



COMSATS UNIVERSITY ISLAMABAD

DEPARTMENT OF COMPUTER SCIENCE

Project Name

Android Based E-Voting System

By

Muhammad Nadeem CIIT/FA20-BSE-035

Muhammad Usman CIIT/FA20-BSE-055

Supervisor

Supervisor Name

Sir Nashit Ali

Bachelor of Science in Computer Science (2020-2024)



**COMSATS University Islamabad Islamabad,
Pakistan**

Project Name

Android Based E-Voting System

By

Muhammad Nadeem CIIT/FA20-BSE-035

Muhammad Usman CIIT/FA20-BSE-055

Supervisor

Supervisor

Name Sir Nashit

Ali

Bachelor of Science in Computer Science (2020-2024)

The candidate confirms that the work submitted is their own and appropriate credit has been given where reference has been made to the work of others.

DECLARATION

We hereby declare that this software, neither whole nor as a part has been copied out from any source. It is further declared that we have developed this software and accompanied report entirely on the basis of our personal efforts. If any part of this project is proved to be copied out from any source or found to be reproduction of some other. We will stand by the consequences. No Portion of the work presented has been submitted of any application for any other degree or qualification of this or any other university or institute of learning.

Student Name 1

Student Name 2

CERTIFICATE OF APPROVAL

It is to certify that the final year project of BS (SE) “Android Based E-Voting System” was developed by **Muhammad Nadeem (CIIT/FA20-BSE-035)** and **Muhammad Usman (CIIT/FA20-BSE-055)** under the supervision of “Sir Nashit Ali” and that in his opinion; it is fully adequate, in scope and quality for the degree of Bachelors of Science in Computer Sciences.

Supervisor

External Examiner

Head of Department

(Department of Computer Science)

Executive Summary

The Android-based e-voting system is a secure and efficient way to cast votes electronically.

It uses Android devices to enable voters to select candidates or options easily.

The system ensures the integrity and secrecy of votes through end-to-end encryption and secure authentication.

Votes are stored and counted electronically, reducing the risk of human error.

The system provides real-time results and can increase voter turnout, especially among remote or disabled voters.

Acknowledgement

All praise is to Almighty Allah who bestowed upon us a minute portion of His boundless knowledge by virtue of which we were able to accomplish this challenging task.

We are greatly indebted to our project supervisor “Sir Nashit Ali”. Without his personal supervision, advice and valuable guidance, completion of this project would have been doubtful. We are deeply indebted to him for their encouragement and continual help during this work.

And we are also thankful to our parents and family who have been a constant source of encouragement for us and brought us the values of honesty & hard work.

Student Name 1

Student Name 2

Abbreviations

SRS	Software Require Specification
PC	Personal Computer
APIs	Application Programming Interfaces
PKI	Public Key Infrastructure

Table of Contents

1	INTRODUCTION	4
1.1	System Introduction	4
1.2	Background of the System	4
1.3	Objectives of the System	4
1.4	Significance of the System	5
2	REQUIREMENT SPECIFICATIONS	6
2.1	Product Scope	6
2.2	Product Description	6
2.2.1	Product Perspective	6
2.2.2	Product Functionality	6
2.2.3	Users and Characteristics	7
2.2.4	Operating Environment	8
2.3	Specific Requirements	8
2.3.1	Functional Requirements	8
2.3.2	Behavioral Requirements	10
2.3.3	External Interface Requirements	12
2.3.4	Other Interfaces (if any)	23
2.4	Non-functional Requirements	25
2.4.1	Performance Requirements	25
2.4.2	Safety and Security Requirements	26
2.4.3	Software Quality Attributes	26
3	DESIGN SPECIFICATIONS	28
3.1	Explanation	28

3.2	Composite Viewpoint.....	29
3.3	Logical Viewpoint.....	30
3.4	Information Viewpoint.....	31
3.5	Interaction Viewpoint.....	32
3.6	State Dynamics Viewpoint.....	32
3.7	Algorithmic Viewpoint.....	32
3.6	State Dynamics Viewpoint.....	34
4	DEVELOPMENT AND TOOLS.....	36
4.1	Define.....	36
4.2	Development Plan.....	36
4.3	Development Tools.....	37
4.4	Conclusion and Future Work/Extensions.....	37
5	QUALITY ASSURANCE.....	38
5.1	Define.....	38
5.2	Traceability Matrix.....	38
5.3	Test Plan.....	40
6	USER MANUAL.....	42
6.1	Introduction.....	42
6.2	Hardware/Software Requirements for the System.....	42
6.3	Installation guide for Application.....	42
6.4	Operating Manual.....	4

1 INTRODUCTION

Welcome to the forthcoming of voting. We are about to revolutionize elections with our Android based solution which employs a mobile app to simplify the process. By using facial recognition and fingerprint authentication, we make sure that only valid votes are casted thus ensuring secure voting. Our main aim is to increase voter participation rates by providing convenience together with real-time results through this application. Additionally, it also enables users to verify their votes and in turn enhances transparency as well as accountability.

1.1 System Introduction

The main objective of this concept which is still at its early stages but utilizes Android platform is centered around revolutionizing voting making it easy, safe and accessible through a mobile application. With surveys voted on through facial recognition supported by fingerprint authentication for security purposes among others where users can cast their votes; our system seeks to achieve just that. We want more people to take part in elections hence our desire for convenience plus immediate feedback brought about by these features being available on an app. Equally important, the ability of individuals validating both their vote and election thereof fosters openness while demanding responsibility.

1.2 Background of the System

This project deals with an android-based e-voting solution aimed at solving challenges experienced during traditional paper-based voting systems. It offers user-friendliness, security and accessibility among others; all combined together in this type of application development process. This ensures that even before one understand what exactly takes place when they use such apps like ours; there would be no need for them worry much about how safe or convenient it could be since everything has been taken care of already. For example there is introduction of face identification recognition which makes things more secure than ever before apart from just having password protection alone hence making them very reliable too.

1.3 Objectives of the System

The objectives of the Android-based e-voting system are as follows:

- To create a user-friendly mobile application for eligible voters to cast their votes conveniently.
- To enhance the security of the voting process by implementing advanced authentication and encryption mechanisms.
- To provide real-time results and analytics for election administrators.
- To reduce the administrative burden of managing paper ballots and manual vote counting.
- To improve accessibility for voters with disabilities through inclusive design features.
- To support multiple types of elections, including government, academic, and corporate elections

1.4 Significance of the System

Accessibility and Convenience: E-voting allows citizens to cast their votes from anywhere, reducing the barriers posed by physical distance or mobility issues. This accessibility promotes higher voter turnout and engagement.

Efficiency and Speed: Electronic voting systems can streamline the voting process, providing quick and accurate results. This can lead to faster election outcomes, minimizing delays and uncertainties associated with manual vote counting.

Reduced Errors and Fraud: E-voting systems are designed to minimize human errors in the counting and recording of votes, potentially reducing instances of fraud or miscounting that can occur in traditional paper-based systems.

Enhanced Transparency: Some e-voting systems offer greater transparency by allowing voters to verify their votes and ensuring that votes are recorded accurately. Block chain technology, for instance, can enable a transparent and immutable record of votes.

Cost Efficiency: Over time, e-voting systems can potentially reduce the overall costs associated with conducting elections, including expenses related to printing ballots, staffing polling stations, and manually counting votes.

Environmental Impact: Moving away from paper-based voting towards electronic systems can have positive environmental implications by reducing paper usage and waste.

2 REQUIREMENT SPECIFICATIONS

The Flutter-developed Android-based e-voting system must prioritize user verification. To do this, it should offer secure registration methods such as email/password, face detection or fingerprint authentication. It should provide an intuitive interface for casting votes that can show clear information about candidates or issues and allow easy navigation through the voting process. User data security and integrity of votes casted require strong measures which include encryption techniques and biometric authentication.

2.1 Product Scope

The Android e-voting system made with Flutter needs to focus on making voter registration smooth and secure. It should use strong security measures like encryption and multi-factor authentication to protect against hacking. The interface should be easy to use on any device, and biometric verification can ensure that only eligible voters can cast their votes securely. The system should accurately record votes and keep a clear record of the process for transparency. It should also be accessible to people with disabilities. Handling busy times and following laws about elections and data privacy are important, as is keeping the system up to date with maintenance and support.

2.2 Product Description

2.2.1 Product Perspective

The electronic voting system should work seamlessly alongside current election tools like voter databases as well result calculators; therefore it has incorporate strong safety measures such encrypting votes themselves among other ways ensuring anonymity throughout the process while still remaining easy everyone including those having impairments besides being capable handling many simultaneous connections without crashing down at any given point time thus abiding all rules governing conduct during an election period together with protecting personal information according regular updates provided along technical assistance required keep things running smoothly especially when there are changes made within systems controlled by different administrators who may not necessarily share same level skills required manage them effectively however need arises so much more than what is normally expected from such a setup.

2.2.2 Product Functionality

Voter Authentication: Systems typically require secure login procedures to authenticate eligible voters, often using unique identifiers like voter IDs, passwords, or biometric data.

Ballot Casting: Users can securely cast their votes electronically through an intuitive interface, choosing candidates or options based on the election type.

Ballot Verification: Voters may have the option to review and verify their choices before submitting their ballots to ensure accuracy.

Security Measures: Robust security protocols, including encryption and firewalls, safeguard against unauthorized access, tampering, or hacking attempts.

Audit Trail: Systems often generate comprehensive audit trails, recording every action taken within the system to ensure transparency and accountability.

Tabulation and Tallying: Automated tabulation processes calculate votes accurately and promptly, minimizing manual errors and delivering quick results.

Accessibility Features: E-voting systems should cater to diverse user needs, providing accessibility features for voters with disabilities or language barriers.

Scalability and Reliability: The system should handle large-scale elections smoothly, ensuring reliability even with high traffic and a large number of users.

Anonymity and Privacy: Systems must maintain voter anonymity and protect personal information in compliance with privacy regulations.

User Support and Training: Provision of user support, training materials, and assistance to ensure voters and administrators can use the system effectively.

Testing and Certification: Rigorous testing and certification processes verify the system's integrity, security, and functionality before deployment for an election.

2.2.3 Features of E-Voting Systems:

Accessibility: This characteristic allows people with all abilities to use the product, for example they could vote from home or a mobile device.

Security: Implement strong security measures so that no person can interfere with the election process; also make sure votes are counted as intended but remain secret too.

Accuracy: Ensure that every vote is recorded accurately without any mistakes being made during counting them up later on.

Transparency: Permit stakeholders scrutinize this activity to confirm its fairness and credibility.

Auditability: Make it possible for anyone who wants to check whether things were done right or not by being able to follow each step taken during an election procedure until everything gets completed successfully thereby ensuring accountability throughout.

Usability: Design user-friendly interfaces which can be used easily

Scalability: The ability to handle a large number of votes without performance or security being compromised.

Resilience: The ability to resist technical failures or attacks while still maintaining the integrity of an election.

2.2.4 Operating Environment

Security Measures: The system should have strong security protocols in place to prevent hacking, tampering, or unauthorized access. This may include encryption, multi-factor authentication, and regular security audits among others.

Reliability and Stability: There should be no downtimes in the functioning of the system; it must work consistently without glitches. This requires reliable hardware, software, and network infrastructure.

Accessibility: The system needs to be user-friendly so that all eligible voters can use it regardless of their physical abilities. It should also support different devices and offer multiple languages.

Scalability: The system ought to cater for different numbers of users during peak times without losing its efficiency. It should have the capacity to scale well with many users.

Transparency and Auditability: There has to be transparency throughout the process such that votes can be audited for accuracy and fairness. A verifiable record such as a paper trail should be provided for this purpose too.

Regulatory Compliance: All applicable laws, regulations, standards regarding elections and data protection must be followed by this system.

Backup and Recovery: Ensure there is a strong backup system in place so that data is not lost when systems fail. Also have recovery plans for any issues arising from other contingencies.

2.3 Specific Requirements

2.3.1 Functional

Requirements

Administrator module:

Admin Login through facial recognition authentication,

Adding Voters Data,

Editing/deleting candidates' particulars,

Creating a poll/election,

Providing database having voters' information,

Editing/deleting voter's information in the database,

Providing database having total number of casted votes,

Reporting or analyzing statistically total casted votes and voting ratio etc.in graphical view,

Producing detailed report demographically based on male/female voting ratio among others

Voter Module:

Connecting to the system through Android application,

Signing up to cast vote o Logging in for voting with facial recognition authentication,

Casting a vote in favor of a candidate/subject

Voting Module:

Viewing results after polling time,

Reporting total casted votes and voting ratio,

Reporting based on demographics and male/female voting ratio among others

Publishing Results.

Real-time Updates:

- Showcasing real-time updates about election progress like number of votes cast so far, provisional results etc.

Data Backup and Recovery:

- Regularly backing up voting data and implementing reliable data recovery mechanism in case of system failure.

2.3.2 Behavioral Requirements

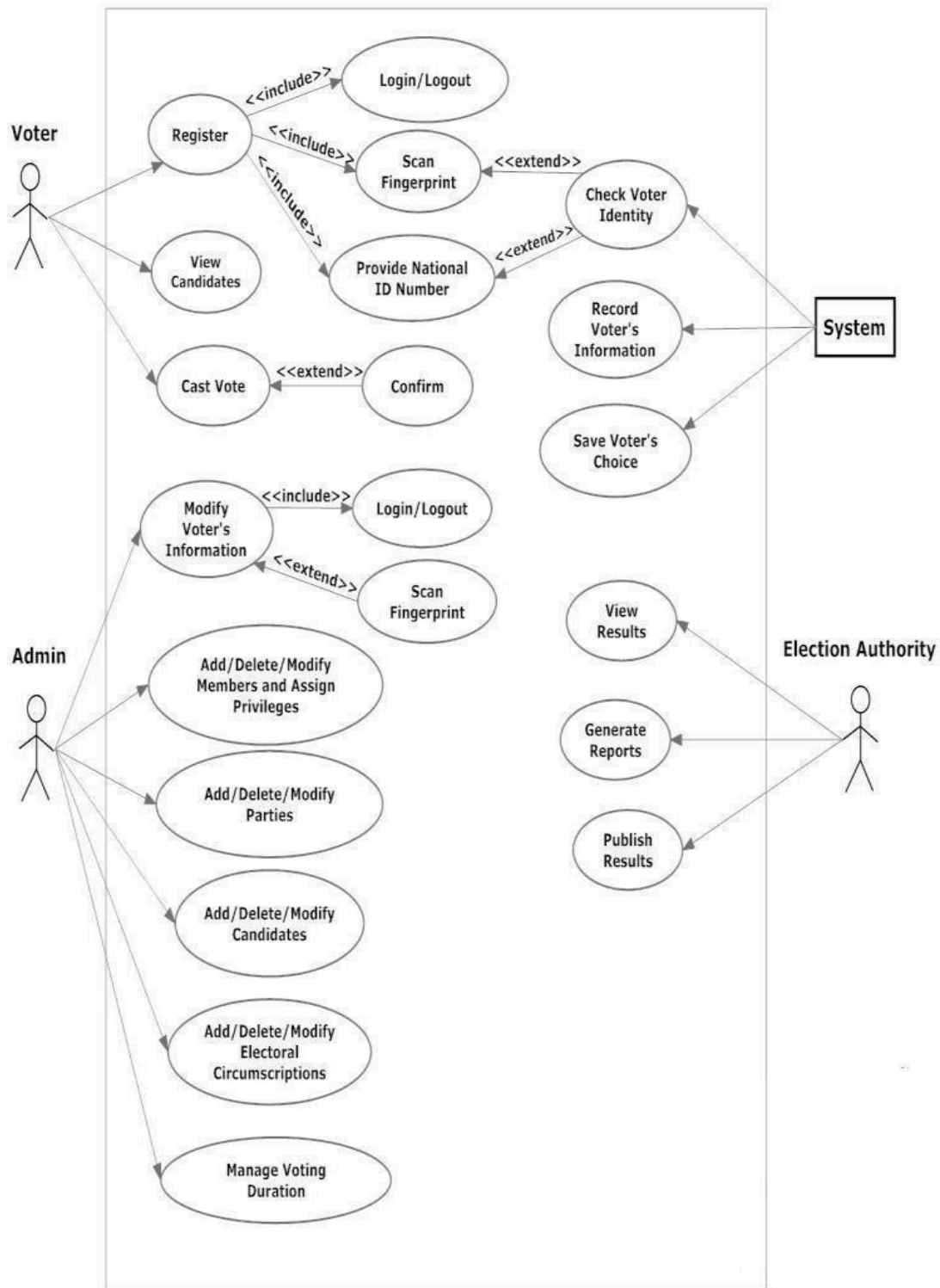


Figure 1: Use Case Diagram

User Verification and Authorization:

The procedure should confirm the identity of individuals correctly, allowing only registered voters.

Monitoring & Reporting in Real-Time:

Officials must be given tools that can enable them to monitor the number of people voting alongside other related activities.

Generate detailed reports on results achieved after finishing voting process

Feedback and Support for Users:

Should have a system where users can report any problem or give their comments concerning the system.

Create quick responsive support systems to address user questions as well as technical challenges.

2.3.3 External Interface Requirements

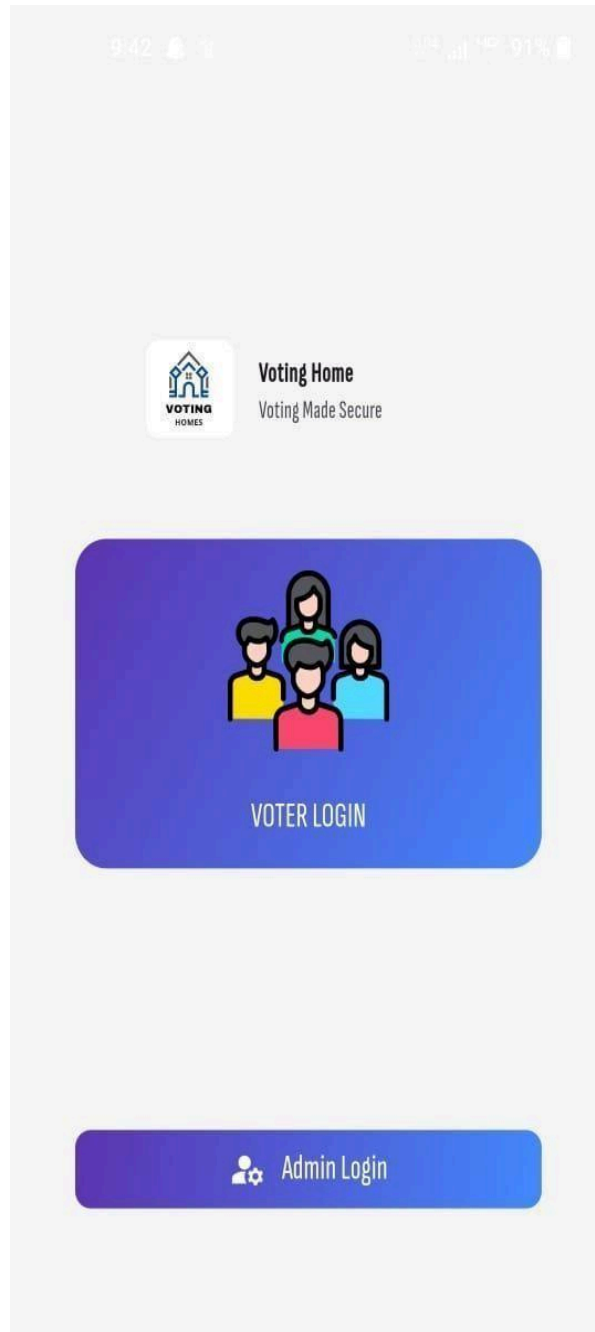


Figure 2: Home Page

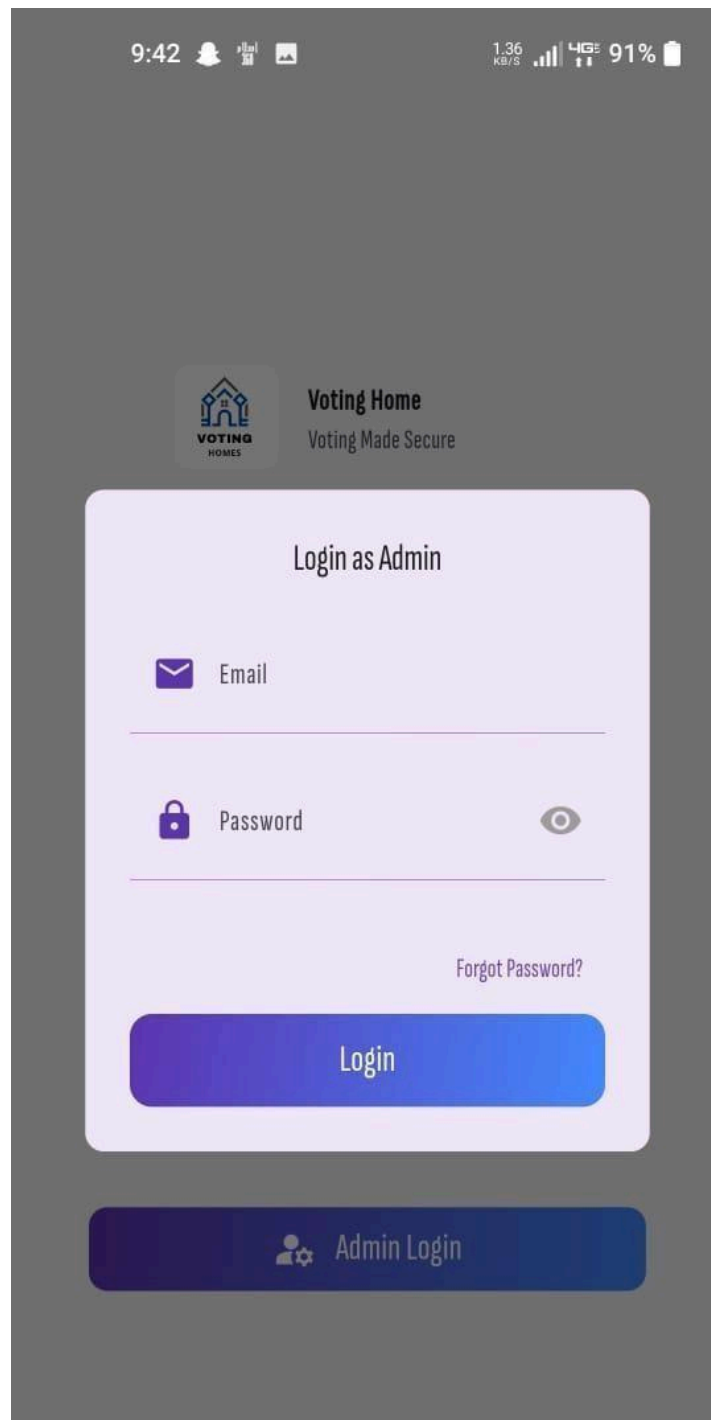


Figure 3: Admin Login Page

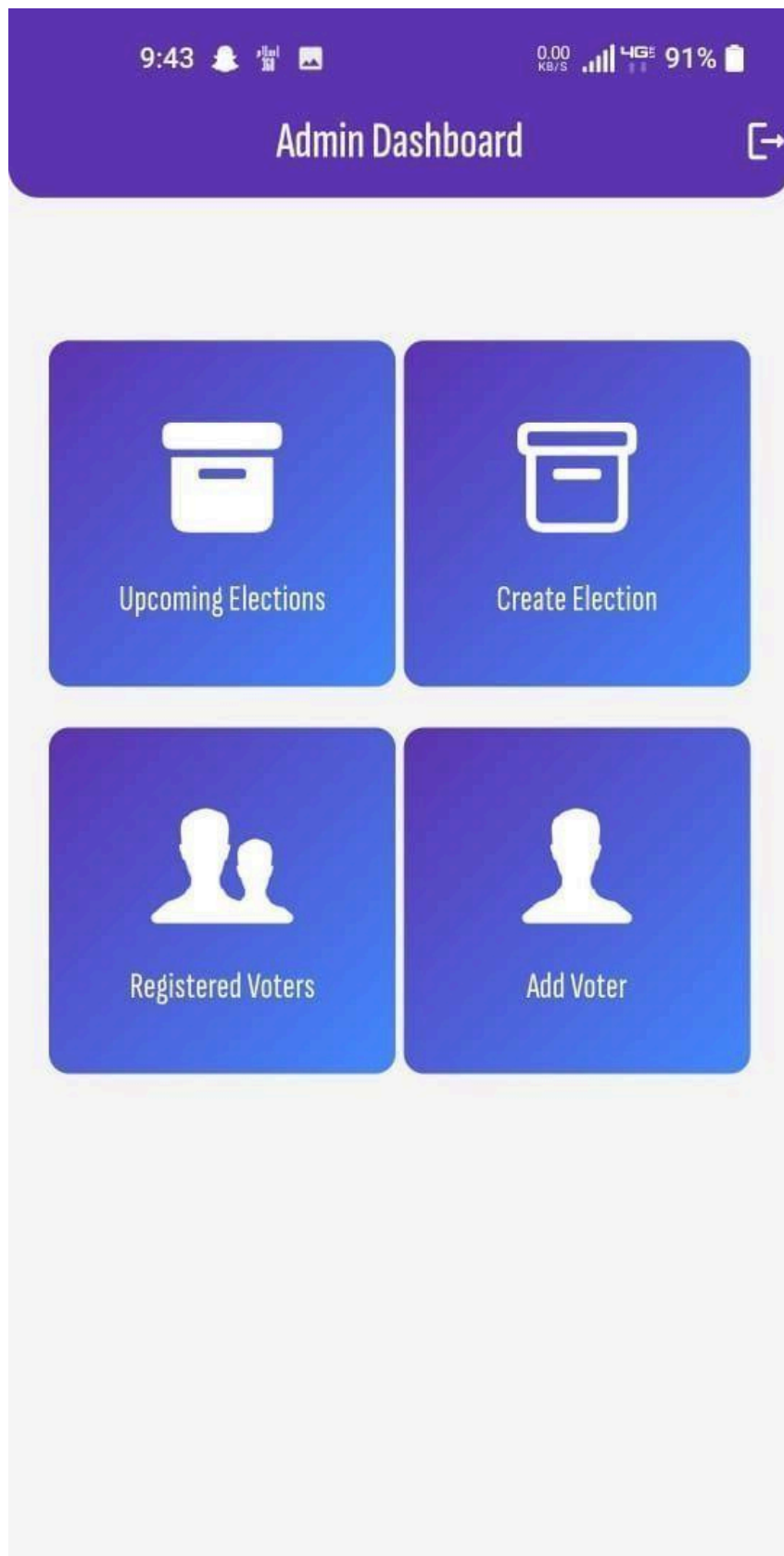


Figure 4: Admin Dashboard

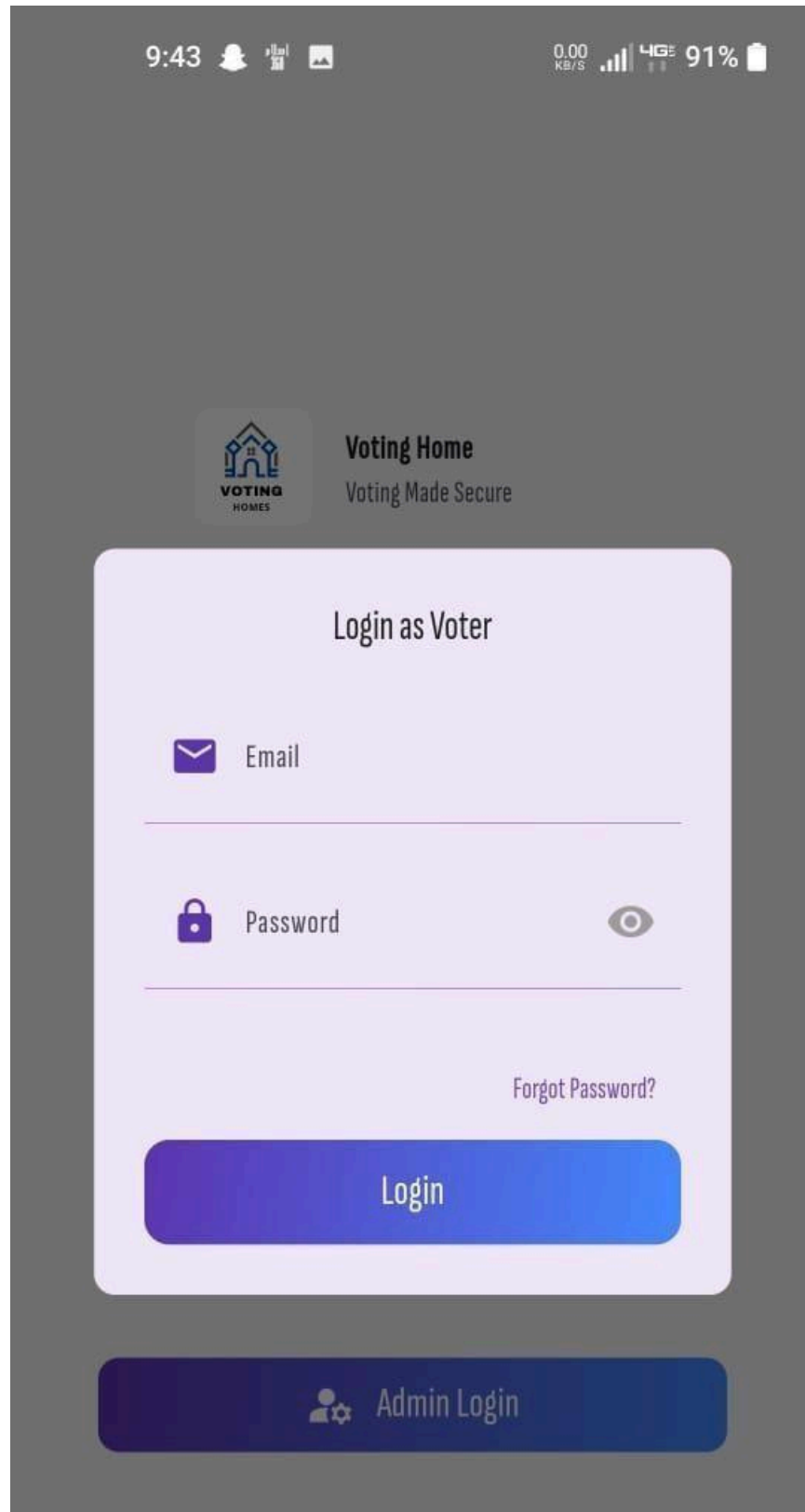


Figure 5: Voter Login Page



Admin Registration

Enter Admin Details



Email



Password



User Name



Position






Designation




Department

SUBMIT


9:43




0.00 KB/s



91%






Add Election


Election Title

Election Type



Select Date:

2024-5-1



Select Time:

21:43

Candidates:

Candidate Name

Party

Add Candidate

Save Election

Figure 7: Add Election



Figure 8: Register User

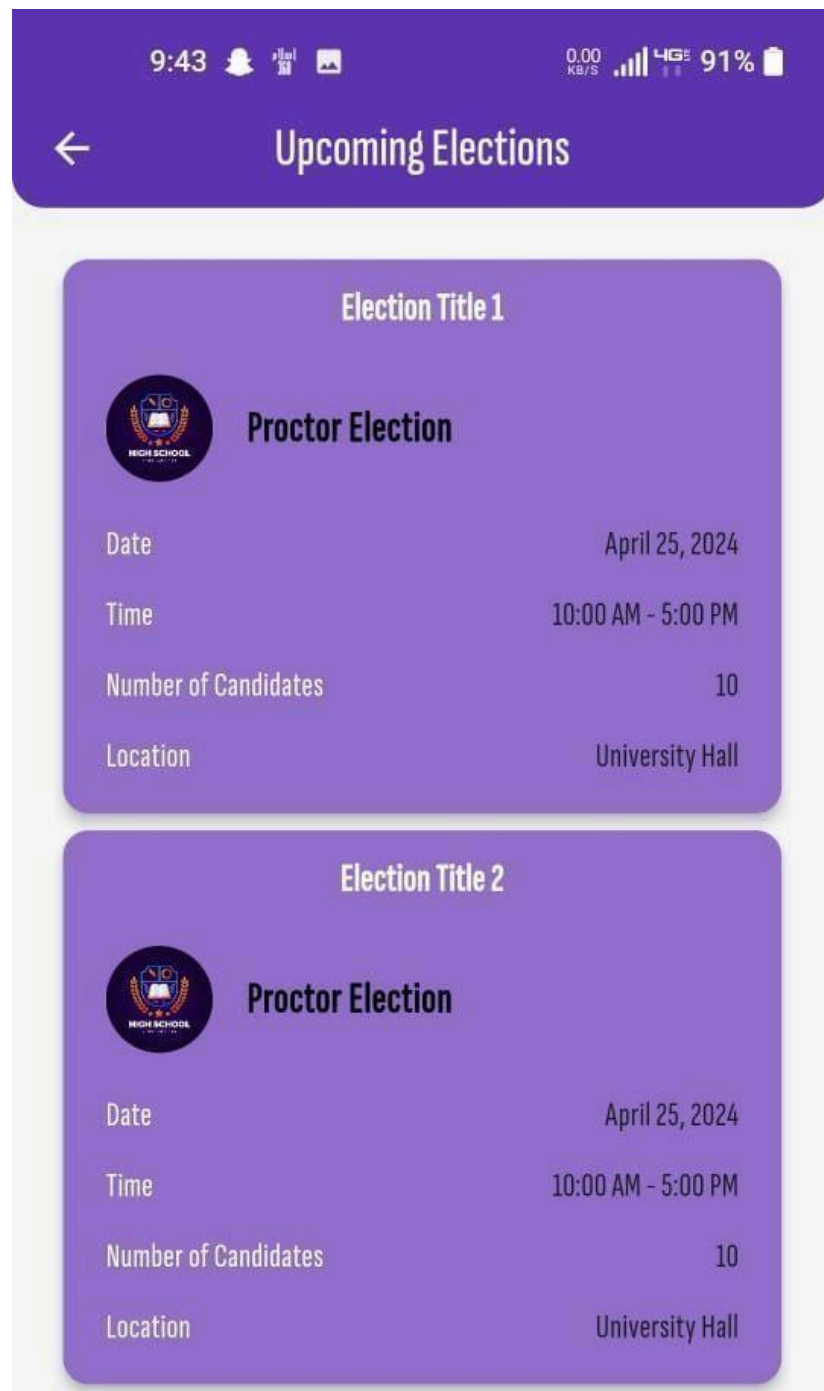







Figure 9: Upcoming Election





Figure 10: Election Details






Figure 11: Voter Home Screen


9:44    0.00 KB/S  91% 

 Voter Profile

 User Name

 Email

 Password 

 CNIC

Register User

Figure 12: Voter Profile

2.3.3.1 User Interface

Voter registration interface:

The system should work with a secure voter registration database that ensures the validity and confirmation of voter details during the process of registering.

Biometric verification systems:

It should be possible to combine with biometric verification systems which could be fingerprint or retina scans for more validation on voter identities.

Government identity databases:

These should connect with government identity databases such as national identification systems in order to crosscheck if voters are eligible.

2.3.4 Other Interfaces (if

any) Hardware

Interfaces:

Voting Machines or Devices:

Describe the hardware specifications and standards which will guide the production of voting machines or devices used by voters in casting their electronic votes.

Specify compatibility requirements so that e-voting software may function well with designated hardware.

Smart Cards or Tokens:

If smart cards or tokens are used by this system for additional authentication, outline necessary interfaces needed to support such devices at hardware level.

Define communication standards between e-voting software and smart card readers.

Network Infrastructure:

Specify server types required; routers; switches etcetera together with other network equipment necessary for supporting e-votes through an efficient network infrastructure.

Software Interfaces:**Voter Registration System:**

- Define software interfaces with voter registration systems to validate and verify voter information during the registration process.
- Specify data exchange formats and communication protocols for seamless integration

Biometric Verification Software:

- Interface with biometric verification software for identity authentication.
- Define communication protocols for exchanging biometric data securely.

Election Commission Database:

- Establish software interfaces with the election commission database to obtain official candidate lists, electoral district information, and other relevant data for ballot configuration.
- Specify data exchange formats and synchronization intervals.

External Authentication Services:

- Integrate with external authentication services (e.g., multi-factor authentication providers) to enhance the security of user logins and access.
- Define communication standards for secure authentication processes.

Notification Services:

- Interface with notification services (e.g., email or SMS) to notify registered voters about upcoming elections, polling locations, and other relevant information.
- Specify messaging formats and delivery mechanisms.

Security Software:

- Interface with security software for continuous monitoring, threat detection, and prevention of unauthorized access.
- Define communication protocols for real-time security alerts.

Mobile App Interfaces (if applicable):

- If the e-voting system has a mobile application, define software interfaces for seamless communication between the mobile app and the main system.
- Specify data exchange formats and synchronization methods.

Communication Interfaces:**Network Protocols:**

- Specify the network protocols used for communication between the various components of the e-voting system, such as TCP/IP, HTTP, or other secure communication protocols.

Web Services/APIs:

- Define web services or APIs (Application Programming Interfaces) for seamless communication between different modules of the e-voting system, ensuring interoperability.

Encryption Standards:

- Specify encryption standards (e.g., SSL/TLS) for securing data transmission between clients (voters), servers, and other system components, ensuring the confidentiality and integrity of data.

Message Queues (if applicable):

- If the system uses message queuing for asynchronous communication between components, define the message queue protocols and standards.

Wireless Communication (if applicable):

- Specify communication standards for wireless technologies if the e-voting system supports mobile voting or any wireless components.

Data Formats:

- Define standard data formats (e.g., JSON, XML) for data exchange between different components of the system to ensure consistency in information flow.

Push Notifications (if applicable):

- If the system supports push notifications to voters, define the communication protocols and standards for delivering notifications securely.

Public Key Infrastructure (PKI):

- Specify how the system will utilize PKI for secure key management, digital signatures, and encryption to ensure the authenticity and integrity of data.

2.4 Non-functional Requirements

2.4.1 Performance

Requirements Scalability:

- For the purpose of elections, this system should be able to scale appropriately when there are changes in number of users at different stages.
- It also needs to have scalability because it can handle high traffic during voting periods without becoming unresponsive.

Response Time:

- To voters who interact with the system, they should not experience delays hence response time should be minimized.
- The software algorithms and network configurations must be optimized for quick processing of votes and providing real-time updates.

Concurrency Handling:

- Even when many people are using the system concurrently, it should still remain responsive and stable therefore include mechanisms for handling concurrent user access.

Reliability and Availability:

- Failover mechanisms as well as redundancy features ought to be included within the design so as to enhance reliability.
- During critical phases of election ensure that the service cannot fail easily by making sure that there is high availability across all components involved in service delivery.

Data Transfer Efficiency:

- Optimize transfer speed between client computers or devices connected through a network which will help reduce load on networks hence improving efficiency throughout whole infrastructure performance optimization process.
- Efficiency in data exchange formats can also help optimize data transfer between clients and servers thus enhancing overall system performance.

2.4.2 Safety and Security Requirements

Authentication and Authorization: There is need for strong verification systems that can confirm whether a person is registered as a voter or not. This may require biometric identification methods among others like secure login credentials unique identifiers etcetera

Resources scaling: Make sure that the base facilities are capable of expanding to take more loads without any drop in efficiency. This is important for ensuring the performance does not reduce even with an increase in workload.

Modularity: The design should have separate modules which can be updated, fixed, or improved independently without affecting other parts.

Documentation: Documents should be provided to cover all aspects of the system.

Standards Compliance: Follow appropriate standards for data formats, communication protocols and security so as to work with other systems outside and in the future.

Integration with External Systems: Make it easy for this service to work together with other systems which might include voter registration databases and election commission systems among others.

Performance:

Response Time: The speed at which a system responds should be optimized for smoothness and efficiency of voting experience.

Throughput: Ensure that the system can handle many concurrent transactions or votes without compromising on performance level.

2.4.3 Software Quality

Attributes Security:

- **Confidentiality:** Ensure that voter information and ballots remain confidential, employing robust encryption and access controls.
- **Integrity:** Guarantee the integrity of the voting process, preventing tampering with ballots or election results.

- **Authentication:** Implement secure user authentication mechanisms to verify the identity of voters and prevent unauthorized access.

Reliability:

- **Fault Tolerance:** Design the system to tolerate and recover from hardware or software failures, ensuring uninterrupted service during the voting process.
- **Availability:** Ensure high availability, minimizing downtime and providing consistent access to the system throughout the election period.

Usability:

- **User Interface Design:** Create an intuitive and user-friendly interface that accommodates voters with varying levels of technical expertise.
- **Accessibility:** Ensure the system is accessible to voters with disabilities, meeting relevant accessibility standards.

Scalability:

- **Performance Scaling:** Design the system to handle a scalable number of users and votes, especially during peak voting periods.
- **Resource Scaling:** Ensure that the underlying infrastructure can scale to accommodate increased loads without degradation in performance.

Maintainability:

- **Modularity:** Design the system with modular components, making it easier to update, maintain, and enhance without affecting the entire system.
- **Documentation:** Provide comprehensive documentation for system components, APIs, and configurations to facilitate maintenance and troubleshooting.

Interoperability:

- **Standards Compliance:** Adhere to relevant standards for data formats, communication protocols, and security, ensuring interoperability with external systems and future technologies.
- **Integration with External Systems:** Facilitate seamless integration with external systems, such as voter registration databases and election commission systems.

Performance:

- **Response Time:** Optimize the system to deliver fast response times, ensuring a smooth and efficient voting experience.
- **Throughput:** Design the system to handle a high volume of concurrent transactions and votes without compromising performance.

3 DESIGN SPECIFICATIONS

3.1 Explanation

This part contains design specifications about the "Android Based E-voting System" project. It outlines sections like introduction, composite viewport, logical viewpoint, information viewpoint, interaction viewpoint and state dynamic viewpoint. These give an overall idea of how the system will look like visually represented; its structure in terms of logic; organization of data storage within it; user interactions expected from them when using such a software package plus what happens during different states showing dynamic behavior etcetera hence contributing towards better understanding the project's design considerations as well implementation strategies thereof by readers who may be interested in this particular area of study.

3.2 Composite Viewpoint:

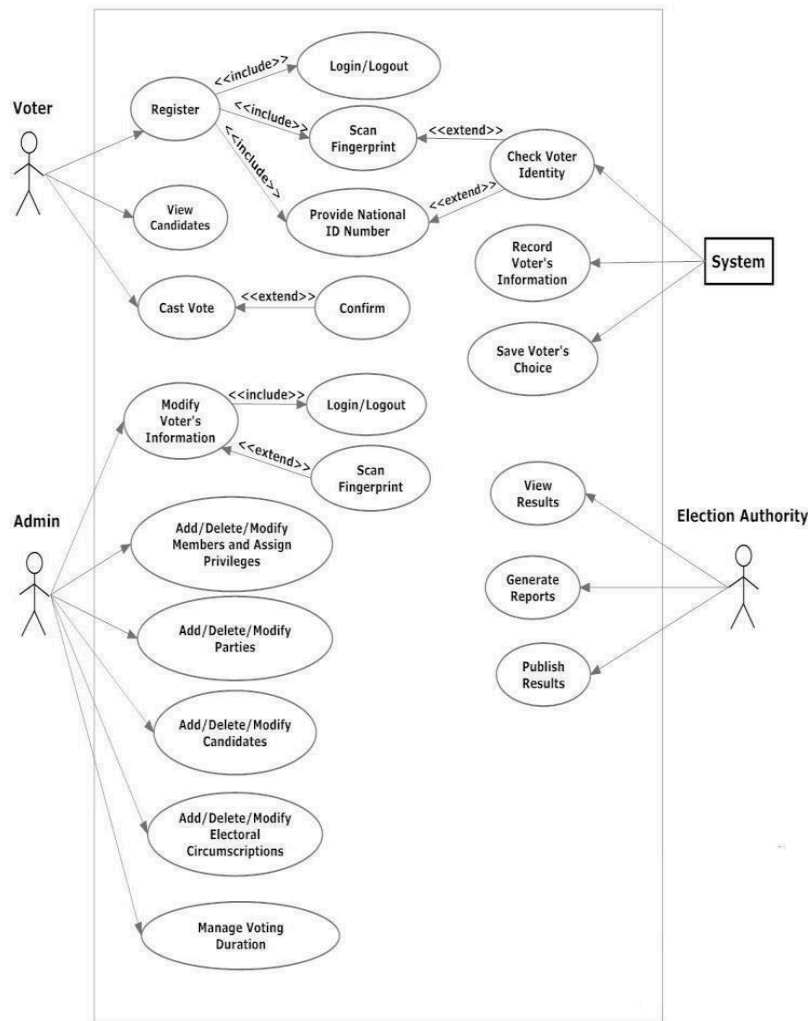


Figure 13: Use Case Diagram

3.3 Logical Viewpoint:

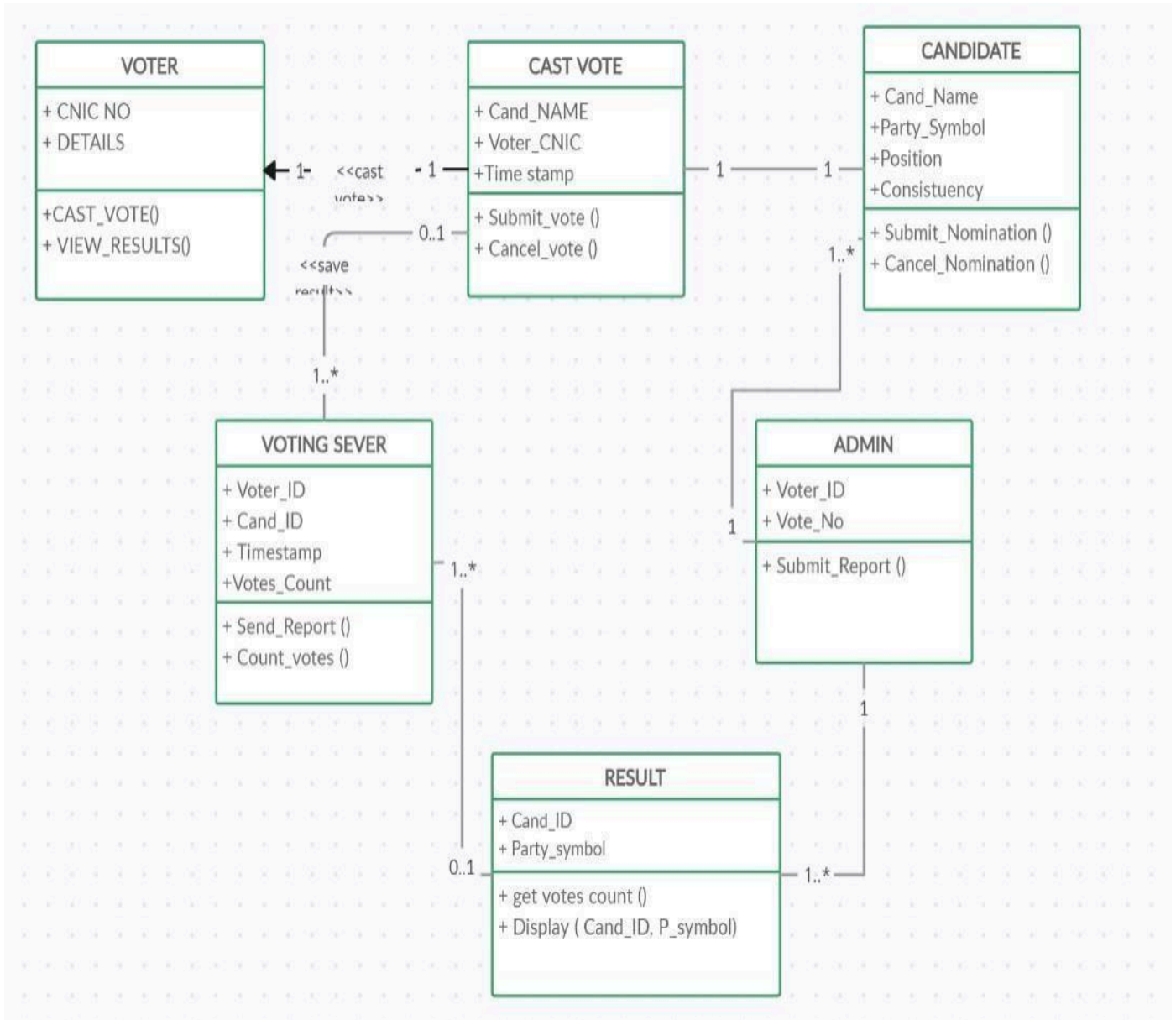


Figure 14: Class Diagram

3.4 Information Viewpoint:

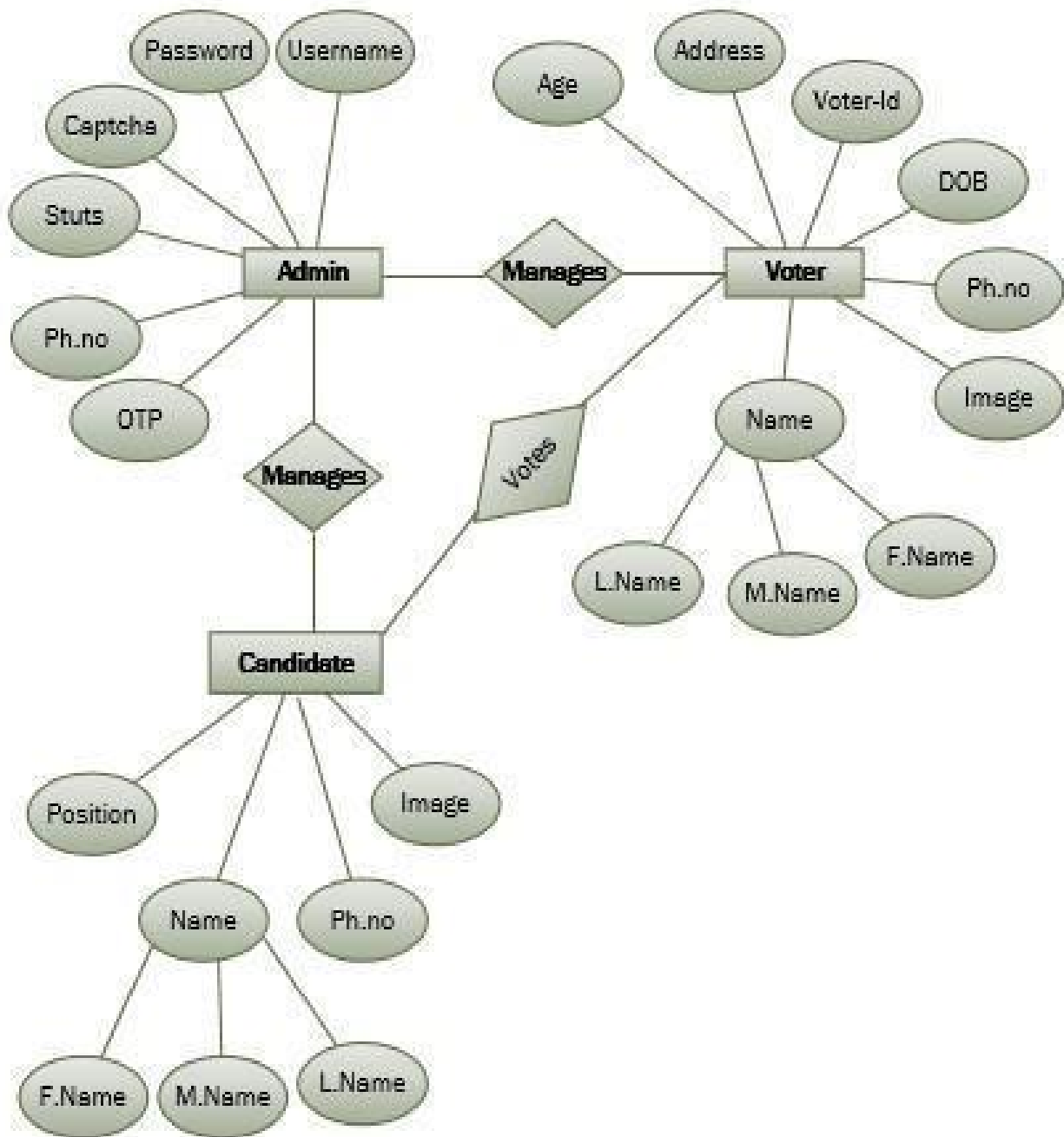


Figure 15: ER Diagram

3.5 Interaction Viewpoint:

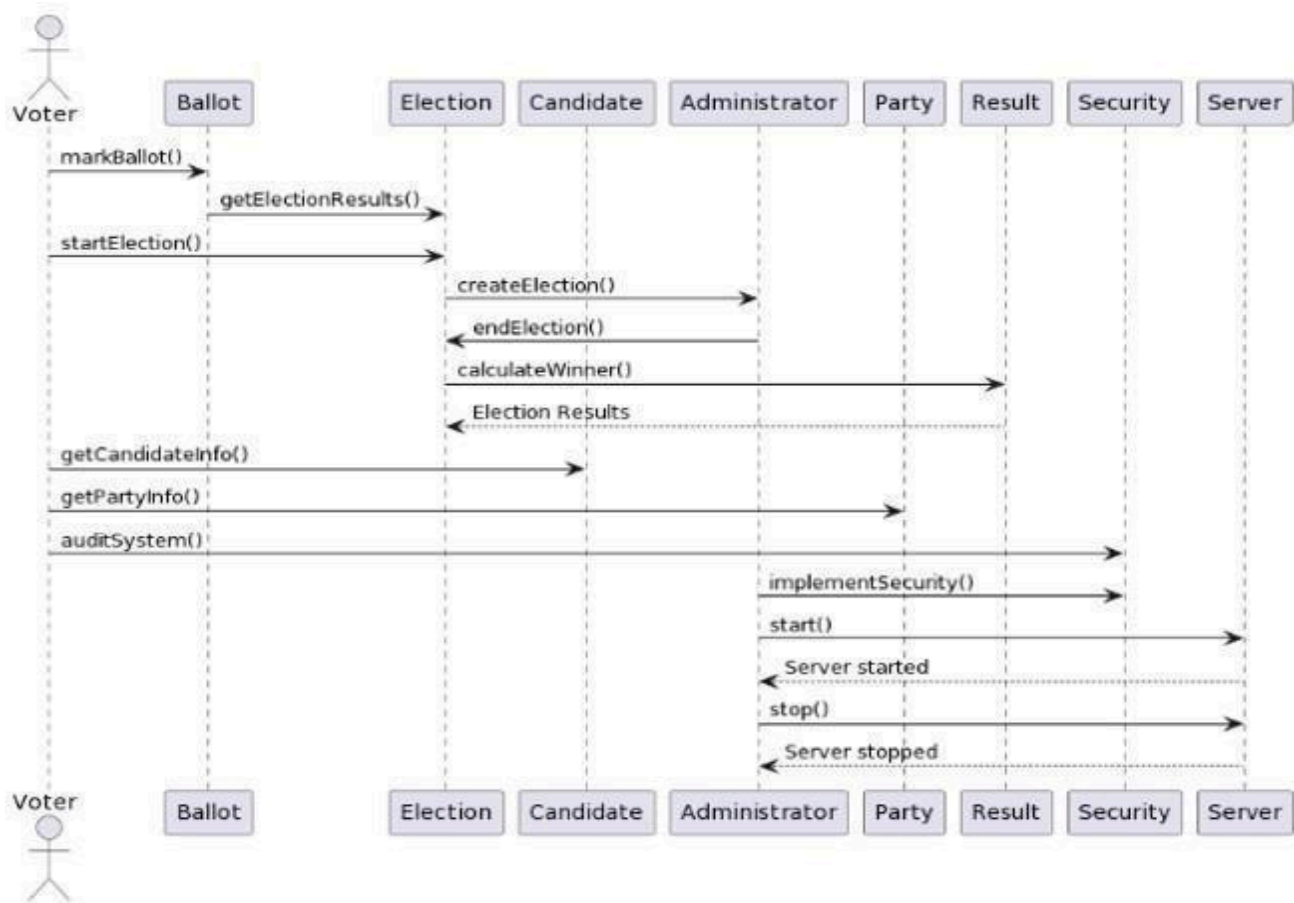


Figure 16 : Sequence Diagram

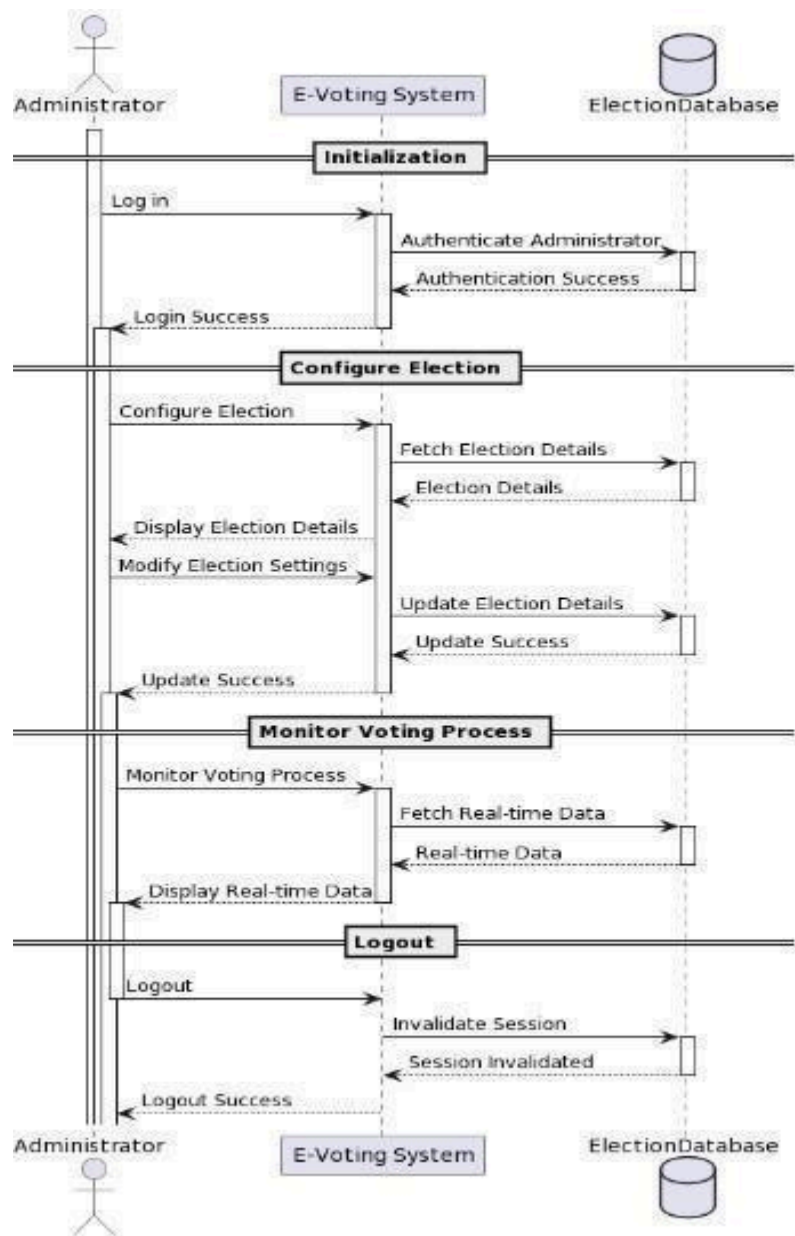


Figure 17: Sequence Diagram

3.6 State Dynamics Viewpoint:

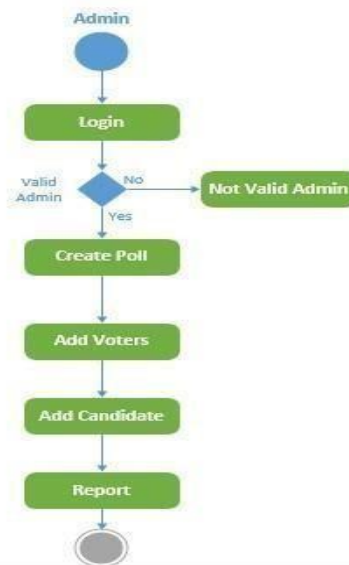


Figure 18: Admin State Diagram

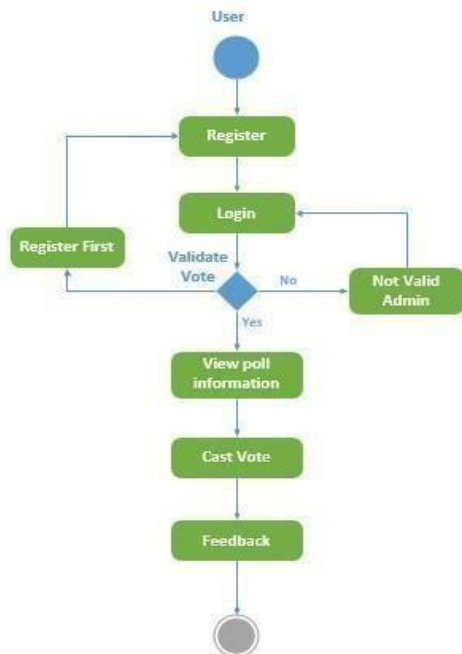


Figure 19: User State Diagram

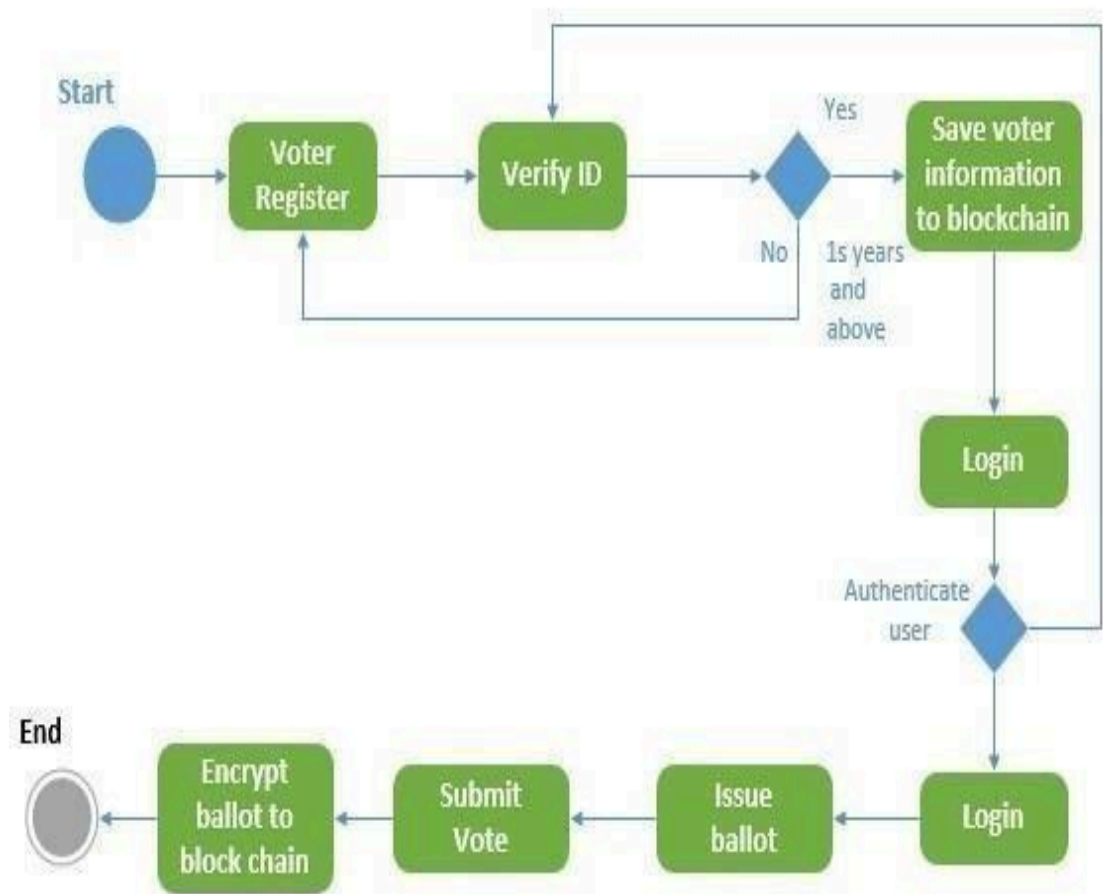


Figure 20: Voter State Diagram

4 DEVELOPMENT AND TOOLS

This chapter is an exploration of development and tools used in making the “Android Based E-Voting System”. It gives information about the team, plan for development and how work was distributed among members.

4.1 Define

In this chapter we present a development plan, tools used and future extensions of the “Android Based E-voting system project”. We discuss team members involved in the project, development plan which includes activities/workload distribution as well as tools/IDEs used for development. Further we wind up by giving some hints on what can be done next time or in future so that this project may have more features or wider coverage.

4.2 Development Plan

This project is developed by a team of two members.

1. Muhammad Nadeem
2. Muhammad Usman

Gant Chart/Table

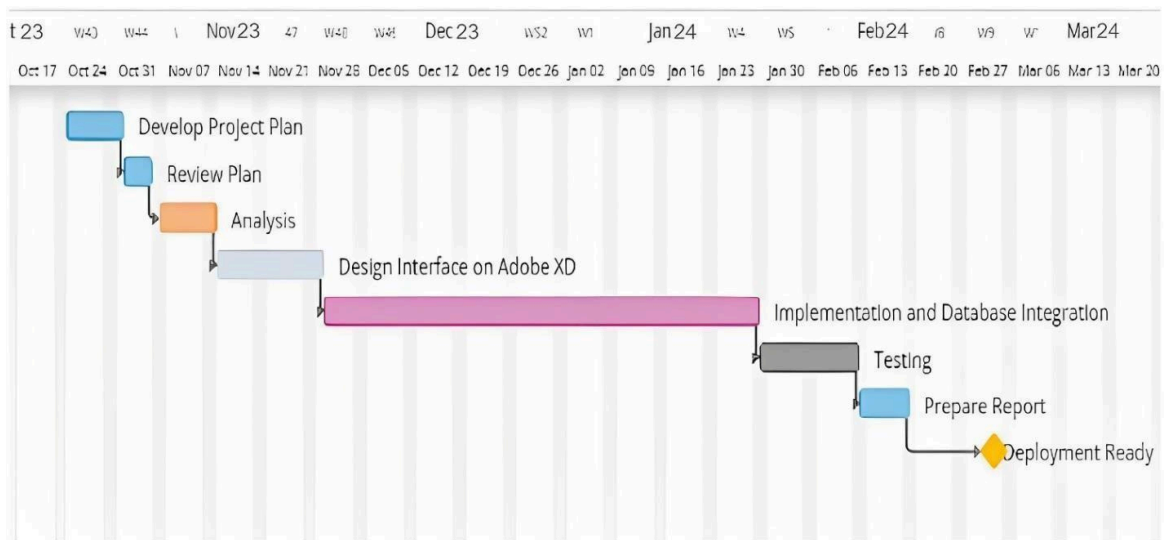


Figure 21: Gant Chart

4.3 Development Tools

To create the “Online Web Based Examination Portal” project, we used the following tools and IDEs:

Programming Languages:

Dart: Used in frontend development.

Frameworks:

Flutter: Employed to build the frontend user interface.

Database Management System:

Firebase: Selected as the relational database system for data storage and retrieval.

Version Control:

Git: Used for version control and collaborative development.

Integrated Development Environments (IDEs):

Visual Studio Code: Used as the main IDE for coding and debugging.

Android Studio: Used for Flutter app development and testing on Android devices.

4.4 Conclusion and Future Work/Extensions

The Android based e-voting system has proven successful in creating a secure platform that is easy to use during democratic elections. We can make it better by connecting with government databases to simplify voter registration, using blockchain for more security when voting, making votes recorded on blockchain open to public scrutiny through transparency enhancement added via machine learning which understands voter behavior even better. These improvements will enhance its strength while enhancing user-friendliness which are among top requirements demanded by modern democratic processes today.

5 QUALITY ASSURANCE

This chapter discusses about quality assurance of the "Android Based E-Voting system "project with focus on reliability, accuracy and usability of the application. It highlights testing strategies, methodologies employed in validating functionalities performance of the system. Each aspect under quality assurance is looked into keenly so as to come up with possible problems that may arise thus ensuring seamless user experience through functional testing up to performance plus security testing.

5.1 Define

The chapter defines quality assurance phase within Test Plan for "Android Based E-Voting system" It covers various types of testing methodologies employed towards achieving reliability and accuracy of an application hence delivering error free experience at users ends. A requirement traceability matrix was developed against each test case vis a vis desired functional requirements of a system while taking into consideration wide test coverage against what users expect since satisfaction remains their primary aim here.

5.2 Traceability Matrix

Table : 1

Test Case ID	Requirement	Admin	Voting	Reports	Real-time Updates	Data Management
TC001_Admin	Admin Login with facial recognition	✓	<input type="checkbox"/>			
TC002_Voting	Add Voters Data		✓			
TC003_Reports	Edit/Delete Candidates' Particulars		<input type="checkbox"/>			
TC004_Reports	Create a Poll/Election					
TC005_Data Management	Provide a database having voters' information		<input type="checkbox"/>			
TC006_Data Management	Edit/Delete voter's information in the database					

TC007_Reports	Provide database having the total number of casted votes			1		
TC008_Reports	Produce a report or analysis based on the total casted votes and voting ratio statistically in a graphical view					1
TC009_Reports	Produce detailed report demographically and based on the male/female voting ratio					
TC010_Reports	Connect to the system through the Android application					1
TC011_Voting	Sign up for cast vote		✓			
TC012_Voting	Login for voting with facial recognition authentication		✓			
TC013_Voting	Cast a vote in favor of a candidate/subject		✓			
TC014_Reports	View results after polling time			✓		
TC015_Reports	Produce a report on total casted votes and voting ratio					

5.3 Test Plan

Table : 2

Test Case ID	Test Objective	Steps	Expected Result	Actual Result	Pass/Fail	Requirement
TC001	Admin Login with facial recognition	1. Open the admin login page.	Admin login page is displayed.	Admin face is recognized, and access is granted.	Pass	Authentication and Access Control
TC002	Add Voters Data	1. Access the "Add Voters" section.	Add Voters form is displayed.	Voter information is added to the database.	Pass	Voter Management
TC003	Edit/delete candidates' particulars	1. Navigate to the candidates list.	Candidates list is displayed.	Edit/delete options are available for selected candidate.	Pass	Candidate Management
TC004	Create poll/election	1. Access the "Create Poll/Election" section.	Poll/Election creation form is displayed.	Poll/Election is successfully created.	Pass	Election Management
TC005	Provide a database having voters' information	Ensure the database is accessible.	Database containing voters' information is available.	Database is accessible.	Pass	Data Availability
TC006	Edit/delete voter's information in the database	Access voter database.	Options to edit/delete voter information are available.	Voter information is successfully edited/deleted.	Pass	Voter Management
TC007	Provide database having the total number of casted votes	Access voting database.	Database containing total number of casted votes is available.	Database is accessible.	Pass	Data Availability

TC009	Produce detailed report demographically and based on the male/female voting ratio	Generate demographic voting report.	Report detailing male/female voting ratio is generated.	Report is generated with male/female voting ratio.	Pass	Reporting and Analysis
TC010	Connect to the system through the Android application	Open the Android app.	Login screen of the voter module is displayed.	Login screen is displayed.	Pass	System Accessibility
TC011	Sign up for cast vote	1. Access the sign-up page.	Sign-up page is displayed.	Voter account is created successfully.	Pass	Voter Management
TC012	Login for voting with facial recognition authentication	Access the login page.	Login page is displayed.	Voter face is recognized, and access is granted.	Pass	Authentication and Access Control
TC013	Cast a vote in favor of a candidate/subject	1. Access the voting section.	List of candidates/subjects is displayed.	Vote is cast successfully.	Pass	Voting Process
TC014	View results after polling time	1. Access the "View Results" section.	Results page is displayed.	Results of the election are shown.	Pass	Reporting and Analysis
TC015	Produce a report on total casted votes and voting ratio	Generate voting report.	Report with total casted votes and voting ratio is produced.	Report is generated with expected data.	Pass	Reporting and Analysis

6 USER MANUAL

This chapter is a user guide for the application “Android Based E-Voting System.” It gives an introduction which explains why this manual was made, hardware and software requirements of the system, installation guide and operational manual which provides detailed instructions on how to use different functions of the application.

6.1 Introduction

This part gives an overview of what should be expected from reading through this document by explaining its purpose and structure. It also prepares one’s mind about what is contained in the book hence acting as a setting for other chapters that follow.

6.2 Hardware/Software Requirements for the System

User Devices: Smartphones.

Server Hardware: Sufficient processing power, memory capacity and storage space.

Database Management Systems (DBMS): Relational databases such as Firebase.

Operating Systems: Windows 10 or later versions; macOS 10.12 or later versions; Linux distributions.

6.3 Installation guide for Application

Download application from designated source.

Run setup wizard then follow displayed instructions accordingly.

Make necessary changes on app settings during configuration stage.

Finish installing it before launching app afterwards.

6.4 Operating Manual

Sign in into e-voting application using provided login credentials.

Go through various parts like voter registration section, casting votes area or viewing election results segment among others shown on screen while navigating within them all steps must be followed as per instruction given thereon in course of performing ballot exercise during polls process do not forget to use features like verifying your identity through biometric authentication simply choose who you want vote for but ensure that you cast it within set time frame also visit your profile where personal details can be edited together with some settings if need be