**Bachelor of Software Engineering**

**Centre for IT Education (CITES)**

**Department of Electrical and Computer Engineering**

**The Open University of Sri Lanka**

**EEX 4189 – Software Design in Groups**

**DETAILED SOFTWARE REQUIREMENTS SPECIFICATION**

**Fingerprint Voting System**

24.06.2024

**Group Members:**

**D.A.T.M Abenayaka - 219213375 - s19004189**

**P.M Abeyrathna - 221426574 - s92066574**

**G.D.D Senath - 222515500 - s22009991**

**N. W. L. U. R. D. Nanayakkara - 321424374 - s92064374**

**Table of Contents**

[**1 Introduction 3**](#_gjdgxs)

[Purpose 4](#_30j0zll)

[Summary 4](#_a64og3v6dc9u)

[Company Overview 4](#_56u19omz31g0)

[Project Overview 4](#_mgh1sl94ypey)

[Scope 5](#_a4ixftqedp89)

[Assumptions 5](#_h5jrhpvfe7cf)

[Definitions, Acronyms and Terminology 7](#_cc4331iyebbd)

[**2 Project Scope and Impact 8**](#)

[Project Scope Inclusions 8](#_a9rpbmlm81na)

[Project Scope Exclusions 8](#_yf52jqv7w00r)

[Impact on other systems 9](#_1l775wodi0hv)

[**3. Functional Requirements 10**](#)

[Function 1: User Registration 10](#_9bm5rtpet1pp)

[Function 2: User Authentication 11](#_qhna7uui8edu)

[Function 3: Vote Casting 12](#_a1gf7mlvwm7r)

[Function 4: Vote Verification 13](#_9bm6tjm9k8pa)

[Function 5: System Administration 14](#_wd2s2t6c2en1)

[Function 6: Reporting Requirements 15](#_eyczpp6w0c9g)

[Function 7: Help and Support Interface 16](#_gyiyk153666)

[Function 8: Error Handling and Feedback 17](#_z2puw1giuhcp)

[Function 9: Notification Service 18](#_l6x059qm2w4w)

[Function 10: Data Archival and Retention 19](#_pzy7sgp5381w)

[**4 Non-Functional Requirements 20**](#)

[Performance and Load Requirements 20](#_2xcytpi)

[Compatibility Requirements 20](#_3whwml4)

[External Interface Requirements 21](#_mhonhs6izl1t)

[Security and Authentication requirements 22](#_z73se13hnwf5)

[Quality Assurance Requirements 23](#_e7a7igs9cfbp)

[Development Requirements 23](#_8uaftxep4m7l)

[Deployment Requirements 25](#_nu5vrga87pkm)

[Applicable Standards 26](#_iie54s1v27us)

[**5 Future Requirements 27**](#)

[**6 Appendix - Survey Results 28**](#_2lwamvv)

[Results 28](#_cqo202oe70tx)

[Analysis 28](#_idzy5pk41orq)

[Conclusions 29](#_y4ep48s6cjac)

**Document Revisions**

| Date | Version | Description | Author |
| --- | --- | --- | --- |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |

**Document Approval**

Group E6 has reviewed this document and hereby agree that the contents herein are accurate. Any changes to this document must be communicated in writing and signed off by both parties.

| Signature | Signature |
| --- | --- |
| Date: | Date: |
| Name: | Name: |
| Customer: | Virtusa Corporation |

# Introduction

## Purpose

The purpose of this document is to identify and outline the system requirements for the Fingerprint Voting System (FVS). This document aims to ensure a comprehensive understanding and agreement on all necessary requirements before transitioning into the design phase. It serves as a foundational reference for the engineering team, guiding them in the development of a secure, efficient, and user-friendly fingerprint-based voting system. The ultimate goal is to provide a reliable and tamper-proof voting solution that enhances the integrity and accessibility of the electoral process.

## Summary

The current voting systems face significant challenges, including susceptibility to fraud, voter impersonation, and accessibility issues. These deficiencies undermine the integrity and trustworthiness of the electoral process. The Fingerprint Voting System (FVS) project aims to address these problems by implementing a secure, tamper-proof, and user-friendly voting solution based on biometric fingerprint recognition. This project will enhance the accuracy of voter identification, reduce the risk of fraud, and improve the overall efficiency and accessibility of voting. By identifying and agreeing on system requirements upfront, we ensure a clear path forward for designing and implementing an effective solution that corrects the current deficiencies.

## Company Overview

**Group E6** is a leading provider of innovative and reliable software solutions, dedicated to enhancing business processes and improving operational efficiency for clients across various industries. With over two decades of experience, Quality Software has established a reputation for delivering high-quality, customizable products that meet the unique needs of each client.

Our team of highly skilled professionals is committed to delivering exceptional service and support throughout the entire project lifecycle, from initial consultation and requirements gathering to implementation and ongoing maintenance.

We are excited to collaborate with your organisation on the Fingerprint Voting System (FVS) project, leveraging our expertise and innovative approach to create a secure, efficient, and user-friendly voting solution that addresses current deficiencies and enhances the integrity of the electoral process.

## Project Overview

The Fingerprint Voting System (FVS) project aims to develop and implement a biometric-based voting solution that addresses current challenges such as voter fraud, impersonation, and accessibility issues. This project will enhance the accuracy of voter identification, ensuring a secure and efficient electoral process

## Scope

### Products or Components to be Updated

1. **Voter Registration System**: Integration with biometric fingerprint recognition to ensure accurate and unique voter identification.
2. **Voting Machines**: Upgrade existing machines to incorporate fingerprint scanning technology, ensuring secure and tamper-proof voting.
3. **Database Management System**: Enhance the current database to securely store and manage biometric data alongside traditional voter information.
4. **User Interface**: Develop an intuitive and user-friendly interface for voters and election officials to interact with the new fingerprint-based system.

### Projects or Components Affected by These Changes:

1. **Voter Authentication Process**: Changes to the current authentication procedures to include biometric verification.
2. **Data Security Protocols**: Updates to security protocols to ensure the protection of biometric data.

## Assumptions

### Availability of Biometric Data

It is assumed that biometric data (fingerprints) of voters will be readily available or can be collected during the voter registration process.

### Legal and Regulatory Compliance

The implementation of biometric voting systems will comply with all relevant legal and regulatory requirements, including data privacy laws.

### Technological Infrastructure

Existing technological infrastructure, such as databases and servers, can support the integration and storage of biometric data without requiring significant upgrades.

### Voter Accessibility

Voters will have reasonable access to fingerprint scanning devices during both the registration and voting processes.

### Hardware Compatibility

Current voting machines can be retrofitted or upgraded to include fingerprint scanning capabilities without needing complete replacement.

### Network and System Security

Sufficient security measures, including encryption and secure data transmission protocols, will be in place to protect biometric data.

### Operational Continuity

The transition to the new system will be smooth, with minimal disruption to ongoing electoral processes.

### Scalability

The system will be designed to handle the scale of current and future voter populations without performance degradation.

### Testing and Validation

Sufficient time and resources will be allocated for comprehensive testing and validation of the system before full-scale deployment.

## 

## Definitions, Acronyms and Terminology

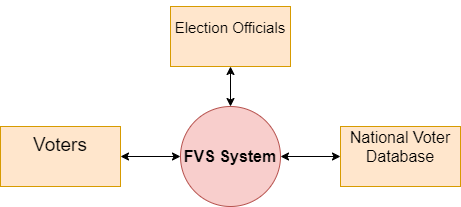
### Industry-Specific Terms

| **Term** | **Definition** |
| --- | --- |
| **Biometric Authentication** | A security process that uses individuals' physical characteristics (such as fingerprints) for identity verification. |
| **Voter Registration System** | A database and software system used to register eligible voters and maintain their records. |
| **Election Management Software** | Software used to organise and manage various aspects of the electoral process, including voter registration, voting, and result tabulation. |
| **Tamper-Proof** | A system designed to prevent unauthorised access or modifications. |
| **User Interface (UI)** | The means by which a user interacts with a computer system or software, including screens, menus, and commands. |
| **Data Encryption** | The process of converting data into a code to prevent unauthorised access. |
| **Interoperability** | The ability of different systems and organisations to work together (inter-operate). |

### Abbreviations

| **Abbreviation** | **Definition** |
| --- | --- |
| **FVS** | Fingerprint Voting System |
| **UI** | User Interface |
| **CRM** | Customer Relationship Management |
| **FISMA** | Federal Information Security Management Act |
| **GDPR** | General Data Protection Regulation |
| **API** | Application Programming Interface |
| **DBMS** | Database Management System |
| **SSL** | Secure Sockets Layer |
| **VPN** | Virtual Private Network |

# Project Scope and Impact



## Project Scope Inclusions

1. **Development of Fingerprint Voting System (FVS):**

Design and implementation of biometric authentication using fingerprint recognition technology.

1. **Integration with Voting Process:**.

Development of interfaces for user-friendly voting experience across different platforms (e.g., desktop, mobile).

1. **Enhancement of Voting Security:**

Implementation of measures to prevent voter fraud, impersonation, and unauthorised access.

1. **Improvement of Voting Efficiency:**

Automation of voter identification and vote counting processes to reduce administrative burden and human errors.

## Project Scope Exclusions

1. **Hardware Development:**

Excludes the development or manufacturing of fingerprint scanning hardware devices.

1. **Legislative and Policy Frameworks:**

Excludes the establishment of legal frameworks or policies related to voting procedures.

1. **Network Infrastructure:**

Excludes the development or enhancement of network infrastructure for data transmission.

## 

### **Affected by Other Systems**

The Fingerprint Voting System (FVS) interacts with and is affected by several other systems such as ;

1. **Biometric Database System:**
   * The FVS relies on a biometric database system to store and verify fingerprint data securely.
2. **Network Infrastructure:**
   * Requires a stable network infrastructure for real-time communication between voting stations and the central database.
3. **Voter Registration System:**
   * Receives voter registration information from existing voter databases to authenticate eligible voters.

### **Effects on Other Systems**

The implementation of the Fingerprint Voting System will impact the following systems, necessitating modifications or considerations:

1. **Data Security Systems:**
   * Existing data security measures must be enhanced to protect biometric and voter information from unauthorized access.
2. **User Support Systems:**
   * Helpdesk and support systems may need additional resources to assist voters and election officials in using the new technology effectively.
3. **Administrative Systems:**
   * Election administration systems may require updates to integrate with the FVS for voter registration and result tabulation.

# 3. Functional Requirements

## Function 1: User Registration

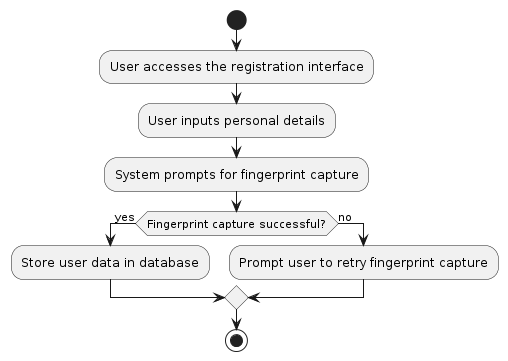
### Description

Users must be able to register on the system using their personal details and biometric data. The process includes capturing the fingerprint, verifying the user identity, and storing the data securely.

### Process Flow

* 1. User accesses the registration interface.
  2. User inputs personal details.
  3. System prompts for fingerprint capture.
  4. Upon successful entry of data, user data is stored in the database.

### Activity Diagram



## 

## Function 2: User Authentication

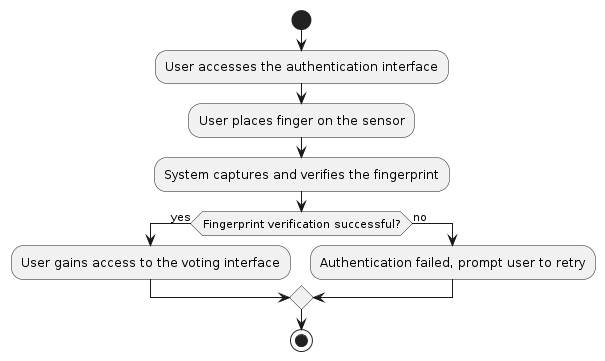
### Description

Users must authenticate using their fingerprints to access the voting system. This ensures secure and efficient user identification.

### Process Flow

* 1. User accesses the authentication interface.
  2. User places finger on the sensor.
  3. System captures and verifies the fingerprint.
  4. Upon successful verification, the user gains access to the voting interface.

### Activity Diagram



## 

## Function 3: Vote Casting

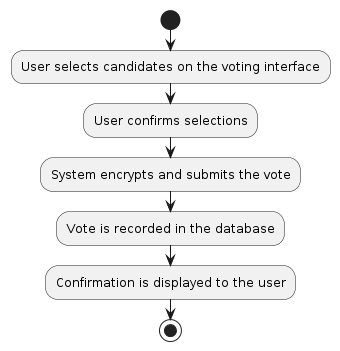
### Description

Users must be able to cast their votes through the system after successful authentication.

### Process Flow

* 1. User selects candidates on the voting interface.
  2. User confirms selections.
  3. System encrypts and submits the vote.
  4. Vote is recorded in the database and a confirmation is displayed to the user.

### Activity Diagram



## 

## Function 4: Vote Verification

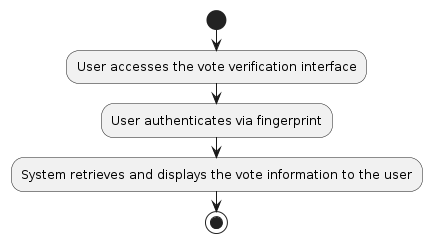
### Description

Users must be able to verify that their vote was cast correctly.

### Process Flow

* 1. User accesses the vote verification interface.
  2. User authenticates via fingerprint.
  3. System retrieves and displays the vote information to the user.

### Activity Diagram



## 

## Function 5: System Administration

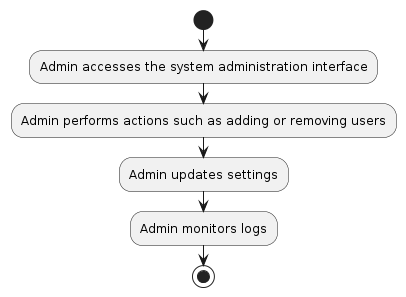
### Description

Admins must be able to manage the system, including user accounts, system settings, and monitoring system health.

### Process Flow

* 1. Admin accesses the system administration interface.
  2. Admin performs actions such as adding or removing users, updating settings, and monitoring logs.

### Activity Diagram



## 

## Function 6: Reporting Requirements

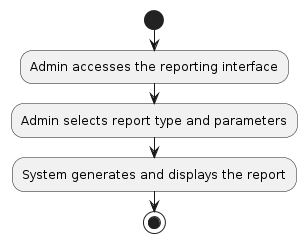
### Description

The system must generate reports for election results, user activity, and system performance.

### Process Flow

* 1. Admin accesses the reporting interface.
  2. Admin selects report type and parameters.
  3. System generates and displays the report.

### Activity Diagram



## 

## Function 7: Help and Support Interface

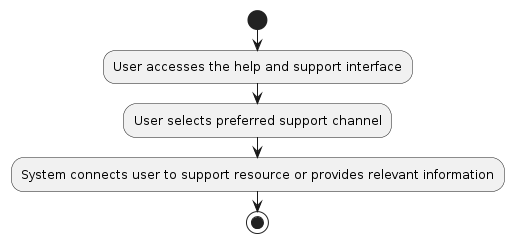
### Description

Users must have access to help and support through various channels, including phone support, email, and online chat.

### Process Flow

* 1. User accesses the help and support interface.
  2. User selects the preferred support channel.
  3. System connects users to support resources or provides relevant information.

### Activity Diagram



## 

## Function 8: Error Handling and Feedback

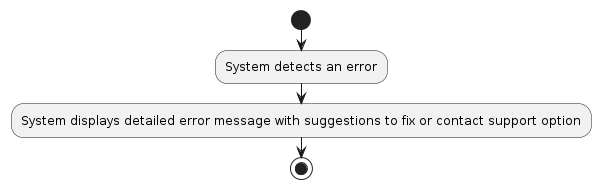
### Description

The system must provide detailed feedback and suggestions to users in case of errors.

### Process Flow

* 1. System detects an error.
  2. System displays detailed error messages with suggestions to fix or contact support options.

### Activity Diagram



## 

## Function 9: Notification Service

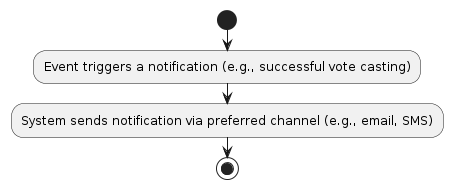
### Description

The system must send notifications to users regarding their voting status and other important updates.

### Process Flow

* 1. Event triggers a notification (e.g., successful vote casting).
  2. System sends notification via preferred channel (e.g., email, SMS).

### Activity Diagram



## 

## Function 10: Data Archival and Retention

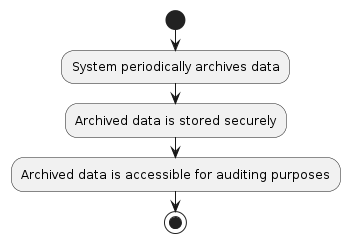
### Description

The system must archive voting data securely and ensure it is retained according to legal requirements.

### Process Flow

* 1. System periodically archives data.
  2. Archived data is stored securely and is accessible for auditing purposes.

### Activity Diagram



# 4 Non-Functional Requirements

## Performance and Load Requirements

| **Screen Load Time** | All static web pages should load within 3 seconds. |
| --- | --- |
| **Page Interaction Response Time** | Pages must respond in less than a 1 second |
| **Current User Load** | 5 |
| **Expected User Load** | 10,000 during peak voting season |
| **Expected Growth** | 20% (at least 12,000 next election cycle) |
| **Peak Concurrent Users** | 15,000 combined election officials and voters |
| **Maximum Transaction Size** | Should not exceed 500 KB |
| **Average Processing Time** | Data processing and serving requests should remain less than 1 second |

## Compatibility Requirements

| **Mobile Operating Systems to be supported** | Android 8.0 (Oreo) models from 2017 afterwards and above, and iOS 16.0 |
| --- | --- |
| **Fingerprint Sensors** | The application must support fingerprint sensors compliant with FIDO (Fast Identity Online) standards. |
| **Backend Server** | The backend server will run on Node.js 14.0 or later with Express.js 4.0 or later framework. |
| **Database Versions to be supported** | The system will use a NoSQL database provided by Firebase |
| **Communication Protocol** | All communications between the client and server will use HTTPS to ensure data security during transmission. |
| **API Standards** | The biometric API must adhere to the ISO/IEC 19794-2 standard for biometric data interchange formats. |
| **Development Frameworks** | The mobile application will be developed using React Native 0.64 or later to ensure cross-platform compatibility. |
| **Biometric API** | The system must integrate with standard biometric APIs available on Android (FingerprintManager) and iOS (LocalAuthentication). |
| **Network Requirements** | The system must operate efficiently on mobile networks with a minimum of 3G connectivity. |
| **External Services** | The system must be compatible with external logging and monitoring services such as ELK Stack (Elasticsearch, Logstash, Kibana) version 7.10 or later. |
| **Hardware Requirements** | The system must operate on mobile devices with a minimum of 2GB RAM and quad-core processors to ensure smooth performance. |

## External Interface Requirements

### User Interfaces

| **Requirement** | **Description** | **Protocol** | **Ports** | **Encryption** | **Remarks** |
| --- | --- | --- | --- | --- | --- |
| Administrative Interface | Interface for election officials to access administrative functions and monitor the voting process. | HTTPS | 443 | TLS 1.2 or later | Administrative portal URLs specified in the deployment configuration |

### Hardware Interfaces

| **Requirement** | **Description** | **Protocol** | **Ports** | **Encryption** | **Remarks** |
| --- | --- | --- | --- | --- | --- |
| Biometric Data Capture Interface | Interface for capturing biometric data using fingerprint sensors on mobile devices. | Local API calls | N/A | N/A | Android FingerprintManager, iOS LocalAuthentication |

### Software Interfaces

| **Requirement** | **Description** | **Protocol** | **Ports** | **Encryption** | **Remarks** |
| --- | --- | --- | --- | --- | --- |
| Mobile Application to Backend Communication | Interface for communication between the mobile application and the backend server for data transmission. | HTTPS | 443 | TLS 1.2 or later | Backend server IP addresses specified in the deployment configuration |
| Backend to Database Communication | Interface for the backend server to communicate with Firebase for storing and retrieving votes and biometric data. | HTTPS | 443 | TLS 1.2 or later | Firebase endpoints specified in the deployment configuration |
| Third-party Biometric API Interface | Interface for integrating third-party biometric APIs that comply with ISO/IEC 19794-2 standards. | HTTPS | 443 | TLS 1.2 or later | Third-party service URLs |
| External Monitoring and Logging Interface | Interface for sending logs and monitoring data to external logging and monitoring systems such as ELK Stack. | HTTPS | 443 | TLS 1.2 or later | External logging service URLs |

### 

### Communications Interfaces

| **Requirement** | **Description** | **Protocol** | **Ports** | **Encryption** | **Remarks** |
| --- | --- | --- | --- | --- | --- |
| Notification Service Interface | Interface for sending notifications to voters regarding their voting status and other relevant information. | HTTPS | 443 | TLS 1.2 or later | Notification service URLs (e.g., Firebase Cloud Messaging) |

## 

## Security and Authentication requirements

### Data Storage Security

| **Database Security** | Use Firebase's built-in security rules and AES-256 encryption for data at rest. |
| --- | --- |
| **Access Control** | Implement role-based access control (RBAC) and unique credentials for all users and applications. |
| **Data Integrity** | Use Firebase real-time capabilities and data validation rules to ensure accurate data storage. |
| **Backup and Recovery** | Regularly back up encrypted data and develop a disaster recovery plan. |

### Data Communication Security

| **Secure Communication Protocols** | Use HTTPS with TLS 1.2 or later and secure WebSockets (WSS) for all data transmissions. |
| --- | --- |
| **Data Encryption** | Encrypt all transmitted data and use POST methods for API calls. |
| **Authentication and Authorization** | Use Firebase Authentication and JWT for secure user identity management and token transmission. |
| **Session Management** | Implement short-lived tokens, refresh tokens, and session expiration. |

## 

## Quality Assurance Requirements

### QA Test Scope

| **Test Type** | **Description** |
| --- | --- |
| Functional Testing | Verify all functional requirements and process flows. |
| Security Testing | Ensure data protection and perform penetration testing. |
| Performance Testing | Conduct load and stress testing to assess system behavior under different conditions. |
| Compatibility Testing | Test across various devices and OS to ensure seamless operation. |
| Usability Testing | Assess user-friendliness and gather feedback for UI improvements. |

### QA Environment

| **Requirement** | **Description** |
| --- | --- |
| Software Requirements | Tools: Selenium (functional testing), JMeter (performance), OWASP ZAP (security), Firebase Test Lab, version control (Git). |
| Hardware Requirements | Variety of smartphones, desktops and/or laptops |
| Environmental Requirements | Test environments replicating production settings, isolated environments for different tests. |

### QA Data

| **Data Type** | **Description** |
| --- | --- |
| Key Data Entities | Test user accounts, biometric data, and simulated vote data. |
| Populated Databases | Firebase database with test user info, biometric data, and vote records. |

## Development Requirements

### Development Environment

| **Requirement** | **Description** |
| --- | --- |
| Software Requirements | Development tools: Visual Studio Code or IntelliJ IDEA, Android Studio, Xcode for iOS, Firebase CLI. |
| Hardware Requirements | Development machines with minimum 16GB RAM, quad-core processors, and SSD storage. |
| Environmental Requirements | Local and cloud-based environments that replicate production settings, including network configurations. |

### 

### Development Data

| **Data Type** | **Description** |
| --- | --- |
| Key Data Entities | Test user accounts, biometric data samples, and initial voting data for development and testing. |
| Populated Databases | Firebase database instances pre-populated with test data for development purposes. |

### Coding Standards

| **Standard** | **Description** |
| --- | --- |
| Code Style Guidelines | Follow the project's coding standards document, which includes guidelines for naming conventions, code structure, and documentation. |
| Naming Conventions | Use camelCase for variables and functions, PascalCase for classes and components, and consistent prefixing for modules (e.g., auth\_ for authentication-related modules). |
| Documentation | Code must be thoroughly documented using JSDoc for JavaScript/TypeScript and standard documentation practices for other languages. |

### Implementation Packaging Requirements

| **Requirement** | **Description** |
| --- | --- |
| JavaScript/TypeScript Modules | Group functionality into specific modules, such as auth, vote, user, and admin, ensuring logical separation of concerns. |
| Mobile Application Packaging | Android app must follow package naming conventions like com.fvs.android, and iOS app like com.fvs.ios. |
| Deployment Packages | Backend services should be containerized using Docker, with each microservice having its own container image. |
| Versioning | Follow semantic versioning (MAJOR.MINOR.PATCH) for all releases and module updates. |

## 

## Deployment Requirements

### Installation Packaging Requirements

The Fingerprint Voting System (FVS) will be packaged for installation using modern containerization and packaging tools to ensure ease of deployment and scalability. The key installation packaging requirements are:

1. **Docker Containers**: Each component of the FVS, such as the backend server, database, and supporting services, will be containerized using Docker. This ensures consistency across different environments and simplifies the deployment process.
2. **Mobile Application Packaging**: The mobile application for Android will be packaged as an APK file, and for iOS, as an IPA file. These packages will be distributed via official app stores (Google Play Store and Apple App Store).
3. **Web Application Packaging**: If a web interface is provided, it will be packaged as a set of static files (HTML, CSS, JavaScript) and served via a web server (e.g., Nginx).

### Deployment Requirements

The deployment of the Fingerprint Voting System (FVS) will follow a structured approach to ensure reliability and accessibility for all users. The deployment requirements include:

1. **Server Locations and Infrastructure**:
   * **Primary Data Center**: The main backend servers and databases will be hosted in a primary data centre with high availability and redundancy.
   * **Cloud Providers**: The deployment will utilise cloud providers (e.g., AWS, GCP) to ensure scalability and resilience.
2. **Access by User Type**:
   * **Voters**: Voters will access the system via the mobile application..
   * **Election Officials**: Admins and election officials will access the system through a secure web portal, with enhanced access controls and logging.
3. **OS Versions and Machine Types**:
   * **Backend Servers**: Running on Linux (Ubuntu 20.04 LTS) virtual machines
   * **Mobile Devices**: Supporting Android 8.0 (Oreo) and above, and iOS 16.0 and above.
4. **Version Management**:
   * The system will use semantic versioning to manage updates and ensure compatibility.
   * Multiple versions of the mobile application may need to be maintained to support different device capabilities and OS versions.
   * Versioning will be managed using the Git version control tool.

### 

### Documentation Requirements

Comprehensive documentation will be provided to support the deployment, maintenance, and usage of the Fingerprint Voting System (FVS). The documentation requirements may include:

1. **Technical Documentation**:
   * System Architecture Documentation
   * API Documentation
   * Deployment Guides
2. **End User Documentation**:
   * User Manuals
   * FAQ and Troubleshooting Guides
   * Support Contact Information

## Applicable Standards

| **Standard** | **Applicable Sections** | **Description** |
| --- | --- | --- |
| Sri Lanka Data Protection Act (2022) | Data processing, consent, data security | Protects personal data and privacy for users in Sri Lanka. |
| Electronic Transactions Act (ETA) | Electronic signatures, electronic records | Ensures legality of electronic records and signatures in the voting process. |
| Transport Layer Security (TLS) | TLS 1.2 and later | Ensures secure communication over the internet through data encryption. |
| Hypertext Transfer Protocol Secure (HTTPS) | RFC 2818 | Protects data integrity and confidentiality in web-based interactions. |
| Android Platform Compliance | Android Compatibility Definition Document (CDD) | Ensures compatibility and consistent user experience on Android devices. |
| iOS Platform Compliance | Apple Human Interface Guidelines | Ensures adherence to Apple's design and technical standards for iOS devices. |
| ISO/IEC 27001 | ISMS requirements | Framework for managing and protecting sensitive information, ensuring data security and integrity. |
| Web Content Accessibility Guidelines (WCAG) | WCAG 2.1 AA | Ensures accessibility for users with disabilities, providing guidelines for web content accessibility. |

## 

# 5 Future Requirements

As the Fingerprint Voting System (FVS) continues to evolve and improve, several future requirements have been identified to ensure the system remains robust, secure, and user-friendly. These requirements focus on enhancing the current functionality, expanding system capabilities, and addressing emerging needs.

### **1 Advanced Technology**

* **Improved Biometrics**: Continuous advancements in biometric technology for better accuracy and speed.
* **AI and Machine Learning**: Use of AI to enhance fingerprint recognition and fraud detection.

### **2 Improved User Experience**

* **Localized Language Support**: Expand language support to include multiple local languages, making the system more accessible to a broader user base

### **3 Security**

* **Disaster Recovery Plans**: Comprehensive plans to handle system failures or disruptions
* **Network Security**: Measures to protect the system from cyber threats and unauthorized access.

### **4 Scalability and Performance**

* **Scalable Cloud Infrastructure**: Enhance cloud infrastructure to support dynamic scaling, ensuring system performance during peak voting times.
* **Distributed Data Storage**: Implement distributed data storage solutions to improve data redundancy, availability, and access speeds.

### **5 Regulatory and Compliance Updates**

* **Compliance with New Regulations**: Continuously update the system to comply with new and evolving legal and regulatory requirements regarding data privacy and election security.
* **Audit Trails and Reporting**: Enhance audit trails and reporting features to provide comprehensive logs and reports for regulatory compliance and transparency.

# 6 Appendix - Survey Results

## Results

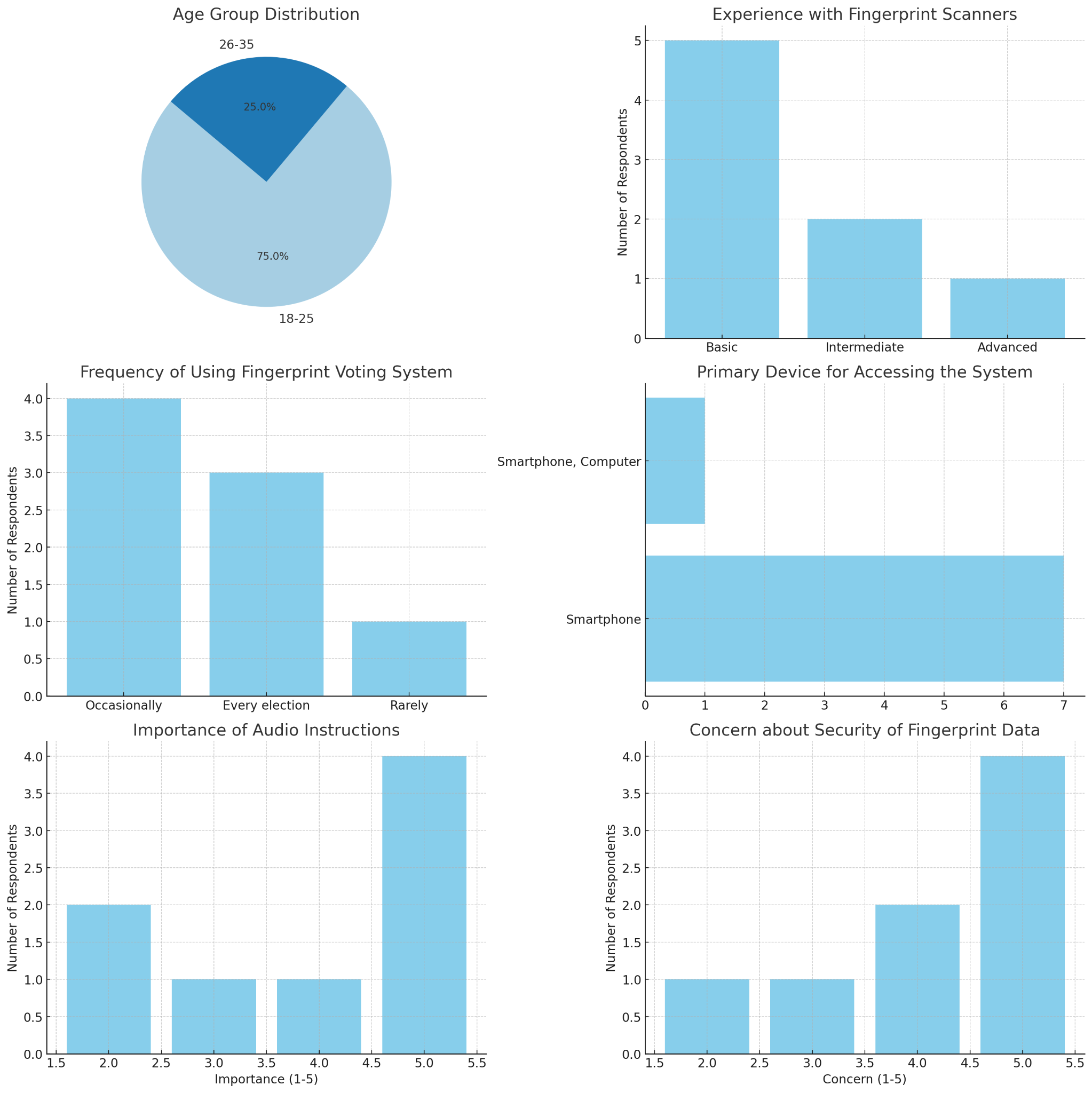
<https://docs.google.com/spreadsheets/d/1-ZDiSQdI8CZR5byxwBP9zWeSAiw5XihY/edit?usp=drive_link&ouid=104631278897196746049&rtpof=true&sd=true>

## Analysis

| **Question** | **Summary** |
| --- | --- |
| Age Group | Majority are aged 18-25, with some in the 26-35 age group. |
| Experience with Fingerprint Scanners | Most users have Basic experience, with a few having Intermediate or Advanced experience. |
| Frequency of Using Fingerprint Voting System | Most users expect to use the system every election, while some will use it occasionally or rarely. |
| Primary Device for Accessing the System | Majority will use Smartphones, with a few indicating both Smartphones and Computers. |
| Importance of Audio Instructions for Visually Impaired Users | Responses vary from 2 to 5, indicating mixed importance. |
| Concern about Security of Fingerprint Data | Majority are highly concerned (rating 5). |
| Importance of Review and Confirm Vote Option | Majority consider it Very Important or Important. |
| Preferred Features on Voting Confirmation Screen | Summary of choices and Confirmation button are the most desired features. |
| Preferred Method of Receiving Help or Support | Online chat and Phone support are the most preferred methods, with some preferring Email support. |
| Preferred Types of Feedback for Errors | Detailed message and Contact support option are the most useful feedback types. |

## 

## Conclusions



The following conclusions were drawn from analysing the survey results.

1. **User Demographics**:
   * The primary users are young adults aged 18-25, followed by those aged 26-35.
   * Most users have basic experience with fingerprint scanners, suggesting a need for user-friendly guidance.
2. **Usage Frequency**:
   * The system will see high usage during elections.
3. **Device Preference**:
   * The majority of users prefer to access the system via smartphones, indicating the need for a mobile-optimised user interface.
4. **Accessibility**:
   * While the importance of audio instructions for visually impaired users varies, it's crucial to include this feature to cater to all user needs.
5. **Security Concerns**:
   * Security of fingerprint data is a major concern for users, necessitating robust data protection measures.
6. **Confirmation of Votes**:
   * Users prioritise the ability to review and confirm their votes, highlighting the need for a clear and intuitive confirmation process.
7. **Support Preferences**:
   * Providing multiple support channels, particularly online chat and phone support, will be essential to assist users effectively.
8. **Error Feedback**:
   * Detailed error messages and options to contact support are critical for user satisfaction and problem resolution.