Software Requirements Specification

for

DE-Vote

Version 1.0

Prepared by Muhammad Waris, Mujeera Noor, Saifullah Soomro, Inamullah Memon and Muhammad Idrees

Group 02

02-02-2019

Table of Contents

Revision History

Name	Date	Reason For Changes	Version

1. Introduction

1.1 Purpose

The purpose of this document is to present detailed description of a Decentralized E-Voting System based on Blockchain (DE-Vote). It will explain the purpose, features and interfaces of the system, plus the constraints the constraints under which it must operate.

1.2 Intended Audience and Reading Suggestions

This project is a protype for the DE-Vote and is restricted within the university premises for now. And this document is primarily intended to be a reference for developing the prototype of DE-Vote for the development team.

1.3 Product Scope

The purpose of developing DE-Vote is to make voting easier, as the voter can go to nearest polling station to cast his vote. Minimize the paper work in order to put end to paper waste and reduce the human resource needed for carrying out the election.

More specifically this system is designed to eliminate political and unfair business and promote transparency in casting votes since it is decentralized the voting system will not be in control of any third party.

1.4 References

- K. Wüst, "Do you need a Blockchain?," in 2018 Crypto Valley Conference on Blockchain Technology, 2018.
- D. P. Moynihan, "Building Secure Elections: E-Voting, Security, and Systems Theory," Public Administration Review, vol. 64, no. 5, pp. 515-528, 2004.
- A. M. A. HTET NE OO, "A Survey of Different Electronic Voting Systems," International Journal of Scientific Engineering and Technology Research, vol. 3, no. 16, pp. 3460-2464, 2014.
- B. R. Rifa Hanifatunnisa, "Blockchain Based E-Voting Recording System," in 11th International Conference on Telecommunication Systems Services and Applications (TSSA), Lombok, 2017.

2. Overall Description

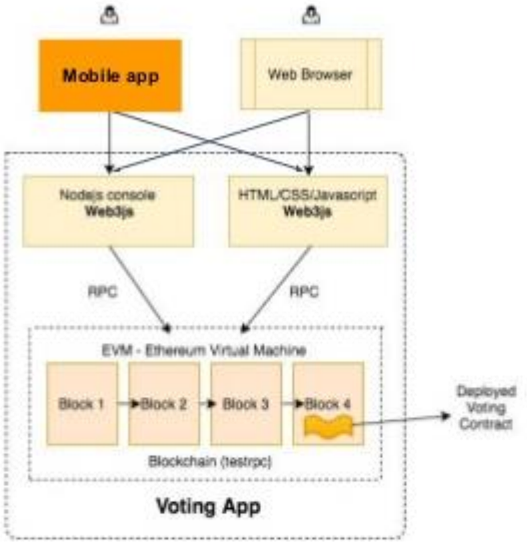
This section will give an overview of the whole system. The system will be explained in its context to show how the system interacts with other systems and introduce the basic functionality of it. It will also describe what type of stakeholders that will use the system and what functionality is available for each type. At last, the constraints and assumptions for the system will be presented.

2.1 Product Perspective

This system will be deployed on two platforms, both Web and Mobile.

- The mobile application will be deployed on Android Tablets that would be placed in every polling station and will be used to cast votes.
- The web portal will be showing the results after the election is carried out.

Since this is a data-centric product it will need somewhere to store the data. For that, a distributed ledger will be used. Both the mobile application and web portal will communicate with the blockchain's distributed ledger, however in slightly different ways. The web portal will only use the database to get the data and show the results after the election ends. While the mobile device will have to be a node so, the voter can add data (Vote) to the ledger. All the communication will go over the Blockchain network.



2.2 Product Functions

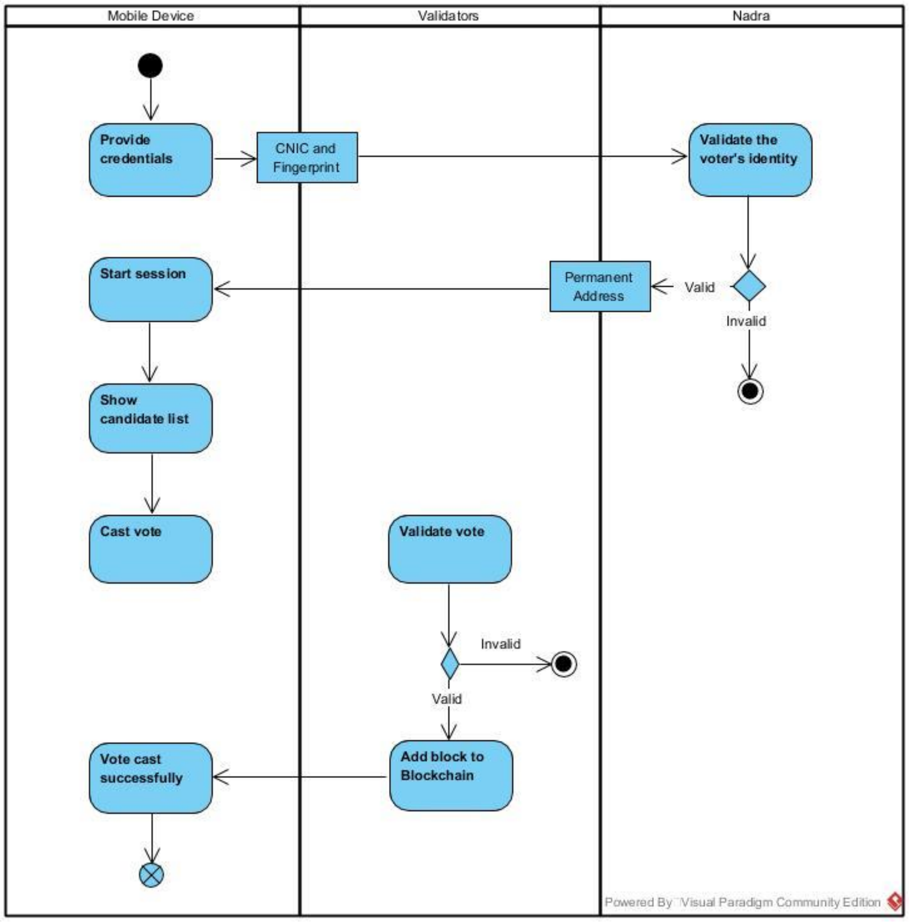
Our application will require the following information from the voter:

1. Voter's CNIC number which will be inputted from the Android device.
2. Voter's fingerprint using biometric fingerprint scanner which will be embedded in the device.
3. Voter's choice.

Voter's CNIC number and fingerprint will be used verify his/her identification and whether he/she is registered or not

And after the voter is verified, the voter will be provided with a list of candidates based on his permanent address provided by him/her in the CNIC. Then voter must provide a choice for the candidate he/she is willing to vote for

After the voter makes his choice, the vote is validated by the nodes (the devices present on other polling stations) present in the Blockchain network to prevent double voting.



2.3 User Classes and Characteristics

- **Administrator:** Creates the voting lobby, sets lobby to public or private, adds candidates to lobby, and sets the date and time the voting period will end. Administrators can vote during the voting period.
- **Voter:** Receives a voting token upon registration, votes before time ends.
- **Candidate:** Added to lobby by administrator, tokens are received by this address during voting phase, candidates can vote

2.4 Operating Environment

The application will be deployed on an Android device with which a biometric fingerprint scanner would be attached.
In case of Operating System, the application will run on all the devices with Android version 5.0 and later.

2.5 Design and Implementation Constraints

The mobile application is constrained by the internet connection. Since the application fetches data from the distributed ledger over the network through the internet. It is crucial that there is an internet connection for the application to function.
There is another limitation to this project that is the Authentication server for Biometric identity verification of the voter, which is not easily accessible to everyone so, we will have to make a dummy server so that we can perform verification of the voter.

2.6 User Documentation

The UI/UX of the application is quite simple and easy to interact but, still the user would be guided using video tutorials beforehand.

2.7 Assumptions and Dependencies

There will be no third-party or commercial components used while implementing the project, but if there comes an issue of scalability, we might opt for Microsoft Azure / Amazon Web Services BaaS (Blockchain as a Service) to scale our project.

3. External Interface Requirements

This section provides a detailed description of all inputs into and outputs from the system. It also gives a description of the hardware, software and communication interfaces and provides basic prototypes of the user interface.

3.1 User Interfaces

The voter first sees the “Welcome” screen where he/she will be prompted to enter his/her CNIC number and put their thumb on the Biometric fingerprint scanner in order to verify his/her identity, see Figure 1.
If the voter’s identity is verified, he/she should be able to see the candidate list and select the candidate for whom he/she desires to vote for, see Figure 2.

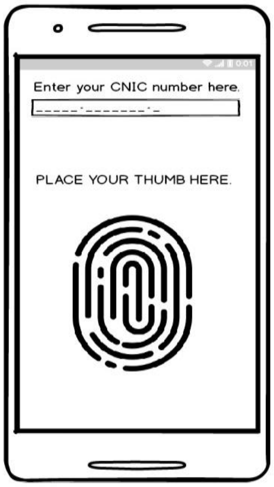


Figure 1

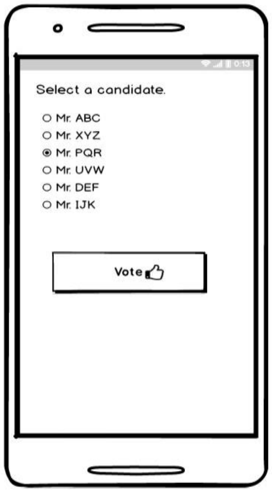
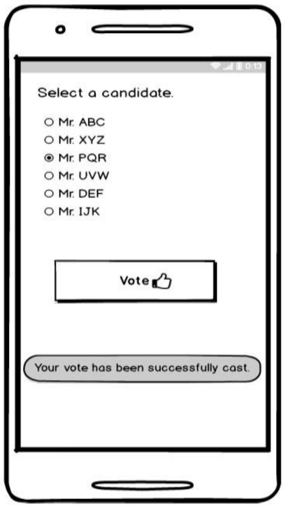


Figure 2

And after the vote gets validated , the user should see a pop up which says, “Your vote has been successfully cast”, see Figure 3.
Voters will be able to see the results on the web portal after the election ends, see Figure 4.



Your vote has been successfully cast.

Figure 3

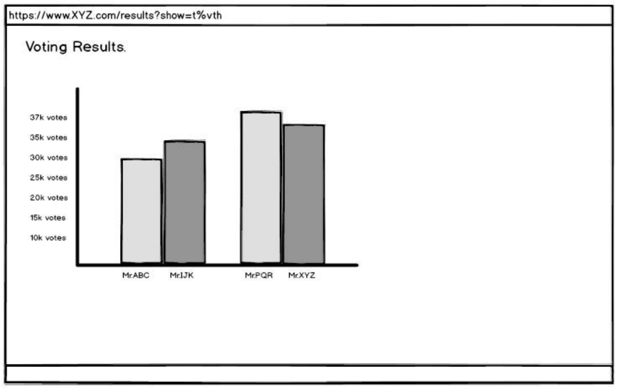

https://www.XYZ.com/results?show=t%vth

Figure 4

3.2 Hardware Interfaces

Since neither the mobile application nor the web portal have any designated hardware, it does not have any direct hardware interfaces. The hardware connection to the distributed ledger is managed by the underlying operating system on the mobile device and the web server.

3.3 Software Interfaces

The mobile application fetches the permanent address of the voter after his/her identity verification in order to get the candidate list of that constituency so that the voter votes for the candidate running election from the area of voter's permanent address, see Figure 2.
The communication between the blockchain and the Mobile Application consists of operation concerning both reading and writing the data, while the Web portal consists of only reading operations.

3.4 Communications Interfaces

Since all the transactions are being carried out in a Blockchain network (P2P network), there must be certain protocols used for transfer and integrity of data. We'll need a Consensus protocol so that all the nodes can come to an agreement that who gets to add the block to the chain. Smart contracts as an intermediate between the Application layer and the blockchain itself.

4. Functional Requirements

4.1 User Class 1 – The User (Voter)

4.1.1 Functional requirement 1.1

ID: FR1
TITLE: Provide credentials.
DESC: The user must provide his/her CNIC and Fingerprint to prove his/her identity.
RAT: In order for a user to see the candidate list according to his permanent address.
DEP: none.

4.1.2 Functional requirement 1.2

ID: FR2
TITLE: Choose a candidate.
DESC: The user must provide his/her choice to select a particular candidate.
RAT: In order for a user to vote a particular candidate.
DEP: FR1

4.2 User Class 2 – The Admin

4.2.1 Functional Requirement 2.1

ID: FR1
TITLE: Add candidates to lobby.
DESC: The list of candidates must be prepared before the voting starts.
RAT: In order for the candidate to receive token (votes)
DEP: none.

4.3 User Class 3 – The Candidate

The candidate has almost all the same functional requirements as the voter except one.

4.3.1 Functional Requirement 3.1

ID: FR1
TITLE: Receive the token (votes)
DESC: Candidates must be able to receive tokens .
RAT: In order to give out the result with the greatest number of tokens(votes).
DEP: none.

5. Other Nonfunctional Requirements

5.1 Software Quality Attributes

5.1.1 Security

ID: QR1
TAG: Communication Security
GIST: Security of the communication between the nodes.
SCALE: The messages should be encrypted for voting information. So, the voters identity remains anonymous.
METER: Attempts to get voter’s identity on 1000 voting session during testing.
MUST: 100% of the votes of a voting session should be encrypted.

Appendix A: Glossary

Term	Definition
User	Someone who interacts with the mobile/web application.
Admin	System administrator who is given specific permission for managing and controlling the system.
Web portal	A web application which will show the election results.
DESC	Description
RAT	Rational
DEP	Dependency
GIST	A short, simple description of the concept.
SCALE	The scale of measure used by the requirement
MUST	The minimum level required to avoid failure
Voter	The user who will vote for the candidate.
CNIC	Computerized National Identity Number
Blockchain	a growing list of records, called blocks, which are linked using cryptography.
Ledger	A file for recording and totaling economic transactions
RPC	Remote Procedure Call
Node	A point/device in a network.
Token	A token is a specific amount of digital resources which you control and can reassign control of to someone else.

Appendix B: Analysis Models

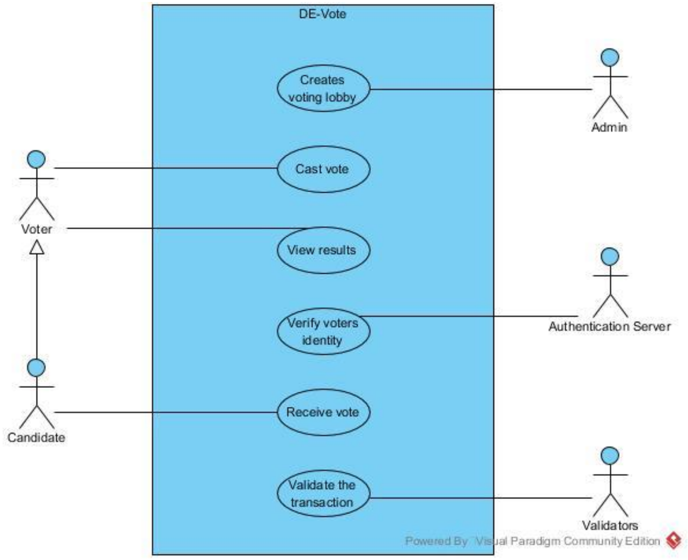


Figure 1Usecase Diagram

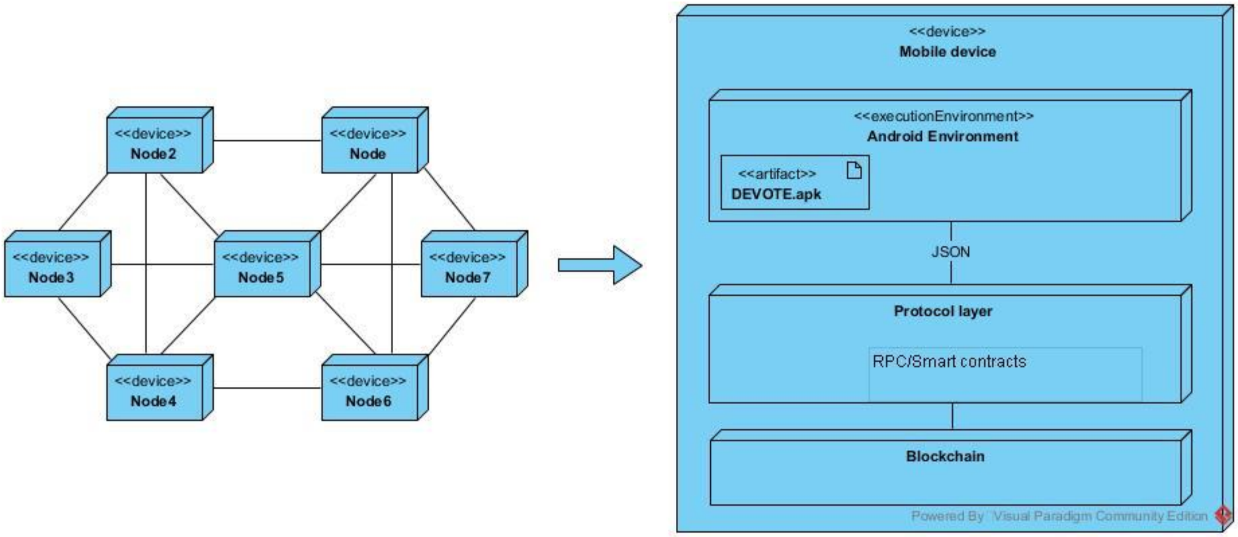


Figure 2 Deployment Daigram

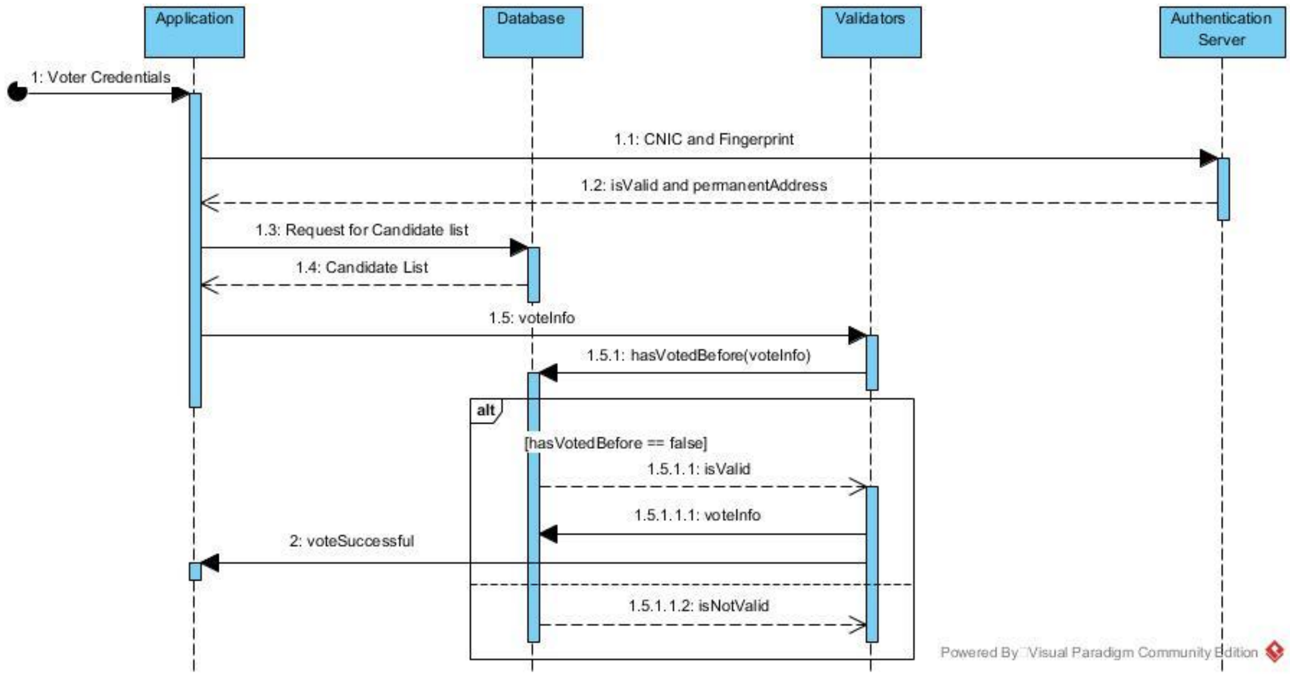

Application
Database
Validators
Authentication Server
1: Voter Credentials
1.1: CNIC and Fingerprint
1.2: isValid and permanentAddress
1.3: Request for Candidate list
1.4: Candidate List
1.5: voteInfo
1.5.1: hasVotedBefore(voteInfo)
alt
[hasVotedBefore == false]
1.5.1.1: isValid
1.5.1.1.1: voteInfo
2: voteSuccessful
1.5.1.1.2: isNotValid
Powered By Visual Paradigm Community Edition

Figure 3 Sequence Diagram

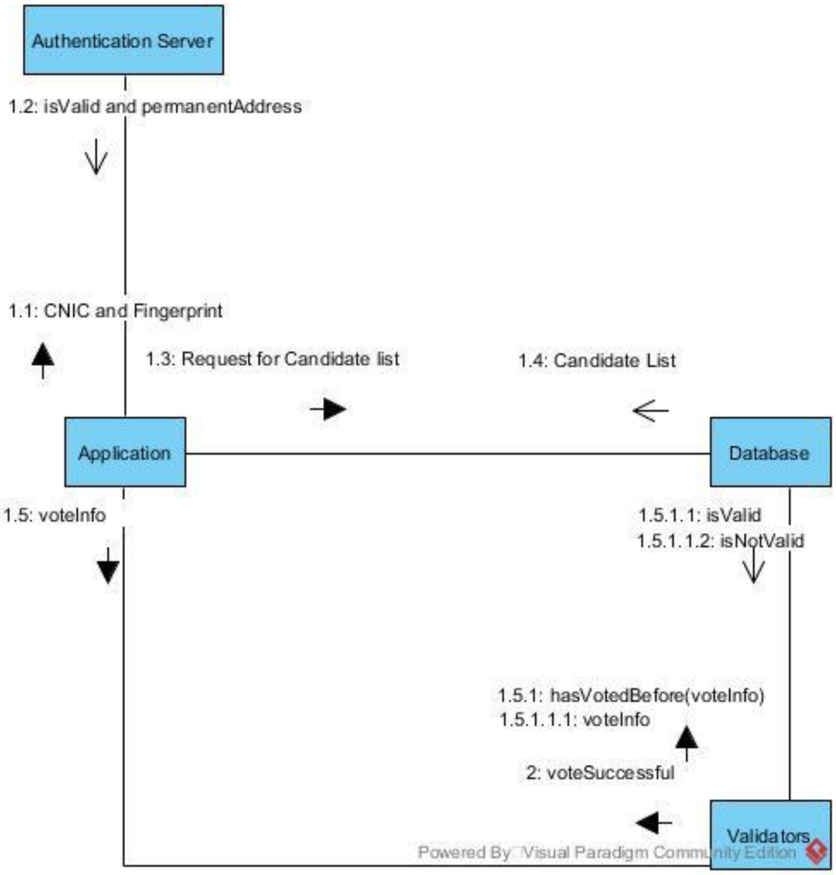

Authentication Server
1.2: isValid and permanentAddress
1.1: CNIC and Fingerprint
1.3: Request for Candidate list
1.4: Candidate List
Application
Database
1.5: voteInfo
1.5.1.1: isValid
1.5.1.1.2: isNotValid
1.5.1: hasVotedBefore(voteInfo)
1.5.1.1.1: voteInfo
2: voteSuccessful
Validators
Powered By Visual Paradigm Community Edition

Figure 4 Communication Diagram

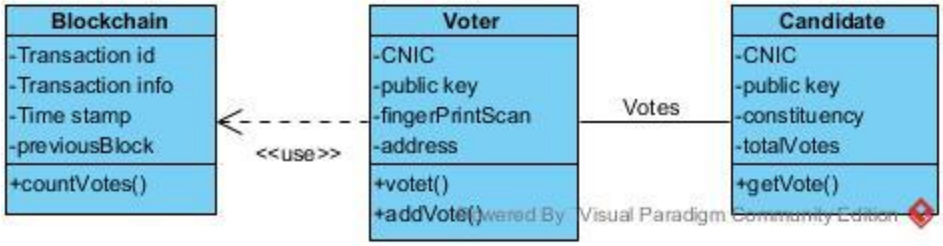


Figure 5 Class Diagram

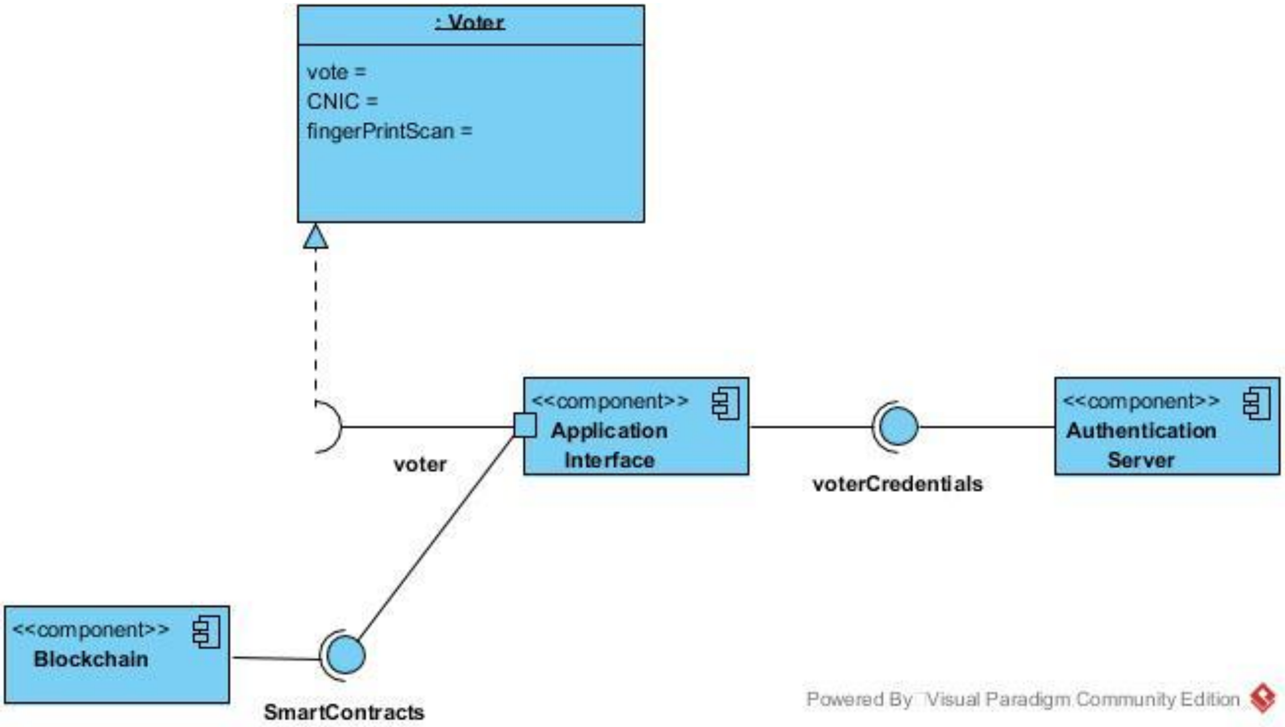


Figure 6 Component Diagram