PUNE INSTITUTE OF COMPUTER TECHNOLOGY,

DHANKAWADI PUNE-43.

A Mini-Project Report On

D-app for E-Voting System

SUBMITTED BY

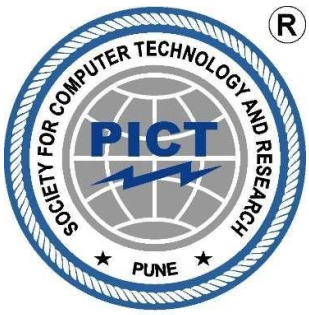
Mehul Oswal
(41444)
Burhanuddin Merchant
(41439)

Under the Guidance of

Prof. Deepika Bhaiya

Batch: R4

CLASS: BE-IV



COMPUTER ENGINEERING DEPARTMENT

Academic Year: 2022-23

PUNE INSTITUTE OF COMPUTER TECHNOLOGY,

DHANKAWADI PUNE-43.

CERTIFICATE

This is to certify that ***Mehul Oswal (41444), Burhanuddin Merchant (41439) students of B.E. (Computer Engineering Department)*** Batch 2022-2023, have satisfactorily completed report on "D-app for E-Voting System" towards the partial fulfilment of the fourth year Computer Engineering Semester VII.

Prof. Yogesh Handge
Internal guide

Date:

Place: Pune

D-app for E-Voting System

Contents

1 PROBLEM STATEMENT : 1

2 ABSTRACT : 1

3 REQUIREMENTS : 1

Hardware Requirements : 1

Software Requirements : 1

4 INTRODUCTION : 2

5 OBJECTIVE : 2

6 THEORY: 2

7 RESULT: 5

8 DEPLOYEMENT: 7

9 CONCLUSION : 7

List of Figures

1 Workflow of voting system application 3

1. PROBLEM STATEMENT:

Develop a Blockchain based application dApp for e-voting system.

2. ABSTRACT:

For long, different e-voting systems have been provided with the goal of increasing security and minimizing cost. Blockchain is a major breakthrough in the technological industry that provide immense secured platform. With the launch of Ethereum, a decentralized platform which runs decentralized applications (DApps) on it, a secured voting system now seems possible.

Many organizations have now shifted their focus on voting through blockchain platforms. There's a very high chance that a normal voting method won't lead to a clear majority. There can be many ways to deal with this issue which includes another voting process to take place which can be quite expensive in terms of time and resources. Here, we introduce the vote-trading concept where the votes can be redistributed to other candidates in case if there is no clear majority and also this 'majority' factor can be set by the organization according to their requirement.

We discuss the design for the blockchain based preferential e-voting system using the Solidity programming language where instead of one vote per candidate, we provide the concept of giving preference to the candidates.

3. REQUIREMENTS:

Hardware Requirements:-

- Laptop with 8 GB RAM
- Intel core i5 processor

Software Requirements:-

- Latest version of Google Browser
- Any online platform to draw UML diagrams like Creately

4. INTRODUCTION:

Unlike nationwide voting where there are millions of voters, in an organization, there are very few voters, and very few nominees. There's a very high chance that a normal voting method won't lead to a clear majority. Thus, another voting process has to take place which can be quite expensive in terms of time and resources or the other option is to have some kind of vote-trading where the votes can be redistributed to other candidates. As many organizations are now shifting their focus to blockchain technology because of its immense security and other reasons as mentioned in , voting through blockchain has been one of the main applications in the blockchain world.

We have introduced an e-voting system based on blockchain technology. The system utilizes smart contracts that make cost-efficient and secure election possible. Not only this, the voters privacy are also guaranteed by this system. We have discussed how the blockchain technology can overcome limitations of e-voting systems. We have discussed how blockchain based e-voting system have layed the ground for transparency and how these systems ensures the election integrity and security. We have proposed a decentralized voting platform which is based on Ethereum Technology. Restriction of multiple votes per mobile (MSISDN) is the main contribution of this platform.

The future work for this system could be to develop the system further to make it more suitable for nationwide elections using some of the technologies located in the voting centers such as fingerprint. We have proposed Votereum, an Ethereum-based E-voting system that utilizes blockchain technology and smart contract. We in our system have introduced the concept of verifiable E-voting platforms and have discussed the evaluation and security concerns to the system. Not only this, the authors have discussed the requirements, architecture and design of the system. We know that Ethereum platform provides immense security to the decentralized applications, thus our main focus have been on how can we deal with the situation when the voting process leads to no clear majority defined by the organization.

5. OBJECTIVE:

- To understand and explore the working of Blockchain technology and its applications.
- To create a blockchain application d-App for e-voting system.

6. THEORY:

Blockchain works by creating decentralized distributed ledgers that are circulated over all devices participating in the system. It allows to share records based on peer-to-peer replication, and processing by all nodes in the network namely, transacting nodes and validating nodes. When records are placed in the ledger, all nodes in the network go through encryption procedures and are processed by all miners. The blockchain provides highly reliable storage
of data and makes use of consensus strategy, digital signatures, and hash of each block.

Blockchain provides many features such as integrity, traceability, safety, and security for protection of data. Blockchain is applied in both private and government sectors. Usually, an organization has a database where it stores information about their employees and workers. The system will use information from this database so that we can have a list of valid voters and valid candidates in our system the system flowchart for user casting the vote. The user would be directed to a login portal where the user would need to enter some credential to authenticate the user. The user interface will be a web application. Once the user enters the credentials correctly, i.e., the user

is successfully able to log in into system, the user would be directed to the voting portal.

In the voting portal, there would be a list of candidates. The voter needs to give preference to the candidates with the lowest number given to the most preferred candidate and the largest number given to the least preferred candidate. In our system, each preferred list by a voter is taken as one transaction.

Thus whenever a user or voter gives preference to the candidates and submit his list, a transaction is added to the blockchain network.

Thus, each voter would be able to cast only one vote.

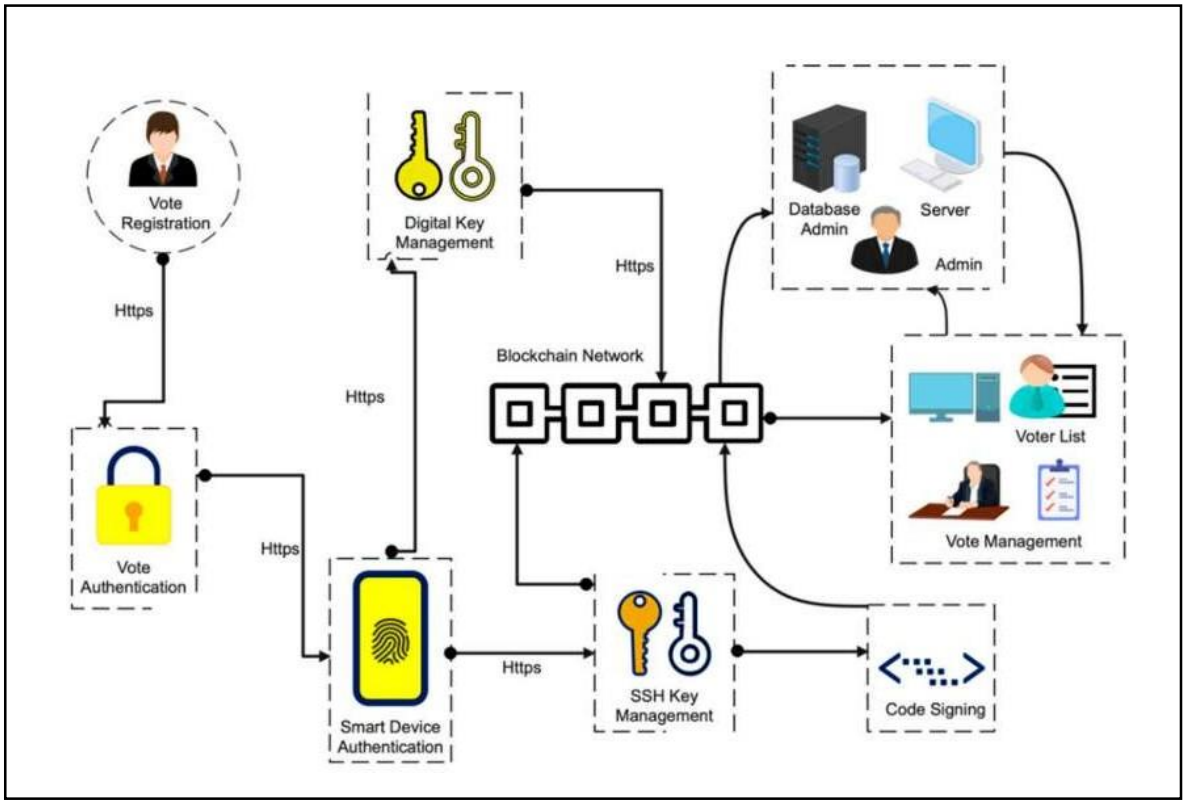


Figure 1: Workflow of voting system application

Description:

- The authority must login first with the provided session ID.
- The voter can now begin the process of voting with proper authentication through OTP(one time password) on the respective linked mobile number.
- If the voter is valid then the system will check for for the voters age and the address to which he can give vote.
- the voting pallete will be opned with candidate names,their parties and logos.
- Now the voter can give his vote by clicking vote button.
- one voter can give his vote only once,i.e after one time voting buttons are disabled and the vote is automatically loged out.
- Same process continiues for many more votters irrespective of their voting wards.

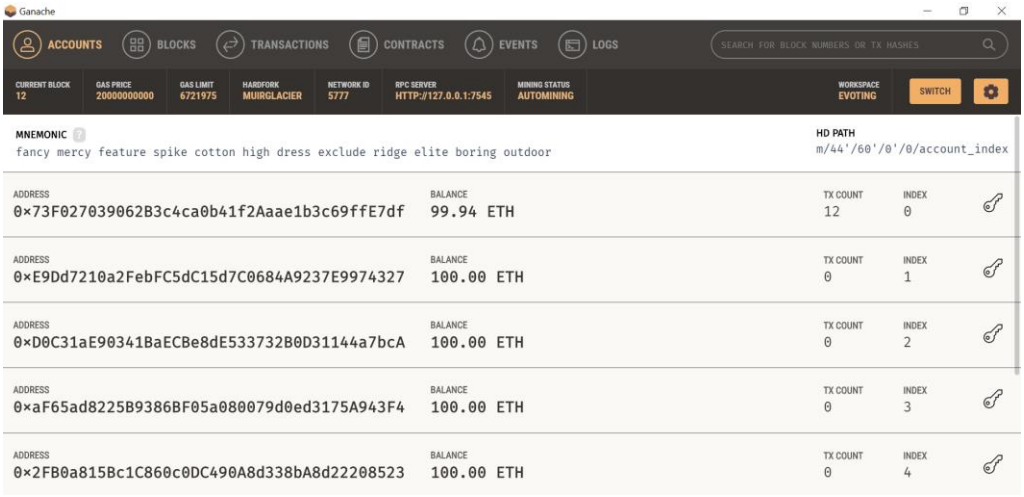
Installing and Running Project

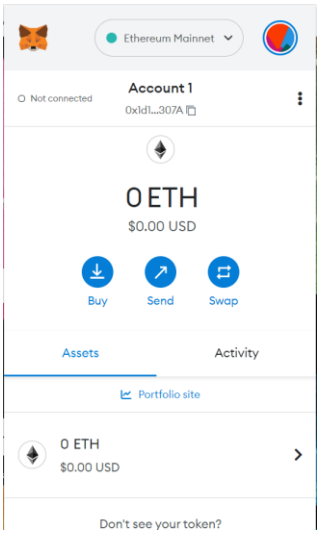
1. Install Ganache and create a workspace.
2. Install Truffle npm package globally by running npm install -g truffle.
3. Run truffle migrate --reset from the command line to deploy the smart contract to the blockchain.
4. Download Metamask Chrome extension for the browser to help interaction between the application and the blockchain.

Running Project

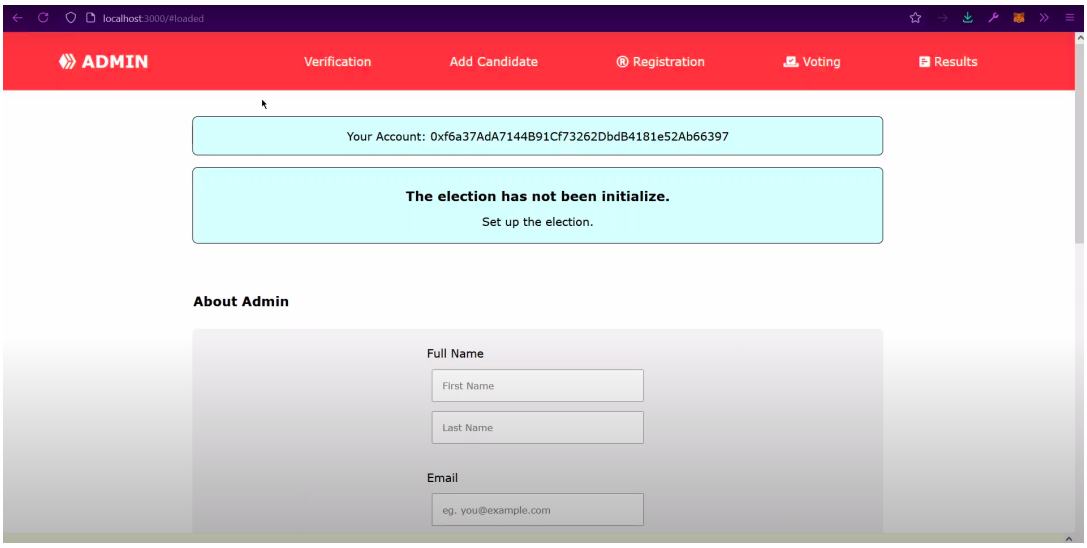
1. **To run development server:**
   a. cd blockchain
   b. npm install
   c. npm start
2. **To run node server:**
   a. cd server
   b. npm install
   c. npm start

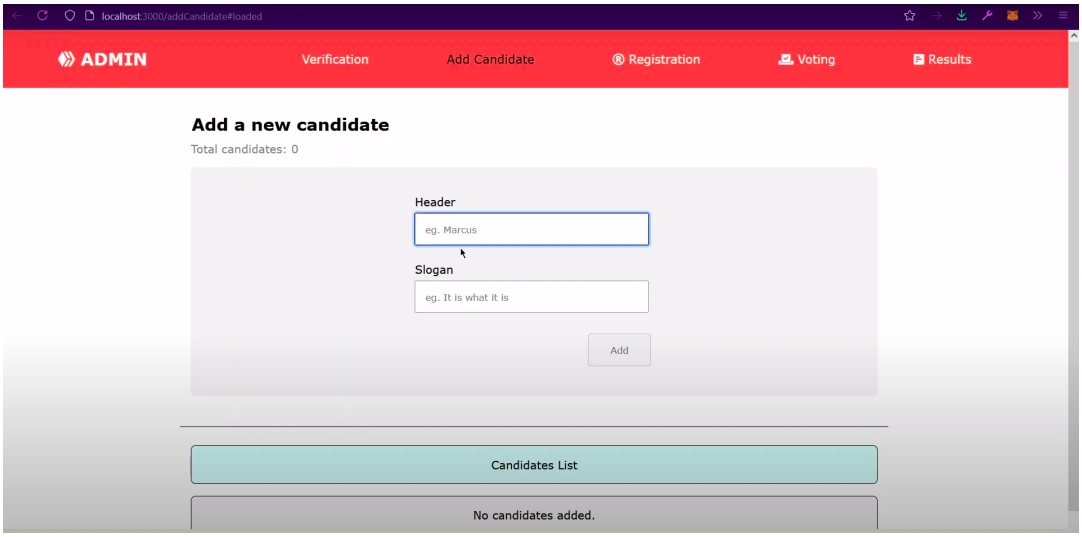
GANACHE and Metamask:

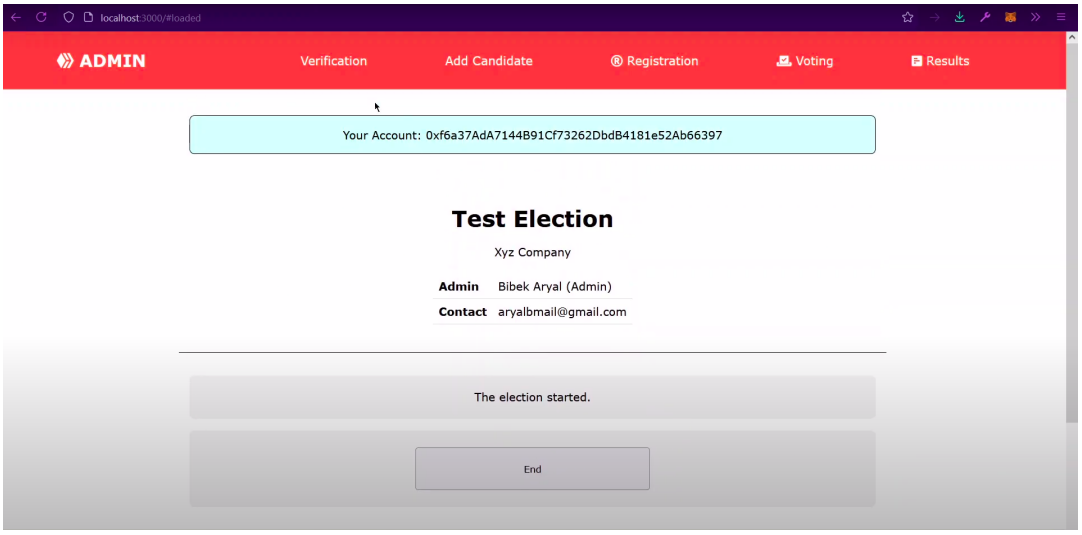


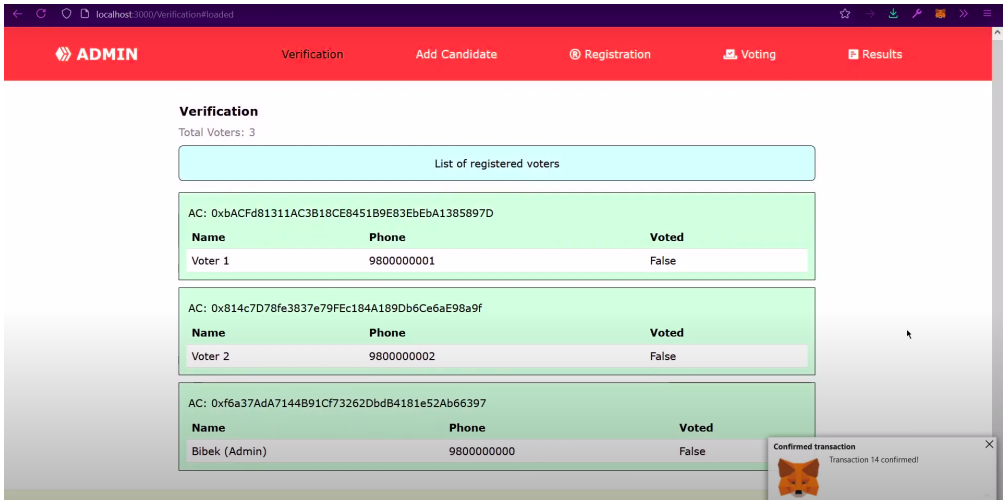


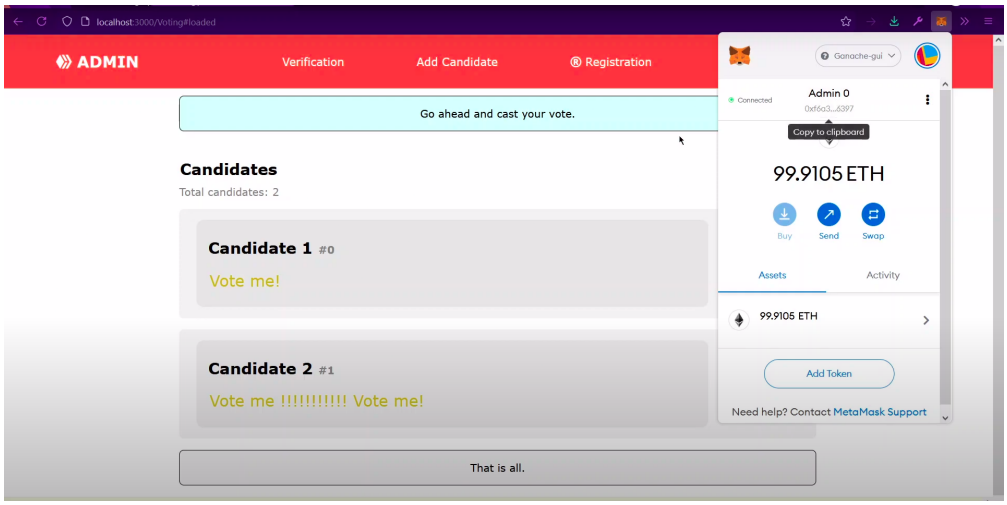
7. RESULT

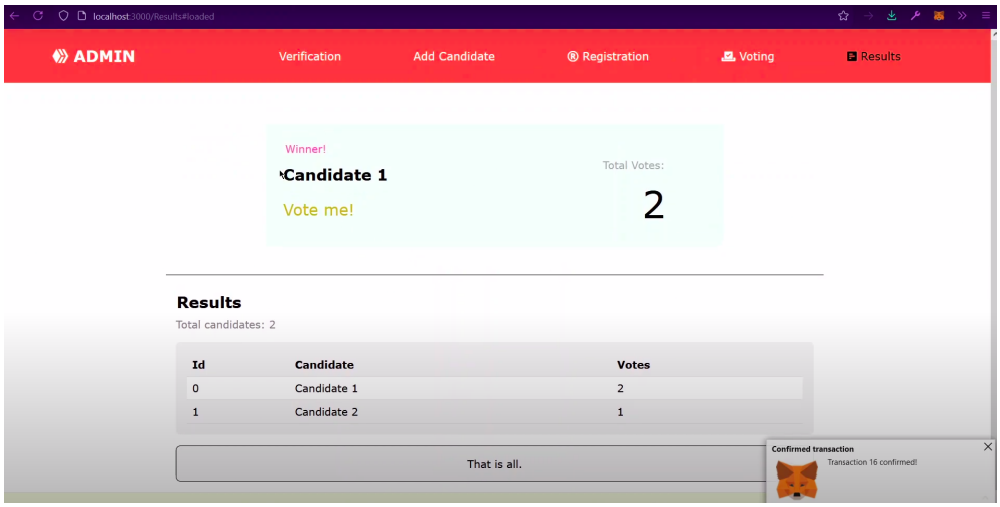

localhost:3000/#loaded
ADMIN
Verification
Add Candidate
Registration
Voting
Results
Your Account: 0xf6a37AdA7144B91Cf73262DbdB4181e52Ab66397
The election has not been initialize.
Set up the election.
About Admin
Full Name
First Name
Last Name
Email
eg. you@example.com


localhost:3000/addCandidate#loaded
ADMIN
Verification
Add Candidate
Registration
Voting
Results
Add a new candidate
Total candidates: 0
Header
eg. Marcus
Slogan
eg. It is what it is
Add
Candidates List
No candidates added.


localhost:3000/#loaded
ADMIN
Verification
Add Candidate
Registration
Voting
Results
Your Account: 0xf6a37AdA7144B91Cf73262DbdB4181e52Ab66397
Test Election
Xyz Company
Admin
Bibek Aryal (Admin)
Contact
aryalbmail@gmail.com
The election started.
End


localhost:3000/Verification#loaded
ADMIN
Verification
Add Candidate
Registration
Voting
Results
Verification
Total Voters: 3
List of registered voters
AC: 0xbACFd81311AC3B18CE8451B9E83EbEbA1385897D
Name
Phone
Voted
Voter 1
9800000001
False
AC: 0x814c7D78fe3837e79FEc184A189Db6Ce6aE98a9f
Name
Phone
Voted
Voter 2
9800000002
False
AC: 0xf6a37AdA7144B91Cf73262DbdB4181e52Ab66397
Name
Phone
Voted
Bibek (Admin)
9800000000
False
Confirmed transaction
Transaction 14 confirmed!


localhost:3000/Voting#loaded
ADMIN
Verification
Add Candidate
Registration
Go ahead and cast your vote.
Candidates
Total candidates: 2
Candidate 1 #0
Vote me!
Candidate 2 #1
Vote me !!!!!!!!!! Vote me!
That is all.
Ganache-gui
Connected
Admin 0
0xf6a3...6397
Copy to clipboard
99.9105 ETH
Buy
Send
Swap
Assets
Activity
99.9105 ETH
Add Token
Need help? Contact MetaMask Support


localhost:3000/Results#loaded
ADMIN
Verification
Add Candidate
Registration
Voting
Results
Winner!
Candidate 1
Vote me!
Total Votes:
2
Results
Total candidates: 2
Id
Candidate
Votes
0
Candidate 1
2
1
Candidate 2
1
That is all.
Confirmed transaction
Transaction 16 confirmed!

8. DEPLOYMENT:

The decentralised based voting system is developed to overcome the flaws of EVM system. So directly EVM will be replaced by touch screen interface having the great user interface and high security.

9. CONCLUSION:

Our voting system uses the concept of preference based voting and makes use of Ethereum platform. The votes are counted, and if we have a clear majority as defined by the organization, then our system will declare the winner. But if we don’t have clear winner, the votes of the last candidate are distributed according to the preferences given by the voters for that candidate.Thus, the last candidate is removed from the competition and its votes distributed to other candidates according to the preferences given in the votes of the eliminating candidates. This process continues as long as we don’t get a clear majority winner.