EAI Endorsed Transactions
on Smart Cities
Research Article EAI.EU

Implementation of Decentralized Blockchain E-voting

Saad Moin Khan[1], Aansa Arshad[1], Gazala Mushtaq[1,*], Aqeel Khalique[1] and Tarek Husein[1]

[1]Department of Computer Science & Engineering, SEST, Jamia Hamdard, New Delhi, India

Abstract

E-voting reduced the cost of election and provided convenience to some extent as compared to the traditional approach of pen and paper but it was considered to be unreliable as anyone having access to the machine physically can obstruct the machine and alter the votes. Also in order to control the entire procedure from electronic voting to electoral results and tracking the outcomes, a central system is required. Voters are not completely secure as vote can be targeted easily. It also possesses a great threat to the right to vote and transparency. This paper provides a solution for removing inconveniences from conventional elections using blockchain that has emerged as an exciting technology for various application due to its unique characteristics that outperform other technologies. The goal of this research is to establish a system for e-voting that is decentralized rather than centralized by using blockchain technology that guarantees protection to electorate's identity, data transfer privacy and verifiability by an open and transparent voting process.

Keywords: E-Voting, Blockchain, Smart Contracts, Ethereum Cryptocurrency.

Received on 29 January 2020, accepted on 21 May 2020, published on 03 June 2020

doi: 10.4108/eai.13-7-2018.164859

*Corresponding author. Email: gazalamushtaq188@gmail.com

1. Introduction

An election is the procedure of espousing a candidate to hold a public office or an official position in order to establish a government through the voters. Elections are considered as one of the founding pillars in any democratic society where the citizens make a decision by voting for the competent candidate to form a healthy democracy. The history of election dates back to ancient Greece, Rome and across the medieval period to select the pope and the holy roman emperor. In India it dates back to the early Vedic period where the *'raja'* (king) was elected by the *'gana'* (people). The modern day elections emerged only after the 16th century across Europe and North America. Modern approach of voting system or EVM replaced the traditional method of voting, which was a monotonous process, demanding arduous and taxing efforts, resulting an ample scope of error and miscalculations. With the techno-advancement, the mechanical system of voting proved far more fluent, serviceable and reduced the human effort, thereby increasing the reliability and accuracy. The proposed system employs the technologies like e-voting, blockchain and smart contract to provide more security and convenience.

E-voting:

E-voting refers to the process of casting and compiling votes using an electronic system. Votes are stored in tape cartridges, diskette, smart cards and sent to a centralized location for compilation process. The various forms of e-voting are DER (direct electronic recording) touch screens, optical scanners. The two main types of e-voting are:

On-site e-voting where electronic voting machines are placed / present in the polling boots with some government official who will supervise the voting process and people have to be in queue for casting the vote.

Remote e-voting; where people need not be present at the polling station instead can cast their vote from any remote location using computers, mobile phones, etc. through internet, sms, or kiosks.

The security community found electronic voting machines inaccurate and untrustworthy based on security issues. The software can be undermined when the device is physically reached which affects the votes on the machine. Elections need to be secure and unimpeachable irrespective of the

organization. People's privacy and voting protection must be secured but it should not take too long for votes to be counted, as it upraises concerns.

Blockchain:

Blockchain proved to be a substitute for the conventional approach by making system unalterable and transparent. Blockchain is an organized data structured that includes blocks where each block is connected to every other block through a chain. The first block is called as genesis block. Each new block will be stacked to form a stack called a blockchain. Each block consists of data, hash and hash of previous block. If any change is being made to the data available in a particular block, consequently the hash of the block also gets changed but the next block will have the same unchanged hash of the previous block which invalidates this block and all other succeeding blocks. This is to avoid tempering because making change in one block you will need to calculate hash for every other following block however hackers now a days can compute hundreds of thousands of hashes in a matter of seconds. In order to avoid this problem it makes use of proof-of-work concept that delays the pace of forming a new block. Moreover it make use of a distributed peer to peer network where no central entity is present. Whenever a new block gets created it is sent to all other nodes present on this network where each node makes sure that no tempering is done by verifying the block after which the new block is added to every other node's blockchain. Every node on the network agrees on whether the block is valid or not by creating a consensus which makes blockchain so secure, safe and reliable.

Smart contract:

A smart contract is a self-imposed contract that is embedded in a blockchain managed computer code. This code includes a set of rules governing the communication and decision on the contract between the parties, the contract will be enforced automatically once the already defined rules are met. Smart contract gives a framework for efficient control between two or more parties of tokenizes assets and access rights [1]. Fig 1 [1]; shows the working principle of smart contract. Blockchain is just a database that cannot be altered, without smart contract, which expands and leverages blockchain

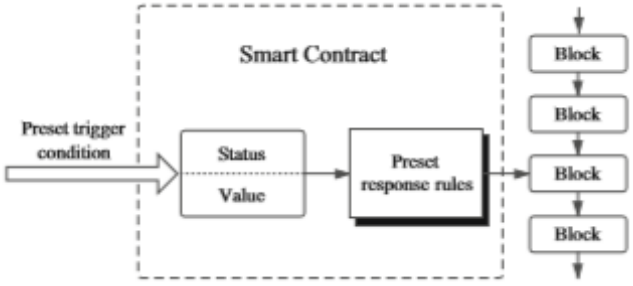

Figure 1. Smart Contract Working Principle.

There are various aspects of smart contract including technical aspect, legal aspect, economic aspect, that can be seen from the Figure 2 [2].

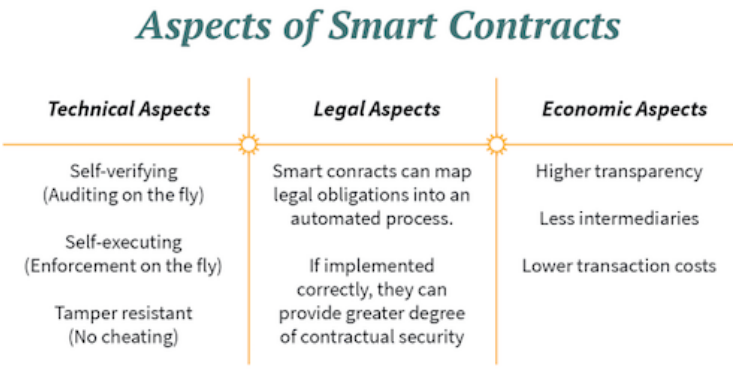

Figure 2. Various Aspects of Smart Contract.

Self-verification of the conditions in a smart contract is done by data interpretation. Each network node will guarantee the proper execution of a single contract, which relief the contract creators from tracking the execution of the contract. Smart contracts are self-executing, where the conditions of the agreement between different parties are written into the code. This means that legal obligations can be mapped using smart contracts into automated process. The execution of the contract can be automatically invoked by a trigger like expiration date. In this paper, we implement blockchain based e-voting system which overcome the problems encountered in e-voting and builds trust among voters for legitimate voting. Moreover, it will also be a helpful step towards the development of smart governance. This paper contains various sections; section 2 presents recent related work in block chain technology and e-voting system, section 3 presents proposed e-voting system based on blockchain, section 4 gives implementation details and results. Finally, section 5 concludes our work

2. Related Work

E-voting system was a great advancement over traditional pen and paper approach and became very popular in many countries. Several countries introduced e-voting system in their election process. Although it provided a number of advantages like increased voter turnout, auditability, low cost, convenient and accessible elections, etc. but there were several important challenges and issues associated with it.

Abdelwehab et al. [3] in their study discussed a number of challenges in e-voting system including legal challenges, social and cultural challenges, technical challenges, attacks, etc.

Diego F. Aranha et al. [4] in their work identified an experiment to test the validity of election results and to enhance transparency and voter participation within electronic elections. This proposal was based upon two aspects i.e. distributed collection of pole tape, made by mobile devices by voters and crowdsourcing election data

verification by electoral authority.

Kristian Gjosteen and Anders Smedstuen [5] believed that if voters make use of voting protocol correctly then there will be no chance of attack on results of elections and they give a statistical method to improve the security of e-voting.

Budurudhi et al. [6] explores how to properly develop voting machine interfaces to promote the role of electors and electoral administrators and then use such interfaces for complex elections. The problem of relying on a remote voting device is said to be firmly linked to the interface provided which in turn influences votes as verification of voting is an important issue. Much of the work in remote–electronic voting involves cryptography voting protocols design and verification to safeguard desired property.

Neumann et al. [7] states that in order to make specific recommendations on the type of voting system that is best suited to that particular context, they introduced a model for the comparison of voting schemes in any given electoral setting and the model was applied to the specific context of Estonian internet voting.

As there were various problems with electronic voting especially related to physical security, people began to look for solutions to the problems that's when blockchain came into the field of e-voting, initially blockchain was used for bitcoin. Ahmed Ben Ayed [8] in his work discusses how to take the advantages of blockchain technology in the process of e-voting to make it safe, secure, anonymous, etc.

Jen-Ho Hsiao et al. [9] in their work make use of smart contracts in decentralized blockchain technology for e-voting to engage all voters in evaluating and recording ballots. It increases the trust of electorate and decreases the misuse of election capital.

Jonathan Alexander et al. [10] in their study used NetVote for user interface of the program, it uses decentralized application. The dApp admin helps electoral administrators to decide electoral policy, generate voting, register rules, opening and closing of voting. Identification of voters is done by other applications like biometric readers. To test and check the results of election TallydApp is used. This NetVote reinforce three kinds of elections: private election, open election, Token holder elections.

There are many problems in current e-voting system which acts as a hindrance in accomplishing the accurate results like:

a. Prone to hacking.
b. Inefficient auditing
c. Misinterpretation of voter intent.
d. Political biasness on behalf of the manufacturer.
e. Tempering of software programs.
f. Inefficiency in securing the casted votes.
g. Hardware malfunctions, etc.

The main aim of this research is to develop an electronic voting system based on blockchain technology that meets the legislation's longstanding challenge. This model can be implemented at various levels including school, college, offices and even at national level voting. Because elections are always challenged with fraudulent practices like infiltrating machine, alter votes, organization of information campaigns and more. All of these problems will be addressed by our model.

3. Proposed System

The proposed system utilizes several tools namely ganache, truffle framework, npm and metamask. Truffle imports the smart contracts on the blockchain while as ganache operates the internal blockchain and it will be accessed by using metamask. With some Ether i.e. Ethereum's cryptocurrency is required by a user for an account with wallet address. To write the transaction to blockchain, user needs to pay a certain transaction fee which is called as gas. Once votes are cast the process is completed by a number of nodes on the network called as minners. These miners compete with each other to complete the transaction. The miners who succeed in this transaction is awarded ether paid by users to vote. Instead of node we will be using ganache software for mining purpose.

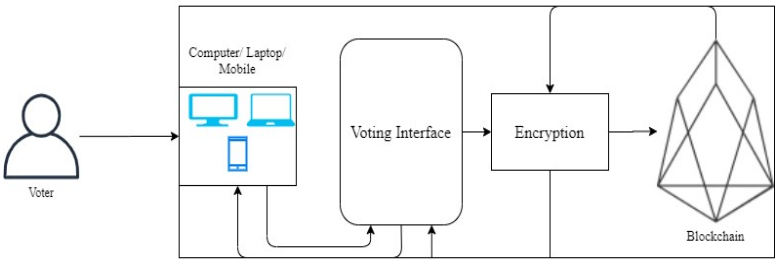

Figure 3. Proposed E-Voting System Based on Blockchain

Preliminaries:

Our proposed model can be implemented by using 64-bit hardware/ machine, windows 7 onwards, NMP dependencies, Truffle framework, Metamask, solidity toolkit and Ganache.

1. Dependency NPM(Node Package Manager)
2. Truffle framework
3. Ganache
4. Metamask
5. Coding language; solidity, HTML, JavaScript, CSS

NPM (Node Package Manager):
NPM is package manager that manages, installs, updates or uninstalls the node.js packages in an application. It is a command line based tool. It operates in two modes: local mode

and global mode. In global mode all node.js application are affected and in local mode only particular directory of an application gets affected [11].

Truffle framework:
Truffle is a powerful tool to work with ethereum smart contracts. It is used for compilation, deploying and linking of smart contracts, provides testing platform for automated contracts, manages networks and packages, etc. [12].

Ganache:
It was previously known as Testrpc and comes in both forms command line and UI. A virtual blockchain establishes ten standard Ethereum addreses with all and private key preloading them with simulated hundred ether each. With ganache there is no mining rather it automatically confirms every transaction. It is convenient for operating systems like windows, Linux and mac [13].

Metamask:
Metamask is an open source, user friendly tool having a graphical user interface for doing transactions in ethereum. Ethereal Dapps can run without having a complete ethereum node running your system browser. Metamask is essentially a bridge between browser and blockchain ethereum [12].

Solidity:
Solidity is a high-level language with JavaScript style syntax for contracts. It is a method for generating EVM machine-level code and converts it into simple instructions. It has four value types namely: Boolean, Integer, Address and String but has same operators as that of JavaScript [14].

Working:

The voter can log on to the voting website, then he has to log in with the Chrome Extension of Metamask to connect with the local blockchain. Once the user is connected, the page is refreshed and the user can see the candidates and the current votes. Below that is the option to select the candidate to vote, the voter selects the candidate and click on vote, a metamask pop-up comes up which tells the Ethereum transaction that has to be made, once the user clicks on Vote, the vote is given to the selected candidate provided that the voter hasn't voted before. If the user has already voted and attempts to vote again, a failed transaction will occur and vote will not be accounted.

A local blockchain is deployed using Ganache and metamask is set up to connect with it. Truffle framework allows to migrate the smart contracts created on solidity to the local blockchain. When the user clicks to vote, metamask allows to move Ether from one account to another. Every user is given a unique ID that is Ethereum Address and a private key and exact amount of Ether is distributed to all the voters' accounts. Once the user votes, the Ether is transferred from the voter's account to the Candidate's account, and all the transactions goes through the blocks, all the transactions will be visible to everyone once we launch the project. This will give voters complete transparency and they can cross-check their votes. Once the user has voted, the address will not contain the same amount of Ether, therefore if the user attempts to vote again, the transaction won't be completed and the vote will not be accounted. Mining is performed by all the other nodes but here, we have given Ganache the power to auto-mine on behalf of other nodes. The flowchart below explains the voter side of the process.

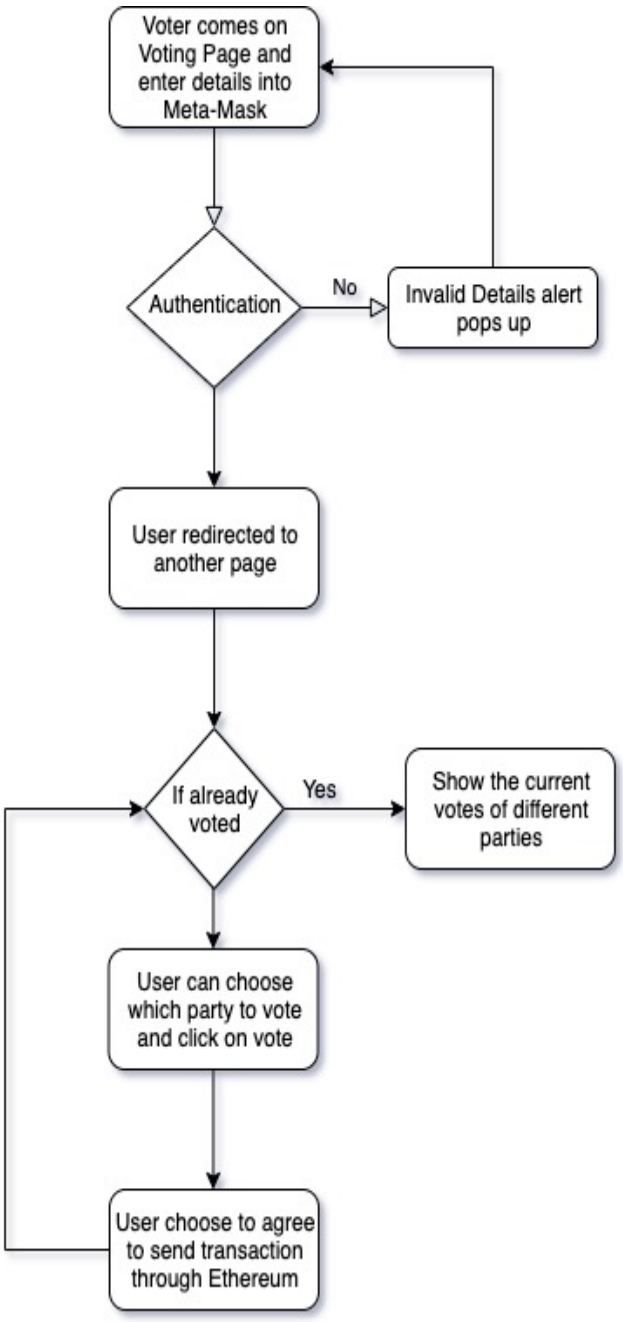

Figure 4. Flow model of the E-voting system based on blockchain

4. Implementations and Results

1. Setting up:

The first thing that we need to do is run local blockchain by starting up Ganache.

After setting up ganache there will be not any transaction as we have not done any transaction yet. As we can see from the snapshot below there is no transaction.

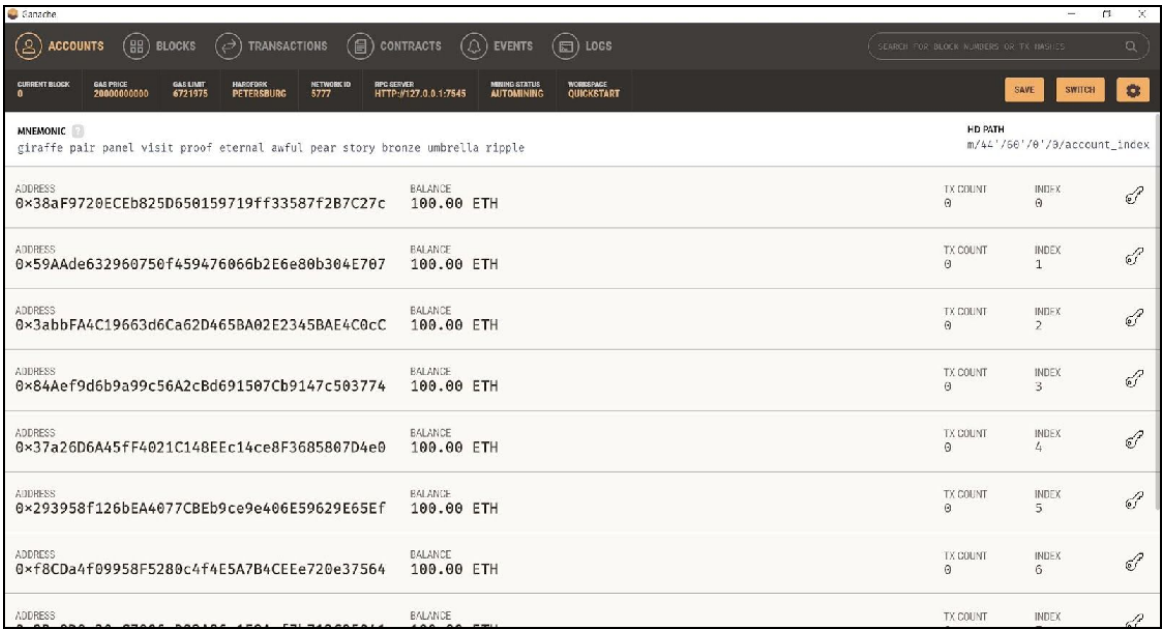

Figure 5. Screenshot of Setting up Ganache.

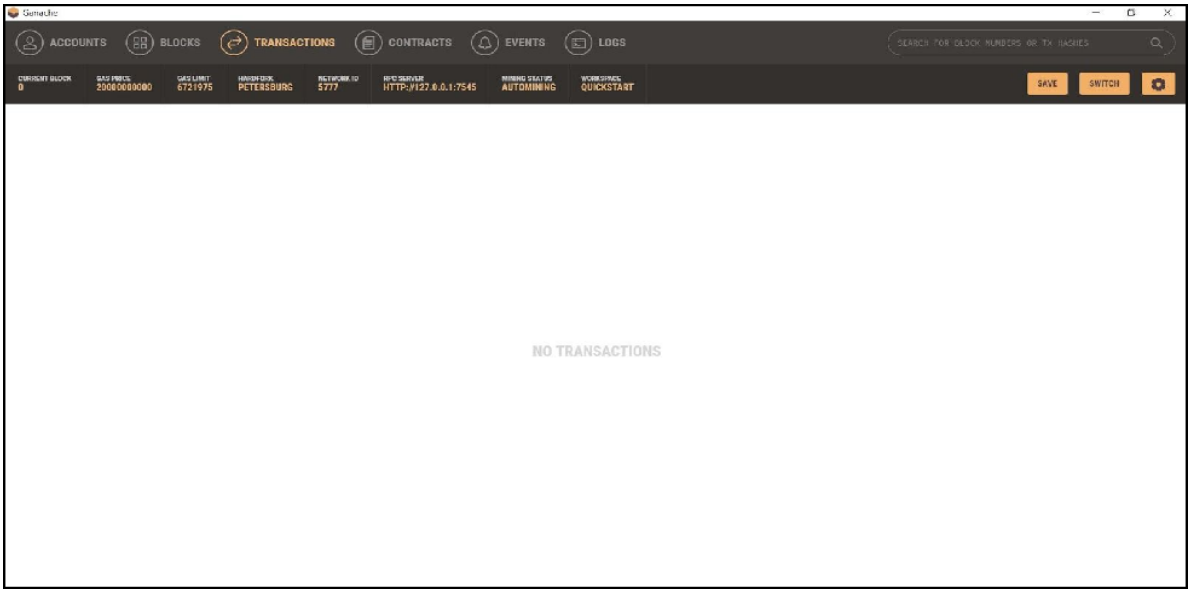

Figure 6. Snapshot of No Transaction.

Now we use truffle framework to transfer the smart contract to the blockchain by giving command on the command line. We have also used NPM directory by cmd. Following commands are being used for this purpose:

After migrating the smart contract, we start the project using NPM directory by cmd.

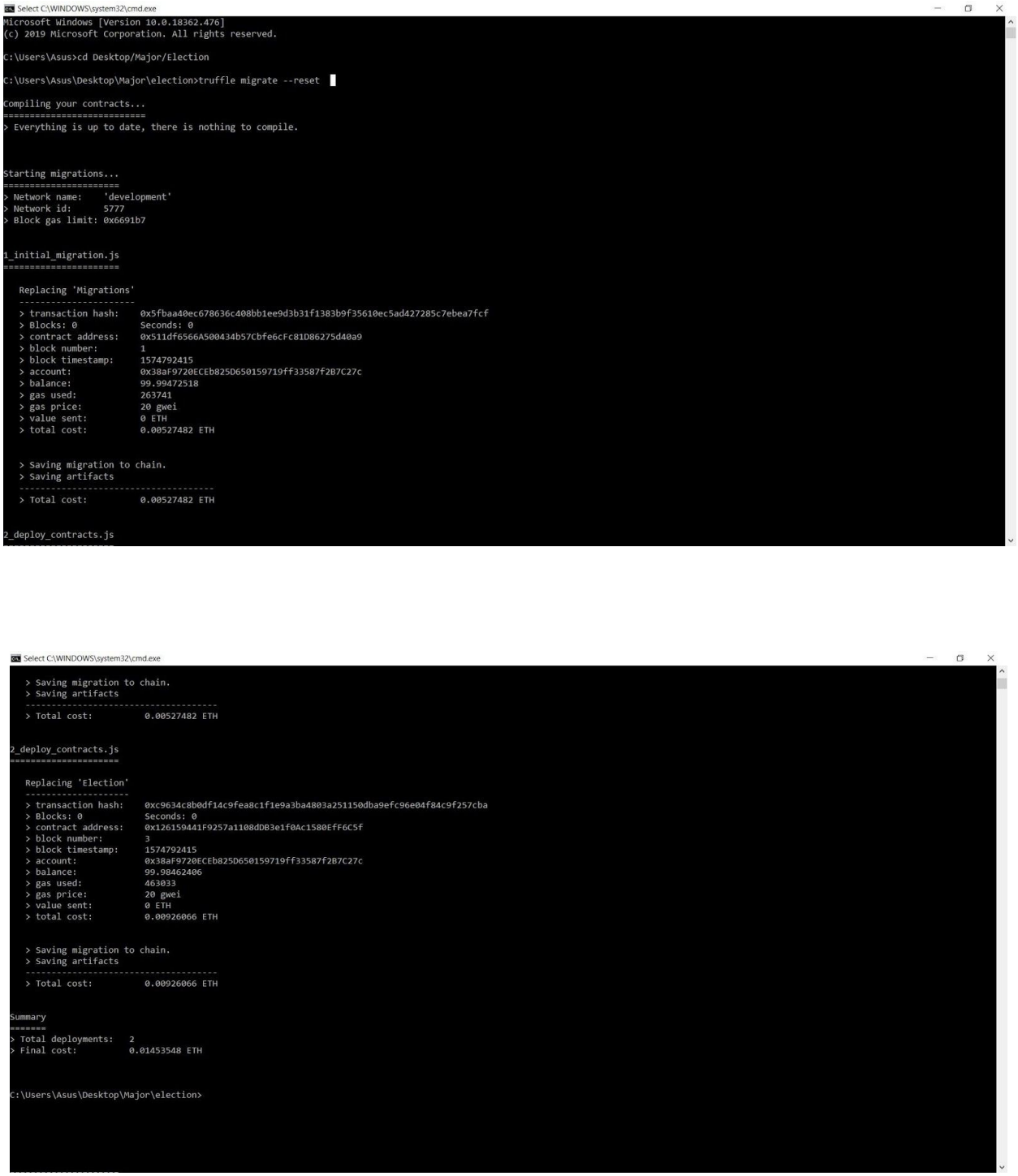

Figure 7. Snapshot of Command Line for Truffle Framework.

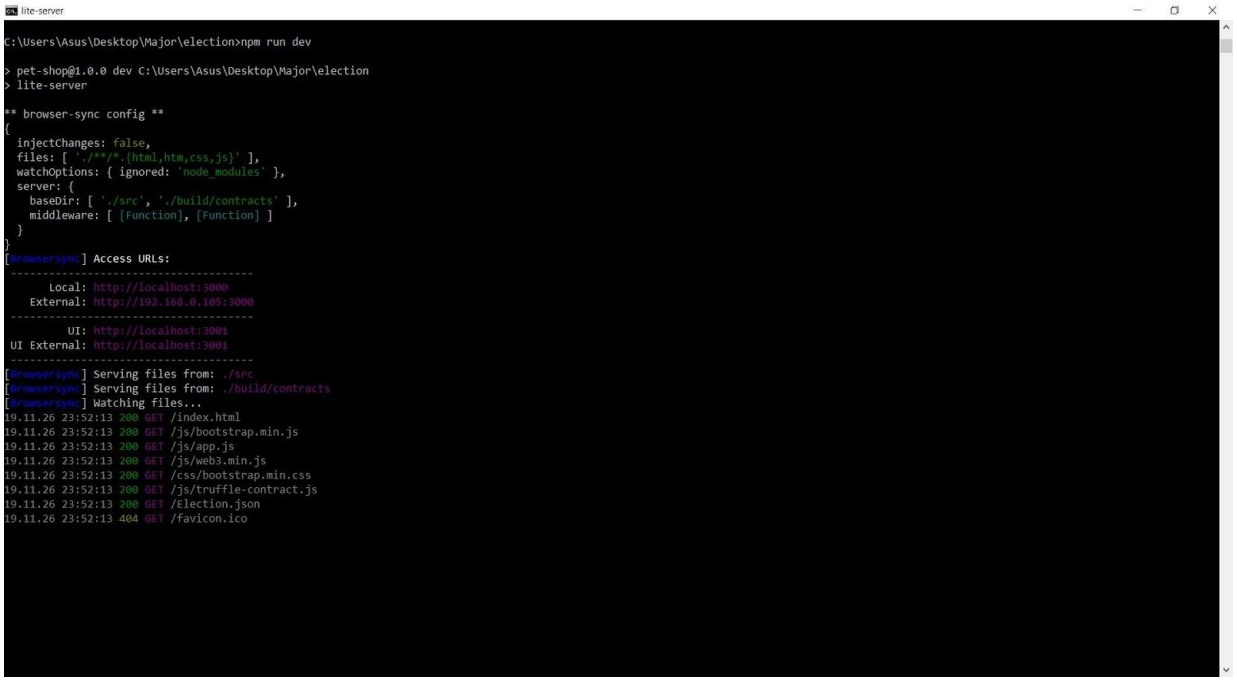

Figure 8. Command Line of NPM Directory

2. User Interface:

User interface is through which users can interact with the e-voting system. The picture bellow is how user will see the interface. The loading screen will continue to display loading until the electorate login through metamask.

Below secreen will be displayed when the electorate is logging in through metamask

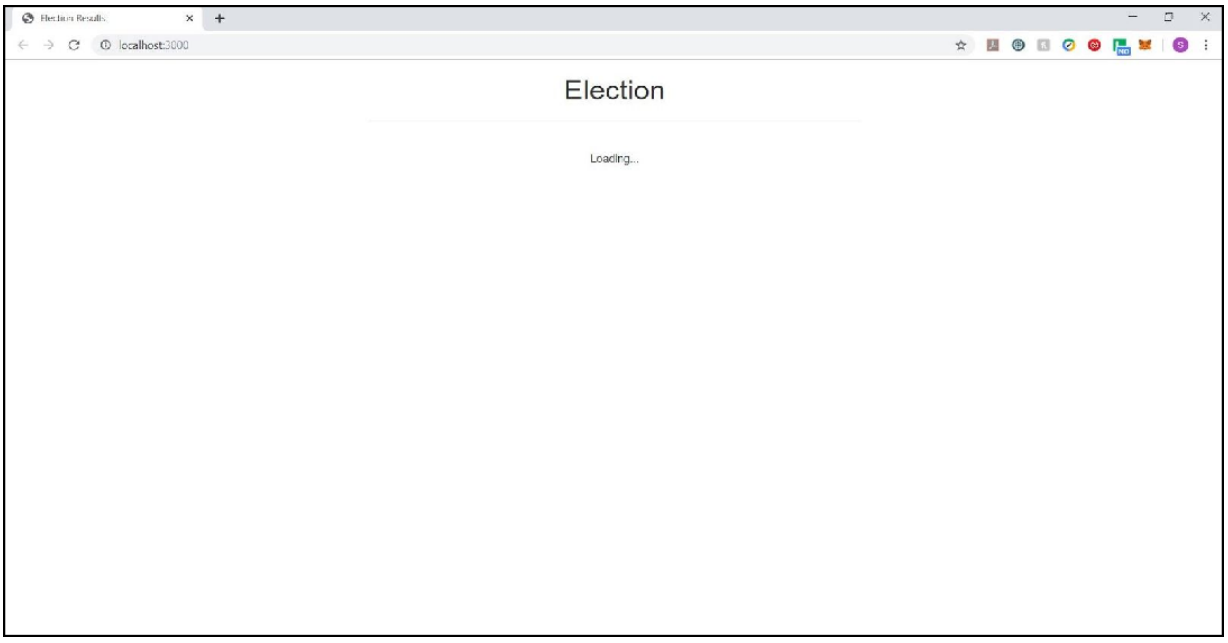

Figure 9. Snapshot of Loading Screen.

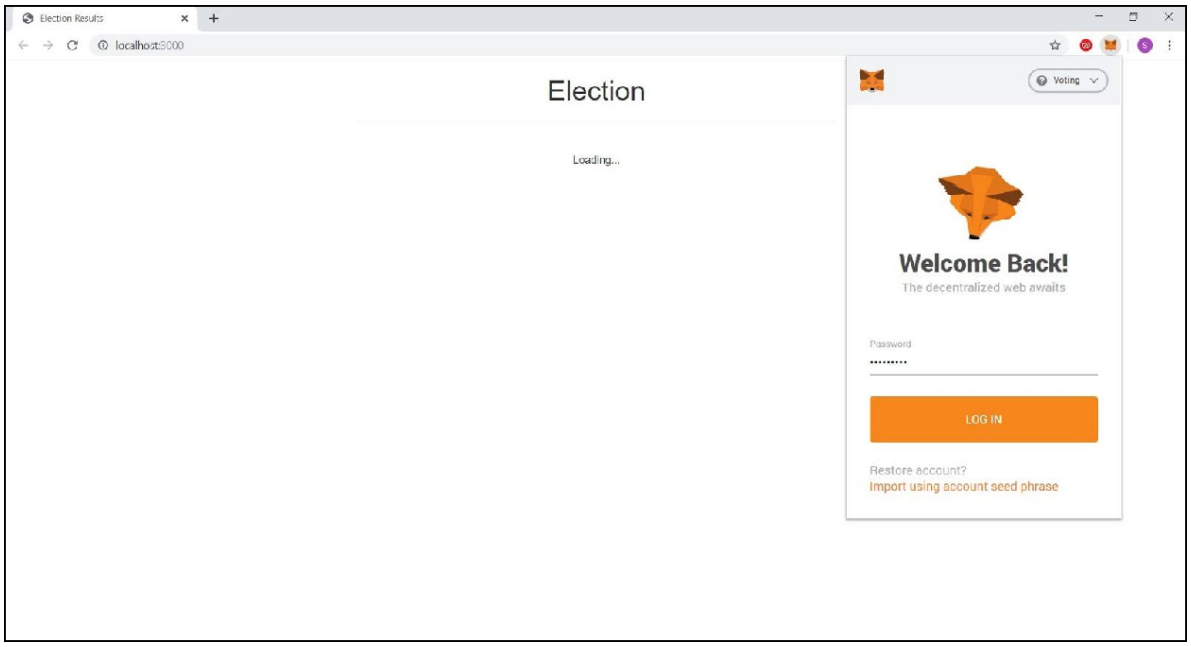

Figure 10. Constituency Logging In Via Metamask.

After the user has logged in, main screen comes up with zero vote, the user cannot vote until they import their account by entering private key.

The private key is given in advance to the user that will look like Figure 12.

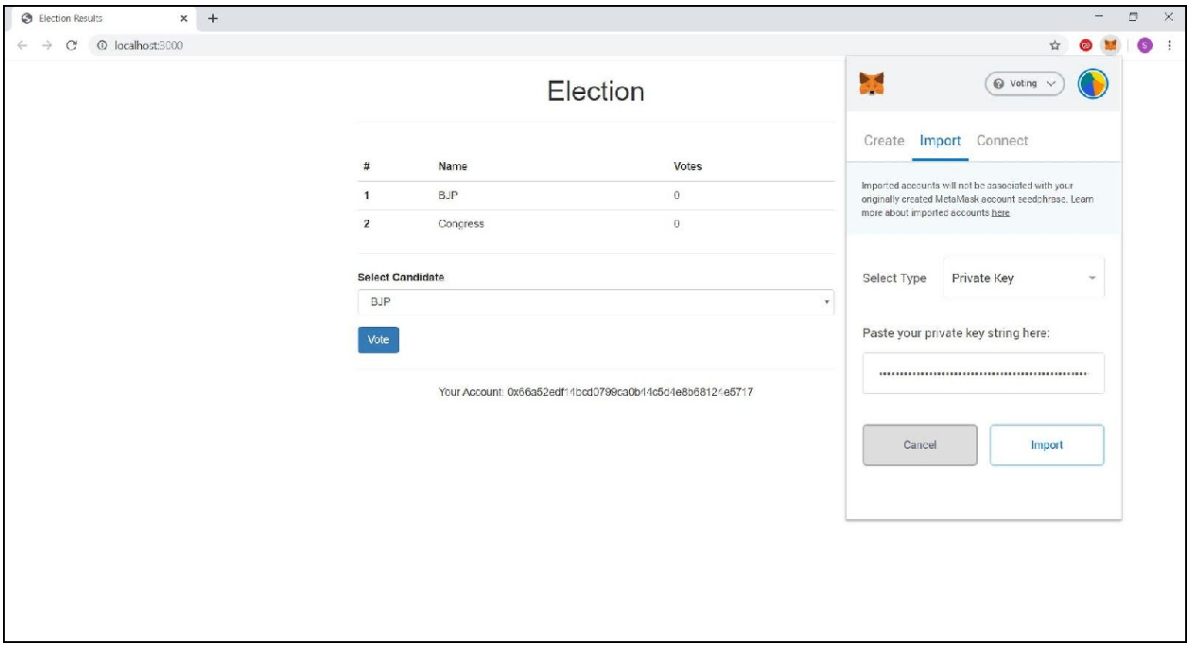

Figure 11. Main Screen

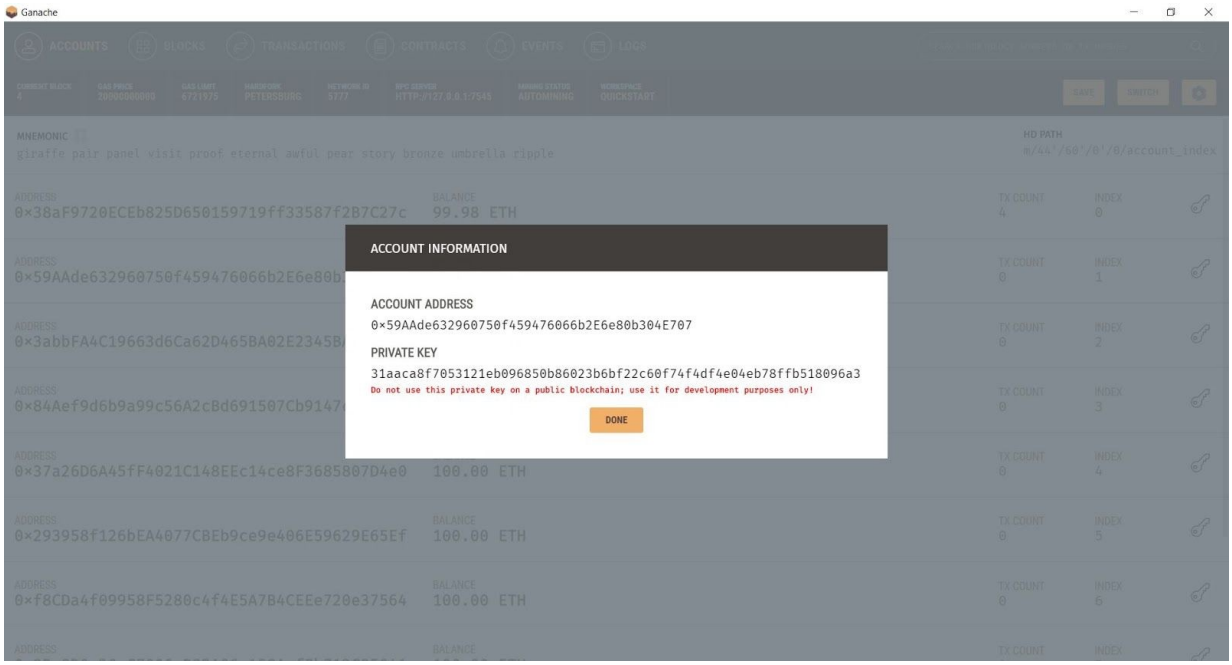

Figure 12. Private Key.

Voter imports their account by entering the private key above.

The electorates choose candidate of their choice, the metamask pop-up gets open when clicked on vote to confirm the transaction

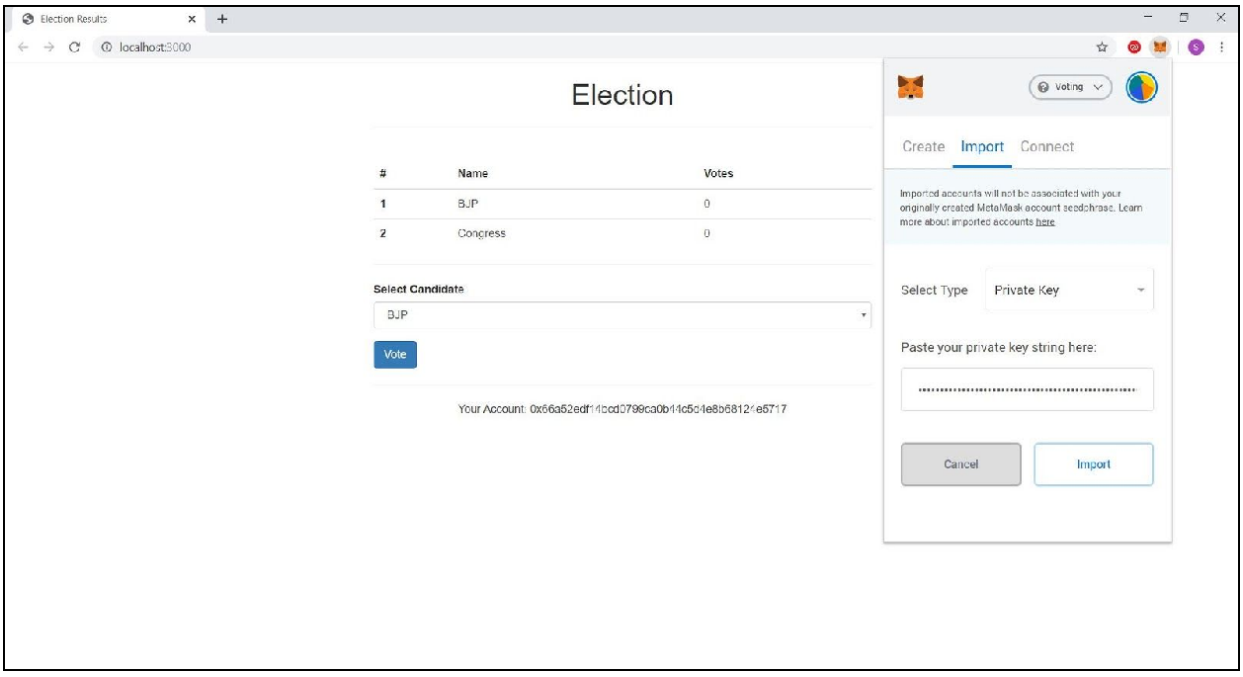

Figure 13. Snapshot of Importing Account.

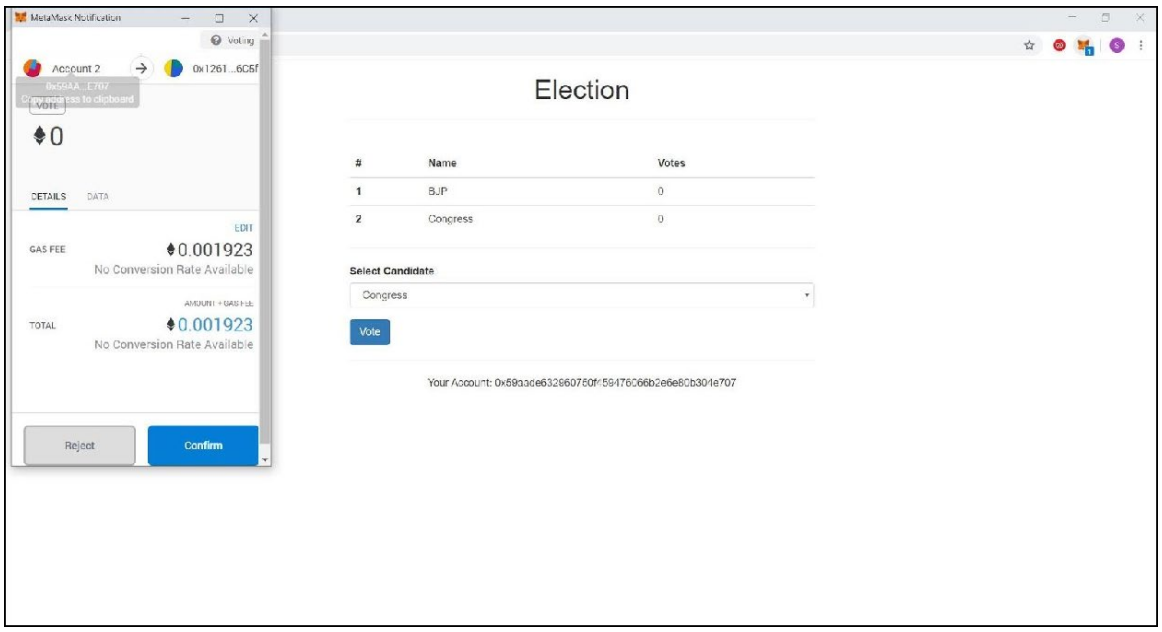

Figure 14. Snapshot of Confirming Transaction.

Once confirmation is done, the voter gets redirected to the main page where only results are visible but now you can't vote. In the similar manner, others can also vote by importing their account.

3. Checking the Transactions:

The transaction list will be available publicly to provide the user with convenience to tally their votes respectively. The users can check their votes given by them by looking into the transaction list. An entire transaction list will be like one given in fig 16.

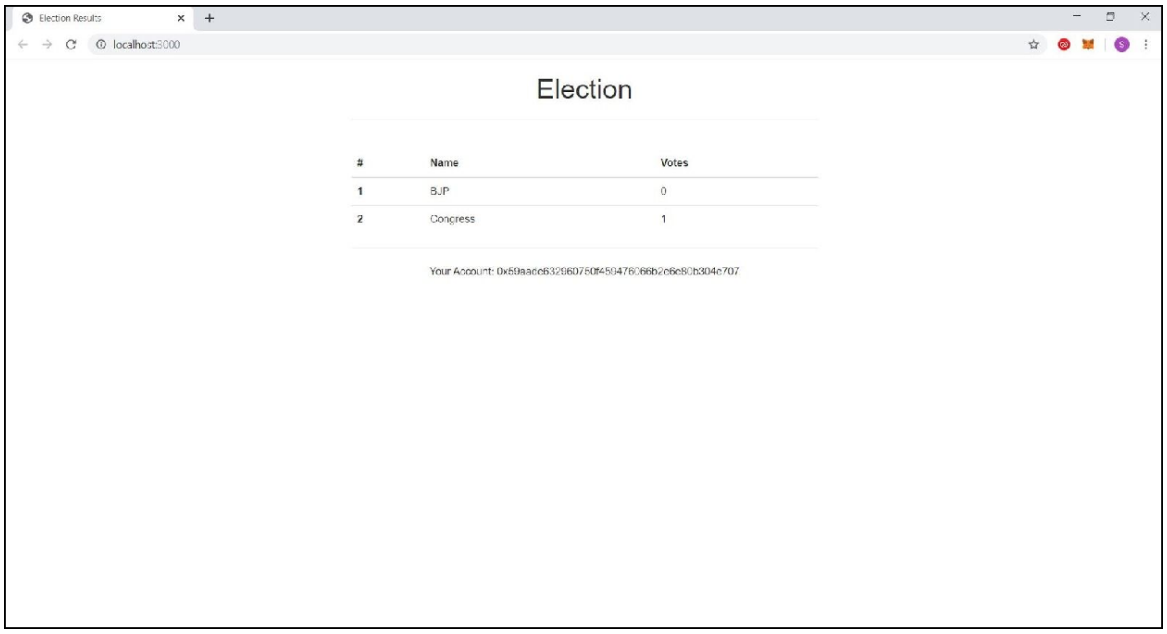

Figure 15. Snapshot of Result on Main Page.

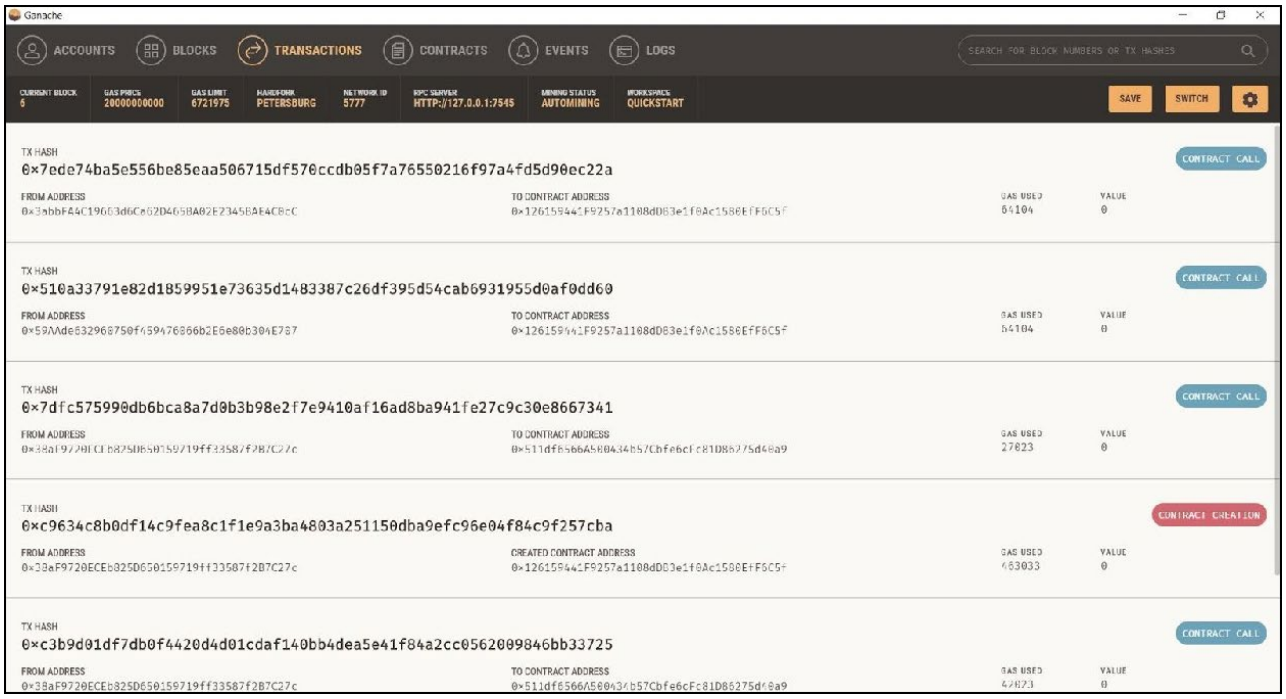

Figure 16. Snapshot of Transaction List.

4. Conclusion

The recent development in the area of voting system includes Blockchain technology, which not only proved to be time and cost efficient but is also safe and secure, hence is more reliable and precise than the earlier approaches. In this paper we have used blockchain based e- voting using smart contract which includes a set of rules governing the communication and decision on the contract between parties. Various tools like Ganache, Truffle framework, NPM and metamask were used for implementation purpose. As blockchain technology is decentralized due to which tempering and alteration in such system is quite attainable. Our proposed system provides convenience to the voters by allowing them to connect to the system having easy-to-use user interface, through which they can cast their vote by importing their account and can easily review their vote. It creates a sense of trust among voters, that there vote is being computed and kept in a safe custody.

References

[1] https://shermin.net/token-economy-book/

[2] Zhang, S., Wang, L. & Xiong, H. Int. J. Inf. Secur. (2019) Chaintegrity: blockchain-enabled large-scal e-voting system with robustness and universal verifiability. International Journal of Information Security. https://doi.org/10.1007/s10207-019-00465-8

[3] E. Elewa, A. AlSammak, A. AbdElRahman, T. ElShishtawy, "Challenges of Electronic Voting-A Survey", Advances in Computer Science: an International Journal, vol. 4, no. 6, pp. 98-108, 2015.

[4] Aranha DF, Ribeiro H, Paraense ALO (2016) Crowdsourced integrity verification of election results. Annals of Telecommunications:1–11. doi:10.1007/s12243-016-0511-1

[5] Gjøsteen K, Lund AS (2016) An experiment on the security of the norwegian electronic voting protocol. Annals of Telecommunications:1–9. doi:10.1007/s12243-016-0509-8

[6] Budurushi J, Renaud K, Volkamer M, Woide M (2016) An investigation into the usability of electronic voting systems for complex elections. Annals of Telecommunications pp 1–14. doi:10.1007/s12243-016-0510-2

[7] Neumann S, Volkamer M, Jurlind B, Prandrini M (2016) Secivo: a quantitative security assessment model for internet voting schemes. Annals Telecommunication pp 1–14

[8] Ayed, A.B. (2017). A Conceptual Secure Blockchain Based Electronic Voting System. International Journal of Network Security & Its Applications (IJNSA) Vol.9, No.3, May 2017

[9] Hsiao JH, Tso R., Chen CM., Wu ME. (2018) Decentralized E-Voting Systems Based on the Blockchain Technology. Advances in Computer Science and Ubiquitous Computing. CUTE 2017, CSA 2017. Lecture Notes in Electrical Engineering, vol 474. Springer, Singapore.

[10] Jonath Alexander, Steven Lander and Ben Howerton (2018). Netvote: A Decentralized

Voting Network Available at: https://netvote.io/wp-content/uploads/2018/02/Netvote-White-Paper-v7.pdf

[11] https://www.tutorialsteacher.com/nodejs/what-is-node-package-manager

[12] https://www.edureka.co/blog/developing-ethereum-dapps-with-truffle

[13] https://www.codementor.io/@swader/developing-for-ethereum-getting-started-with-ganache-l6abwh62j

[14] https://www.edureka.co/blog/solidity-tutorial/