DESIGN OF BLOCKCHAIN BASEDANNONYMOUS SECURE VOTING SYSTEM USING SMART CONTRACT

SUBMITTED BY

BISWAJIT DAS
EXAMINATION ROLL NUMBER: M4SWE19007
REGISTRATION NUMBER: 140965 of 2017-2018

A THESIS SUBMITTED TO
THE FACULTY OF ENGINEERING & TECHNOLOGY OF JADAVPUR UNIVERSITY IN
PARTIAL FULFILLMENT OF THE REQUIREMENTS FOR THE DEGREE OF

MASTER OF ENGINEERING
IN
SOFTWARE ENGINEERING

UNDER THE SUPERVISION OF

MR. UTPAL KUMAR RAY
ASSISTANT PROFESSOR

DEPARTMENT OF INFORMATION TECHNOLOGY
JADAVPUR UNIVERSITY
2019

Department of Information Technology
Faculty of Engineering & Technology

Jadavpur University

Certificate of Submission

*I hereby recommend that the thesis, entitled "**Design of Blockchain based anonymous secure voting system using smart contract**", prepared by **Biswajit Das** (Registration Number:140695 of 2017-2018) under my supervision, be accepted in partial fulfilment of the requirements for the degree of **Master of Engineering** in **Software Engineering** from the **Department of Information Technology** under **Jadavpur University**.*

______________________________-

Mr. Utpal Kumar Ray,
Assistant Professor,
Department of Information Technology,
Jadavpur University

Countersigned by:

______________________________-

Head of the Department,
Information Technology,
Jadavpur University

______________________________-

Dean,
Faculty of Engineering and Technology,
Jadavpur University.

JADAVPUR UNIVERSITY

DEPARTMENT OF INFORMATION TECHNOLOGY
FACULTY OF ENGINEERING AND TECHNOLOGY

CERTIFICATE OF APPROVAL

The thesis at instance is hereby approved as a creditable study of an Engineering subject carried out and presented in a manner satisfactory to warrant its acceptance as a prerequisite to the degree for which it has been submitted. It is understood that by this approval the undersigned does not necessarily endorse or approve any statement made, opinion expressed or conclusion drawn therein, but approve this thesis for the purpose for which it is submitted.

Signature of External Examiner

Prof. Dr. Saswat Chakrabarti
G. S. Sanyal School of Tele-Comm,
IIT Kharagpur, PIN - 721302,
West Bengal
E-Mail : saswat@ece.iitkgp.ac.in

Signature of the Supervisor

Mr. Utpal Kumar Ray
Assistant Professor
Department of Information Technology
Jadavpur University

DECLARATION OF ORIGINALITY AND COMPLIANCE OF ACADEMIC ETHICS

I hereby declare that this thesis contains literature survey and original research work by me, as a part of my Master of Engineering in Software Engineering studies.

All information in this document have been obtained and presented in accordance with academic rules and ethical conduct.

I also declare that, as required by these rules and conduct, I have fully cited and referenced all material and results that are not original to this work.

Name: **BISWAJIT DAS**
Roll Number:**M4SWE19007**
Thesis Title: DESIGN OF BLOCKCHAIN BASED ANNONYMOUS SECURE VOTING SYSTEM USING SMART CONTRACT

________________________________-

Signature (with date)

ACKNOWLEDGEMENTS

I would like to express my heartfelt gratitude to ***Prof. Utpal Kumar Ray,*** *Assistant Professor, Faculty, Department of Information Technology, Jadavpur University, for the help, cooperation and guidance that he has bestowed during the course of this thesis work. I am highly indebted to him for providing a mentally stimulating environment and for all those hours of enjoyable discussions which helped me to complete my thesis work.*

My sincere obligation goes to ***Dr. Bhaskar Sardar****, Head of The Department, Department of Information Technology, Jadavpur University for his constant support.*

I would also like to thank the staffs of the Department of Information Technology, Jadavpur University, and my friends for their help.

(BISWAJIT DAS)
M.E. in Software Engineering
JADAVPUR UNIVERSIT

ABSTRACT

The voting system is one of the great backbone for a democratic country for which the conventional systems or procedures are being followed from centuries. Either the ballot paper is used for voting organized by Election Commission or the Electronic Voting Machine are being used. However, none of the systems are beyond the suspicion of transparency. The Electronic voting machine or ballot paper results in long queue for common people and elderly person with the possibility of duplicate voting or booth capturing. These scenarios demand a design and implementation of efficient electronic voting system that can't be tampered at all.

The technological advancements are leading towards the invention of newer technologies that is revolutionizing various field with its socio-economic impacts. Blockchain is one of these emerging technologies that have dragged all the attention of researchers for its capability of security and transparency. Blockchain refers to the growing list of record linked together with the use of cryptographic algorithm. The secure hashing techniques of SHA are used to generate hash with timestamp, transaction data and hash of previous block. This cryptographically linked list forms a distributed decentralized database of records in a peer to peer network where every peer holds a public ledger. The consensus of a majority of the participants verifies each and every transactions taking place in the network leading towards the tamper proof system from where transaction data once done can't not be modified.

Smart contracts, on the other hand, are a set of instructions or lines of codes deployed in the Blockchain platform which executes when the predefined terms and conditions are satisfied. The smart contract code facilitates, verifies, and enforces the negotiation or performance of an agreement or transaction. It is the simplest form of decentralized automation.

This thesis intends to carry an in-depth and thorough study on Blockchain technologies and its implications. The thesis also investigates the issues and problems present in the online and offline voting system based on the current scenario and used infrastructure. The study in thesis also proposes an efficient online voting system leveraged by the non-tampering infrastructure provided by the Blockchain technology.

Keywords: Blockchain, smart contract, solidity, Ethereum

CONTENT

LIST OF FIGURES

CHAPTER 1 INTRODUCTION

Chapter gist: In this chapter we discuss the motivation, focuses and organization of the thesis.

Chapter contents:

1.1MOTIVATION

The foundation of a democratic society is laid by its voting mechanism. The traditional ballot enables the general public to visit allocated polling station and vote for candidate. Votes are recorded in a ballot and checked once the whole process is complete. According to the survey done by Heritage Foundation, approximately 1,199 cases are found to be proven instances of the voter fraud across the US [1].

Estonia was the first in the world to introduce an electronic voting system for their national elections. Soon after electronic voting was adopt by Norwegian for its council election. But both Estonia and Norwegian electronic voting systems problem is the secrecy of critical part of code. Estonia I-voting system is made close what raise question about transparency and security.

The real voter fraud case that happened in the Washington, US. During 2008 election issues is ballot fraud, illegal double voting.

Preventing such election fraud is crucial to safeguarding the virtue of the voting process. The traditional voting system does offer anonymity to the voter but the counting process is not transparent. Only election officials know what happens to your vote once you punch it, People have only choice to trust the voting result, declare by an election commission or government entities.

There are various issues related to the current voting system such as voting stuffing, booth capturing, outdated voting machines and big queues at polling places.

In comparison with the current system, the Blockchain voting system is fair, secure, and transparent and offer real time counting and processing.

In elections, we usually have a central authority who records, counts and checks all the votes and with Blockchain, the process is decentralized. So everyone can hold a copy of the full voting record on their own devices. The data is encrypted to protect the identity of individual voters. Illegitimate votes cannot be added and the historical record cannot be changed because everyone holds a copy and check that all the votes comply with the rules and counted properly.

1.2.FOCUS

The aim of this project is to gain knowledge of the Blockchain Technology and focus on real world use cases. We are briefly discuss design of electronic voting system which provides voting system is fair, secure, and transparent and offer real time counting and processing with recent technology.

In order to deal with the practice of Blockchain, we describe related terminology associated with it. We describe core concept like secure hash function, cryptography, decentralization, ledger and miner. We also describe smart contract and solidity programming language that is used to write our voting contract. We try to give an overview of basic Blockchain related mustknow theory.

Then we briefly describe and compare the different voting process or techniques namely one paper ballot, two electronic voting.We also discuss a newly proposed voting scheme improving upon the works of smart contract written in solidity programming language underline technology is Blockchain.

1.3.THESISORGANIZATION

The outline of the thesis is as follows:

In chapter 1, we discuss motivation towards the thesis and what actually I focus to achieve our require things.

In chapter 2, we state some introductory theories and history on Blockchain. We touch upon the component, advantage over centralized system and explain types of cryptography. We give a brief definition of hash function, Merkle tree and cryptography. We also explain Ethereum Blockchain, compared Ethereum architecture to webapp architecture for better understanding of Ethereum architecture and discuss about core component of Etrhereum.

In chapter 3, we give a brief idea about application and use cases of Blockchain in both financial and non-financial aspect. First we explain some financial application or uses cases like "Digital currencies and global payment system" and "Compliance and mortgage". Then we explain some non-financial use cases i.e. Blockchain in healthcare, decentralized data storage, voting system and digital identification.

In chapter 4, we first state what we mean by smart contract. Then we give brief idea how smart contracts works with an example. We explain benefit of smart contract and how to write a contract by solidity programming language.

In chapter 5 we briefly demonstrate about e-voting result in Rinkeby Ethereum Blockchain. We also discuss how smart contract implement to principle of voting.

In chapter 6, we explain proposed work related info. Firstly we go through literature survey and the drawbacks and issues in present system. We give a proposed architecture of our voting system. We briefly discuss election stages and implementation voting contract. Lastly we demonstrate whole thing in details in result and discussion section

In chapter 7, we states conclusion and future work in this domain.

CHAPTER

2

BLOCKCHAIN TECHNOLOGY

Chapter gist:In this chapter we give a brief idea of the history of Blockchain technology, how Blockchain works, core component of Blockchain and Ethereum Blockchain.

Chapter contents :

2.1. BriefHistory Of Blockchain Technology

When Satoshi Nakamoto, Whose true identity is still unknown, released the whitepaper Bitcoin: A Peer to peer Electronic Cash system in 2018 that describes a purely peer to peer version of electronic cash known as Bitcoin. Blockchain technology made its public debut. Blockchain the technology that runs Bitcoin has developed over the last decade into one of the today's biggest technology.

We have given a brief history in the following diagram.

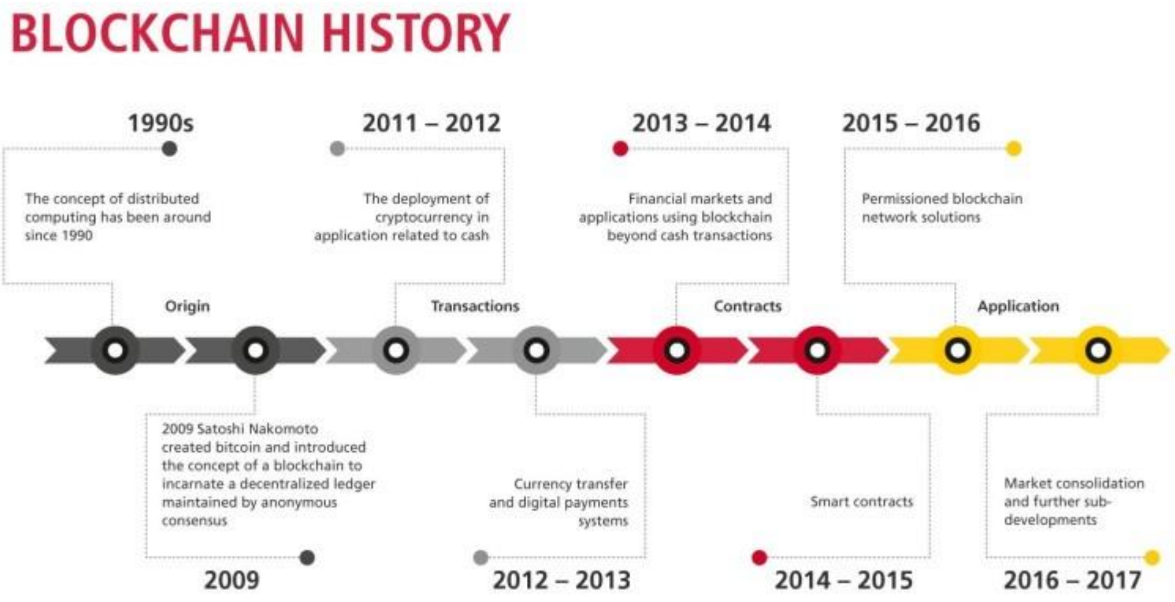

BLOCKCHAIN HISTORY
1990s
The concept of distributed computing has been around since 1990
2011 – 2012
The deployment of cryptocurrency in application related to cash
2013 – 2014
Financial markets and applications using blockchain beyond cash transactions
2015 – 2016
Permissioned blockchain network solutions
Origin
Transactions
Contracts
Application
2009 Satoshi Nakomoto created bitcoin and introduced the concept of a blockchain to incarnate a decentralized ledger maintained by anonymous consensus
2009
Currency transfer and digital payments systems
2012 – 2013
Smart contracts
2014 – 2015
Market consolidation and further sub-developments
2016 – 2017

Fig 1: Blockchain history

2.2.What Is Blockchain??

Blockchain is a growing list of records, called blocks, which are linked using cryptography. Each block contains a cryptographic hash of the previous block, a time-stamp, and transaction data. Inothers word a Blockchain is essentially a distributed database of records, or public ledger of all transactions or digital events that have been executed and shared among participating parties. Each transaction in the public ledger is verified by consensus of a majority of the participants in the system once entered, information can never be erased. The Blockchain contain certain and verifiable record of every single transaction ever made.

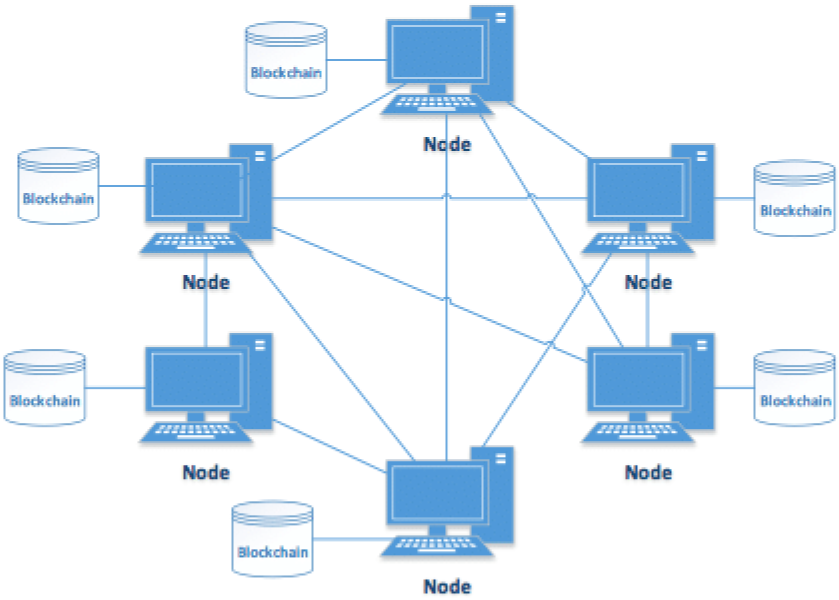


Fig 2:Peer to peer network in Blockchain

Bitcoin [2] is the most popular example that is intrinsically tied to Blockchain technology. It is also the most controversial one since it helps to enable a multibillion-dollar global market of anonymous transactions without any governmental control. Hence it has to deal with a number of regulatory issues involving national governments and financial institutions.

However, Blockchain technology itself is non-controversial and has worked flawlessly over the years and is being successfully applied to both financial and non-financial world applications.

Current digital economy is based on the reliance on a certain trusted authority. All online transactions rely on trusting someone to tell us the truth— it can be an email service provider telling us that our email has been delivered; it can be a certification authority telling us that a certain digital certificate is trustworthy; or it can be a social network such as facebook telling us that our posts regarding our life events have been shared only with our friends or it can be a bank telling us that our money has been delivered reliably to our dear ones in a remote country. The fact is that we live our life precariously in the digital world by relying on a **third entity** for the security and privacy of our digital assets. The fact remains that these third party sources can be hacked, manipulated or compromised.

This is where the Blockchain technology comes handy. It has the potential to revolutionize the digital world by enabling a distributed consensus where each and every online transaction involving digital assets, past and present, can be verified at any time in the future. It does this without compromising the privacy of the digital assets and parties involved.

2.3. Component Of Blockchain

Blockchain is emerging technology build with different technologies:

2.3.1. Cryptographic Secure Hash Function

Basic fundamental which is very useful is context of Blockchain, cryptographic secure Hash function is one of them.

Hash Function:

Hash function is a function that map any sized data to a fixed size. For example, if you define hash function like **H(x) = x%n**, where x and n are integers and % is the modular operations (remainder after division by n). So, if we define a hash function like this way; that means, in this case you can see that whatever be the value of x the value of **H(x)** will be in between 0 and n-1. So, this type of functions we call it a kind of hash function.

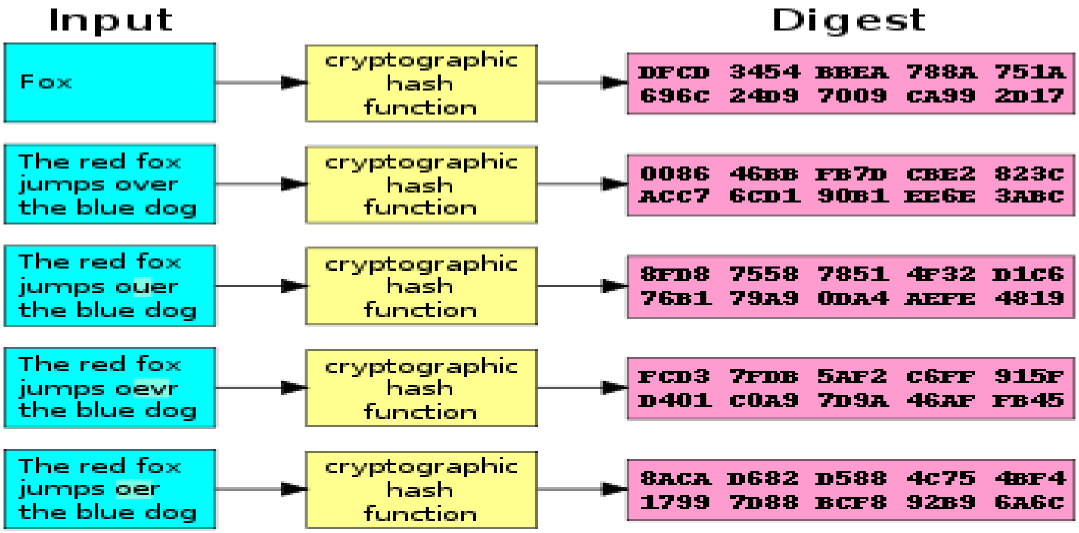

Input
Digest
Fox
cryptographic hash function
DFCD 3454 BBEA 788A 751A 696C 24D9 7009 CA99 2D17
The red fox jumps over the blue dog
cryptographic hash function
0086 46BB FB7D CBE2 823C ACC7 6CD1 90B1 EE6E 3ABC
The red fox jumps ouer the blue dog
cryptographic hash function
8FD8 7558 7851 4F32 D1C6 76B1 79A9 0DA4 AEFE 4819
The red fox jumps oevr the blue dog
cryptographic hash function
FCD3 7FDB 5AF2 C6FF 915F D401 C0A9 7D9A 46AF FB45
The red fox jumps oer the blue dog
cryptographic hash function
8ACA D682 D588 4C75 4BF4 1799 7D88 BCF8 92B9 6A6C

Fig 3: Hash Function

PROPERTIES:

- Hash function is a **one way,** cannot be reversed; that means, given an x and n you can compute H(x). But if H(x) is given then you cannot say that what is the corresponding value of x, you cannot say it uniquely.
- It is infeasible to find two different messages with the same hash value i.e. hard to find collisions. For different x1 and x2 ,H(x1) and H(x2) should be different.
- Output does not reveal information on input.
- A small change to a message should change the hash value so extensively that the new hash value appears uncorrelated with the old hash value. This is so called avalanche effect.

So, that is the property of a hash function and this concept of hash function is widely used in the concept of Blockchain or indeed Blockchain is nothing, but on data structure which is built upon this concept of hash function.

2.3.2 MERKLE TREE:

A hash tree or Merkle tree [3] is a tree in which every leaf node is labelled with the hash of a data block, and every non-leaf node is labelled with the cryptographic of the labels of its child nodes. The concept of hash trees is named after Ralph Merkle who patented it in 1979 [4].Hash trees allow efficient and secure verification of the contents of large data structures. In block in Blockchain transactions are represented as a leaf node of Merkle tree.

In the Blockchain context every transaction has a hash associated with it .In a block, all of the transaction hashes in the block are themselves(sometimes several times-the exact process is complex), and the result is the **merkle root**. In other words, the merkle root is the hash of all the hashes of all the transaction in the block. The merkle root is a part of the block header.

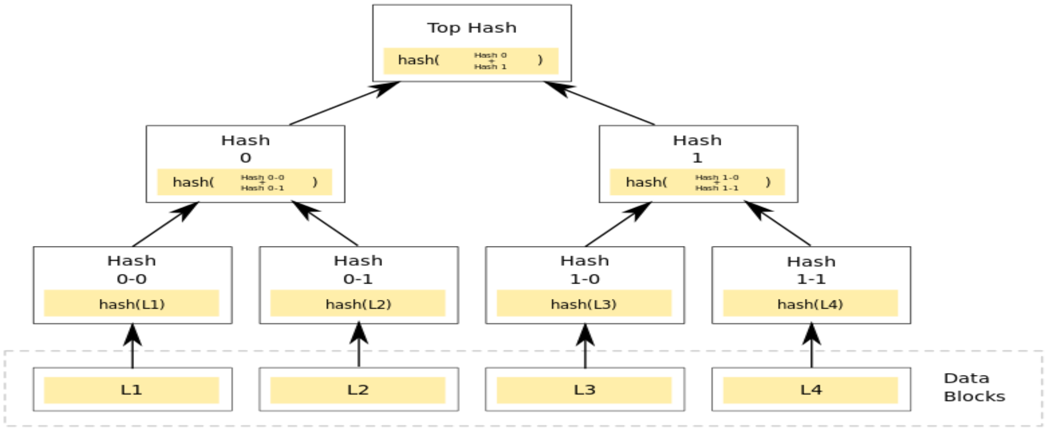


Fig 4: Merkle tree

This is an example of a binary hash tree. Hashes 0-0 and 0-1 are the hash values of data blocks L1 and L2, respectively, Hashes 1-0 and 1-1 are the hash values of data blocks L3 and L4, respectively. Hash 0 is the hash of the concatenation of hashes 0-0 and 0-1 and Hash 1 is the hash of the concatenation of hashes 1-0 and 1-1. Merkle root hash is the hash of the concatenation of hashes 0 and 1.

2.3.3.Block

The Block contains the ledger/batches of valid transactions that are hashed and encoded into a Merkle tree. The block data can be hashed by cryptographic hash functions. Each block includes the cryptographic of the prior block in the Blockchain, linking the two. The linked blocks form a chain.

Basic essential structure of a block

- Block No
- Timestamp
- Data

  Transactions/ledger as a merkle tree
- Previous block hash
- Nonce
- Block hash

Block hash: Hash identifies a block and all of its contents and it's always unique, just as a fingerprint. Once a block is created, its hash is being calculated. Changing something inside the block will cause the hash to change.

So in another word: hashes are very useful when you want to detect changes in blocks. If the fingerprint of a block changes, it no longer is the same block.

Hash of previous block: The hash of the previous block effectively creates a chain of blocks and it’s this techniquethat makes a Blockchain more secured.

Data: The data that is stored inside the block depends on the type of Blockchain.In Bitcoin Blockchain transactions are data.

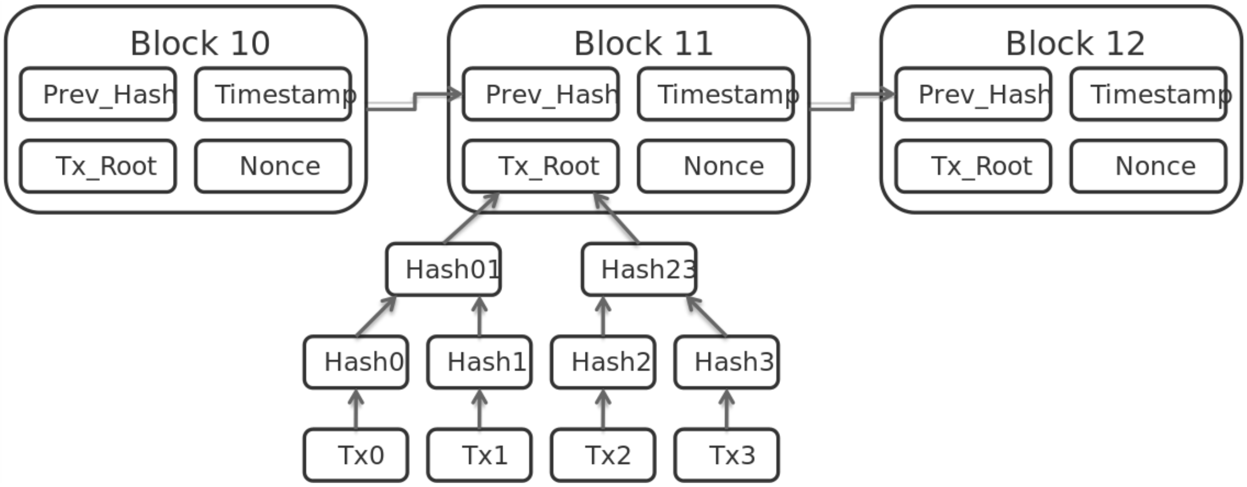


Fig 5: Blockchain (hash of the previous block effectively creates a chain of blocks)

Nonce: a number that can only be used once - in cryptography is a one-time code, pseudorandom manner, which is used to preventing to take the power of reproduction.

2.3.4.CRYPTOGRAPHY

The type of cryptography used in Blockchain, namely public key cryptography, also known as asymmetric cryptography.

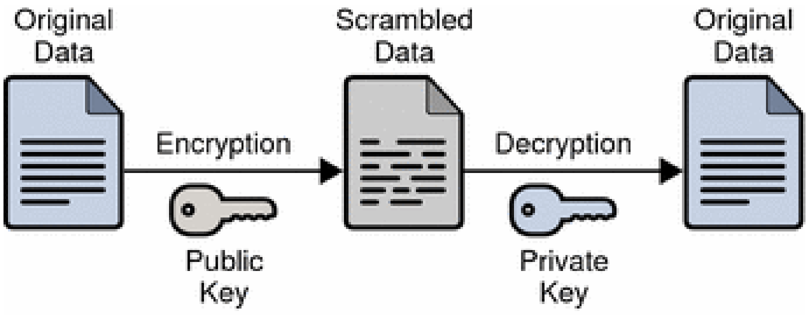


Fig 6: public key cryptography

Blockchain technology utilizes cryptography as a means of ensuring transactions are done safely, while securing all information and storages of value. Therefore, anyone using Blockchain can have complete confidence that once something is recorded on a Blockchain, it is done so legitimately and in a manner that preserves security.

In Blockchain, cryptography is primarily used for two purposes:

I. Securing the identity of the sender of transactions.

II. Ensuring the past records cannot be tampered with.

2.4 TYPE OF BLOCKCHAIN

There are various (often conflicting) categorizations of Blockchain types, and for the purposes of this we will focus on the different types of Blockchain according to whether

Authorization is required for network nodes which act as verifiers, and whether access to the Blockchain data itself is public or private [5].

For the 1st categorization we have:

- ❖ **Public Blockchain**, where anyone can read and submit transactions to the Blockchain.
- ❖ **Private Blockchain**, where this permission is restricted to users within an organization or group of organization

For the second categorization we have:

- ❖ **Permission less Blockchain**, Where anyone can participate in the verification process, i.e. no prior authorization is required and a user can contribute his/her computational power, usually in return for a monetary reward.
- ❖ **Permissioned Blockchain,** where verification nodes are preselected by a central authority or consortium.

In reality, most permissionless Blockchain feature public access, while the intention of most permissioned Blockchain is to restrict data access to the company or consortium of companies that operate the Blockchain.

2.5.HOW BLOCKCHAIN WORKS

Just like many other technologies for the internet, Blockchain rely on public key cryptography to protect users from having unauthorized persons take control of their accounts. The private and public key pairs enable people to encrypt information to transmit to each other, where the receiving party would then be able to determine whether the message actually originated from the right person, and whether it had been tampered with. This is critical when one needs to communicate to a network that a transaction between two parties has been agreed.In addition, the presence of an ability to identify the integrity of the data.

However, the Blockchain technology is applicable to any digital asset transaction exchanged online. We explain the concept of the Blockchain by explaining how Bitcoin works since it is intrinsically linked to the Bitcoin.

Internet commerce is exclusively tied to the financial institutions serving as the trusted third party who process and mediate any electronic transaction. Bitcoin uses cryptographic proof instead of the trust-in-the- third-party mechanism for two willing parties to execute an online transaction over the Internet. Each transaction is protected through a digital signature, is sent to the “public key” of the receiver, and is digitally signed using the “private key” of the sender. In order to spend money, the owner of the cryptocurrency needs to prove his ownership of the “private key”.

Each transaction is broadcasted to every node in the Bitcoin network and is then recorded in a public ledger after verification. Every single transaction needs to be verified for validity before it is recorded in the public ledger. The verifying node needs to ensure two things before recording any transaction:

1. Spender owns the cryptocurrency, through the digital signatureverification on the transaction.
2. Spender has sufficient cryptocurrency in his account, through checking every transaction against the spender’s account, through checking every transaction against the spender’s account, or “public key” that is registered in the ledger. This ensures that there is sufficient balance in hisaccount before finalizing the transaction.

However, there is question of maintaining the order of these transactionsthat are broadcasted to every other node in the Bitcoin peer-to-peer network. Need to develop a mechanism so that the entire Bitcoin network can agree regarding the order of transactions in a distributed system. The Bitcoin solved this problem by a mechanism that is now popularly known as Blockchain technology. The Bitcoin system orders transactions by placing them in groups called blocks and then linking these blocks through what is called Blockchain. These blocks are linked to each-other (like a chain) in a proper linear, chronological order with every block containing the hash of the previous block.

There still remains one more problem: Any node in the network can collect unconfirmed transactions and create a block and then broadcast it to the rest of the network as a suggestion as to which block should be the next one in the Blockchain. There can be multiple blocks created by different nodes at the same time. How does the network decide which block should be next in the Blockchain?

Bitcoin solves this problem by introducing a mathematical puzzle: each block will be accepted in the Blockchain provided it contains an answer to a very special mathematical problem. Chain provided it contains ananswer to a very special mathematical problem. The first node, to solve the problem, broadcasts the block to the rest of the network. Occasionally, however, more than one block will be solved at the same time, leading to several possible branches. However, the math needed to be solved is verycomplicated and hence the Blockchain quickly stabilizes: after this, everynode is in agreement about the ordering of blocks. The nodes donating their computing resources to solve the puzzle and generate blocks are called “miner” nodes” and are financially awarded for their efforts.

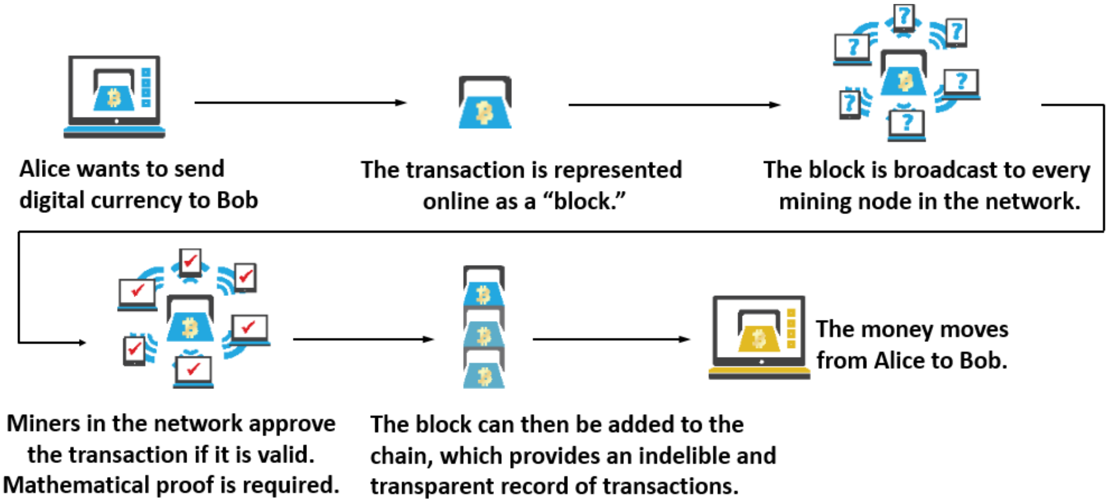

Alice wants to send digital currency to Bob
The transaction is represented online as a "block."
The block is broadcast to every mining node in the network.
Miners in the network approve the transaction if it is valid. Mathematical proof is required.
The block can then be added to the chain, which provides an indelible and transparent record of transactions.
The money moves from Alice to Bob.

Fig 7: Blockchain – Flow diagram

2.6.ETHEREUM BLOCKCHAIN

2.6.1.WHAT IS ETHEREUM

Ethereum [6] is an open source, public Blockchain based distributed decentralized platform that runs smart contracts. Ethereum is a public, Blockchain based distributed computing platform. It can be thought of as **one big computer** made up of small computers around the world. You can write applications and run them on this global computer. The platform guarantees that your application will always run without any downtime, censorship, and fraud or third-party interference. Apart from running applications, Ethereum Blockchain can also transfer money between 2 parties without a central authority.

All these computers (also called nodes) are connected to one another and have a **full copy of the code and data**. When you deploy your code on to the Ethereum Blockchain, the code is **replicated across all the nodes** in the network. When your application stores any data, even that **data is replicated** across all the nodes. There are thousands of nodes in the network and it is almost impossible for anyone to stop all the nodes. This insures your application to be always accessible.

Ethereum was proposed in late 2013 by Vitalik Buterin, a cryptocurrency researcher and programmer. Development was funded by an online crowdsale that took place between July and August 2014[7].

Ether is a token whose Blockchain is generated by the Ethereum platform. *Ether* can be transferred between accounts and used to compensate participant mining nodes for computations performed. Ethereum provides a decentralized virtual machine, the Ethereum Virtual Machine (EVM), which can execute scripts using an international network of public nodes.

2.6.2. WEBAPP ARCHITECTUREVS. ETHEREUM ARCHITECTURE

One of the best ways to understand Ethereum is by comparing it with traditional client/server architecture. A typical web application consists of server side code which is usually written in a programming language like Java, C#, Ruby, Python etc. The frontend code is implemented using HTML/CSS/JavaScript. This entire application is then hosted on a hosting provider like AWS, Microsoft Azure, Google Cloud Platform, Heroku or a VPS.

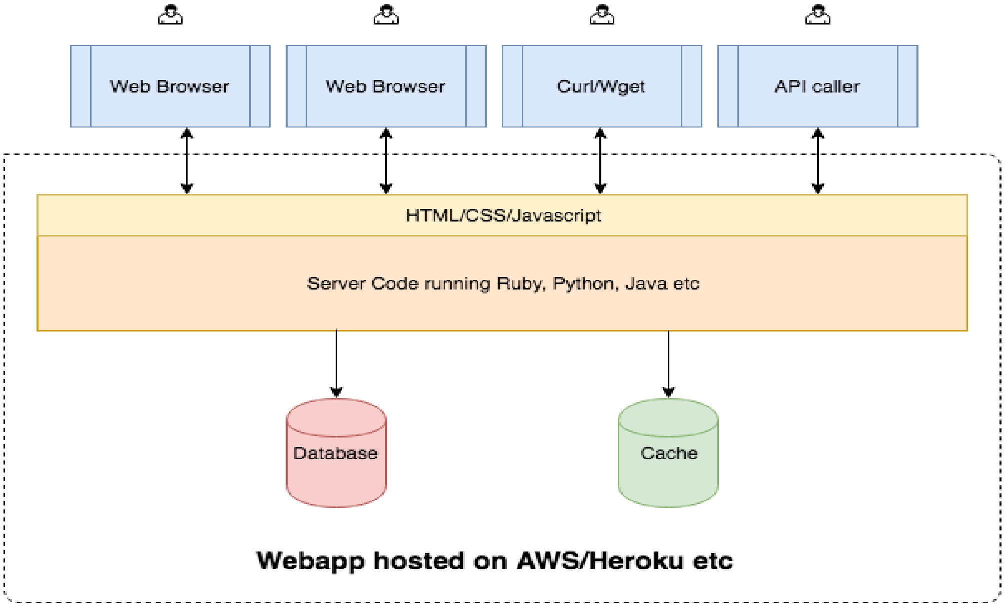

Web Browser
Web Browser
Curl/Wget
API caller
HTML/CSS/Javascript
Server Code running Ruby, Python, Java etc
Database
Cache
Webapp hosted on AWS/Heroku etc

Fig 8: Webapp Architecture

If you notice the diagram on the right, every client (browser) communicates with its own instance of the application. There is no central server to which all clients connect to. This means, in an **ideal decentralized world**, every person who wants to interact with a dapp (Decentralized Application) will need a full copy of the Blockchain running on their computer/phone etc. That means, before you can use an application, you have to download the entire Blockchain and then start using the application.

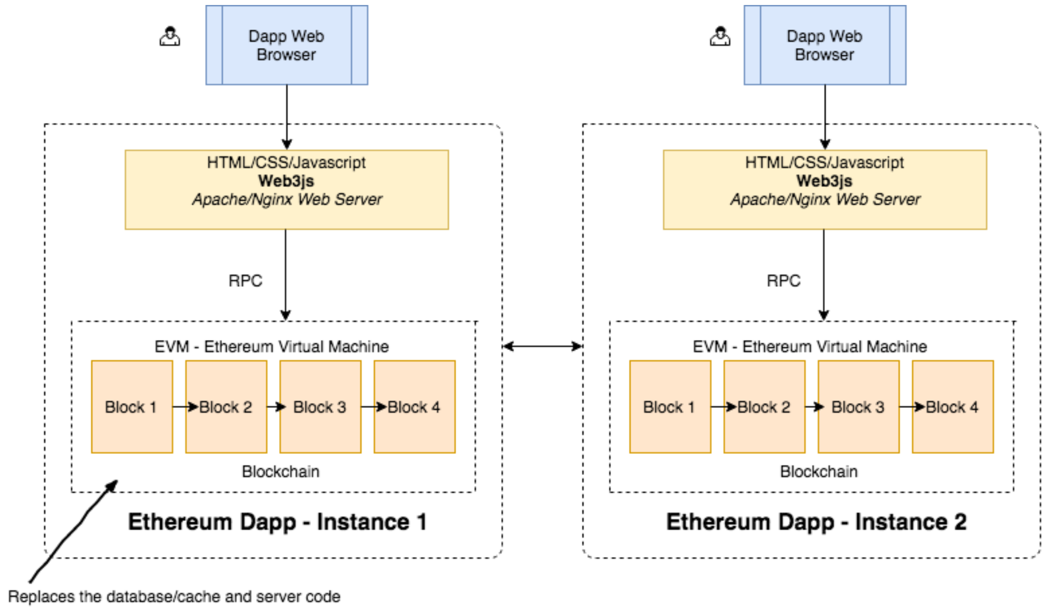


Fig 9: Ethereum Architecture

2.6.3. Core Component Of Ethereum

- **Contracts**

A contract is written insolidity, a smart contract language, and is a collection of code and data that resides at a specific address on the Ethereum Blockchain. It's very similar to a class in Object Oriented Programming, where it includes functions and state variables. Smart Contracts, along with the Blockchain, are the basis of all Decentralized Applications. They are, like Blockchain, immutable and distributed, which means upgrading them will be a pain if they are already on the Ethereum Network. Fortunately.

- **The Ethereum Virtual Machine (EVM)**

EVM handles the internal state and computation of the entire Ethereum Network. Think of the EVM as this massive decentralized computer that contains "addresses" that are capable of executing code, changing data, and interacting with each other.

- **Web3.js**

Web3.js is a JavaScript API that allows you to interact with the Blockchain, including making transactions and calls to smart contracts. This API abstracts the communication with Ethereum Clients, allowing developers to focus on the content of their application.

- **DECENTRALIZED APPLICATION(DApp)**

Decentralized applications or DApps are computer applications that operate over a blockchain (or have their own blockchain) enabling direct interaction between end users and providers (e.g. connecting buyers and sellers in some marketplace, owners and storers in file storage).

In Ethereum's purview, a DApp can be anything that enables transactions over the Ethereum framework, typically interfacing with users via an HTML/Javascript web application using a Javascript API to communicate with the blockchain.

- **ETHER**

Ether is a necessary element—a fuel—for operating the distributed application platform Ethereum. Every time a contract is executed, Ethereum consumes tokens which is termed as '**gas**' to run the computations.Ether can be used for executing contracts or manage transactions over Ethereum. To put it another way, ether is the incentive ensuring that developers write quality applications (wasteful code costs more), and that the network remains healthy (people are compensated for their contributed resources).

CHAPTER 3 APPLICATION OFBLOCKCHAIN

Chapter gist: In this chapter we give a brief idea about application and use cases of Blockchain in both financial and non-financial aspect.

Chapter contents :

3.1. Smart Contract And Automated Transaction

The advantages of Blockchain technology outweigh theregulatory issues and technical challenges. One key emerging use case of blockchain technology involves “smart contracts”. Smart contracts are basically computer programs that can automatically execute the terms of a contract. When a preconfigured condition in a smart contract among participating entities is met then the parties involved in a contractual agreement can be automatically made payments as per the contract in a transparent manner.

Smart Property is another related concept which is regarding controlling the ownership of a property or asset via blockchain using Smart Contracts. The property can be physical such as car, house or smartphone, or it can be non-physical such as shares of a company. It should be noted here that even Bitcoin is not really a currency: Bitcoin is all about controlling the ownership of money.

The development of smart contracts is expanding rapidly. Over the past several months, a number of open source projects—such as Ethereum, Counterparty and Mastercoin—have been developed to create programming languages that enable the creation of increasingly sophisticated smart contracts. Using these programming languages, smart contracts could be used to enable employees to be paid on an hourly or daily basis with taxes remitted to a governmental body in real time. The technology could be employed to create smart contracts that automatically check state death registries and allocate assets from a testator’s estate, send applicable taxes to governmental agencies without the need of administering the will through probate [8]

3.1. Blockchain In Financial Services

3.1.1 Digital Currencies And Global Payment System

One of the earliest applications for Blockchain technology has been digital currencies such as Bitcoin. Released in 2009 by Satoshi Nakamoto (a pseudonymous individual or group) [9], Bitcoin relies on a decentralized Blockchain to establish a digital currency that, unlike the US dollar, does not depend on any bank or government. As explained by Nakamoto, the system is “completely decentralized, with no central server or trusted parties, because everything is based on crypto proof instead of trust.”[10]. It is powering an entirely new payments system that allows for the seamless transfer of funds around the globe. Unlike existing payments systems, which generally take days to transfer funds, Bitcoin can be sent across the world in a little over seven minutes at fees that are drastically lower than those imposed by existing money transmitters, such as Western Union. All that is needed is an Internet connection and a computer or a simple mobile device [11].

Adoption of Bitcoin has spread rapidly [12], and the currency—as well as its many imitators [13]—has the potential to be the first breakthrough Application that relies on Blockchain technology.

Unlike Bitcoin there are many crpyptocurrency available in market works through distributed ledger technology, typically a Blockchain and strong cryptography to secure financial transactions. For example Litecoin(LTC),Ethereum(ETH),Ripple(XRP) Peercoin(PPC), Namecoin(NMC) etc.[14].

3.1.2. Compliance And Mortgage

Blockchain could improve compliance procedure, improve regulatory efficiency and provide faster and more accurate reporting that draws immutable data source [15].

Buying a house is the largest and most complex financial transaction that most people will ever complete. In the United States, 65% of homeowners have a mortgage, and for most of them, the process was likely complex, expensive, and time consuming.

Putting mortgages on the blockchain has the potential to transform the mortgage process. Blockchain’s distributed ledgers and smart contracts will remove friction and save money for both lenders and homeowners. The decentralized nature of the blockchain will also extend the possibility of home ownership to people who might currently be excluded based on traditional methods of obtaining financing.

Save time and money

Recording mortgage agreements on the blockchain makes it possible for lenders and buyers to interact directly, without the need for an intermediary to manage the transaction. Instead of moving physical documents, all parties access a single source of information[16].A recent study by Capgemini indicates that adopting blockchain in the mortgage loan industry could result in savings of US$480 to US$960 per loan for consumers. In addition, banks could potentially cut costs from US$3B to US$11B annually through lowered processing costs [17], and some of those savings would ultimately extend to home buyers.

Information sharing

Distributed ledgers allow for information to be transferred seamlessly between parties. Because smart contracts execute automatically when agreed-upon terms are completed, errors and delays in processing are eliminated. With no need for paper documents, the possibility of introduced errors is also removed, and the immutable nature of blockchain records generates a built-in level of trust and objectivity. Transactions are visible to all parties and are verified automatically, with no possibility of tampering.

Regulatory compliance

The distributed ledger upon which blockchain records reside is both indelible and transparent. Placing mortgage records on DLT allows lenders to prove that loan estimates sent within guidelines and that all regulations have been followed. In the United States, for example, the Dodd-Frank Act requires mortgage brokers to be more transparent in their dealings with borrowers. Recording contract details on the blockchain makes it easier to meet these regulatory obligations. Blockchain also provides easy access to information that proves documents and data have been through the necessary compliance checks [18].

Block66 is a blockchain platform that will use DLT to reduce the risks of mortgage fraud. By placing the complete history of every transaction on the blockchain, Block66 not only removes inefficiencies in the mortgage application and approval process, but also creates a transparent audit trail [19].

3.2. BLOCKCHAIN IN GOVERNMENT SERVICES

3.2.1 HEALTH RECORD STORAGE AND ACCESS

A decentralized database which is consistently held up to date presents many advantages to the healthcare industry. These advantages become especially interesting, when many different parties need access to the same information [20] [21] [22]. Medical treatment processes, for example in the area of elderly care or chronic diseases, are predestined fields of Application where Blockchain technology can create added value. A variety of involved parties (e.g., general practitioners, medical specialists, hospitals, therapists, etc.) and the amount of media disruptions involved during the treatment of a patient(e.g., change of communication media, various medical health records, incompatible IT interfaces, etc.) can lead to time consuming and resource intensive authentication and information processes for all medical stakeholders involved.

This network is thus an example of a Blockchain approach that provides all relevant medical stakeholders transparent and clear access to the latest treatment information. On the one hand, this can limit medical negligence due to out-dated information and thereby prevent health issues in an early stage. On the other hand, it allows medical experts involved to track the interactions between the patient and all physicians which have taken place in the past. Consequently, the entire treatment of a patient is characterized in a transparent manner, whereby a completely new information and confidence level between all medical stakeholders is created.

3.2.1 DECENTRALIZED DATA STORAGE

A distributed cloud-storage [23] consists of a peer-to-peer decentralized cloud storage solution. It protects your files, both on the nodes and in transmission, by using Blockchain technology and cryptography to encrypt files. In principle, a distributed cloud storage system where every aspect of cloud storage such as transport, processing, or storage of data is entered into the Blockchain. Data centers are the hub of cloud storage capabilities for cloud giants like AWS, Microsoft Azure, and Drop box. But, these data centers come with a high price tag for cloud developers, providers, and users. Moreover, they come with an even higher cost of data failures and security breaches. From networking equipment and physical servers to other infrastructure demands like electricity, cloud service providers are spending billions of dollars every quarter just to maintain or grow their service offerings.

The Blockchain is revolutionizing cloud storage by putting the user back in control over their data and devices. The decentralized aspect of Blockchain means that there are no central servers to be compromised, and because of the use of client-side encryption, only the end users have complete access to their un-encrypted files and encryption keys. Thus, distributed cloud storage enables users to store data in a secure and decentralized manner. This is done by using Blockchain features such as transaction ledgers, cryptographic hash functions, and public/private key encryption. Major benefits of distributed cloud storage are Tamper-proof data, Verifiability and No more middlemen. Backup and storage specialists have proved that stored data has not been tampered with, when an effective and verifiable backup was created. Distributed cloud storage based on Blockchain technology stores only hashes of its data blocks. And the encrypted and distributed hashes are enough to verify these data blocks Blockchain does not just store data in a distributed and encrypted form, but also provides for a sequential chain in which every block contains a cryptographic hash of the block.

Storj [24] is based on Blockchain technology and peer-to-peer protocols to provide the most secure, private, and encrypted cloud storage. Emercoin,Factor,Nxt and Eris[25] are cloud data storage which behind technology is Blockchain.

3.2.3. VOTING SYSTEM

Election has a great power in determining the fate of a nation or organization. The aspect of security and transparency is a threat from still widespread election with the conventional system. General election still use a centralized system, there is organization that manages it. Some of the problems that can occur in traditional election systems are with an organization that has full control over the database and system. It is possible to tamper with the database of considerable opportunities.

Blockchain technology is one of solution; because it embraces a distributed system and the entire database are owned by many users. By adopting Blockchain in the distribution of databases on electronic voting can reduce one of the fraud sources of database manipulation.

Followmyvote is world’s first open-source online voting solution based on Blockchain.

3.2.4. DIGITAL IDENTITY VERIFICATION

Digital identity is a critical component in service delivery. However, relying on physical identity documents and conventional approaches in a digital age is not only cumbersome and tedious, but also expensive.

Furthermore, as we spend more and more time online, fraudsters and hackers are gradually discovering new ways to access sensitive personal information and use it for selfish gains at the expense of legitimate users as well as their online service providers.

These challenges call for new approaches in identity verification. Let’s dive in and explore how Blockchain will disrupt this industry:

SELF-SOVEREIGN ID

Blockchain has an appealing trait for identity. It offers an avenue where only you and you alone are in control of your identity. You will be able to update it at any time; and you can choose to make it public or private.

Your identity is available to you at any time and it cannot be accessed by anyone without your consent. And best part of it is that no third party will ever be involved when it comes to your identity. A good example here is VALID.

REDUCE IDENTITY THEFT

Thirdly, identity theft is an ugly experience. It takes you from a hundred to zero in a matter of minutes. Then you have to go on resetting everything. Call your credit card companies to have all your cards changed. Dispute charges that you know nothing about. It is not only unfair, but very frustrating.

Blockchain powered solutions are decentralized and have no single point of failure, this means there is no way a hacker will be able to hack into the system and access your information. Civic, is a good example of a Blockchain based solution that aims to reduce fraud and protect users from identity theft.

KYC AND DIGITAL IDENTITY

Blockchain makes it possible for financial institutions to access customers’ information without always starting the process of due diligence again and again. It will also allow for the creation a digital identity for customers. With a digital identity, financial institutions will be able to monitor your transactions, as a customer, access relevant information about you and reduce false positives. Furthermore, financial institutions that identify fraudulent transactions will be able to notify other financial institutions globally, preventing more fraudulent activities [26].

CHAPTER
4
SMART CONTRACTS IN E-VOTING CONTRACT

Chapter gist: In this chapter we will discuss about smart contract, how it works and some benefits of smart contract. We also describe solidity programming language which is used for writing a contract.

Chapter Content:

4.1. WHAT ARE SMART CONTRACTS?

Smart contracts are lines of code that are stored on a Blockchain and automatically execute when predetermined terms and conditions are met.If and when the pre-defined rules are met, the agreement is automatically enforced. The smart contract code facilitates, verifies, and enforces the negotiation or performance of an agreement or transaction. It is the simplest form of decentralized automation [27].

The term smart contract was first used by Nick Szabo in 1997, long before Bitcoin was created.Nick Szabo is a computer scientist, law scholar & cryptographer. He wanted to use a distributed ledger to store contracts. Smart contracts are just like contract in the real world. The only difference is that they are completely digital. Smart contract is actually a tiny computer program that is stored inside the blockchain.

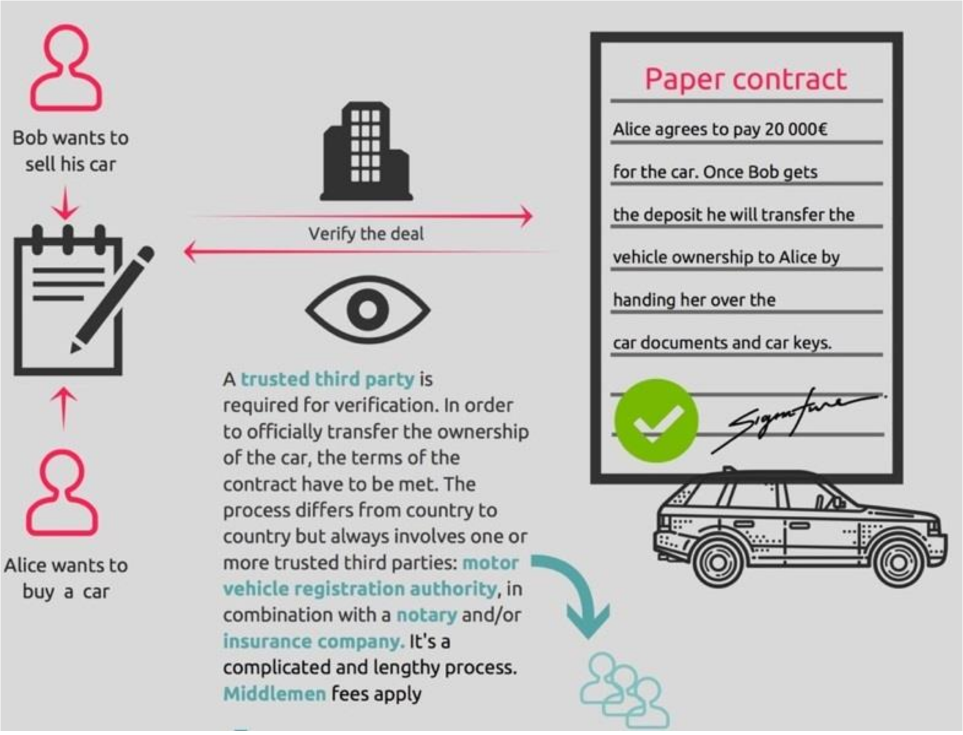
Let’s take an example to understand how smart contracts work.

4.2. HOW SMART CONTRACTS WORKS

If A and B don’t know and don’t trust each other, they usually need a trusted third party to serve as an intermediary to verify transactions and enforce them. With smart contracts &blockchain, you don’t need those trusted intermediaries anymore for clearing or settlement of your transactions. Take the example of buying and selling a car:

First we see how a traditional contract works and then we see how a smart contracts works with an example.

If Alice wants to purchase a car from Bob, a series of trusted third parties are required to verify and authenticate the deal. The process differs from country to country but always involves at least one, but usually more, trusted third parties: motor vehicle registration authority, in combination with a notary and/or insurance company. It is a complicated and lengthy process, and considerable fees for these middlemen apply.

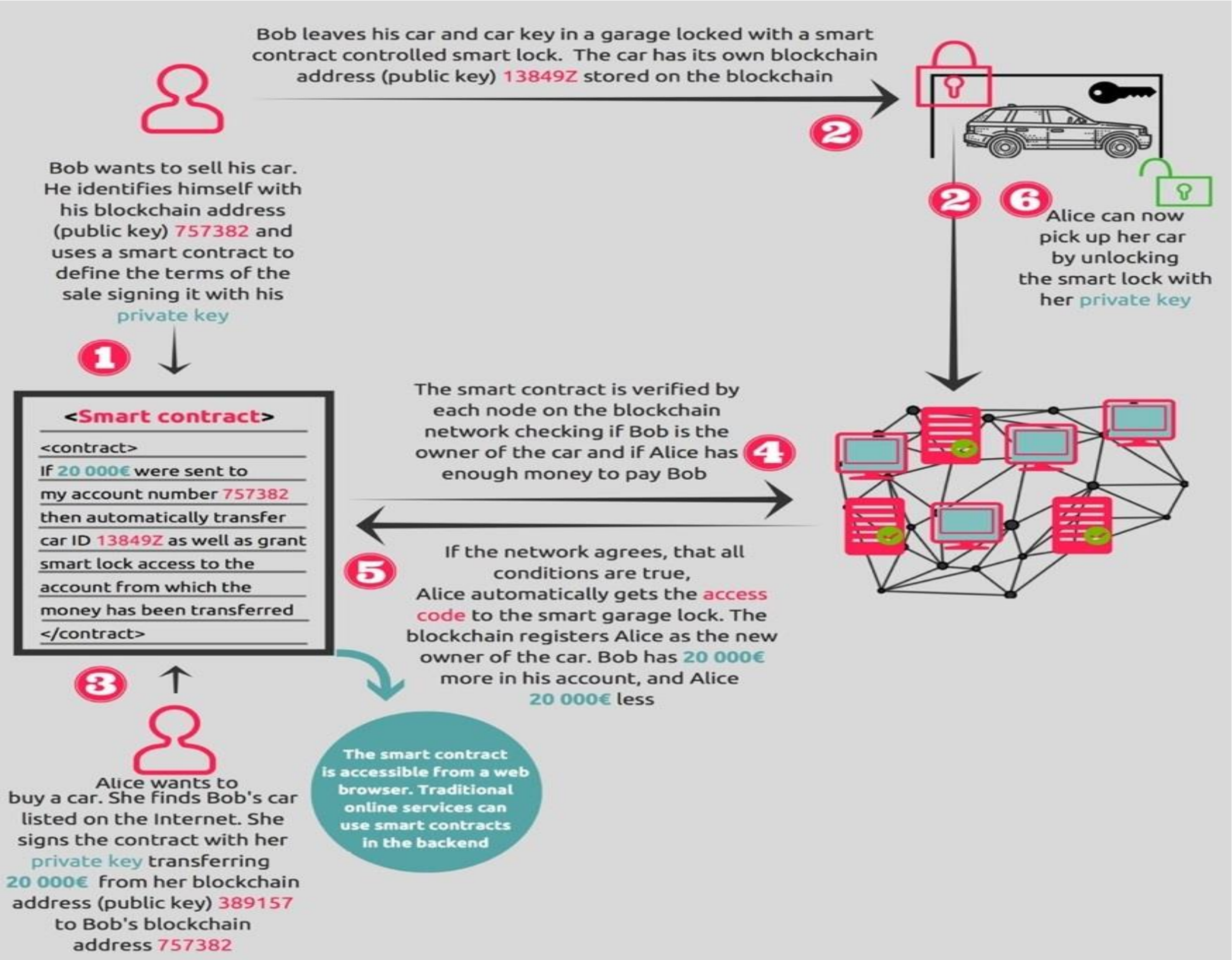


Alice agrees to pay 20 000€ for the car. Once Bob gets the deposit he will transfer the vehicle ownership to Alice by handing her over the car documents and car keys.

A trusted third party is required for verification. In order to officially transfer the ownership of the car, the terms of the contract have to be met. The process differs from country to country but always involves one or more trusted third parties: motor vehicle registration authority, in combination with a notary and/or insurance company. It's a complicated and lengthy process. Middlemen fees apply

Fig 10:– Traditional contract

On the Blockchain, once all involved authorities and companies are on a blockchain, a smart contract could be used to define all the rules of a valid care sale. If Alice wanted to buy the car from Bob using a smart contract on the blockchain, the transaction would be verified by each node in the Blockchain Network to see if Bob is the owner of the car and if Alice has enough money to pay Bob.



Bob wants to sell his car. He identifies himself with his blockchain address (public key) 757382 and uses a smart contract to define the terms of the sale signing it with his private key

Bob leaves his car and car key in a garage locked with a smart contract controlled smart lock. The car has its own blockchain address (public key) 13849Z stored on the blockchain

Alice can now pick up her car by unlocking the smart lock with her private key

```
<Smart contract>
<contract>
If 20 000€ were sent to
my account number 757382
then automatically transfer
car ID 13849Z as well as grant
smart lock access to the
account from which the
money has been transferred
</contract>
```

The smart contract is verified by each node on the blockchain network checking if Bob is the owner of the car and if Alice has enough money to pay Bob

If the network agrees, that all conditions are true, Alice automatically gets the access code to the smart garage lock. The blockchain registers Alice as the new owner of the car. Bob has 20 000€ more in his account, and Alice 20 000€ less

Alice wants to buy a car. She finds Bob's car listed on the Internet. She signs the contract with her private key transferring 20 000€ from her blockchain address (public key) 389157 to Bob's blockchain address 757382

The smart contract is accessible from a web browser. Traditional online services can use smart contracts in the backend

Fig 11: Smart contract

If the network agrees that both conditions are true, Alice automatically gets the access code to the smart lock for the garage. The blockchain registers Alice as the new owner of the car. Bob has € 20,000 more on his account, and Alice € 20,000 less. No middlemen required.

On the Blockchain, who owns what is transparent and at the same time anonymous or pseudonymous. This means that every computer running the blockchain protocol could check whether a certain person is the rightful owner of the car or not.

Why should we used smart contract?

Smart contracts are stored on a Blockchain, they inherit some interesting properties.

- Immutable:

  Being immutable means that once a smart contract is created, it can never be changed again. So no one can go behind your back and tamper with the code of your contract.

- Distributed

  Being distributed means that the output of your contract is validated by everyone on the network. So single person cannot force the contract to release the founds because of other people on the network will spot this attempt and mark it as invalid. Tampering with smart contracts becomes almost impossible.

4.3. Benefit Of Smart Contracts

The benefits of smart contracts go hand-in-hand with blockchain.

- **Speed and accuracy**: Smart contracts are digital and automated, so you won’t have to spend time processing paperwork or reconciling and correcting the errors that are often written into documents that have been filled manually. Computer code is also more exact than the legalese that traditional contracts are written in.
- **Trust**: Smart contracts automatically execute transactions following predetermined rules, and the encrypted records of those transactions are shared across participants. Thus, nobody has to question whether information has been altered for personal benefit.
- **Security**: Blockchain transaction records are encrypted, and that makes them very hard to hack. Because each individual record is connected to previous and subsequent records on a distributed ledger, the whole chain would need to be altered to change a single record.
- **Savings**: Smart contracts remove the need for intermediaries because participants can trust the visible data and the technology to properly execute the transaction. There is no need for an extra person to validate and verify the terms of an agreement because it is built into the code.

4.4.Language Require for Smart Contracts

Here are several kinds of Blockchain platforms on which to deploy smart.

Ethereum, Ripple, Codius, Mastercoin, Hyper ledger fabric.

- **Solidity**

Solidity is a contract-oriented, high-level language for implementing smart contracts. It was influenced by C++, Python and JavaScript and is designed to target the Ethereum Virtual Machine (EVM).

Solidity is statically typed, supports inheritance, libraries and complex user-defined types among other features. Solidity is the primary kind of smart contract language on Ethereum**.**

4.5 E-VOTING SMART CONTRACT

In this section we will develop a smart contract for e-voting which is written in solidity programming language.

Our contract will include:

State Variables—variables that hold values that are permanently stored on the Blockchain. We will use state variables to hold number of Voters and Candidates and owner of the contract i.e. administrator.

Function -- Functions are the executable of smart contracts. They are what we will call to interact with the Blockchain, and they have different levels of visibility, internally and externally. Keep in mind that whenever you want to change the value/state of a variable, a transaction must occur—costing Ether. You can also make calls to the Blockchain, which won't cost any Ether because the changes you made will be destroyed.

Event-- Whenever an event is called, the value passed into the event will be logged in the transaction's log. This allows Javascript call-back functions or resolved promises to view the certain value you wanted to pass back after a transaction. This is because every time you make a transaction, a transaction log will be returned. We will use an event to log the ID of the newly created Candidate, which we'll display.

Struct type -- This is very similar to a C struct. Structs allow you to hold multiple variables, and are awesome for things with multiple attributes. Candidate will only have their name and vote count, but you can definitely add more attributes to them.

Mappings—think of these like hash-maps or dictionaries, where it has a key-value pair. We will use two mappings.

There are a couple more types that aren't listed here, but some of them are a little more complicated.

Basically, we have two Structs (types that hold multiple variables) that describe a Voter and a Candidate.To keep track of Voters, we put them into mappings where they are integer indexed.

Write the solidity contract code by .sol extension .Our contract is election.sol. After compilation contract two files created.

1. Election.bin: This is the bytecode you get when the source code in Election.sol is compiled. This is the code which will be deployed to the blockchain.

2. Election.abi: This is an interface or template of the contract (called abi) which tells the contract user what methods are available in the contract. Whenever you have to interact with the contract in the future, you will need this abi definition. You can read more details about ABI

Web3js is a library which lets you interact with the blockchain through RPC. We will use that library to deploy our application and interact with it.

CHAPTER 5 DESIGN METHODOLOGY OF PROPOESD SYSTEM

Chapter gist: In this chapter we give proposed scheme of our voting system and all necessary information regarding our thesis design and implementation. Chapter contains mainly literature survey, proposed system architecture and implementation.

Content list:

5.1. OVERVIEW

In today’s world, widespread mistrust towards the government and interference in countries’ processes by external actors have made the democratic process of voting more critical than ever. The voting system in India as well as in some countries abroad is flawed and can be easily manipulated and hampered by those with power to suit their personal benefits. It allows people with money to buy the votes or tamper the machine that record it.

The pitfalls of the current system of ballot voting are being taken advantage of by people or organizations looking to gain power. In the African countries of Uganda and Kenya there has been widespread controversy over their elections in recent years. The election of 1946 in

Romania was heavily rigged. The communists took over Romania and abolished the multi-party system to gain complete control of the country [28].

These instances of controversial elections could all have been avoided if the counting process was fair, transparent and verifiable. The current ballot system does offer anonymity to the voter but the counting process is not transparent. People are supposed to trust the result which is provided by an Election commission or a government body. This makes the process of counting, a major vulnerability in the current process. There are also other major electoral scams such as voter fraud, ballot stuffing and booth capturing. All these make it very difficult for organizers of an election to distinguish between the actual votes and votes added without authorization.

5.2. LITERATURE REVIEW

5.2.1. RELATED WORK

Electronic voting systems have been being used in many countries. Estonia became the first nation to hold legally binding an electronic voting system for its national elections [**28]**.

Springall, Drew, et al. [**29]** proposed a security analysis of **EstoniaI-Voting system.** In Estonia I-voting, citizens were able to cast their vote using only the Internet and an electronic national identification card. The ID card used in the elections was designed to run on an integrated circuit, a Java chip platform, and also protected with 2048 bit PIN [30]. The card is able to create signatures using SHA1/SHA2 [30]. The card is easily usable for authentication, encryption, and signatures. The voter has to download the voting application, authenticate using the electronic ID, and if the voter is eligible to vote a list of candidates will be displayed and a vote could be cast. The vote will be encrypted using the election’s public key and signed with the voter private key. As soon as the vote is cast it will be a vote storage server controlled by the Estonian government [31]. Voters could vote multiple times, and only the last vote will be considered valid. This is done to prevent vote buying.

Though being considerably successful nearly 30% penetration rate during recent elections, the Estonian model has some drawbacks, too. By its nature, this is a centralized solution, creates a single-point-of-failure and is open to hacking/hijacking attempts. In example, Distributed Denial of Service (DDoS) attacks can harm the software, servers or databases used. The administrators of such a system may act malicious and steal, if cannot manipulate, some valuable information during an election. The scalability of this system is another question.

Gebhardt Stenerud et al [32] In 2011 Norway used an electronic remote voting system for the country council election. Norwegian electronic voting system is fairly simple from the voter perspective. The voter authentication done by their MinID. After verification voter can go to voting interface, interface showing the ballot. The voter makes her selections and submits them to a Java applet. The applet encrypted and digitally signs the vote and then sends it to the central voting servers.

In 2014 Norwegian has discontinued I-Voting project due to securities issues [33]. Critics of the Norwegian I-Voting system faced were the fear of votes going to public in case of a cyber-attack. New

In the South Wales State election [34] eligible citizens placed their vote using iVote system in 2017 about 280,000 eligible voters. There is a four step to casting a vote and 2 steps out of four are optional. The Following step is…

1)The voter has to register ,received a 8 digit voter card ID and choose 6 digit PIN.

2) Using voter ID and PIN voter can logged in to the server and cast a vote and received a 12-digit receipt number as a separate verification service.

3) This step is optional. The voter enters his ID, PIN, and receipt number to verify that his vote.

4) This step is optional. After the election is over, the voter is still can use his 12-digit receipt to check if his vote was included in the final count. If the vote was not counted a reason will be displayed.

In the iVote online voting system that would have allowed a malicious attacker to expose voters' secret ballots, substitute replacement votes, and sidestep the verification mechanism. This is serious flaws in I-Vote system.

Wolchok[34], Scott, et al. in 2010, Washington, D.C. developed an Internet voting pilot project that was allow overseas voters to cast their ballots using a website. The D.C. incursion illustrates how Internet voting can be attacked from anywhere. Most complex software systems have an abundance of vulnerabilities, with attackers needing to exploit just one. Many critical issues were found; therefore the project was cancelled and never used in any official elections.

5.2.2.SECURITYISSUESANDDRAWBACKS

Both Estonia and Norwegian electronic voting system main problem is the secrecy of the critical parts of the code. That's why the Estonia I-Voting system is terminated what raise question about transparency. An open source e-voting system is a must for a trusted election.

Intelligence Agencies has access to a wide range of network traffic and enough computing power to analyze voting data for a potential alteration. Even with enhanced security, State level attacks are possible in all previously motioned systems.

The centralization is one biggest issue of the I-Voting system makes it vulnerable to DDOS attacks what could make the elections inaccessible to voters.

5.3. PROPOSED SYSTEM

5.3.1. INTRODUCTION

The system we are going to propose in this thesis will address all these security concerns by using open source code to develop our electronic voting system, and rely on Blockchain technology to secure votes, and decentralize the system.

Our e-Voting solution will include four main requirements that can be illustrated as shown below:

- **Voter eligibility:**Only eligible voters are allowed to vote. Our system will not support a registration process. Registration usually requires verification of certain information and documents to comply with current laws, which could not be done online in a secure manner. Therefore, the system should be able to verify voters' identities against a previously verified database, and then let them vote only once.
- **Anonymity:**The voter has to remain anonymous during and after the election.
  The electronic voting system should not allow any links between voters' identities and ballots.
- **One man, one vote:**Every voter votes once and the voting system must be able to reconcile the total number of votes to the total number of voters.
- **Transparency:**The vote counting process is fixed, rules are well established, known to voters and withstands public scrutiny.

- **Accuracy**: Every vote is accurate and vote counting is consistent. Every vote should be counted, and can’t be changed, duplicated or removed. Vote counts are audit-able.

5.3.2.Representation Of The Electronic Voting system

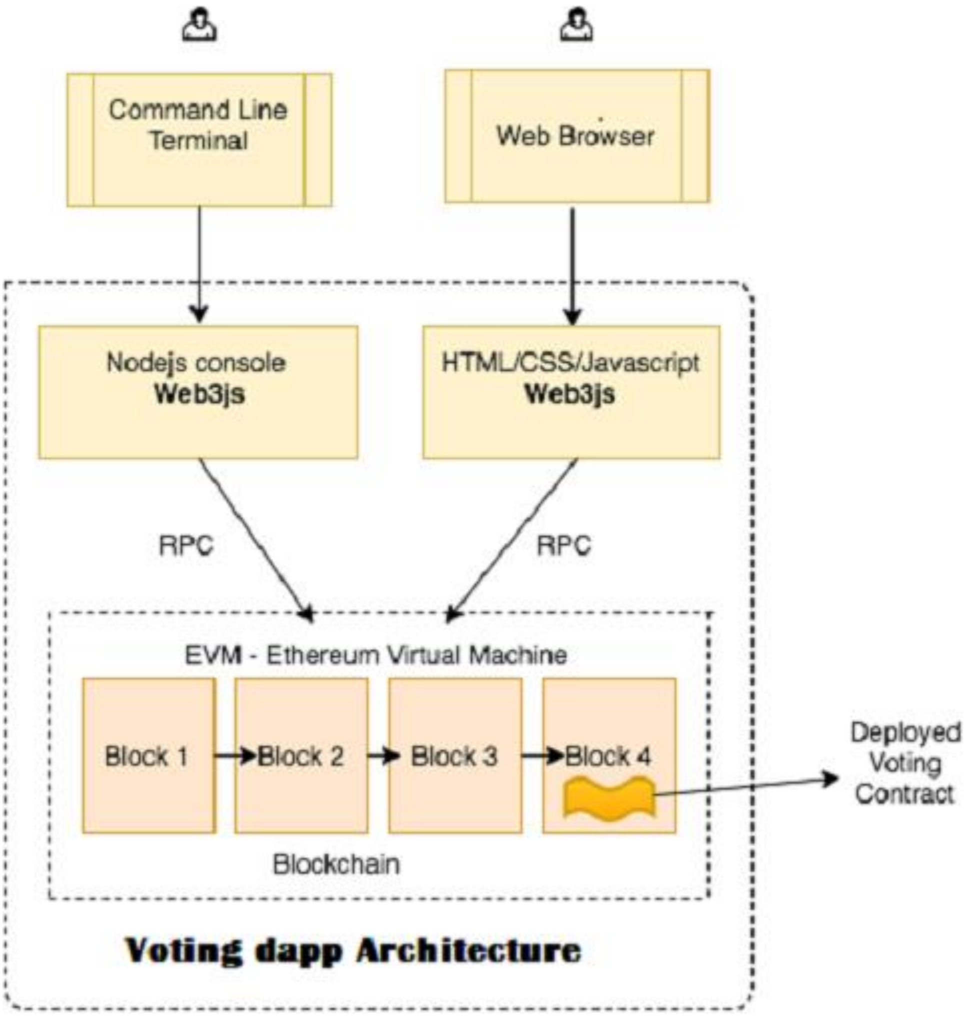


Fig 12: Architecture of proposed system

Above the figure show the basic architecture of our proposed model.

We already known that what is Blockchain and how it’s works. Now we are discuss our proposed voting scheme works in the EthereumBlockchain. The administrator creates a voting contract that deploy into the Blockchain as a first transaction added to new the block. Any data change in the deployed contract creates a transaction added a new block into the Blockchain. By casting votes as transactions, we can create a Blockchain which keeps track of the tallies of the vote.

We used platform as an existing EthereumBlockchain.Blockchain is an ordered data structure that contains blocks of transactions. Each block in the chain is linked to the previous block in the chain. The first block in the chain is referred to as the foundation of the stack called genesis block. Each new block created gets layered on top of the previous block to form a stack called a Blockchain.

In deployment process we first compile contract by solidity compiler. After compilation contract byte code and Application Binary Interface (ABI) generate. Contract byte code deploy into ethereum blockchain.

TERMONOLOGY

- **Blockchain**: Distributed database that stores the code of the Smart contract and the interaction between the users and the application in the form of transactions.
- **Smart contract**: A piece of code that is executed on the Ethereum Blockchain.
- **Ether (ETH):** The cryptocurrency used in Ethereum.
- **Gas**: The amount of ether that you have to pay to have your transactions accepted in the Ethereum Blockchain.

5.3.3. ELECTION STAGES

Stockholders involved in the Blockchain voting platform:

- **Voter** – Who will vote for the candidate standing in the elections based on their capabilities.
- **Candidate** – A person who participates in the election to represent to a specific political party.
- **Election Administration** – Who manage the entire process of election.

The Blockchain voting system could have the following steps:

- **Step 1: Candidate's Registration**
  Candidate can register on the Blockchain enable platform to receive the votes during elections. For registration, candidates need to submit personal identifiable information that would be stored on the public Blockchain.

Candidates provide a valid ID to Election Authority. All candidate unique names are publicly listed in the blockchain using the function “addCandidate”.

- **Step 2: Voter’s Registration**

  To register on the platform, a voter would need to submit personal identifiable Information and proof of their citizenship. And all the user’s information stored in the IPFS [35]. The information submitted by the voter would be verified through existing Government-approved systems. If the documents are checked and found to be correct, Then they are allowed to create their account on the system.

  Voters show a valid ID and their ethereum address to election authority. All voters ethereum address are publicly listed in the Blockchain by election authority using the function “Authorize” voters.

- **Step 3: Voting on Blockchain**

  On the Election Day, participating candidates can receive the votes in their account on the Blockchain voting platform. Eligible voter can cast the vote to their choice of candidate.

  Voters vote for their preferred candidate's name publicly listed in the Blockchain using the function "vote".

  Step 4: Verifying the vote

  On the Blockchain system, the voter could enter the public key and verify whether their vote was counted.

- **Step 5: Counting the vote**

  Voting could be transparent with the Blockchain system as it makes easy to follow and evaluate votes in the real-time. Each voter can vote once for the candidate of their choice. And, the candidate who has the highest number of votes wins the election.

- **Step 6: Election’s results**

  Using the Blockchain voting platform, the election’s result is instant. The predefined rules built in the smart contracts notify the stakeholder that voting has been closed along with election result.

CHAPTER 6 RESULT ANDDISCUSSION

Chapter gist: In this chapter we give some screen shoot of our e-voting system that shows the result. Also we analyse and discuss each and every stages of election how smart contract implement.

Chapter Content:

6.1. REQUIREMENT

Here's how I implemented smart contract for voting.Metamask is an Ethereum web browser extension that acts as an Ethereum wallet and an interface for Ethereum-based dApps. MetaMask allows users to sign smart contracts and interface with Ethereum dApps (distributed Ethereum-based applications) without running a full Ethereum node.In other words, MetaMask allows users to store Ethereum related data like public addresses and private keys like any other Ethereum wallet, *and* it allows users to interact with websites running Ethereum-based apps and smart contracts (essentially turning your browser into an Ethereum browser).

REQUIREMENT

- All actors of the election should be able to have an Ethereum account.
- All actors of the election should be able to have a little amount of ether to pay for the gas needed to execute the Smart contract.
- All actors of the election should be able to secure their voting computer.

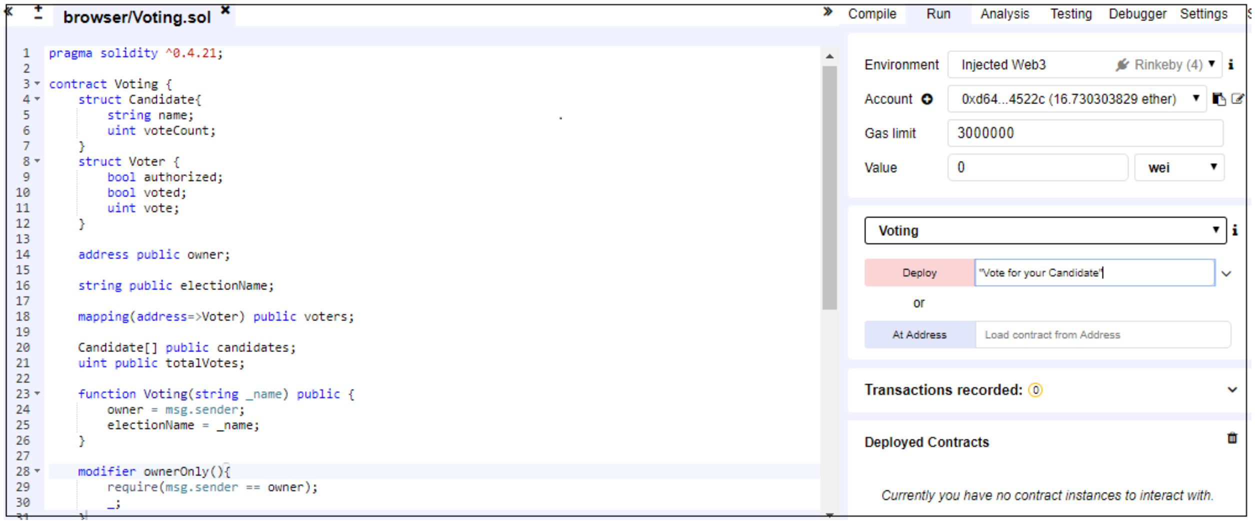
We demonstrate, create 4 accounts in Metamask to represent the following participants:

1. Administrator – manage the all voting process
2. Anurag – Voter -Public key-0x1128EaDFEEBC54f16f1EeA25227c976bB8220572
3. Bikash – Voter -Public key - 0xe2b6De6f1D8F9Ebb6AA7e9521Afca2dF465B83CA
4. Piyush – Voter-public key -0x3330774Cb282E6B8f0E2ac6D5705D1Ba73C08c28

We written our voting smart contract in remix solidity IDE. Remix is a web browser based IDE that allows you to write solidity smart contracts, then deploy and run the smart contract.

6.2. Result With each E-voting Stage

In this demonstration, the administration sets up a voting process which group of voters to vote, for example "Vote for your Candidate". Only Administrator can add candidate, authorize the voter who can vote their favourite candidate.



```
pragma solidity ^0.4.21;

contract Voting {
    struct Candidate{
        string name;
        uint voteCount;
    }
    struct Voter {
        bool authorized;
        bool voted;
        uint vote;
    }

    address public owner;

    string public electionName;

    mapping(address=>Voter) public voters;

    Candidate[] public candidates;
    uint public totalVotes;

    function Voting(string _name) public {
        owner = msg.sender;
        electionName = _name;
    }

    modifier ownerOnly(){
        require(msg.sender == owner);
        _;
```

Environment: Injected Web3 — Rinkeby (4)

Account: 0xd64...4522c (16.730303829 ether)

Gas limit: 3000000

Value: 0 wei

Voting

Deploy: "Vote for your Candidate"

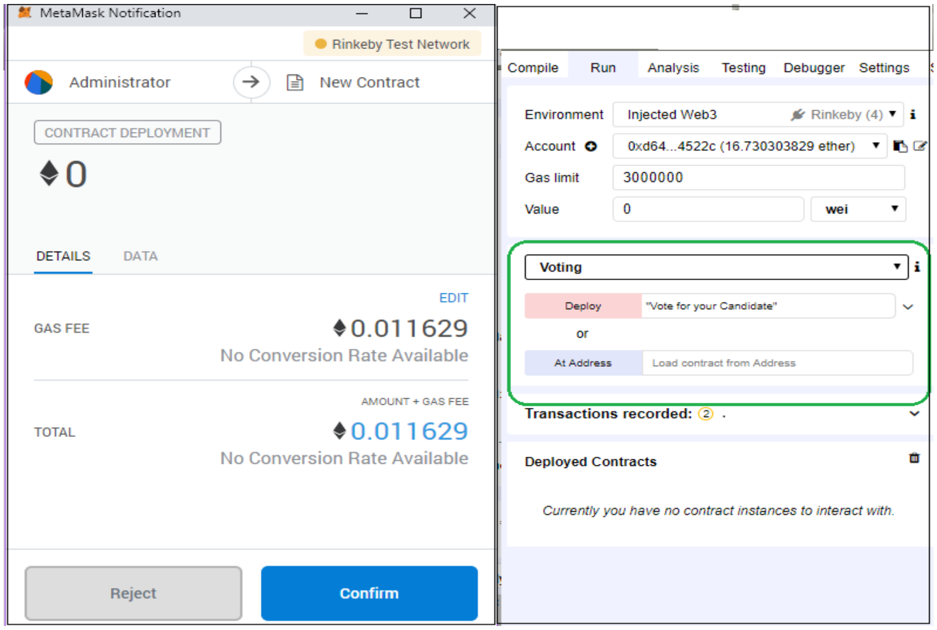
Transactions recorded: 0

Deployed Contracts

Currently you have no contract instances to interact with.

Fig 13- Voting contract deploy in remix solidity IDE

In Remix, switch to the Administrator's MetaMask wallet account to deploy the Voting contract. The Administrator compiles a list of wallet addresses of eligible votes. The decision about who is eligible to vote is an offline process beyond the scope of Blockchain.



MetaMask Notification — Rinkeby Test Network

Administrator → New Contract

CONTRACT DEPLOYMENT

0

GAS FEE: 0.011629 — No Conversion Rate Available

TOTAL (AMOUNT + GAS FEE): 0.011629 — No Conversion Rate Available

Environment: Injected Web3 — Rinkeby (4)

Account: 0xd64...4522c (16.730303829 ether)

Gas limit: 3000000

Value: 0 wei

Voting

Deploy: "Vote for your Candidate"

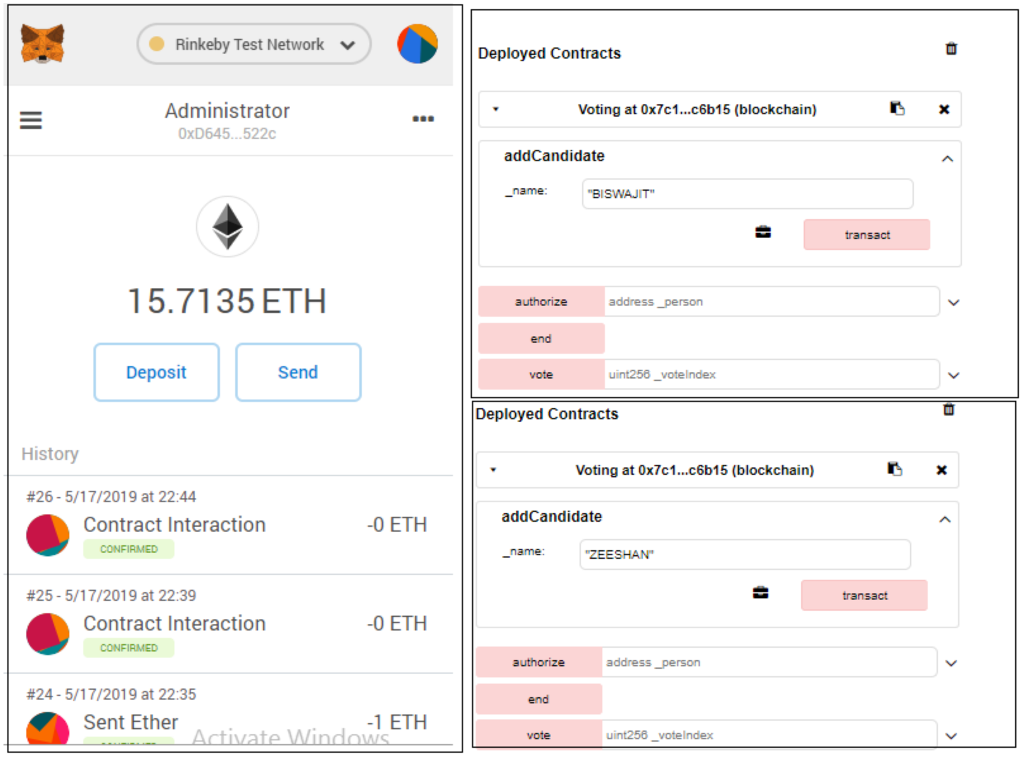
Transactions recorded: 2

Deployed Contracts

Currently you have no contract instances to interact with.

Fig 14– Deploy contract in Rinkeby Ethereum network

Candidate Registration



Rinkeby Test Network

Administrator

0xD645...522c

15.7135 ETH

History

#26 - 5/17/2019 at 22:44 — Contract Interaction — CONFIRMED — -0 ETH

#25 - 5/17/2019 at 22:39 — Contract Interaction — CONFIRMED — -0 ETH

#24 - 5/17/2019 at 22:35 — Sent Ether — -1 ETH

Deployed Contracts

Voting at 0x7c1...c6b15 (blockchain)

addCandidate — _name: "BISWAJIT"

Deployed Contracts

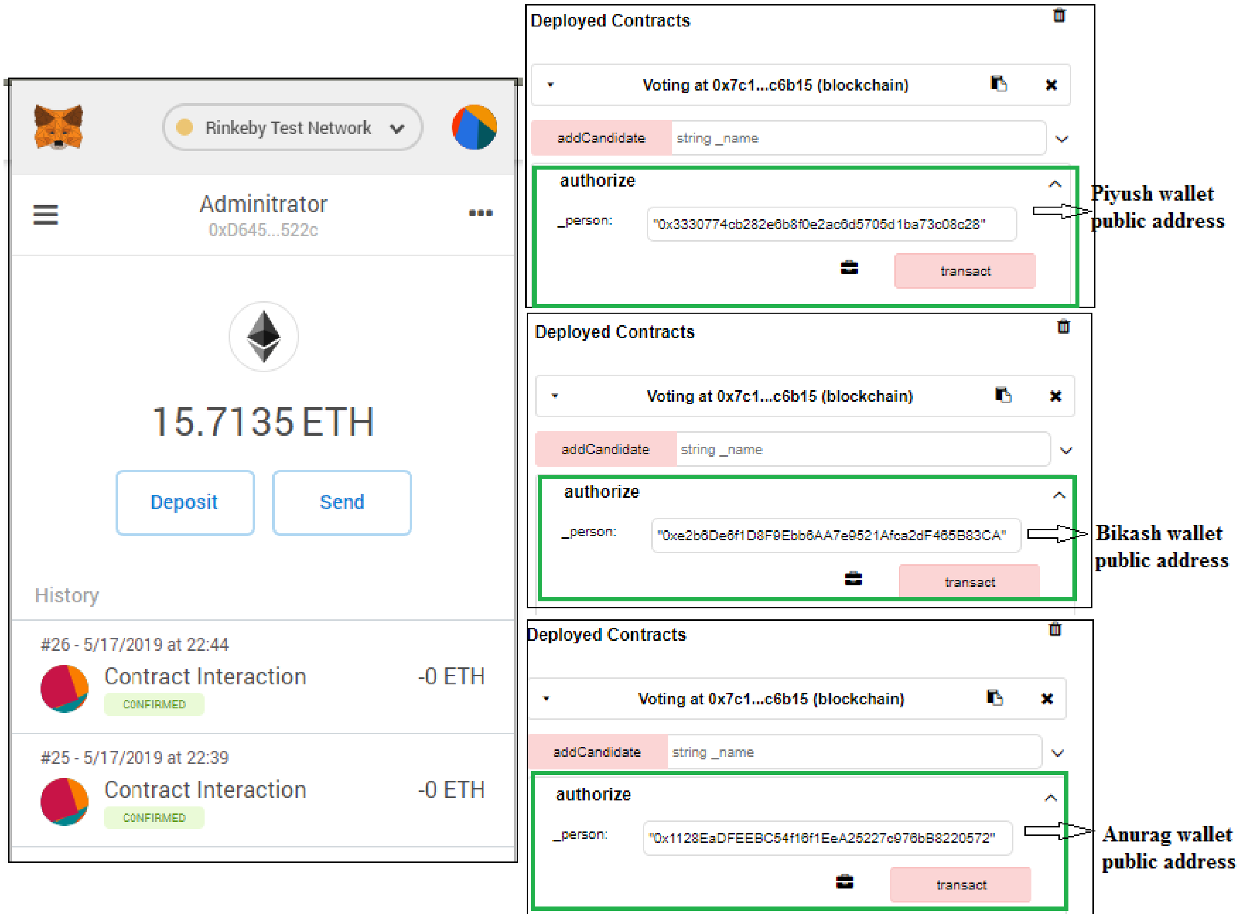
Voting at 0x7c1...c6b15 (blockchain)

addCandidate — _name: "ZEESHAN"

Fig 15:– Add candidate by contract owner i.e. Administrator

Voters Registration

In Remix, switch to the Administrator's wallet address. Then add the wallet addresses of Anurag, Bikah and Piyush using the authorize function.



#26 - 5/17/2019 at 22:44 Contract Interaction CONFIRMED -0 ETH

#25 - 5/17/2019 at 22:39 Contract Interaction CONFIRMED -0 ETH

_person: "0x3330774cb282e6b8f0e2ac6d5705d1ba73c08c28"

Piyush wallet public address

_person: "0xe2b6De6f1D8F9Ebb6AA7e9521Afca2dF465B83CA"

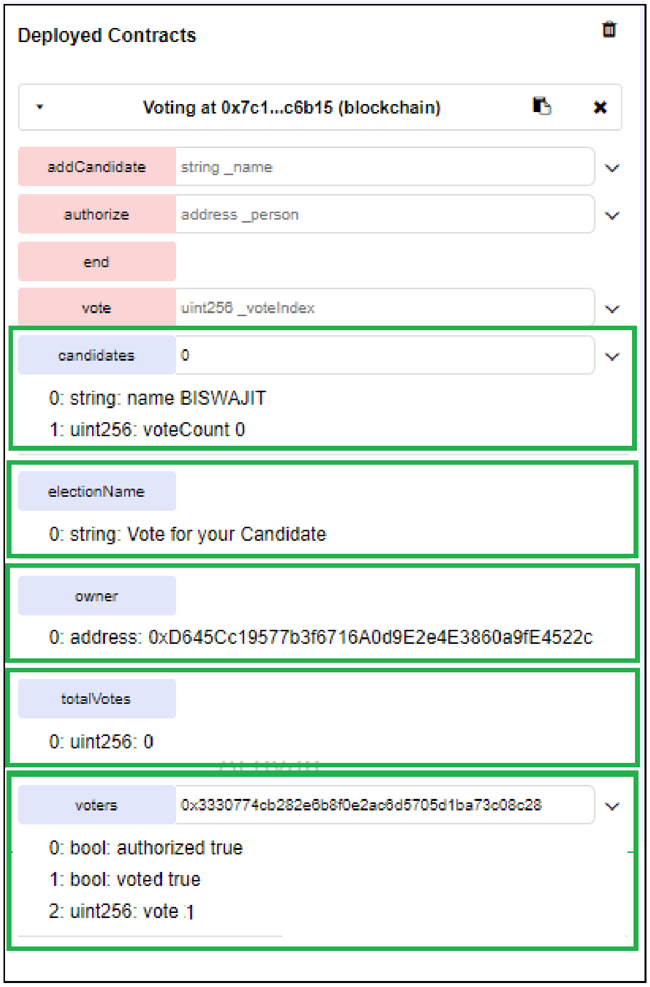
Bikash wallet public address

_person: "0x1128EaDFEEBC54f16f1EeA25227c976bB8220572"

Anurag wallet public address

Fig 16: Authorization of voters

At any time you can click on the respective buttons below to read the ballot contract's public variables which include information about who the Administrator address is, what the proposal says, the candidate and the status of votes and whether or not they have voted and authorized to vote. This ensures transparency.



```
Deployed Contracts

Voting at 0x7c1...c6b15 (blockchain)

addCandidate    string _name
authorize       address _person
end
vote            uint256 _voteIndex
candidates      0
    0: string: name BISWAJIT
    1: uint256: voteCount 0
electionName
    0: string: Vote for your Candidate
owner
    0: address: 0xD645Cc19577b3f6716A0d9E2e4E3860a9fE4522c
totalVotes
    0: uint256: 0
voters          0x3330774cb282e6b8f0e2ac6d5705d1ba73c08c28
    0: bool: authorized true
    1: bool: voted true
    2: uint256: vote 1
```

To find out candidate name and number vote received,enter its position(0 for first candidate, 1 for second candidate)

Anyone can read election name.

Anyone can read the total number of vote is given

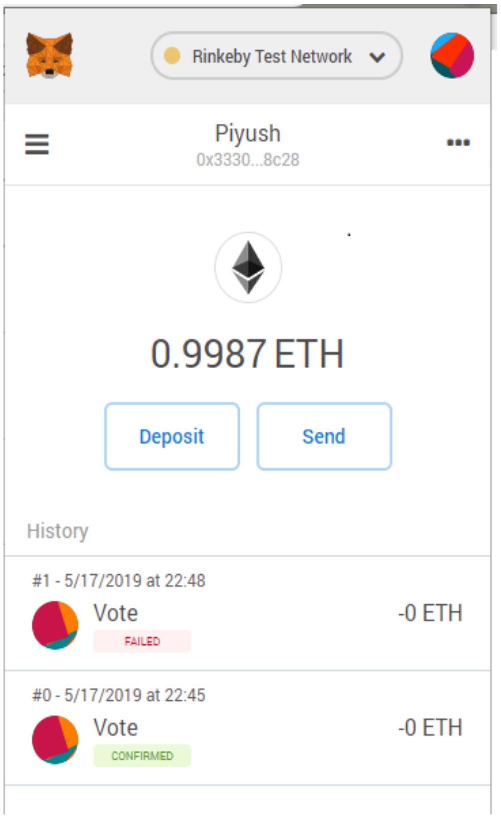
Enter any public wallet address to see the person is authorized to vote or not, and voted or not.

Fig 17: Function of smart contract

Voting

Piyush attempts to vote. The contract first compares Piyush's MetaMask wallet address to her record in the voters' array is authorized. If Piyush's contract address is not found in the array, she's deemed an ineligible voter and her vote will not be accepted. If Piysuh's wallet address is found in the voters' array, the contract checks the voters array to confirm if she has voted previously. If she has, the ballot contract will refuse to accept her vote. If she hasn't, her vote is accepted, and her status is updated to say that she has voted. Once she has voted, the contract will no longer allow Piyush to vote again.

Similarly Anurag and Bikash cases also same condition checked before attempt to vote.



Rinkeby Test Network

Piyush
0x3330...8c28

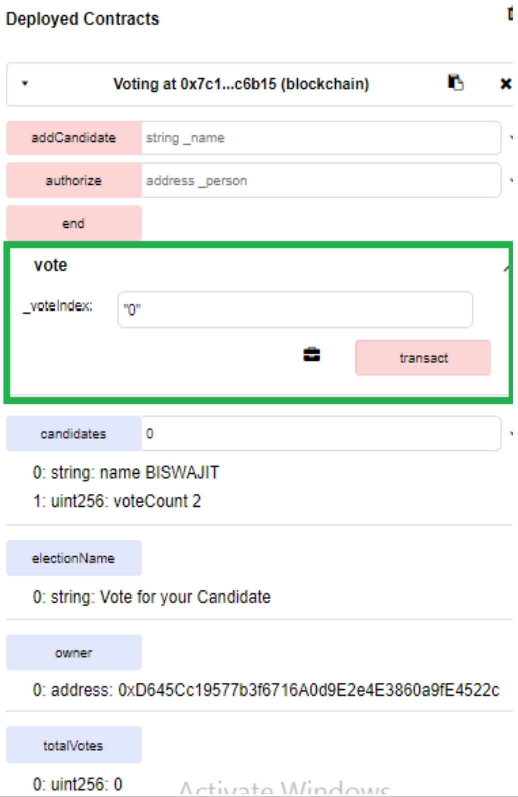
0.9987 ETH

Deposit Send

History

#1 - 5/17/2019 at 22:48
Vote -0 ETH
FAILED

#0 - 5/17/2019 at 22:45
Vote -0 ETH
CONFIRMED



Deployed Contracts

Voting at 0x7c1...c6b15 (blockchain)

addCandidate string _name

authorize address _person

end

vote

_voteIndex: "0"

transact

candidates 0

0: string: name BISWAJIT

1: uint256: voteCount 2

electionName

0: string: Vote for your Candidate

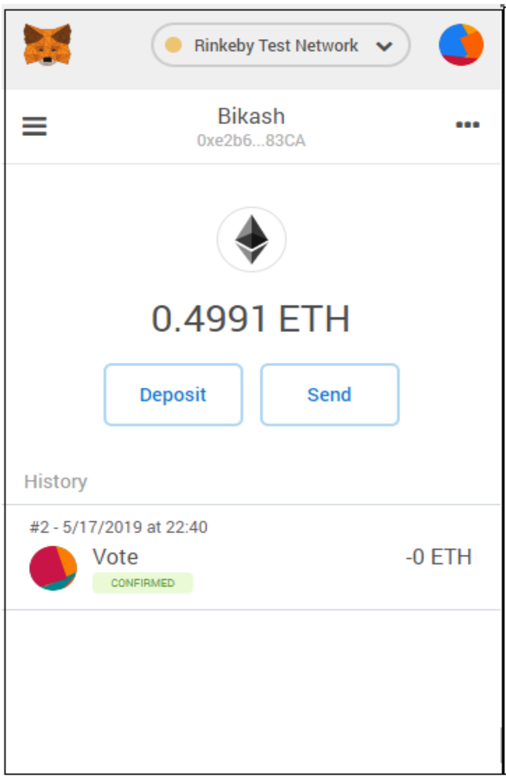
owner

0: address: 0xD645Cc19577b3f6716A0d9E2e4E3860a9fE4522c

totalVotes

0: uint256: 0

Activate Windows



Rinkeby Test Network

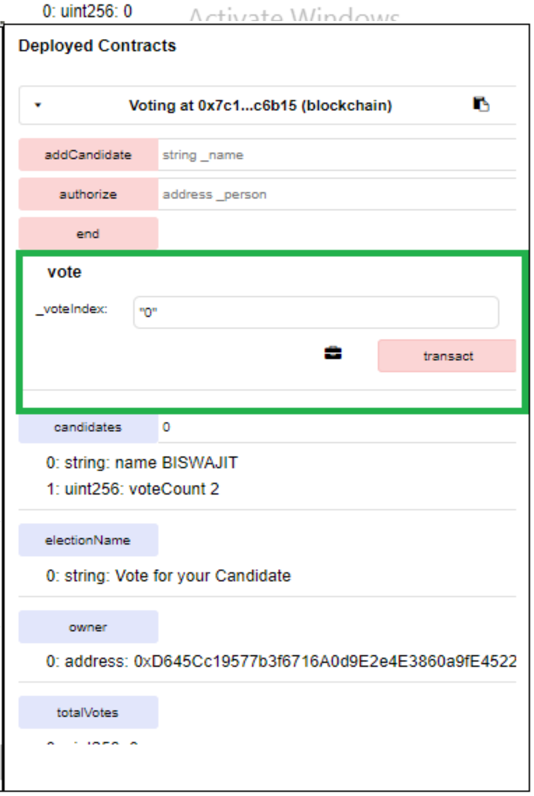
Bikash
0xe2b6...83CA

0.4991 ETH

Deposit Send

History

#2 - 5/17/2019 at 22:40
Vote -0 ETH
CONFIRMED



Deployed Contracts

Voting at 0x7c1...c6b15 (blockchain)

addCandidate string _name

authorize address _person

end

vote

_voteIndex: "0"

transact

candidates 0

0: string: name BISWAJIT

1: uint256: voteCount 2

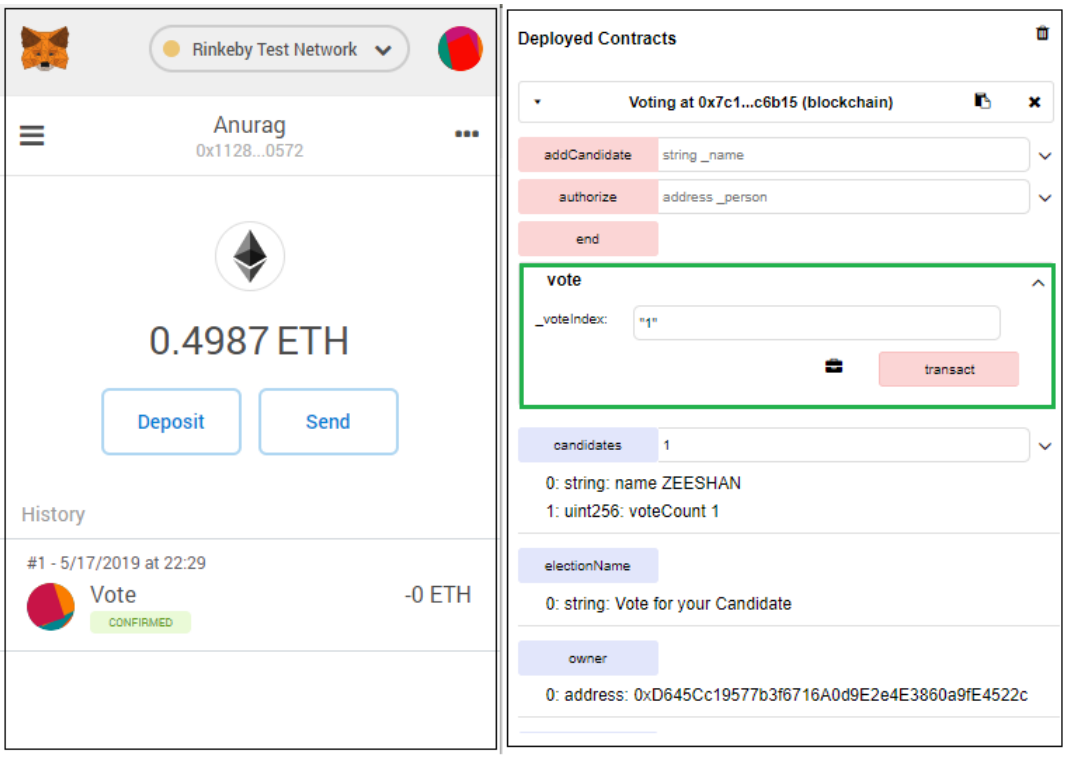
electionName

0: string: Vote for your Candidate

owner

0: address: 0xD645Cc19577b3f6716A0d9E2e4E3860a9fE4522

totalVotes



_voteIndex: "1"

candidates: 1

```
0: string: name ZEESHAN
1: uint256: voteCount 1
```

```
0: string: Vote for your Candidate
```

```
0: address: 0xD645Cc19577b3f6716A0d9E2e4E3860a9fE4522c
```

Fig 18: Vote

6.3. HOW DOES THE E-VOTING CONTRACT IMPLEMENT THE PRINCIPLE OF VOTING?

One man one vote:

The voters array stores a list of voters who have voted. It ensures that no one can vote the second time. Once a voter votes, his status changes to “voted” and the ballot contract checks to ensure that he does not vote again

Voter Eligibility

Voter eligibility is determined by assembling a voters’ array of wallet address before voting begins. You need to vote with your MetaMask Wallet whose address matches the one that the administrator before voting begins.

Wait a moment, doesn’t that mean that someone who have stolen my MetaMask wallet will be able to vote on my behalf? That’s correct, on a Blockchain, the only thing that separates you from an identity thief is your wallet’s private key.

But how does the administrator ensure that it’s really me who’s casting the vote? Facial recognition? That’s definitely a possibility—builds a ballot app to allow the wallet to be unlocked to vote only if you pass a fingerprint or facial recognition test will work.

Transparency

This is one of the things that Blockchain does extremely well. Every action taken and every record etched on the Blockchain is immutable. On a public Blockchain, collusion is close to impossible as described here.

Reliability

There is no single point of failure on a Blockchain as every node in the chain participates in keeping the Blockchain running. A smart contract, once deployed on the Blockchain is immutable. The business logic of the smart contract once deployed is cast in stone. There’s no way the administrator can change the rules, say, from a one man one vote to one man two vote once the contract is deployed.

CHAPTER 6 CONCLUSIONAND FUTURE WORK

The blockchain offers many potentially world-changing opportunities and has been one of the most hyped technologies of the last few years. However, the spotlight is often on applications its applications in the developed world. That spotlight is much less often on developing countries that could potentially benefit an incredible amount from Blockchain technology, and specifically blockchain voting.

Additionally, the blockchain is a trust-less system that proves the ownership of 47 assets without being corrupt, something incredibly valuable and useful in the developing world.

We have presented a smart contract implementation for the Open Vote Network that runs on Ethereum. Our implementation was tested on the official Ethereum test network with four simulated voters.To address voter tampering, blockchains generate cryptographically secure voting records. Votes are recorded accurately, permanently, securely, and transparently. So, no one can modify or manipulate votes. Furthermore, blockchains preserve participants' anonymity while still being open to public inspection. Although nothing is totally secure, tampering is nearly impossible with blockchains.

In future work, we will investigate the feasibility of running a large-scale election over the Blockchain.As far I study it is only possible, when its implementation will almost certainly require a dedicated Blockchain.This new blockchain can have a larger block size to store more transactions on-chain. We want to develop full fledge voting Decentralized Application (DApp) and explore many tools related this technology.

REFERENCES

[1] https://www.heritage.org/voterfraud

[2] Nakamoto, S., 2008. Bitcoin: A peer-to-peer electronic cash system.

[3] Merkle, R. C. (1980, April). Protocols for public key cryptosystems. In *1980 IEEE Symposium on Security and Privacy* (pp. 122-122). IEEE.

[4] Merkle, Ralph C. "A digital signature based on a conventional encryption function." *Conference on the theory and application of cryptographic techniques*. Springer, Berlin, Heidelberg, 1987.

[5] https://blog.ethereum.org/2015/08/07/on-public-and-private-blockchains/

[6] Wood, Gavin. "Ethereum: A secure decentralised generalised transaction ledger." *Ethereum project yellow paper* 151 (2014): 1-32.

[7] Buterin, Vitalik. "Ethereum white paper." *GitHub repository*(2013): 22-23.

[8] Contracts, BITCOIN FOUNDATION WIKI, https://en.bitcoin.it/wiki/Contracts.

[9] https://bitcoin.org/bitcoin.pdf

[10] Satoshi Nakomoto, Bitcoin Open Source Implementation of P2P Currency, P2P FOUNDATION (Feb. 11, 2009), http://p2pfoundation.ning.com/forum/topics/bitcoin-opensource

[11] Average Transaction Confirmation Time, BLOCKCHAIN.INFO, https://blockchain.info/charts/avgconfirmationtime?timespan=all&showDataPoints=false &daysAverageString=1&sho_header=true&scale=0&address= showing that the average confirmation time for a Bitcoin transaction has hovered around 7.5 minutes in 2014).

[12] As of the end of 2014, nearly 8 million Bitcoin accounts (wallets) have beencreated—a number that has been growing at a 200% yearly rate. See State of Bitcoin 2015, COINDESK at Slide 6, http://www.slideshare.net/CoinDesk/state-of-bitcoin-2015 . Moreover, over 82,000 merchants accept Bitcoin as a form of payment, including Microsoft, Dell, Time Magazine, Overstock, and Braintree, a division of Paypal. Id. At Slide 44

[13] Tom Simonite, Bitcoin Isn’t the Only Cryptocurrency in Town, MIT TECH REVIEW (Apil15,2013),http://www.technologyreview.com/news/513661/bitcoin-isnt-the onlycryptocurrency-in-town/; Ariel Schwartz, Bitcoin 2.0: Can Ripple MakeDigital Currency Mainstream? FAST CO. EXIST, (May 14, 2013),http://www.fastcoexist.com/1682032/bitcoin-20-can-ripple-make-digital-currency-mainstream

[14] Madise, Ü. Madise and T. Martens, “E-voting in Estonia 2005. The first practice of country-wide binding Internet voting in the world.”,Electronic voting, 2nd International Workshop, Bregenz, Austria,(2006) August 2-4

[15] Anjum, A., Sporny, M., & Sill, A. (2017). Blockchain standards for compliance and trust. *IEEE Cloud Computing*, *4*(4), 84-90

[16] Catalini, C., & Gans, J. S. (2016). *Some simple economics of the blockchain* (No. w22952). National Bureau of Economic Research

[17] https://www.capgemini.com/beyond-the-buzz/blockchain/#smart-contracts-from-hype-to-reality

[18] Anjum, A., Sporny, M., & Sill, A. (2017). Blockchain standards for compliance and trust. *IEEE Cloud Computing*, *4*(4), 84-90.

[19] https://bitsonline.com/blockchain-platform-mortgage-fraud/

[20] Madise, Ü. Madise and T. Martens, “E-voting in Estonia 2005. The first practice of country-wide binding Internet voting in the world.”,Electronic voting, 2nd International Workshop, Bregenz, Austria,(2006) August 2-4.

[21] J. Gerlach and U. Grasser, “Three Case Studies from Switzerland: E-voting”, Berkman Center Research Publication, (2009).

[22] I. S. G. Stenerud and C. Bull, “When reality comes knocking Norwegian experiences with verifiable electronic voting”, Electronic Voting. Vol. 205. (2012), pp. 21-33

[23] http://www.allerin.com/blog/distributed-cloud-storage-with-blockchain-technology

[24] See https://storj.io/

[25] Springall, Drew, et al. "Security analysis of the Estonian internet voting system." *Proceedings of the 2014 ACM SIGSAC Conference on Computer and Communications Security*. ACM, 2014.

[26] Rivera, R., Robledo, J. G., Larios, V. M., & Avalos, J. M. (2017, September). How digital identity on blockchain can contribute in a smart city environment. In *2017 International Smart Cities Conference (ISC2)* (pp. 1-4). IEEE.

[27] Kosba, A., Miller, A., Shi, E., Wen, Z., & Papamanthou, C. (2016, May). Hawk: The blockchain model of cryptography and privacy-preserving smart contracts. In *2016 IEEE symposium on security and privacy (SP)* (pp. 839-858). IEEE

[28] Alvarez, R. M., Hall, T. E., & Trechsel, A. H. (2009). Internet voting in comparative perspective: the case of Estonia. *PS: Political Science & Politics*, *42*(3), 497-505

[29] Springall, D., Finkenauer, T., Durumeric, Z., Kitcat, J., Hursti, H., MacAlpine, M., & Halderman, J. A. (2014, November). Security analysis of the Estonian internet voting system. In *Proceedings of the 2014 ACM SIGSAC Conference on Computer and Communications Security* (pp. 703-715). ACM.

[30] Trueb Baltic, “Estonian Electronic ID – Card Application Specification Prerequisites to the Smart Card Differentiation to previous Version of EstEID Card Application.” http://www.id.ee/public/TBSPEC-EstEID-Chip-App-v3_5-20140327.pdf

[31] Cybernetica.“InternetVotingSolution.”https://cyber.ee/uploads/2013/03/cyber_ivoting_ NEW2_A4 _web.pdf

[32] Gebhardt Stenerud, Ida Sofie, and Christian Bull. "When reality comes knocking norwegian experiences with verifiable electronic voting." *5th International Conference on Electronic Voting 2012 (EVOTE2012)*. Gesellschaft für Informatik eV, 2012.

[33] Ministry of Local Government and Modernisation. “Internet Voting Pilot to be Discontinued.”https://www.regjeringen.no/en/aktuelt/Internet-voting-pilot-to-be-discontinued/id764300/

[34] Wolchok, S., Wustrow, E., Isabel, D., & Halderman, J. A. (2012, February). Attacking the Washington, DC Internet voting system. In *International Conference on Financial Cryptography and Data Security* (pp. 114-128). Springer, Berlin, Heidelberg