Smart Card based E-Governance System," *Int. J. Inf. Eng. Electron. Bus.*, vol. 9, no. 2, pp. 45–52, 2017, doi: 10.5815/ijieeb.2017.02.06.

[18] Y. Abuidris, R. Kumar, and W. Wenyong, "A Survey of Blockchain Based on E-voting Systems," in *Proceedings of the 2019 2nd International Conference on Blockchain Technology and Applications*, New York, NY, USA: ACM, Dec. 2019, pp. 99–104. doi: 10.1145/3376044.3376060.

[19] U. Jafar, M. J. A. Aziz, and Z. Shukur, "Blockchain for Electronic Voting System—Review and Open Research Challenges," *Sensors*, vol. 21, no. 17, p. 5874, Aug. 2021, doi: 10.3390/s21175874.

[20] A. Khelifi, Y. Grisi, D. Soufi, D. Mohanad, and P. V. S. Shastry, "M-Vote: A Reliable and Highly Secure Mobile Voting System," in *2013 Palestinian International Conference on Information and Communication Technology*, IEEE, Apr. 2013, pp. 90–98. doi: 10.1109/PICICT.2013.25.

[21] S. A. Adeshina and A. Ojo, "Maintaining Voting Integrity using Blockchain," in *2019 15th International Conference on Electronics, Computer and Computation (ICECCO)*, IEEE, Dec. 2019, pp. 1–5. doi: 10.1109/ICECCO48375.2019.9043225.

[22] A. Ben Ayed, "A Conceptual Secure Blockchain Based Electronic Voting System," *Int. J. Netw. Secur. Its Appl.*, vol. 9, no. 3, pp. 01–09, May 2017, doi: 10.5121/ijnsa.2017.9301.

[23] R. Hanifatunnisa and B. Rahardjo, "Blockchain based e-voting recording system design," in *2017 11th International Conference on Telecommunication Systems Services and Applications (TSSA)*, IEEE, Oct. 2017, pp. 1–6. doi: 10.1109/TSSA.2017.8272896.

[24] P. Baudier, G. Kondrateva, C. Ammi, and E. Seulliet, "Peace engineering: The contribution of blockchain systems to the e-voting process," *Technol. Forecast. Soc. Change*, vol. 162, no. October 2020, p. 120397, Jan. 2021, doi: 10.1016/j.techfore.2020.120397.

[25] N. Kshetri and J. Voas, "Blockchain-Enabled E-Voting," *IEEE Softw.*, vol. 35, no. 4, pp. 95–99, Jul. 2018, doi: 10.1109/MS.2018.2801546.

[26] B. Shahzad and J. Crowcroft, "Trustworthy Electronic Voting Using Adjusted Blockchain Technology," *IEEE Access*, vol. 7, pp. 24477–24488, 2019, doi: 10.1109/ACCESS.2019.2895670.

[27] H. V Patil, K. G. Rathi, M. V Tribhuwan, C. Science, and D. Y. P. A. C. S. College, "A Study on Decentralized E-Voting System Using Blockchain Technology," *Int. Res. J. Eng. Technol.*, vol. 5, no. 11, pp. 48–53, 2018, [Online]. Available: https://www.academia.edu/download/57934860/IRJET-V5I1109.pdf

[28] M. E. Mavungu, "Stay in power whatever it takes-fraud and repression in the 2011 elections in the Democratic Republic of Congo," *J. African Elections*, vol. 12, no. 3, pp. 25–50, 2013, [Online]. Available: https://hdl.handle.net/10520/EJC147382

[29] R. Cooley, S. Wolf, and M. Borowczak, “Blockchain-Based Election Infrastructures,” in *2018 IEEE International Smart Cities Conference (ISC2)*, IEEE, Sep. 2018, pp. 1–4. doi: 10.1109/ISC2.2018.8656988.

[30] F. P. Hjalmarsson, G. K. Hreioarsson, M. Hamdaqa, and G. Hjalmtysson, “Blockchain-Based E-Voting System,” in *2018 IEEE 11th International Conference on Cloud Computing (CLOUD)*, IEEE, Jul. 2018, pp. 983–986. doi: 10.1109/CLOUD.2018.00151.

[31] Ethereum, “Ethereum Goerli test network Explorer,” *Etherscan*, 2022. https://goerli.etherscan.io/ (accessed Nov. 04, 2022).

[32] Ethereum, “Ethereum Mainnet Explorer.” https://etherscan.io/ (accessed Aug. 27, 2022).

[33] M. N. M. Bhutta *et al.*, “A Survey on Blockchain Technology: Evolution, Architecture and Security,” *IEEE Access*, vol. 9, pp. 61048–61073, 2021, doi: 10.1109/ACCESS.2021.3072849.

[34] S. M. Idrees, M. Nowostawski, R. Jameel, and A. K. Mourya, “Security Aspects of Blockchain Technology Intended for Industrial Applications,” *Electronics*, vol. 10, no. 8, p. 951, Apr. 2021, doi: 10.3390/electronics10080951.

[35] R. Anandan and B. S. Deepak, “An Overview of Blockchain Technology: Fundamental Theories and Concepts,” in *The Convergence of Artificial Intelligence and Blockchain Technologies*, WORLD SCIENTIFIC, 2022, pp. 1–22. doi: 10.1142/9789811225079_0001.

[36] O. Ali, A. Jaradat, A. Kulakli, and A. Abuhalimeh, “A Comparative Study: Blockchain Technology Utilization Benefits, Challenges and Functionalities,” *IEEE Access*, vol. 9, pp. 12730–12749, 2021, doi: 10.1109/ACCESS.2021.3050241.

[37] V. Dhillon, D. Metcalf, and M. Hooper, *Blockchain Enabled Applications*, Apress. Berkeley, CA: Apress, 2017. doi: 10.1007/978-1-4842-3081-7.

[38] K. Söze, *BLOCKCHAIN: Novice to Expert*. 2017.

[39] S. Angraal, H. M. Krumholz, and W. L. Schulz, “Blockchain Technology,” *Circ. Cardiovasc. Qual. Outcomes*, vol. 10, no. 9, pp. 1–3, Sep. 2017, doi: 10.1161/CIRCOUTCOMES.117.003800.

[40] S. Squarepants, “Bitcoin: A Peer-to-Peer Electronic Cash System,” *SSRN Electron. J.*, pp. 1–9, 2008, doi: 10.2139/ssrn.3977007.

[41] C. V. Helliar, L. Crawford, L. Rocca, C. Teodori, and M. Veneziani, “Permissionless and permissioned blockchain diffusion,” *Int. J. Inf. Manage.*, vol. 54, no. October 2019, p. 102136, Oct. 2020, doi: 10.1016/j.ijinfomgt.2020.102136.

[42] M. J. W. Rennock, A. Cohn, and J. R. Butcher, “Blockchain Technology Regulatory and Investigations,” *J. Litig.*, no. March, pp. 34–44, 2018, [Online]. Available: https://www.steptoe.com/images/content/1/7/v3/171269/LIT-FebMar18-Feature_Blockchain.pdf

[43] S. Michael G., *Ethereum for dummies*. John Wiley & Sons, Inc., Hoboken, New Jersey, 2019.

[44] S. Anwar, S. Anayat, S. Butt, S. Butt, and M. Saad, “Generation Analysis of Blockchain Technology: Bitcoin and Ethereum,” *Int. J. Inf. Eng. Electron. Bus.*, vol. 12, no. 4, pp. 30–39, 2020, doi: 10.5815/ijieeb.2020.04.04.

[45] A. Donmez and A. Karaivanov, “Transaction fee economics in the Ethereum blockchain,” *Econ. Inq.*, vol. 60, no. 1, pp. 265–292, Jan. 2022, doi: 10.1111/ecin.13025.
[46] D. Kim, D. Ryu, and R. I. Webb, “Determination of Transaction Fees in the Bitcoin Network,” *SSRN Electron. J.*, pp. 1–21, 2022, doi: 10.2139/ssrn.4228897.
[47] M. Javaid, A. Haleem, R. Pratap Singh, S. Khan, and R. Suman, “Blockchain technology applications for Industry 4.0: A literature-based review,” *Blockchain Res. Appl.*, vol. 2, no. 4, p. 100027, Dec. 2021, doi: 10.1016/j.bcra.2021.100027.
[48] Dejan Vujičić and Dijana Jagodić, “2018 17th International Symposium INFOTEH JAHORINA (INFOTEH).,” no. March, pp. 21–23, 2018.
[49] Https://github.com/Pondorasti, “Ethereum official website,” *INTRO TO ETHEREUM*. https://ethereum.org/en/developers/docs/
[50] T. Tani, “Ethereum EVM illustrated,” 2018.
[51] E. Fondation, “Ethereum Virtual Machine,” 2023. https://ethereum.org/en/developers/docs/evm/ (accessed Mar. 10, 2023).
[52] Ethereum, “Mainnet Merge Announcement,” *Ethereum Fondation Blog*, 2022. https://blog.ethereum.org/2022/08/24/mainnet-merge-announcement (accessed Nov. 04, 2022).
[53] CoolWallet, “Ethereum 2.0: Beacon Chain PoS Upgrade Launces,” 2020, [Online]. Available: https://www.coolwallet.io/ethereum-2-beacon-chain-upgrade/
[54] P. R. Nair and D. R. Dorai, “Evaluation of Performance and Security of Proof of Work and Proof of Stake using Blockchain,” in *2021 Third International Conference on Intelligent Communication Technologies and Virtual Mobile Networks (ICICV)*, IEEE, Feb. 2021, pp. 279–283. doi: 10.1109/ICICV50876.2021.9388487.
[55] F. CROUCH, “What you can do for Ethereum 2.0,” *Public Health Nurs.*, vol. 43, no. 10, pp. 573–574, 1951.
[56] M. Gates, *BLOCKCHAIN: Ultimate guide to understanding Blockchain, Bitcoin, cryptocurrencies, Smart contracts and the future of money*. Wise Fox Publishing, 2017.
[57] N. Savchenko, “Decentralized Applications Architecture: Back End, Security and Design Patterns,” *freecodecamp*, 2019. https://www.freecodecamp.org/news/how-to-design-a-secure-backend-for-your-decentralized-application-9541b5d8bddb/
[58] A. Vacca, A. Di Sorbo, C. A. Visaggio, and G. Canfora, “A systematic literature review of blockchain and smart contract development: Techniques, tools, and open challenges,” *J. Syst. Softw.*, vol. 174, p. 110891, Apr. 2021, doi: 10.1016/j.jss.2020.110891.
[59] M. Hashemi Joo, Y. Nishikawa, and K. Dandapani, “Cryptocurrency, a successful application of blockchain technology,” *Manag. Financ.*, vol. 46, no. 6, pp. 715–733, Aug. 2019, doi: 10.1108/MF-09-2018-0451.
[60] S. Chang, “Ethereum Smart Contracts Vulnerable to Hacks: $4 Million in Ether at Risk,” 2019. https://www.investopedia.com/news/ethereum-smart-contracts-vulnerable-hacks-4-million-ether-risk/ (accessed Mar. 17, 2023).
[61] D. Resnick, “Compromise and Contestation: Understanding the Drivers and Implications of Coalition Behaviour in Africa,” *J. African Elections*, vol. 13, no. 1, pp.

43–65, Jun. 2014, doi: 10.20940/JAE/2014/v13i1a3.
[62] A. Cretarola, G. Figà-Talamanca, and C. Grunspan, “Blockchain and cryptocurrencies: economic and financial research,” *Decis. Econ. Financ.*, vol. 44, no. 2, pp. 781–787, Dec. 2021, doi: 10.1007/s10203-021-00366-3.
[63] L. Tredinnick, “Cryptocurrencies and the blockchain,” *Bus. Inf. Rev.*, vol. 36, no. 1, pp. 39–44, Mar. 2019, doi: 10.1177/0266382119836314.
[64] P. Sonpatki, V. A. M., V. A. M, and P. Sonpatki, *ReactJS by Example-Building Modern Web Applications with React*. Packt Publishing Ltd., 2016. [Online]. Available: http://www.it-ebooks.info/
[65] Facebook Inc., “React official website,” *A JavaScript library for building user interfaces*, Mar. 03, 2021. https://reactjs.org/
[66] E. Fondation, “A browser-based IDE with an integrated compiler for Solidiry,” *https://ethereum.org/*, [Online]. Available: https://remix.ethereum.org/
[67] D. Mohanty, *Ethereum for Architects and Developers: With Case Studies and Code Samples in Solidity*, Apress. Noida, Uttrar Pradesh, India: Library of Congress Control Number, 2018. doi: https://dx.doi.org/10.1007/9781484240755.
[68] https://github.com/ChainSafe/web3.js/blob/v1.3.4/docs/getting-started.rst, “Web3.js official documentation,” *Docs: Getting Started*, Mar. 03, 2021. https://web3js.readthedocs.io/en/v1.3.4/getting-started.html
[69] R. Moore, “Ethers.js Official Documentation,” *https://ethers.org/*, 2022. https://docs.ethers.org/v5/ (accessed Dec. 17, 2022).
[70] N. Foundation, “Hardhat: Ethereum development environment for professionals,” 2022. https://hardhat.org (accessed Dec. 12, 2022).
[71] trufflesuite.com, “Ganache (Truffle suite Official Website),” 2022. https://trufflesuite.com/ganache/ (accessed Dec. 17, 2022).
[72] trufflesuite.com, “Truffle - The most comprehensive suite of tools for smart contract development,” 2022. https://trufflesuite.com/ (accessed Dec. 17, 2022).
[73] Microsoft, “Visual Studio Code - Code editing. Redefined,” 2022. https://code.visualstudio.com/ (accessed Jun. 20, 2022).
[74] M. Carr and J. Verner, “Prototyping and Software Development Approaches,” *Prototyp. Softw. Dev. Approaches*, no. 3, pp. 1–16, 2004, [Online]. Available: https://www.google.com.my/url?sa=t&rct=j&q=&esrc=s&source=web&cd=1&cad=rja&uact=8&ved=0CCkQFjAA&url=http://www.cb.cityu.edu.hk/is/getFileWorkingPaper.cfm?id=55&ei=73eDU6aCBo7PlAXNooGQBg&usg=AFQjCNFCEFbDyv9tNk_YuH0VpPfavJPs2A&sig2=wimyHPVpHpp
[75] R. G, “4 Software Prototyping Techniques You Need to Know,” *www.perfomatix.com/*, 2022. https://www.perfomatix.com/4-software-prototyping-techniques-you-need-to-know/ (accessed Mar. 15, 2023).
[76] Y. Glamazdina, “Data Collection: Methods, Definition, Types, and Tools,” 2022. https://brocoders.com/blog/data-collection-methods-definition-types-and-tools/ (accessed Mar. 14, 2023).
[77] S. Z. Iqbal and M. Idrees, “Z-SDLC Model: A New Model For Soware Development Life Cycle (SDLC),” 2017.
[78] A. S. Services, “Changing the SDLC to Produce Secure Applications.” https://affinity-

it-security.com/changing-the-sdlc-to-produce-secure-applications/ (accessed May 07, 2023).
[79] N. B. Ruparelia, “Software development lifecycle models,” *ACM SIGSOFT Softw. Eng. Notes*, vol. 35, no. 3, pp. 8–13, May 2010, doi: 10.1145/1764810.1764814.
[80] Polygon, “Polygon testnet Blockchain explorer (Mumbai testnet).” https://mumbai.polygonscan.com/ (accessed Dec. 10, 2022).
[81] L. Marchesi, M. Marchesi, G. Destefanis, G. Barabino, and D. Tigano, “Design Patterns for Gas Optimization in Ethereum,” *IWBOSE 2020 - Proc. 2020 IEEE 3rd Int. Work. Blockchain Oriented Softw. Eng.*, no. February 2021, pp. 9–15, 2020, doi: 10.1109/IWBOSE50093.2020.9050163.
[82] Corwin Smith, “GAS AND FEES,” *https://github.com/corwintines*. https://ethereum.org/en/developers/docs/gas/ (accessed Jan. 03, 2023).
[83] J. ;Sandeep N. A. A. Kanani, “Polygon Whitepaper.”

Appendices

A. Project timetable and allocated budget

The budget includes the Smart Contract deployment fees (On Ethereum Mainnet and on Polygon Testnet so called Mumbai Testnet [80]) to make our prototype functional, and the cost involved when the application will be used. In the Ethereum concept, the word “gas” [81], [82] refers to the amount of computation needed to produce a certain work on Ethereum Blockchain through the Ethereum Virtual Machine. This computation implies paying a certain amount of fees (Most of the time in Ether, but other tokens EVM compatible can also be used, like MATIC for Polygon Blockchain [83]).

A.1 Project timetable

Our study project's timeline is provided via a Gant diagram, where several stages of the development process are displayed to help the reader better understand the work done.

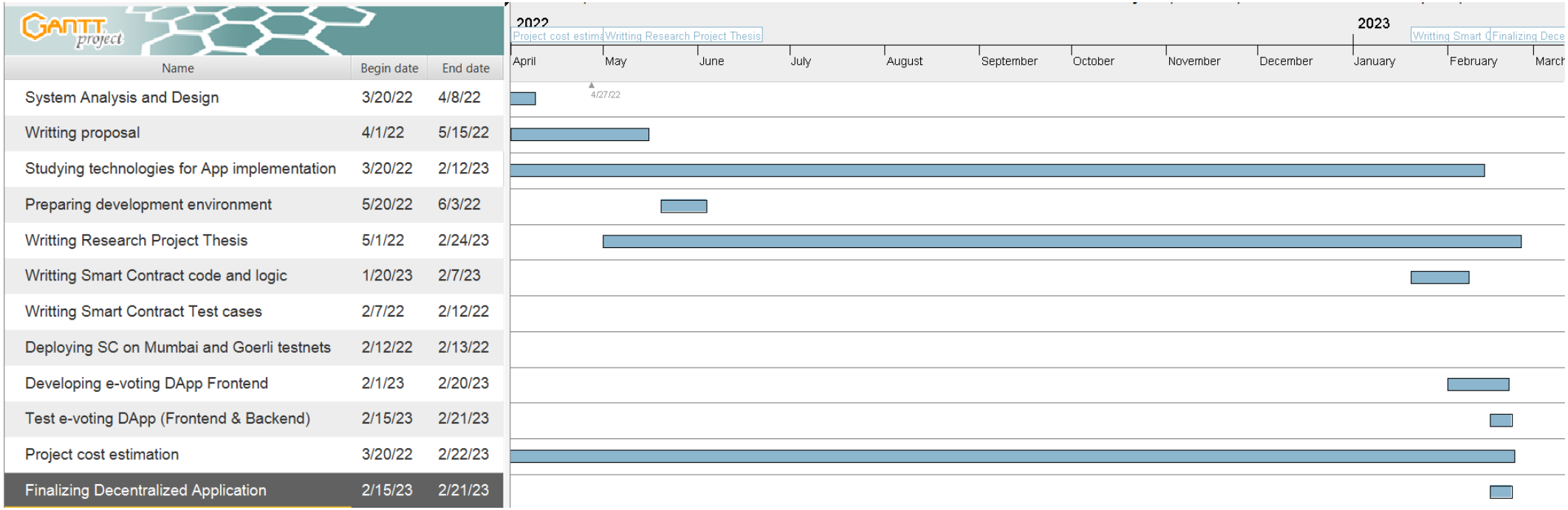


Figure A.1: Timetable on Gantt Diagram

A.2 Project allocated budget

The Allocated budget gives us an idea of the cost of the project regarding all technologies and tools to be used with the complexity of the system. The cost demonstrates the benefits of selecting Polygon Blockchain (Also known as Layer 2 Solution) over the Ethereum Main Blockchain (Called Layer 1) for the implementation in production.

Num	Task Designation	Qty (Days)	Unit Cost (USD)	Total Cost (USD)	Observation
01	System Analysis and Design	20	10	200	Using UML
02	Writting proposal	45	-	-	With Supervisors help
03	Studying technologies for App implementation	-	-	150	
04	Preparing development environment	15	-	-	
05	API Keys for development (Infura & Coinmarketkap)	-	-	-	
06	Internet Connectivity	330	0.5	165	Mobile Internet
07	Writting Research Project Thesis	300	-	-	
08	Writting Smart Contract code and logic	19	30	570	
09	Writting Smart Contract Test cases	06	-	-	
10	Deploying SC on Goerli Testnet	02	-	503.39	0.3282 Ether * $1,539.89
11	Developing e-voting DApp Frontend	20	10	200	
12	Tests with some users (around 50 or less)	-	15	750	$10 per external users
13	Test e-voting DApp (Frontend & Backend)	7	-	-	
Total cost				**2,538.39**	

Table A.1: Project allocated cost without using Polygon Optimization

The next table (On the next page) will use Polygon optimization to reduce the cost involved when transactions are made.

Num	Task Designation	Qty (Days)	Unit Cost (USD)	Total Cost (USD)	Observation
01	System Analysis and Design	20	10	200	Using UML
02	Writting proposal	45	-	-	With Supervisors help
03	Studying technologies for App implementation	-	-	150	
04	Preparing development environment	15	-	-	
05	API Keys for development (Infura & Coinmarketkap)	-	-	-	
06	Internet Connectivity	330	0.5	165	Mobile Internet
07	Writting Research Project Thesis	300	-	-	
08	Writting Smart Contract code and logic	19	30	570	
09	Writting Smart Contract Test cases	06	-	-	
10	Deploying SC on Polygon Testnet	02	-	0.5758	0.4499 MATIC * $1.28
11	Developing e-voting DApp Frontend	20	10	200	
12	Tests with some users (around 50 or less)	-	0.04	2	$0.04 per external users
13	Test e-voting DApp (Frontend & Backend)	7	-	-	
Total cost				**1,287.58**	

Table A.2: Project allocated cost with Polygon Optimization

As we can see, using Polygon Blockchain reduced the expense by an average of ***50 percent***, from **$2,538.39** to **$1,287.58**.

B. Smart Contract writing

To try to adhere to Solidity's 24Kb file size restriction, our Smart Contract logic was divided into three files, which are as follows: ***Voting.sol***, ***WitnessUser.sol***, and ***VotingOfficeUser.sol*** are the files that contain the core logic.

C. Smart contract deployment

The deployment of our smart contract can be done on Ethereum Mainnet or on others test networks. Mumbai Testnet is our pick for this project because the costs are so low, and it is just for test.

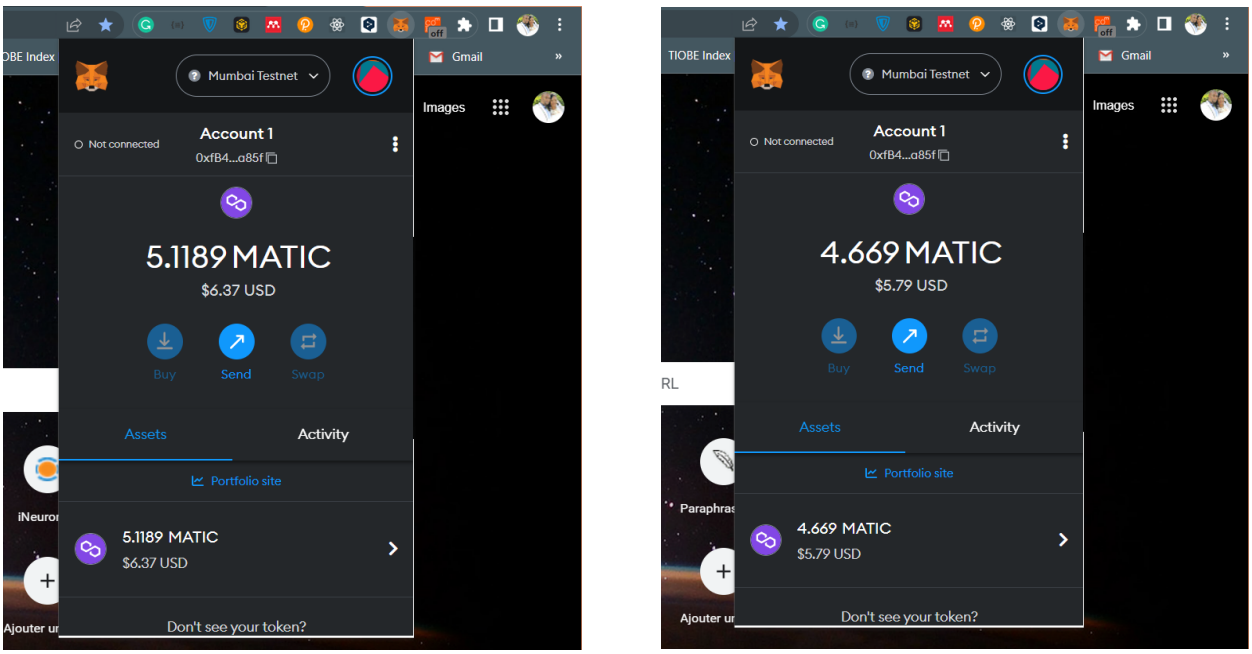
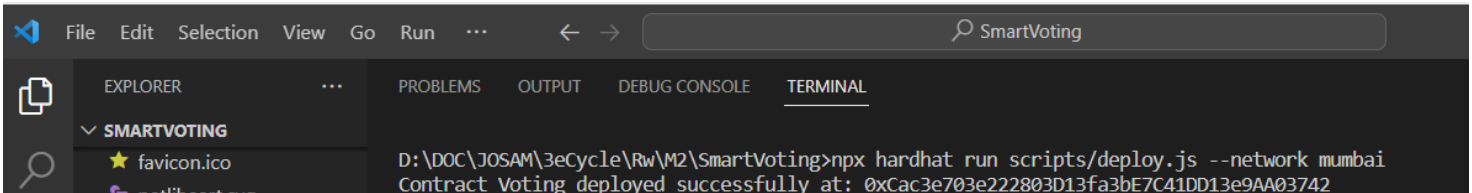


Figure C.1: MetaMask Balance before (On the left side) and after (on the right side) deployment of Smart Contracts on Mumbai Testnet

These deployment addresses are used to track (In figures 7.4, 7.5, and 7.6) our Smart Contracts on Mumbai Blockchain Explorer for respective Contract:

- **Contract Voting**: 0xe215350F4063674d729B4645ad828Be6De53Bd61.
- **Contract VotingOfficeUser**: 0x50256A88d1bf6829b8d8Ca245b48B3a408BC77AA.
- **Contract WitnessUser**: 0xd6124c88160407cBf0B8f19f2136F36aC8b0B5F9.



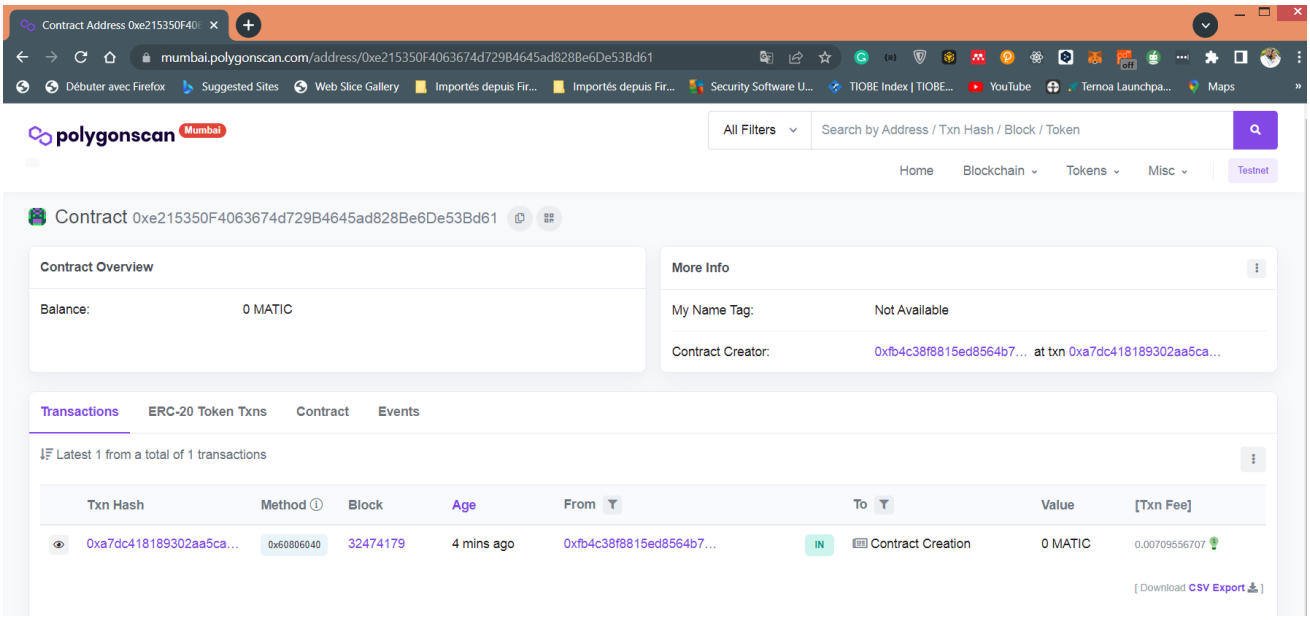


Figure C.3: Voting Smart Contract on Mumbai Blockchain Explorer

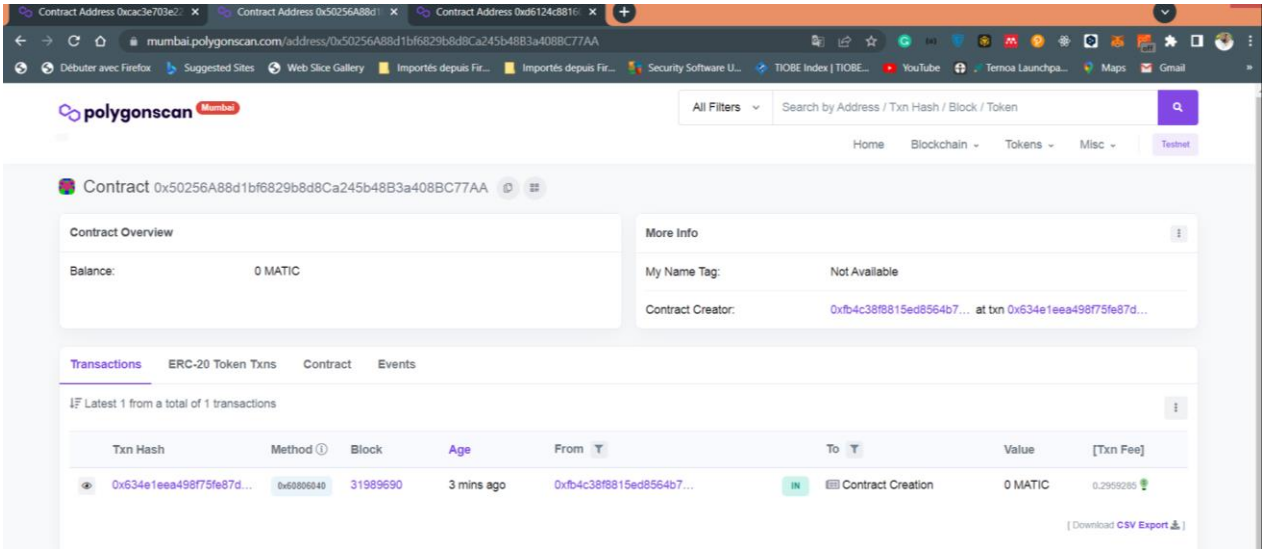


Figure C.4: VotingOfficeUser Smart Contract on Mumbai Blockchain Explorer

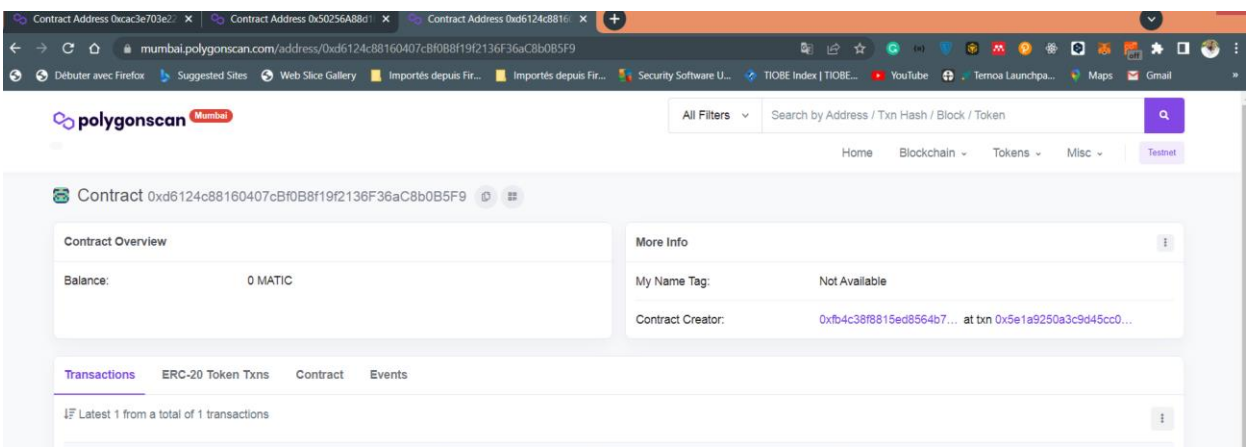


Figure C.5: WitnessUser Smart Contract on Mumbai Blockchain Explorer

To be able to see as well how things work on Goerli Testnet (To see the cost on Ethereum Mainnet), we have deployed our Smart Contracts on this test and observed the real gas fees.

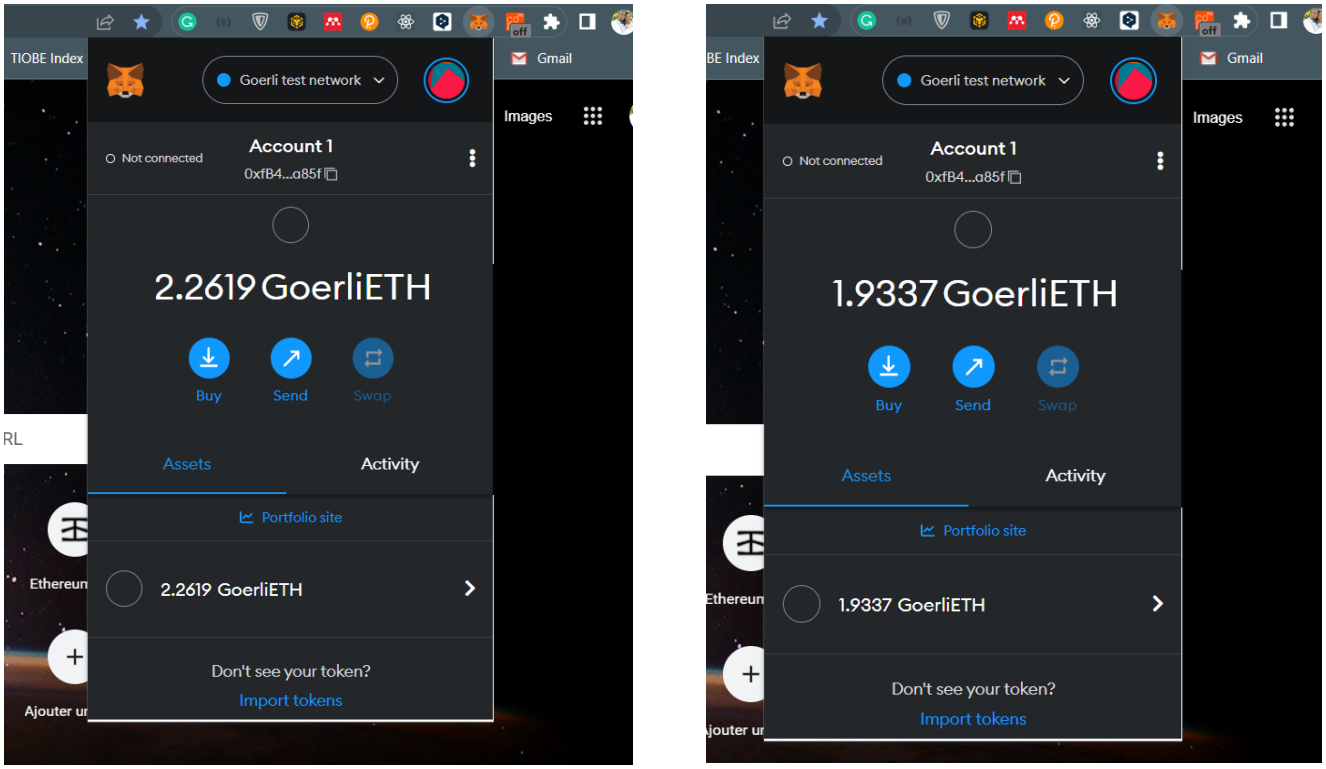


Figure C.6: MetaMask Balance before (On the left side) and after (on the right side) deployment of Smart Contracts on Goerli Testnet

Deployment addresses on Goerli Testnet:

- **Contract Voting**: 0xe4fc9073F42c731d3255BeA228946316d87f7C77.
- **Contract VotingOfficeUser**: 0xEa270453F89A4B64c43ca8aD0084B6F927e011fb.
- **Contract WitnessUser**: 0xA783F908559930CeE6c78C0520c056cAB7B3cac4.

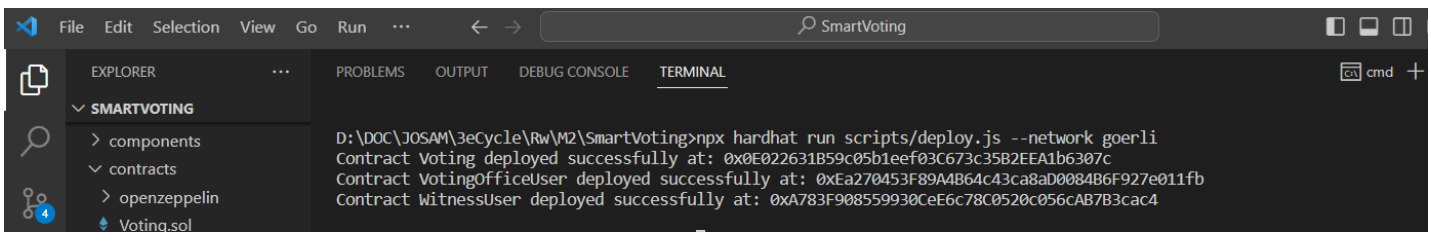


Figure C.7: Smart Contracts deployment on Goerli Testnet

These deployment transactions could be tracked to Goerli Blockchain explorer using contract addresses.

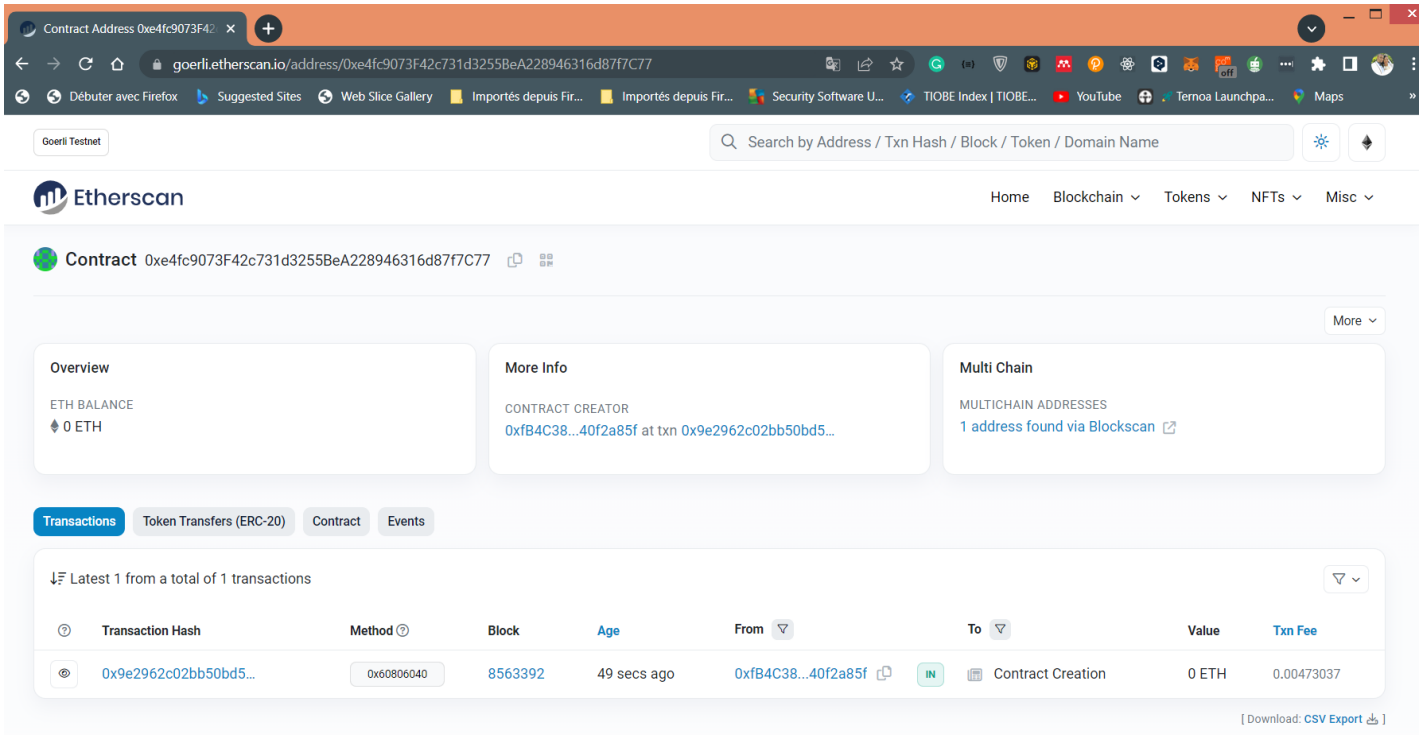


Figure C.8: Voting Smart Contract on Goerli Blockchain Explorer

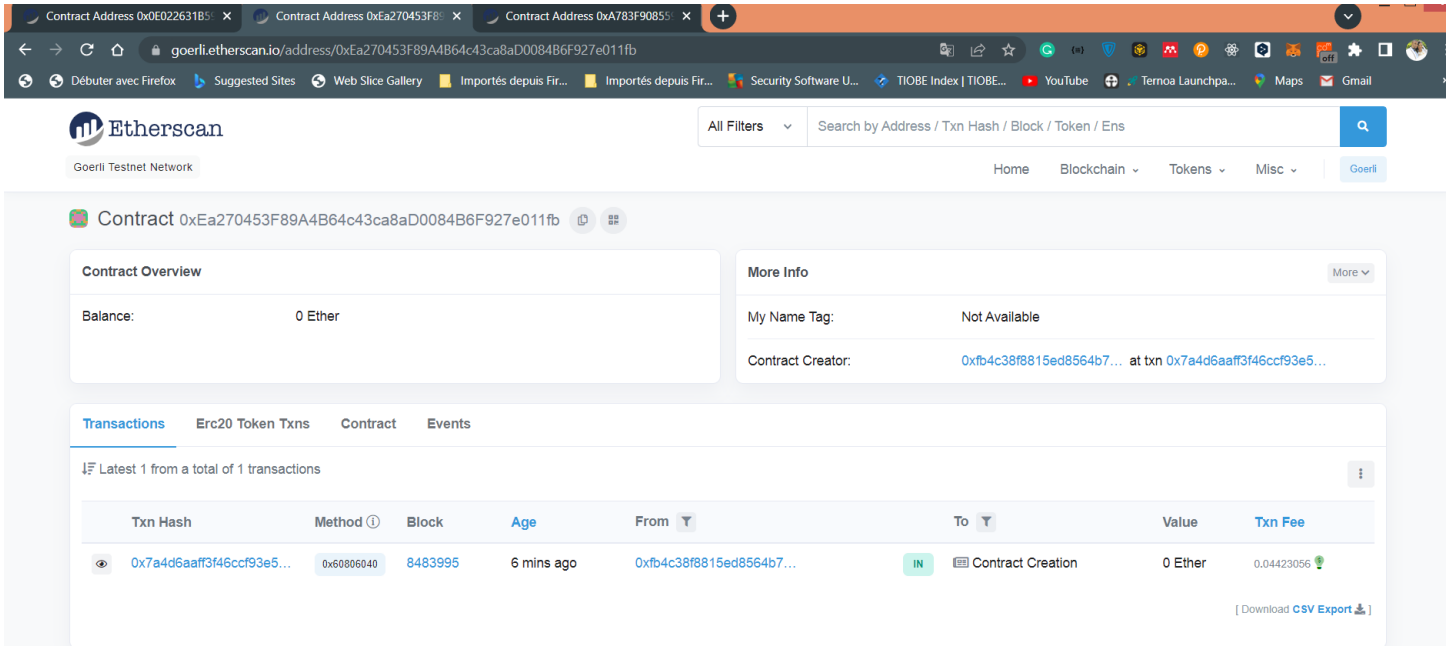


Figure C.9: VotingOfficeUser Smart Contract on Goerli Blockchain Explorer

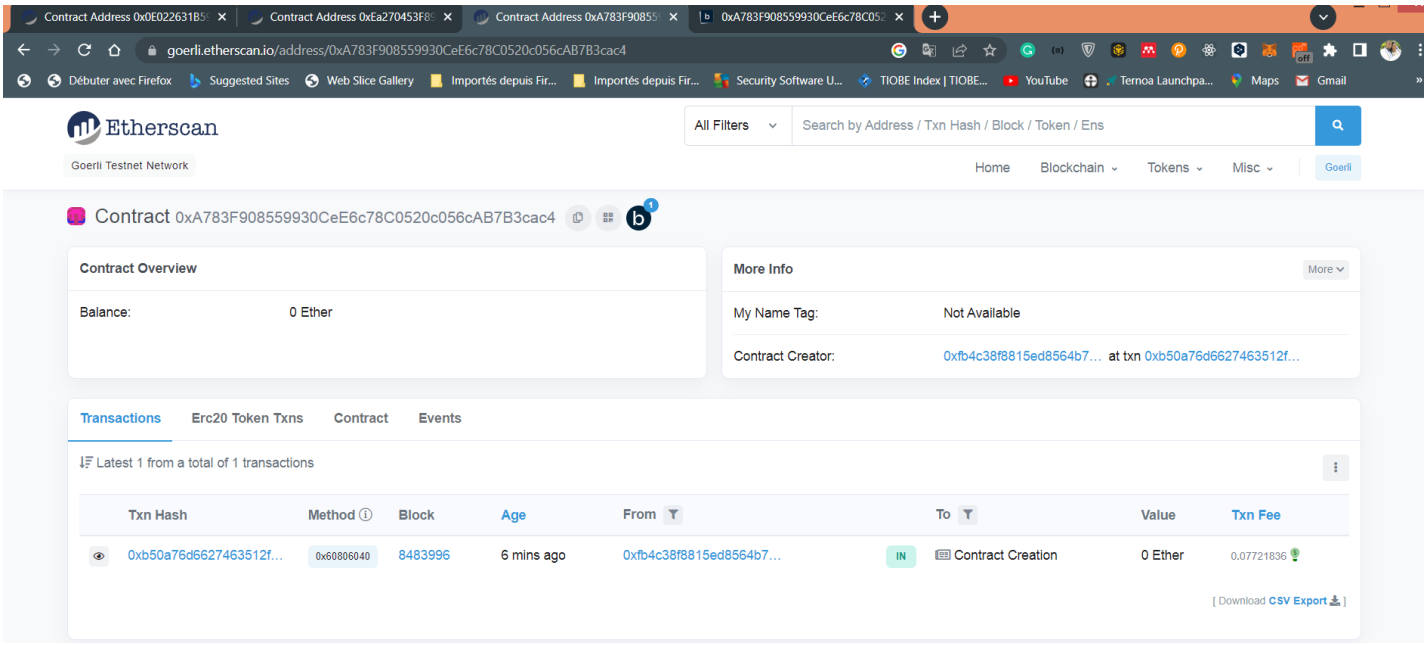


Figure C.10: WitnessUser Smart Contract on Mumbai Blockchain Explorer

D. Smart contract gas optimization with Polygon Blockchain

This Dap's budget includes the Smart Contracts deployment fees, and the Hardhat library makes easy this fees calculation (hardhat-gas-reporter). In this bellow picture, the deployment (If it is made on Ethereum Mainnet) will be *$2.97 + $12.19 + $9.90 + $6.77 + $4.42 + $6.35 + $1.84 + $5.08 + $6.13 + $2.08 + $115.67 + $22.33 + $35.69 =* ***$346.84***.

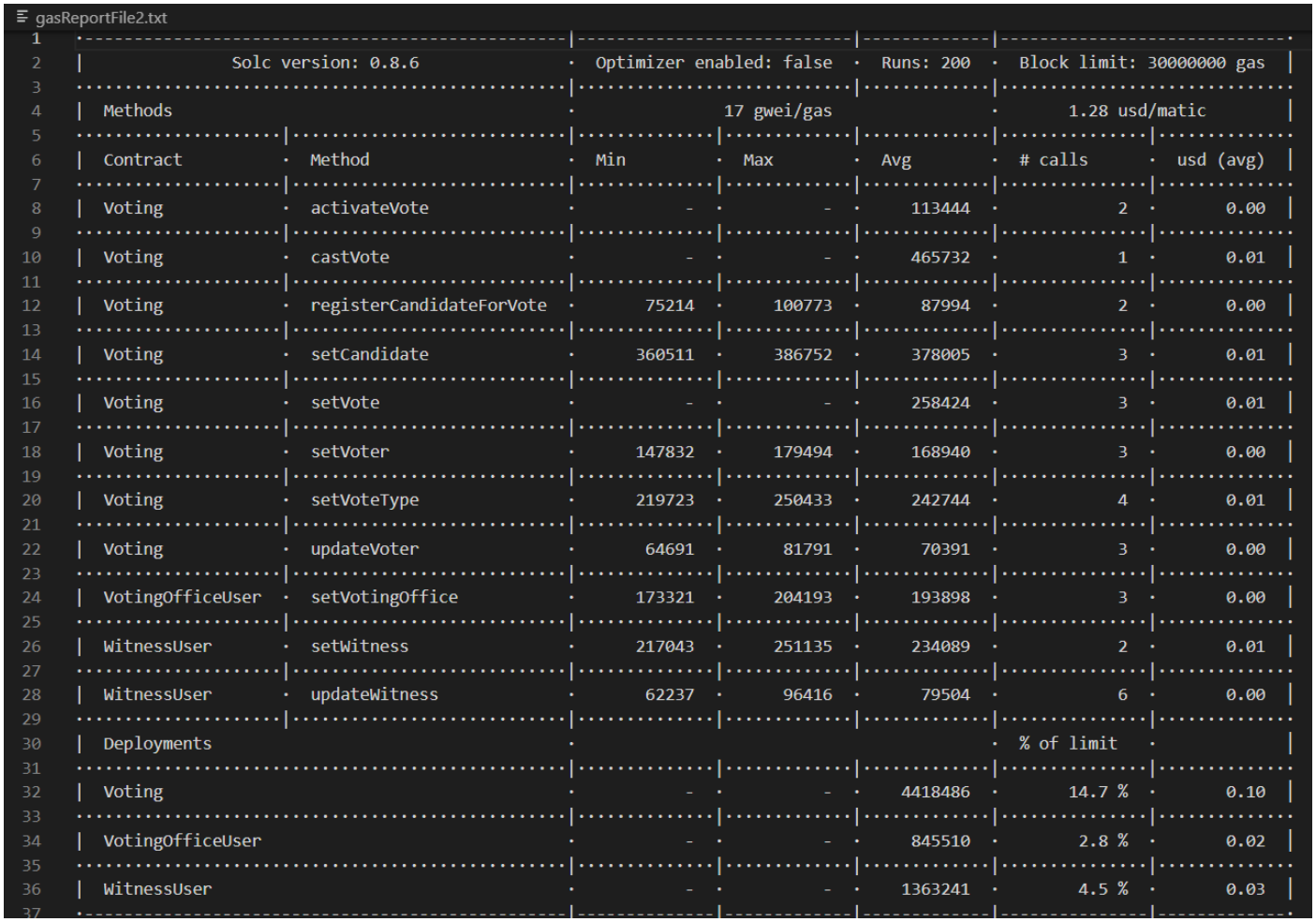
gasReportFile1.txt

Solc version: 0.8.6		Optimizer enabled: false	Runs: 200	Block limit: 30000000 gas
Methods		17 gwei/gas		1539.89 usd/eth
Contract	Method	Min	Max	Avg	# calls	usd (avg)
Voting	activateVote	-	-	113444	2	2.97
Voting	castVote	-	-	465732	1	12.19
Voting	registerCandidateForVote	75214	100773	87994	2	2.30
Voting	setCandidate	360511	386752	378005	3	9.90
Voting	setVote	-	-	258424	3	6.77
Voting	setVoter	147832	179494	168940	3	4.42
Voting	setVoteType	219723	250433	242744	4	6.35
Voting	updateVoter	64691	81791	70391	3	1.84
VotingOfficeUser	setVotingOffice	173321	204193	193898	3	5.08
WitnessUser	setWitness	217043	251135	234089	2	6.13
WitnessUser	updateWitness	62237	96416	79504	6	2.08
Deployments				% of limit	
Voting		-	-	4418486	14.7 %	115.67
VotingOfficeUser		-	-	845510	2.8 %	22.13
WitnessUser		-	-	1363241	4.5 %	35.69

Figure D.1: Gas report evaluation price in USD on Ethereum.

But, with the Mumbai Blockchain scalability, the deployment fees price is *$0 + $0.01 + $0 + $0.01 +$0.01 + $0+ $0.01 + $0+ $0.01 + $0 + $0 +$0.01 +$0.02 + 0.03=* ***$0.2***. As we can see, this fee has been significantly reduced, making DApp cheaper to use.

gasReportFile2.txt



Solc version: 0.8.6	Optimizer enabled: false	Runs: 200	Block limit: 30000000 gas
Methods	17 gwei/gas		1.28 usd/matic

Contract	Method	Min	Max	Avg	# calls	usd (avg)
Voting	activateVote	-	-	113444	2	0.00
Voting	castVote	-	-	465732	1	0.01
Voting	registerCandidateForVote	75214	100773	87994	2	0.00
Voting	setCandidate	360511	386752	378005	3	0.01
Voting	setVote	-	-	258424	3	0.01
Voting	setVoter	147832	179494	168940	3	0.00
Voting	setVoteType	219723	250433	242744	4	0.01
Voting	updateVoter	64691	81791	70391	3	0.00
VotingOfficeUser	setVotingOffice	173321	204193	193898	3	0.00
WitnessUser	setWitness	217043	251135	234089	2	0.01
WitnessUser	updateWitness	62237	96416	79504	6	0.00

Deployments			% of limit	
Voting	-	-	4418486	14.7 %	0.10
VotingOfficeUser	-	-	845510	2.8 %	0.02
WitnessUser	-	-	1363241	4.5 %	0.03

Figure D.2: Gas report evaluation price ins USD on MATIC Blockchain