Design Document (HLD/LLD)

BLOCKCHAIN-BASED E-VOTING SYSTEM

Revision Number: 3.0

Last Date of Revision: 11/11/2021

ANIRUDH BEHERA (B2020009)

NIPUN ALLURWAR (B2020031)

NITIN CHAUHAN (B2020033)

SANDIP GANGULY (B2020045)

ANUPHARI SINGH (B2020055)

Document Version Control

|  |  |  |  |
| --- | --- | --- | --- |
| **Date Issued** | **Version** | **Description** | **Author** |
| 31/10/2021 | 1 | Initial Draft |  |
| 06/11/2021 | 2 | 1st round of Changes |  |
| 11/11/2021 | 3 | 2nd round of Changes |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |

Table of Contents

[Abstract 4](#_Toc87656596)

[1. Introduction 5](#_Toc87656597)

[1.1 Why this Design Document? 5](#_Toc87656598)

[1.2 Scope 5](#_Toc87656599)

[2. General Description 6](#_Toc87656600)

[2.1 Product Perspective 6](#_Toc87656601)

[2.2 Problem Statement 7](#_Toc87656602)

[2.3 Proposed Solution 7](#_Toc87656603)

[2.4 Future Improvements 7](#_Toc87656604)

[2.5 Technical Requirements 8](#_Toc87656605)

[2.6 Data Requirements 8](#_Toc87656606)

[2.7 Programming Used 8](#_Toc87656607)

[2.8 Constraints 8](#_Toc87656608)

[3. Design Details 10](#_Toc87656609)

[3.1 Process Flow 10](#_Toc87656610)

[3.2 Error Handling 11](#_Toc87656611)

[3.3 Performance 11](#_Toc87656612)

[3.4 Reusability 12](#_Toc87656613)

[3.5 Application Compatibility 12](#_Toc87656614)

[3.6 Deployment 12](#_Toc87656615)

[4. Conclusion 13](#_Toc87656616)

# Abstract

Electronic voting or e-voting has been used in varying forms with fundamental benefits over paper-based systems such as increased efficiency and reduced errors. However, there remain challenges to achieving widespread adoption of such systems especially concerning improving their resilience against potential faults. Blockchain is a disruptive technology of the current era and promises to improve the overall resilience of e-voting systems. This paper presents an effort to leverage the benefits of blockchain such as cryptographic foundations and transparency to achieve an effective scheme for e-voting. The proposed scheme conforms to the fundamental requirements for e-voting schemes and achieves end-to-end verifiability. The paper presents details of the proposed e-voting scheme along with its implementation using the Multichain platform. The paper presents an in-depth evaluation of the scheme which successfully demonstrates its effectiveness to achieve an end-to-end verifiable e-voting scheme.

# 1. Introduction

## Why this Design Document?

This Design Document contains both the components

* High-Level Design (HLD)
* Low-Level Design (LLD)

The goal of the High-Level Design Document (HLD) is to supplement the current project description with the essential details and to serve as a reference manual for how modules interact at a high level. We hope to provide a bird' eye view of the architecture and design of the solution we're giving for e-voting system traceability difficulties in this document.

The HLD will:

* Describe the project's constraints and assumptions
* Present all design aspects and characterize them in depth
* Present the process flow
* Describe the asset tracker's performance and reusability

The purpose of the Low-Level Design Document (LLD) is to provide the internal logical design of the actual program code for the E-voting System. LLD describes class diagrams, including methods and relationships between classes, as well as program specifications. It explains the modules in such a way that the programmer can code the program directly from the document.

## Scope

This article aims to outline in as much detail as possible the High-Level features (HLD) of an E-Voting System. The paper combines information from previous discussions, blockchain lectures, and IT skills to create a strategy for a blockchain-enabled e-voting procedure.

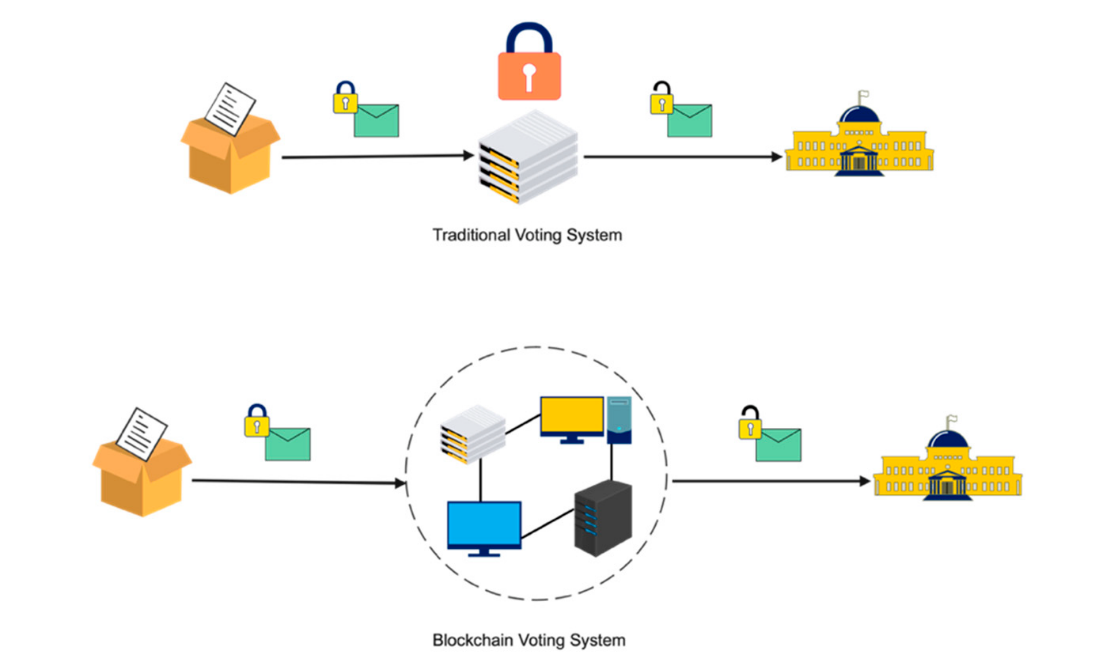
The Low-level design (LLD) is a component-level design approach that is refined in a step-by-step manner. This method can be applied to the creation of data structures, software architecture, source code, and, eventually, performance algorithms. Overall, data organization can be determined at the requirement analysis phase and then refined throughout the data design phase.

# 2. General Description

## 2.1 Product Perspective

Blockchain technology would fix shortcomings in today’s method in elections and would make the polling mechanism clear and accessible, stop illegal voting, strengthen the data protection, and check the outcome of the polling. The implementation of the electronic voting method in the blockchain is very significant. However, electronic voting carries significant risks such as if an electronic voting system is compromised, all cast votes can probably be manipulated and misused. Electronic voting has thus not yet been adopted on a national scale, considering all its possible advantages. Today, there is a viable solution to overcome the risks and electronic voting, which is blockchain technology. In the below Figure, one can see the main difference between both of the systems. In traditional voting systems, we have a central authority to cast a vote. If someone wants to modify or change the record, they can do it quickly; no one knows how to verify that record. One does not have the central authority; the data are stored in multiple nodes. It is not possible to hack all nodes and change the data. Thus, in this way, one cannot destroy the votes and efficiently verify the votes by tallying with other nodes.

If the technology is used correctly, the blockchain is a digital, decentralized, encrypted, transparent ledger that can withstand manipulation and fraud. Because of the distributed structure of the blockchain, a Bitcoin electronic voting system reduces the risks involved with electronic voting and allows for a tamper-proof for voting system. A blockchain-based electronic voting system requires a wholly distributed voting infrastructure. Electronic voting based on blockchain will only work where the online voting system is fully controlled by no single body, not even the government. To sum up, elections can only be free and fair when there is a broad belief in the legitimacy of the power held by those in positions of authority. The literature review for this field of study and other related experiments may be seen as a good path for making voting more efficient in terms of administration and participation. However, the idea of using blockchain offered a new model for electronic voting.



## 2.2 Problem Statement

To ensure a smooth, fair, and democratic election process through the implementation of **blockchain technology in electronic voting machines** by completely eradicating illegal voting, strengthening the data protection, and more accurate verification of the polling outcomes. Whether talking about traditional paper-based voting, voting via digital voting machines, or an online voting system, several conditions need to be satisfied:

* **Eligibility:** Only legitimate voters should be able to take part in voting
* **Unreusability:** Each voter can vote only once
* **Privacy:** No one except the voter can obtain information about the voter’s choice
* **Fairness:** No one can obtain intermediate voting results
* **Soundness:** Invalid ballots should be detected and not taken into account during tallying
* **Completeness:** All valid ballots should be tallied correctly

## 2.3 Proposed Solution

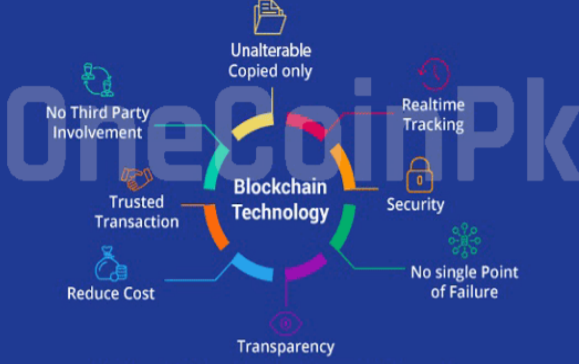
The blockchain is a digital, decentralized, encrypted, transparent ledger that can withstand manipulation and fraud. Because of the distributed structure of the blockchain, a Bitcoin electronic voting system reduces the risks involved with electronic voting and allows for a tamper-proof for voting system. A blockchain-based electronic voting system requires a wholly distributed voting infrastructure. Electronic voting based on blockchain will only work where the online voting system is fully controlled by no single body, not even the government. To sum up, elections can only be free and fair when there is a broad belief in the legitimacy of the power held by those in positions of authority. The literature review for this field of study and other related experiments may be seen as a good path for making voting more efficient in terms of administration and participation. However, the idea of using blockchain offered a new model for electronic voting.

## 2.4 Future Improvements

Blockchain technology achieves significant success in the detection of malleable change in a transaction however successful demonstration of such events has been achieving which motivates us to investigate it further. To this end, we believe an effective model to establish trustworthy provenance for e-voting systems will be crucial to achieving an end-to-end verifiable e-voting scheme. Over time, **research** has highlighted specific problems, such as the need for further work on blockchain-based electronic voting and that blockchain-based electronic voting schemes have significant technical challenges such as-

* Scalability and Processing Overheads
* User Identity
* Transactional Privacy
* Energy Efficiency

## 2.5 Technical Requirements



## 2.6 Data Requirements

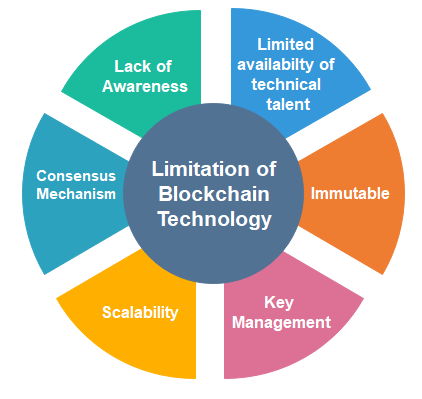
The E-voting System uses data about the vote recorded by the user along with its location details. The timestamp data showing the recording of the vote is used. Other data requirements are Voter ID number, User name, location, flag, and digital signature which is the approval required to send the product.

## 2.7 Programming Used

Python Programming is used with the Hashlib module which is used to implement a common interface to message-digest algorithm and different secure hash. SHA 256 cryptographic algorithm is used in this to add blocks we will also need to flesh out methods for the new\_block(), add\_new\_transaction(), and hash() in blockchain for this Blockchain-based E-Voting System.

## 2.8 Constraints

Blockchain technology has enormous potential in creating trustless, decentralized applications. But it is not perfect. Certain barriers make blockchain technology not the right choice and unusable for mainstream applications. We can see the limitations of blockchain technology in the following image.



* **Lack of Awareness**

There is a lot of discussion about blockchain, but people do not know the true value of blockchain and how they could implement it in different situations.

* **Limited availability of technical talent**

In blockchain technology, there are not so many developers available who have specialized expertise in blockchain technology. Hence, the lack of developers is a hindrance to developing anything on the blockchain.

* **Immutable**

In immutable, we cannot make any modifications to any of the records. immutability also has a drawback. In this case, when you want to make any revisions or want to go back and make any reversals. For example, you have processed a payment and need to go back and make an amendment to change that payment.

* **Key Management**

There are different keys, such as public keys and private keys. When you are dealing with a private key, then you are also running the risk that somebody may lose access to your private key. It happens a lot in the early days when bitcoin wasn't worth that much. People would just collect a lot of bitcoin, and then suddenly forget what the key was, and those may be worth millions of dollars today.

* **Scalability**

Blockchain has consensus mechanisms that require every participating node to verify the transaction. It limits the number of transactions a blockchain network can process.

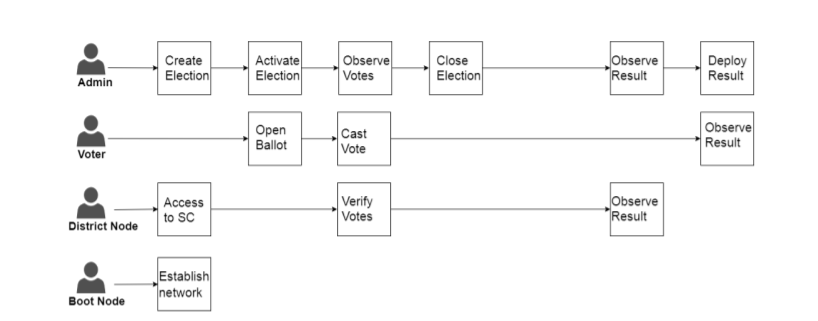
* **Consensus Mechanism**

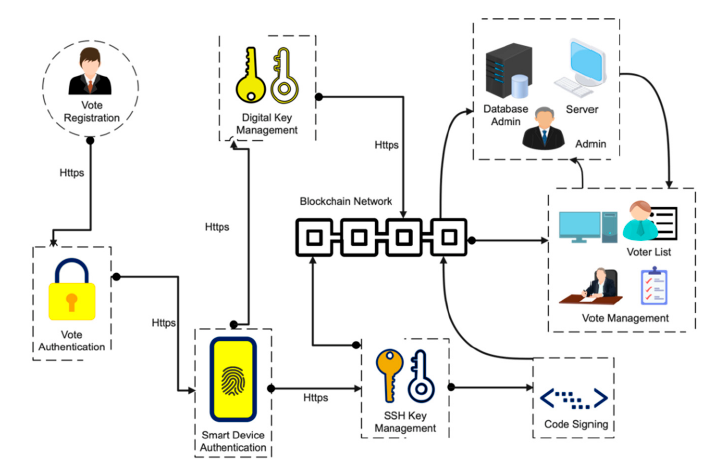
In the blockchain, Depending on the network size and the number of blocks or nodes involved in a blockchain, the back-and-forth communications involved to attain a consensus can consume a considerable amount of time and resources.

# 3. Design Details

## 3.1 Process Flow

We now describe a typical interaction of a user with the proposed scheme based on our current implementation of the system. Typically, a voter logs into the system by providing his/her thumb impression. If the match is found, the voter is then presented with a list of available candidates with the option to cast vote against them. On the contrary, if the match is unsuccessful, any further access would be denied. This function is achieved using an appropriate implementation of the authentication mechanism (fingerprinting in this case) and predefined role-based access control management. Furthermore, it is also envisioned that a voter is assigned to their specific constituency and this information is used to develop the list of candidates that a voter can vote for. The assignment of the voter to a constituency is rendered an offline process and therefore out of the scope of this research. After a successful vote-cast, it is mined by multiple miners for validation following which valid and verified votes are added into the public ledger. The security considerations of the votes are based on blockchain technology using cryptographic hashes to secure end-to-end verification. To this end, a successful vote cast is considered a transaction within the blockchain of the voting application. Therefore, a vote cast is added as a new block (after successful mining) in the blockchain as well as being recorded in data tables at the backend of the database. The system ensures only one-person, one-vote (democracy) property of voting systems. This is achieved by using the voter’s unique thumbprint, which is matched at the beginning of every voting attempt to prevent double voting. A transaction is generated as soon as the vote is mined by the miners which are unique for each vote. If the vote is found malicious it is rejected by miners. After the validation process, a notification is immediately sent to the voter through a message or an email providing the above-defined transaction id by which the user can track his/her vote into the ledger. Although this functions as a notification to the voter, however, it does not enable any user to extract the information about how a specific voter voted thereby achieving the privacy of a voter. It is important here to note that the cryptographic hash for a voter is the unique hash of the voter by which the voter is known in the blockchain. This property facilitates achieving verifiability of the overall voting process. Furthermore, this id is hidden and no one can view it even a system operator cannot view this hash, therefore, achieving the privacy of individual voters.





## 3.2 Error Handling

The different types of errors in blockchain development.

1. Network-based Errors

One of the easiest methods to avoid this is by ensuring the proper installation and starting of the network. You will have to take all the steps according to the network we have.

1. Database Errors

During each line of code, data from different databases may be pulled and put to use. If we encounter database errors while coding for blockchain, we should be much alarmed. The first step is to make sure that there are no corruptions in the database set. Next step, you need to ensure the proper connection between the system and the data.

1. HTTP/API Errors

Depending on the type of error that occurs, certain codes are given for each type. In the world of the world wide web, for instance, HTTP Error 404 means that the page has not been found. Similarly, it is possible to find other types of errors in blockchain development as well.

1. Runtime Errors

We would also have come across runtime errors while dealing with Hyperledger or other types of blockchain development tech. If we see one of these errors, it means that there is something wrong with the process of runtime.

## 3.3 Performance

The primary objective of the evaluation was to assess the performance of the system given the e-voting system requirements presented in section 2 and to identify any considerations with regards to its application in a real-world scenario. The experimentation consisted of multiple steps i.e. conducting multiple transactions, verification of transactions, mining transactions into the blockchain, a reflection of the changes made in the public ledger to all the nodes in the network, and the usability of the system. A test run was made directly at Multichain by starting from asset creation. We can refer to these assets as votes. Since Multichain by default ideally suits cryptocurrency, therefore we wrote our APIs to design it in the context of the vote. To perform the transaction in Multichain, we identified the address and the balance in the address of the node of Multichain from where the asset (vote) will be sent. While sending the asset to the address, the transaction hash was generated carrying the transfer of vote. The balance of the receiving node was incremented by one vote (asset). The transaction becomes a part of the public ledger which shows that it has been mined. A sample transaction within the proposed system. Since our customized API for asset creation is designed in such a way that an address can have at max only one vote (asset), therefore, it will not be possible for a voter to cast multiple votes unless the node receives it from some other address which is only allowed in the case of the candidate.

## 3.4 Reusability

The code written and the components used for the project can be reused and with appropriate changes, we can scale up to meet new requirements as well. The code is written can be further scaled up by importing real-time data into the system.

## 3.5 Application Compatibility

We have designed a webpage that is integrated with the block chain technology that can be accessed on various platforms ranging from smartphones to laptops and computers

## 3.6 Deployment

Heroku/StreamLit - Python code can be deployed into the web application for the E-voting system. Heroku is a platform as a service (PaaS) that enables developers to build, run, and operate applications entirely in the cloud Python apps. StreamLit is an open-source Python library that makes it easy to create and share beautiful, custom web apps for machine learning and data science.

# 4. Conclusion

* The blockchain network has massive scope in the electronic voting process as it helps prevent all kinds of frauds- large as well as a small scale because it is an inherently entire, centralized, open, and consensus-driven technology.
* Since the majority of the implementation takes place at the back end, end users would not notice a significant difference between a blockchain-based voting system and a traditional electronic voting system. Thus, the audience acceptability of this technology can be assumed to be quite high.
* Adopting blockchain voting methods may expose users to unforeseen security risks and flaws, thus a more sophisticated software architecture must be in place which should adequately be complemented with robust managerial expertise.
* In essence, the fact that every vote needs to pass the scrutiny of several validators in the network increases the cost (and decreases the likelihood) of electoral fraud. Indeed, a large number of election stakeholders would need to work together for fraud to occur.