Software Requirements Specification (SRS)

EthioStreetFix Mobile Application

---

1. Introduction

1.1 Purpose

This Software Requirements Specification (SRS) document describes the complete requirements of the **EthioStreetFix mobile application**. The document is prepared in accordance with the **IEEE SRS standard** and integrates **core concepts of Mobile Computing, Mobile Communication, and Mobile & Wireless Security**. It is intended for instructors, supervisors, developers, testers, and stakeholders involved in the project.

1.2 Scope

EthioStreetFix is a **Flutter-based mobile application** that allows citizens to report street infrastructure problems such as potholes, broken streetlights, drainage failures, and road damage. The system operates over **wireless and mobile networks (3G/4G/5G, Wi-Fi)** and relies on **cloud computing** for data processing and storage. Due to its mobile and distributed nature, the system places strong emphasis on **security, privacy, and reliability**.

1.3 Definitions, Acronyms, and Abbreviations

- **SRS**: Software Requirements Specification
- **QoS**: Quality of Service
- **API**: Application Programming Interface
- **GPS**: Global Positioning System
- **CIA**: Confidentiality, Integrity, Availability
- **RBAC**: Role-Based Access Control

1.4 References

- IEEE 830 / IEEE 29148 SRS Standard
- Mobile Computing and Programming – Course Material
- Mobile and Wireless Security – Course Material
- Flutter Official Documentation

1.5 Overview

This document is structured according to IEEE SRS guidelines and covers functional requirements, nonfunctional requirements, architecture, system models, and security considerations with explicit alignment to **mobile computing and wireless security principles**.

---

2. Overall Description

2.1 Product Perspective

EthioStreetFix is a **mobile cloud-based distributed system** consisting of: - Mobile client application (Flutter) - Wireless communication networks (Wi-Fi, 3G, 4G, 5G) - Cloud backend services (application server and database) - Web-based administrative dashboard

2.2 Product Functions

- Secure user registration and authentication
- Street issue reporting with image and GPS location
- Secure data transmission over wireless networks
- Issue tracking and notification delivery
- Authority verification and issue resolution

2.3 User Classes and Characteristics

- **Citizens**: Mobile users reporting and tracking issues
- **Municipal Authorities**: Authorized users managing and resolving issues
- **System Administrators**: Users responsible for security, monitoring, and maintenance

2.4 Operating Environment

- Android OS (version 8.0 and above)
- Wireless and mobile networks
- Cloud infrastructure

2.5 Design and Implementation Constraints

- Variable network bandwidth and latency
- Security threats common to mobile and wireless networks
- Compliance with local data protection regulations

2.6 Assumptions and Dependencies

- Users possess smartphones with internet connectivity
- GPS and camera services are available
- Cloud services remain operational

---

3. System Features (Functional Requirements)

3.1 User Authentication and Authorization

- The system shall authenticate users securely over wireless networks.
- The system shall enforce **role-based access control (RBAC)** for citizens, authorities, and administrators.

3.2 Issue Reporting

- The system shall allow users to capture images using mobile device cameras.
- The system shall attach GPS-based location data to each report.
- The system shall encrypt data before transmission over wireless networks.

3.3 Issue Tracking

- The system shall allow users to track the status of submitted issues.
- The system shall notify users securely of any status updates.

3.4 Issue Management (Authority)

- Authorized personnel shall verify and update issue statuses.
- The system shall log all authority actions for audit purposes.

---

4. Data Requirements

- User identity and authentication data
- Issue reports (images, descriptions, location)
- Status update records
- Security logs and audit trails

---

5. System Architecture Overview

EthioStreetFix adopts a **layered mobile cloud architecture**: - **Application Layer**: Flutter mobile app and web dashboard - **Communication Layer**: Secure wireless and mobile networks (HTTPS/TLS) - **Backend Layer**: Cloud servers handling processing and storage

Security mechanisms are applied across all layers following a **defense-in-depth approach**.

---

6. Nonfunctional Requirements

6.1 Performance Requirements

- The system shall operate efficiently under varying network conditions.
- The application shall minimize bandwidth usage through optimized data transmission.

6.2 Security Requirements

- The system shall ensure **confidentiality, integrity, and availability (CIA)** of data.
- All data in transit shall be encrypted using HTTPS/TLS.
- Sensitive data stored on the device or cloud shall be encrypted.

- The system shall protect against common mobile threats such as malware, eavesdropping, spoofing, and DoS attacks.

6.3 Privacy Requirements

- User location data shall be accessible only to authorized entities.
- The system shall support user anonymity and pseudonymity where applicable.
- Location information shall not be shared with third parties without consent.

6.4 Usability Requirements

- The application shall be easy to use for non-technical users.
- The interface shall remain responsive during mobility and network handoff.

6.5 Reliability and Availability

- The system shall provide high availability through cloud-based fault tolerance.
- The application shall recover gracefully from network disconnections.

6.6 Portability

- The application shall be portable across Android devices.

---

7. External Interface Requirements

7.1 User Interfaces

- Flutter-based mobile user interface
- Web-based administrative interface

7.2 Hardware Interfaces

- Mobile camera
- GPS sensor
- Wireless network interface

7.3 Software Interfaces

- Cloud authentication services
- Mapping and location APIs

7.4 Communication Interfaces

- RESTful APIs over IP-based packet-switched networks
- Optional VPN support for administrative access

---

8. System Models

8.1 Conceptual Architecture

- Mobile Client
- Wireless Network
- Cloud Backend
- Authority Interface

8.2 Security Model (Textual UML Perspective)

- Assets: user data, location data, system services
- Threats: eavesdropping, spoofing, malware, DoS
- Controls: encryption, authentication, authorization, logging

---

9. Other Requirements

- The system shall follow **secure software development life cycle (Secure SDLC)** principles.
- Regular security testing and updates shall be performed.
- The design shall consider Quality of Service (QoS) and scalability.

---

10. Appendix

A. Glossary

- **Confidentiality**: Protection of data from unauthorized access
- **Integrity**: Protection against unauthorized modification
- **Availability**: Ensuring system accessibility when required

B. Future Enhancements

- Multi-factor authentication (MFA)
- Advanced intrusion detection
- AI-assisted threat monitoring