MIST Smart Voting System (MSVS)

Software Requirements Specification Document

Group-02

Group Members

Fahmida Yasmin Rifat-	201714022
Rezwan-A-Rownok-	201714035
Shahriar Rahman Khan-	201714055
Sharmila Rahman Prithula-	201714057
Md. Zakaria Rahman-	201714117

Version: (Final) **Date: (25/01/2021)**

Contents

1 Introduction

It is a mobile application by which MIST students can give their votes easily. In this system vote casting, counting and result publishing made far more convenient than the usual system.

1.1 Purpose

The purpose of the application is to collect and analyze all assorted ideas that have come up to define the MIST central system, its requirements with respect to students', teachers' and the authority's needs. This voting technology intends to speed the counting of ballots, reduce the hardship of counting votes manually and can provide improved accessibility for disabled voters. Results can be reported and published faster. Voters save time and cost by being able to vote independently from their location.

1.2 Scope

According to the current system MIST central issues are handled through manual paper ballot or raising hand technique, where 'fake vote' or 'influenced voting' have become very common. Thus corruption is occurring. Also, their need to be a gathering of students or voters for deciding any central issue, as it is always not possible for all the voters to vote physically. Also it uses a lot of time where time is really a concerning issue for the students. Still, if we consider all the odds for the voters, we aren't hundred percent sure that our votes are counted validly and the published result is accurate.

Our proposed app-based central voting system solves all these problems. Thus central concerning issues like MIST captains selection, club members and designation selection, teacher evaluation, setting food and library reviews, opinions on current issues. Mainly a bridge to share students' needs to the concerning authority. This system ensures accuracy through the whole voting system.

1.3 Overview

MIST central voting system is a biometric enable smart voting system where users or voters will be able to cast the votes sitting in home using smart phones. It is helping to digitalize the vote counting and information storage about the casted vote using a database. To ensure security regarding vote counting, false vote or giving extra vote. The system generates QR code for all the users when entering to the system. This QR code will identify them to get access to voting. Voting events are placed by the admin checking student's reviews and authority's need and gives access to each event by checking fingerprint compared to the secured database where this voting should be done within the given time limit. This will prevent extra votes or fake voting problems. Admin can monitor the whole voting process and generates instant report and publishes result. Students can share their needs and poll based opinion process can be

placed by the system under 5the authority's permission. So, this app based voting system has many advantages over the current manual process of voting in MIST.

2 Overall Description

This section deals in details with the system environment and the functional requirements of the system. The relationship between the stakeholders and the entire system is also briefly mentioned and some non-functional requirements are highlighted at the end.

2.1 System Environment

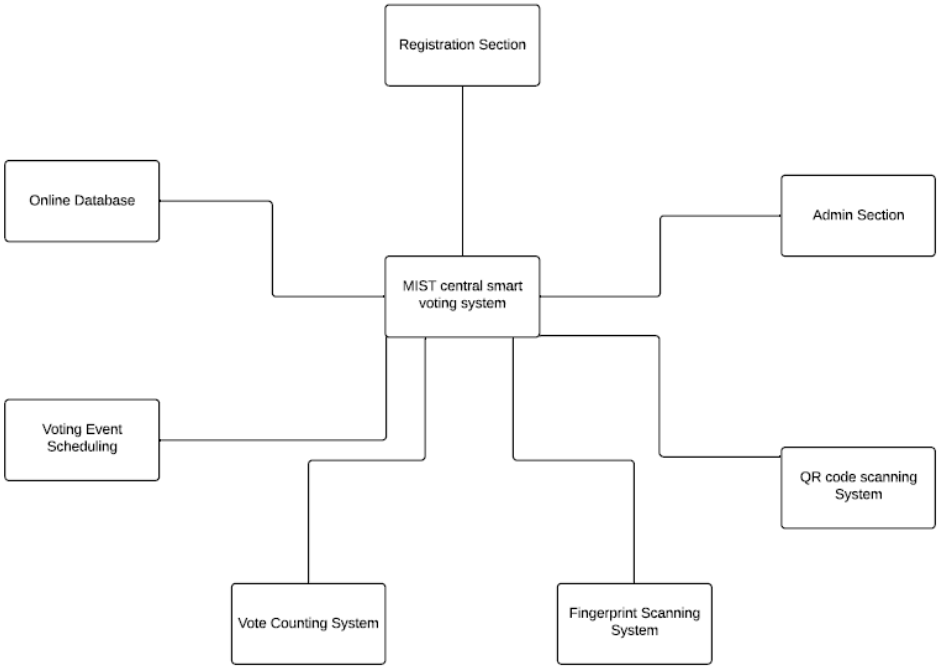


Figure 1: Contextual Diagram

MIST Smart Voting system has five main system, which are Registration sys-

tem, Voting event scheduling system, Vote counting system, Fingerprint scanning & QR code scanning system. Voters and admins need internet to access these features.

2.2 Functional Requirements Specification

2.2.1 Feature List

1. The system uses the barcode of the id card of the students to sign up and sign in.
2. Admin can create vote events, allow the students to become a candidate, choose candidates and voters, publish result, handle the history of the events.
3. Valid voters can see candidate's profile, view results after the voting events.
4. One can give vote to an event only once as it needs fingerprint to cast a vote.
5. Those who don't have fingerprint sensor,can also give vote using confirmation code that is sent via email.
6. One can edit his/her profile.

2.2.2 Use Case Diagram and Tabular Description

Scenario 1 for voter

Initial assumption

Every user is registered with proper biometric information. Every student has individual QR scan code.

Normal flow of Events in the Scenario

A voter logs in the app using QR code. He selects a voting event. Then he watches every candidate's profile of that voting event. This gives him a clear idea of the candidates. Then he gives vote where his vote is verified by fingerprint scanning. He is updated about the current situation of which candidate is winning. After finishing of the voting event, he gets to know the result within no time. Then he exits from the voting application.

What can go wrong

While verifying before giving vote, the fingerprint sensor my not work properly. Database server can be crashed and not working properly. User's QR code may not working properly because of blurred camera of the user.

Other activities

Record maybe consulted, but not edited by other voters while information is being entered.

System state on completion

After completion of the vote count, an overall result will be shown. This result will be added in database. Voting event will be discarded from database after fixed time.

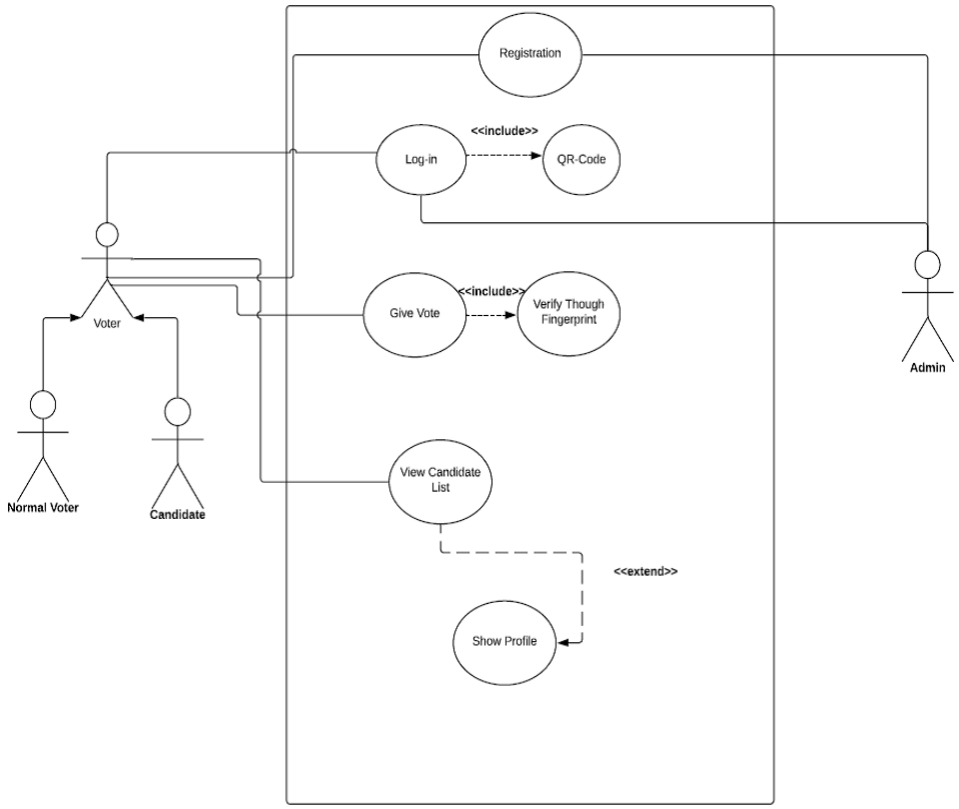


Figure 2: Use Case Diagram for Voter

Table 1: Tabular Description

Use case: Register to the system	
Primary actors	Normal voters
Secondary actors	Candidates
Pre-condition	The system has a secured database having user's biometric information.
Post-condition	Decision tree classifier
Main flow	1. The use-case is activated when the user requests it. 2. System compares QR code with the registered data and gives access to the users. 3. Normal voter can select the offered voting/selections process on current issues. 4. Voter gets access of voting through fingerprint checking. 5. The system starts a timer. 6. Voter can vote within the time limit. 7. Candidate user can update their manifestos and mottos so that the normal voters can choose among them easily(according to their needs) 8. If the normal voter is done with the voting or the timer runs out, the voting is concluded.

Scenario 2 for admin

Initial assumption

The admin has logged in the system by QR code verification and has checked the system.

Normal flow of Events in the Scenario

Admin logs in the system and can create different polls for different kinds of election. For example MIST captain selection, CR selection etc. He observes the profiles of the candidates and the voters. He sees the updates of the ongoing voting and generates report on that process. He checks the feedbacks and reviews and also sees the FAQs from the users. From the reviews and feedbacks he creates poll for general opinions. From the result of the poll of these feedbacks he creates a report and then submits to the authority.

What can go wrong?

QR may not be available for the admin in the database. If any person gives multiple votes then error message will be shown. For any undesirable issue he can cancel the nomination of the candidates. He denied the access of the voters to vote after the allocated time for voting.

Other activities

Personal information of the voters and candidates will be secured.

System state on completion

He publishes the final result of the voting event.

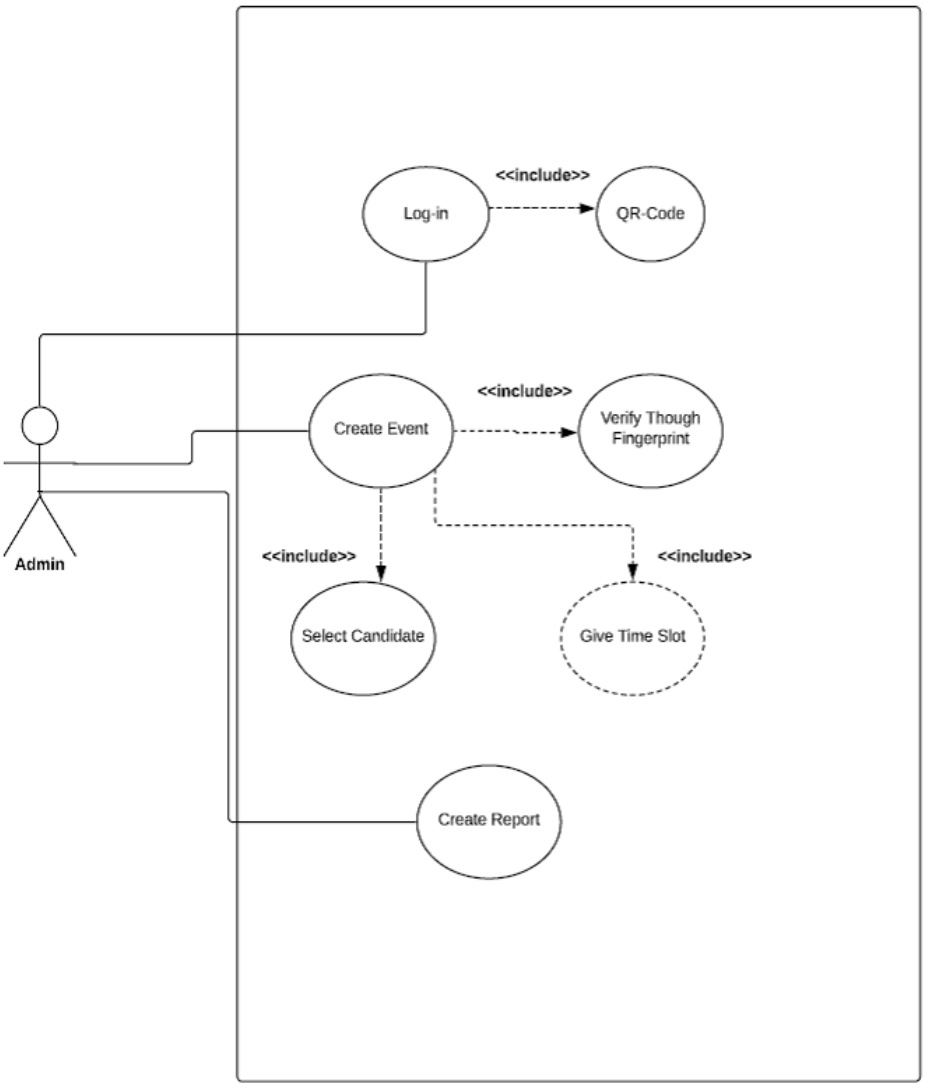


Figure 3: Use Case Diagram for Admin

Table 2: Tabular Description

Use case: Register to the system	
Primary actors	Admin
Secondary actors	The system has a secured database having admin's biometric information.
Pre-condition	The system has a secured database having user's biometric information.
Post-condition	Decision tree classifier
Main flow	1. The use-case is activated when the admin requests it. 2. System compares QR code with the registered data and gives access to the admin. 3. Admin can create different polls for different kinds of election. 4. Admin observes the voter's and candidate's profile. 5. Admin generates report on the updates of a voting process. 6. Admin checks the feedbacks and the reviews and create polls for general opinions. 7. When a voting process is done within the time limit, admin submits the results and the reports to the concerned authority.

2.3 User Characteristics

- Admin or manager should be a responsible person who do know the whole system and can send information to the concerned authority accordingly.
- MIST Students, faculty members, club members who can vote and suggest voting events for happening demand on concerning issues.

2.4 Non-functional Requirements

2.4.1 Performance Requirements

The features to be tested are listed below :

Authentication System

Criteria Assessment: Security
Testing Type: Unit Testing

Vote Event Creation

Criteria Assessment: Performance
Testing Type: Unit and Component Testing

Vote Result Publish

Criteria Assessment: Accuracy and Response
Testing Type: Unit Testing

Accessing Vote Event On Correct Date and Time

Criteria Assessment: Accuracy
Testing Type: Unit Testing

Accuracy of Fingerprint sensor

Criteria Assessment: Accuracy and Performance
Testing Type: Unit Testing

Preventing Same Voter to Cast Vote In an Event More Than One Time

Criteria Assessment: Security
Testing Type: Unit Testing

Synchronizing Fingerprint Sensor With Vote Casting

Criteria Assessment: Accuracy and Performance
Testing Type: Unit and Component Testing

2.4.2 Safety Requirements

- If the id card is not clean enough then the barcode scanner may not work properly.
- If the fingerprint sensor is not clean enough then it may not identify the user's fingerprint.
- If there is any internet issue then the app will not work.

2.4.3 Security Requirements

- If someone steals voter's id card, then he/she will be able to sign in as that voter.
- Admin has to be careful about publishing results, he should never publish the results before the event being finished.

2.4.4 Software Quality Attributes

- Security
- Performance
- Accuracy
- Response time

3 Specific Requirements

3.1 System Requirements

- Registration should be done with proper barcode information, photo and other related information.
- An online database should be maintained in order to store the information of the users.
- Internet access should be required.
- Special access must be given to admin.
- App should be designed simply using java.
- Buttons and functionalities of the app must be easily understood.
- Fingerprint sensor is required to be present in each voter and admin's phone.
- Data analysis should be needed to calculate the result.
- QR scan code is needed to log in into the app.
- Fingerprint sensor system is required to give vote, to avoid false voting.

System Requirement Specification both functional and non-functional requirements of the system is shown in table 3.

Table 3: Functional and Non-functional Requirements

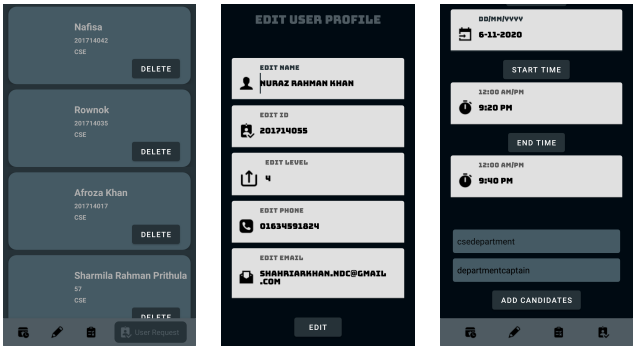
Serial	User Requirements	Type of requirement	
		Functional	Non-Functional
1	Candidate's profile should be able to see by users.	Yes	No
2	Voting event must be able to be created and monitored by admin.	Yes	No
3	User friendly outlook should be designed.	Yes	No
4	Giving votes sitting in individual's places by the voters should be implemented.	Yes	No
5	The result should be calculated fast by the app.	Yes	No
6	Security of the app should be implemented	Yes	Yes

3.2 External Interfaces

3.2.1 User Interface

One can login using the barcode scanner only if he/she is already a valid user. Otherwise he/she can register by giving some authentic information.

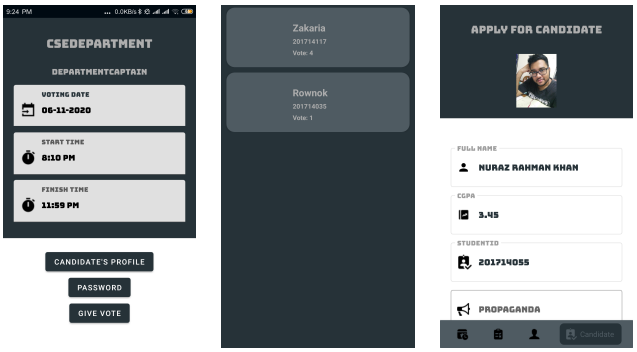
One can choose an Event and that event's particular Posts. If that event is valid then he/she will be able to see the information of the event and selected candidates' profile. Then he/she can select one candidate to give vote. After that, he/she has to give his/her fingerprint to authenticate. If the fingerprint matches then a confirmation message will be shown. So the vote must be confirmed within 5 seconds.



(a) User request (b) Edit user info (c) Create vote event

Figure 4: Screenshots of the admin module.

If a voter does not have fingerprint sensor in his/her phone, then he/she can apply for a one time confirmation code. A confirmation code will be sent to



(a) Cast vote (b) Event result (c) Apply for candidate

Figure 5: Screenshots of the user module.

his/her email address. Then he/she can give vote using that code.

Once a voting event is finished, the result can be seen by choosing the event from the Post.

If anyone wants to become a candidate for a post, then he/she has to submit an application to the admin from Candidate section. If the admin approves the application, then he/she becomes a candidate.

3.2.2 Hardware Interface

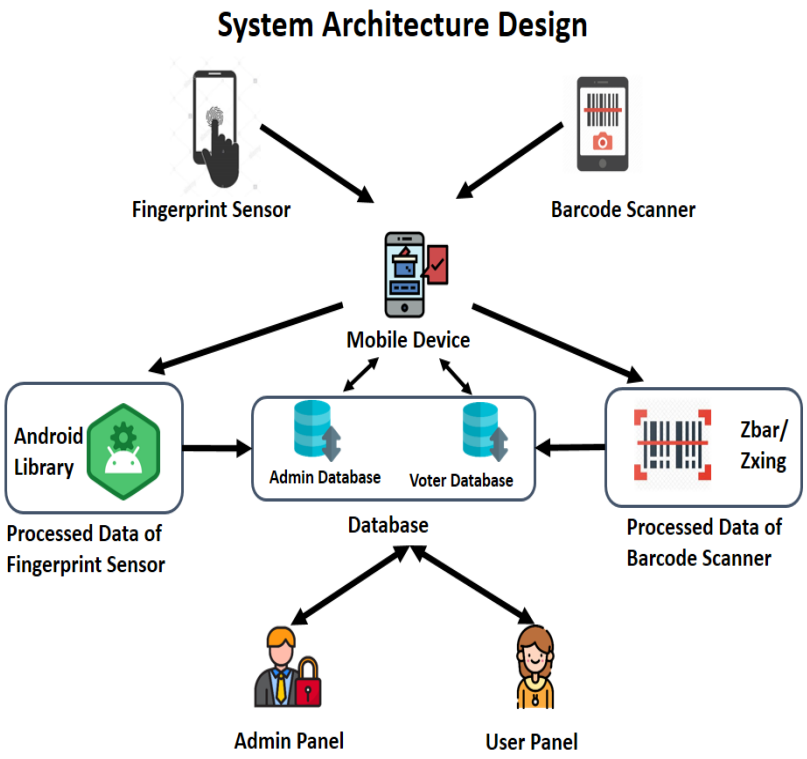


Figure 6: System Architecture

- **Barcode scanner:** A machine-readable code consisting of an array of black and white squares used for storing information for reading by the camera or smart phone.

  Here we use a camera-based barcode scanner where we use built-in video camera to record an image of the barcode and decode the information with digital imaging software. The principle is the same as CCD scanners, with light sensors replaced by the camera itself, which is composed of hundreds of rows of light sensors. In the simplest case it will convert the image to grayscale and threshold at a certain level to create a 1-bit-per-pixel ("black-n-white") image. After this the software looks for the outer edges of a barcode. If it finds something that resembles a barcode it will deskew the image so that the barcode is aligned and framed. It then quantifies the pixels down to logical barcode pixels or modules, so that the recognition algorithm has a "levelled field" of relevant pixels to work on. The software will then run through a range of different barcode symbology detectors to

see which symbology it is, unless it's locked to only one. When that's done it tries to interpret the barcode and convert it to usually human-readable data.

- **Finger Print Sensor:** A fingerprint scanner is a type of technology that identifies and authenticates the fingerprints of an individual in order to grant or deny access to a computer system or a physical facility. It is a type of biometric security technology that utilizes the combination of hardware and software techniques to identify the fingerprint scans of an individual. The most commonly found type of fingerprint scanner used

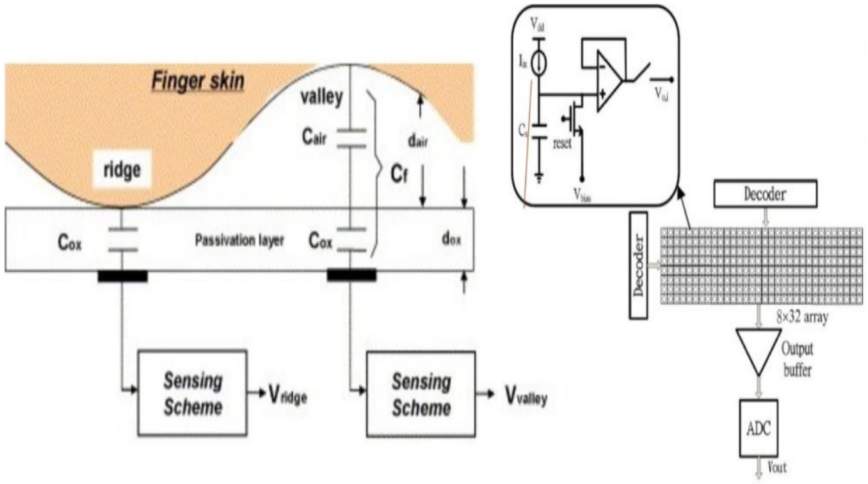


Figure 7: Fingerprint Scanner

today is the capacitive scanner. Instead of creating a traditional image of a fingerprint, capacitive fingerprint scanners use arrays tiny capacitor circuits to collect data about a fingerprint. As capacitors can store electrical charge, connecting them up to conductive plates on the surface of the scanner allows them to be used to track the details of a fingerprint. The charge stored in the capacitor will be changed slightly when a finger's ridge is placed over the conductive plates, while an air gap will leave the charge at the capacitor relatively unchanged. An op-amp integrator circuit is used to track these changes, which can then be recorded by an analogue-to-digital converter.Once captured, this digital data can be analyzed to look for distinctive and unique fingerprint attributes, which can be saved for comparison at a later date. Creating a large enough array of these capacitors, typically hundreds if not thousands in a single scanner, allows for a highly detailed image of the ridges and valleys of a fingerprint to be created from nothing more than electrical signals. Just like the optical scanner, more capacitors results in a higher resolution scanner, increasing the level of security, up to a certain point.

Of course, this information needs to be kept secure on your device and saved well away from code that could compromise it. Rather than uploading this user data online, ARM processors can keep this information securely on the physical chip using its Trusted Execution Environment (TEE) based TrustZone technology. This secure area is also used for other cryptographic processes and to communicate directly with secure hardware platforms, such as a fingerprint scanner, to prevent any software snooping. Approved pieces of none personal information, such as a password key, can only be accessed by applications using the TEE client APIs.

3.2.3 Software Interface

To sign in with barcode scanning and give vote with fingerprint sensor, a specific android version is required. Minimum andfroid SDK version >= 23 is required to access these features.

3.2.4 Communication Interface

Wireless connection is needed to communicate with Firebase subsystem.

3.3 Design Constraints

Specify design constraints that can be imposed by other standards, hardware limitations, etc.

3.3.1 Hardware Constraints

- As we have to sign in using barcode scan, so the device needs high resolution camera to scan the barcode properly.
- The main feature of our system is casting vote using device's fingerprint sensor. So the device must have a fingerprint sensor with great accuracy.

3.3.2 Software Constraints

Our Software part has a constraint as it doesn't run in all android devices. It crashes in the android devices which have SDK version less than 23. So the android SDK version must be greater than or equal to 23.