



FACULTY OF ENGINEERING AND TECHNOLOGY

CEF 440: INTERNET & MOBILE PROGRAMMING

Project Title:

Design and Implementation of a Road Sign and Road State Mobile Notification Application (Smart Roads Ahead).

Task 3:

Requirement Analysis

Group 1	Matricule
ARREY-TABOT PASCALINE	FE22A151
TANUI NORBERT TANGIE	FE22A306
OROCKTAKANG MANYI	FE22A293
NGATTA GEORGE TABOT	FE22A259
FOMECHÉ ABONGACHU SIDNEY	FE22A218

Contents

1. Review and Requirement Analysis.....	3
2. Identification of Ambiguities, gaps and Missing Information.....	4
3. Prioritization of Requirements using MoSCoW (Must have, Should have, Could have, Won't have) Method	5
4. Classification of Requirements.....	9
Functional Requirements	9
Interactive Road Sign Library	9
Personalized Location-Based Notifications.....	10
Real-Time Traffic and Weather Alerts.....	10
Nearby Petrol Stations.....	11
User Feedback Mechanism.....	11
Non-Functional Requirements.....	12
Usability	12
5. Software Requirements Specification (SRS) and validation with stakeholders.	14
Conclusion.....	15

1. Review and Requirement Analysis

Requirement's analysis focuses on identifying and documenting the needs of stakeholders. It serves as a blueprint of the entire development process, ensuring that the final software product meets user expectations and business goals.

The aim of this phase is to analyze, validate, translate the user requirements into technical requirements and document the software requirements based on data gathered during requirement gathering to ensure completeness, clarity, and feasibility.

Categorizing responses observed from surveys and interviews;

>>Common Needs

1. Visual Library of Road Signs

Majority want a "Visual library of road signs and their meanings".

2. Real-Time Alerts

Several mention interest in "Real-Time Traffic alerts", "Weather updates", etc.

3. User-friendly Notifications

Most prefer "Pop-up alerts".

>>Common Suggestions

1. Mobile App Usefulness

Most respondents are open to using a mobile app for road sign education and traffic alerts.

2. Location-Specific Use

Frequent routes include: Buea-Douala, Limbe to Dla, Douala - Yaounde, etc.

>>Common Concerns

1. Unclear Road Signs

Many report encountering "Very frequently" unclear or unfamiliar road signs.

2. Missed or Misunderstood Signs

A large number confirmed they have misunderstood road signs.

3. Main Challenges While Commuting

>>Key concerns

Lack of road sign visibility
Poor road conditions
Traffic congestion

The requirements were reviewed using these metrics:

Completeness: All major user needs (e.g., sign recognition, alerting) are captured.

Clarity: Ambiguous phrases were rewritten to be specific and measurable.

Feasibility: Each requirement was compared against available Android features and our team's skill set.

Dependency Relationships: We traced dependencies like Camera access ML processing Notification to ensure logical coherence.

2. Identification of Ambiguities, gaps and Missing Information

Ambiguities or Conflicts:

1. Some respondents say they "don't drive" but still express needs like real-time alerts — clarify if they're passengers or learners.
2. Others check traffic via "word of mouth" or not at all — confirm if they'd be open to tech-based updates.

Gaps:

1. Limited mention of accessibility features (e.g., voice notifications for visually impaired users).
2. No detail yet on how user feedback/reporting of road issues would be handled.

Initial technical requirements:

- **Alert user quickly:** Alert user within 2 seconds using audio + vibration.
- **Have an easy-to-use interface:** One-tap UI with labeled icons and large buttons.

- **High recognition accuracy:** Recognition accuracy 90% in good lighting.
- **Usability:** Easy to use with an intuitive interface for all users.
- **Performance:** Optimum efficiently, providing quick responses without lag.
- **Scalability:** Increasingly accommodating more functionalities (updates and upgrades), users and data without performance degradation.
- **Offline mode** for basic navigation
- **French/English multilingual support**
- **99.99% service availability**
- **Robust data security for users'** information ensuring secure data transmission and storage.

3. Prioritization of Requirements using MoSCoW (Must have, Should have, Could have, Won't have) Method

This approach helps to categorize features based on their user impact and feasibility, ensuring that the development efforts are focused on delivering the most critical functionalities first.

1. Road Sign Education

This core function aims to educate drivers about the various road signs commonly found in Cameroon, improving their understanding and adherence to traffic regulations.

Potential Features:

- Visual Road Sign Directory: A comprehensive catalog of road signs categorized by type (e.g., regulatory, warning, guide). For each sign, provide a clear visual representation, the official name (if applicable in local context), a detailed explanation of its meaning, and potential consequences of non-compliance.
- Search Functionality: Allow users to search for specific road signs by keyword, category, or visual characteristics (e.g., color, shape).

- Offline Access: Enable users to access the road sign directory even without an active internet connection, crucial in areas with limited connectivity.
- Multi-Language Support: Provide explanations and names of signs in multiple languages commonly spoken in Cameroon (e.g., English, French, and potentially major local languages).
- Voice Assistance: Allow users to query information about a road sign using voice commands.
- Integration with Camera: Feature to allow users to take a picture of a road sign, and the application identifies it and provides information.

MoSCoW Prioritization:

- **Must have:**
 - **Visual Road Sign Directory**: Addressing the problem of drivers lacking access to sign information. Simply showing an image is insufficient. Clear explanations are crucial for understanding the meaning and implications of each sign.
 - **Search Functionality**: With a potentially large number of road signs, efficient search is essential for users to quickly find the information they need.
 - **Offline Access**: Given the potential for limited internet connectivity in various parts of Cameroon, offline access to the core road sign information is a critical requirement for accessibility and reliability.
- **Should have:**
 - **Quizzes and Practice Tests**: These interactive elements can significantly enhance user engagement and knowledge retention. Hence, greatly improving the educational value and user experience.

- **Multi-Language Support:** Providing information in multiple languages will significantly increase the application's reach and usability for a wider demographic.
- **Could have:**
 - **Voice Assistance:** This feature would add convenience and accessibility, allowing users to get information hands-free.
 - **Integration with Camera:** While innovative, accurately identifying road signs from a camera feed can be complex, It's a desirable but not essential feature for the initial release.
 - **Personalized Learning:** Tracking progress and providing personalized feedback can enhance the learning experience.
- **Won't have (for initial release):**
 - Advanced features like augmented reality overlays on live camera feeds showing sign information or integration with driving schools' curricula are considered beyond the scope of the initial release and can be considered for future iterations based on user feedback and resources.

2. Real-Time Traffic/Weather Updates

This function aims to provide drivers with timely information about current road conditions, enabling them to make informed decisions and avoid potential hazards.

Potential Features:

- Traffic Congestion Maps: Visual representation of traffic flow on major roads, indicating congestion levels (e.g., using color-coded overlays).
- Accident Reports: Real-time notifications and location marking of reported traffic accidents.
- Road Closure Alerts: Information about planned or unplanned road closures and alternative routes if available.
- Weather Hazard Warnings: Alerts for adverse weather conditions such as heavy rain, fog, storms, or landslides that may affect road safety.

- Crowdsourced Incident Reporting: Feature allowing users to report traffic incidents, road hazards (e.g., potholes), or unusual conditions.
- Integration with Traffic Cameras: Display live feeds or snapshots from traffic cameras along major routes.
- Integration with Weather APIs: Fetch real-time weather data and forecasts relevant to the user's location and routes.
- Historical Traffic Data: Access to historical traffic patterns to help users plan their journeys based on typical congestion times.
- Predictive Traffic Information: Utilizing data analytics to predict potential traffic congestion based on historical data, events, and real-time conditions.

MoSCoW Prioritization:

- **Must have:**
 - **Traffic Congestion Maps**: This is a fundamental aspect of real-time updates, providing a quick visual overview of traffic conditions. Integration with reliable data sources is crucial.
 - **Accident Reports**: Timely notification of accidents is critical for drivers to be aware of potential delays and hazards. Accurate location information is essential.
 - **Road Closure Alerts**: Information about road closures is vital for effective travel planning and avoiding unexpected detours.
 - **Weather Hazard Warnings**: Alerts about severe weather conditions directly impact safety and are a crucial component of real-time updates. Integration with reliable weather APIs is necessary.
- **Should have:**
 - **Crowdsourced Incident Reporting**: Allowing users to contribute to real-time data can significantly enhance the accuracy and coverage of information, especially in areas where official data

might be limited. A robust moderation system would be needed to ensure data quality.

- **Integration with Traffic Cameras:** Visual confirmation of traffic conditions through camera feeds can be very helpful for drivers. However, the availability and accessibility of traffic camera data might vary.
- **Could have:**
 - **Historical Traffic Data:** This can be a valuable tool for planning but is not essential for providing immediate real-time updates.
 - **Predictive Traffic Information:** Implementing accurate traffic prediction models can be complex and might require significant data and processing power. It's a desirable future enhancement.
- **Won't have (for initial release):**
 - Advanced features like personalized route optimization based on real-time conditions or integration with vehicle telematics for automated data collection are considered beyond the scope of the initial launch.

4. Classification of Requirements

Functional Requirements

Interactive Road Sign Library

The application must provide a comprehensive, searchable database of road signs with detailed explanations in both French and English. This library should support real-time updates to ensure users always have access to the latest sign information.

Key Features:

- High-quality images of road signs with zoom functionality.
- Textual descriptions of each sign's meaning, legal implications, and usage context.
- Categorization of signs (e.g., regulatory, warning, informational) for easier navigation.

- Search functionality allowing users to find signs by keywords or categories.

Implementation Approach:

To achieve this, the app will use a cloud-based NoSQL database (Firestore) for storing sign data, ensuring real-time synchronization. For offline access, a local SQLite database will cache the most frequently accessed signs. The user interface will include a search bar with filters and voice-command support for hands-free operation.

Personalized Location-Based Notifications

Users must receive real-time alerts about traffic conditions, weather hazards, and road closures based on their current location or selected route. Notifications should be customizable to match user preferences.

Key Features:

- GPS integration to track the user's location and route.
- Geofencing to trigger alerts when approaching hazardous areas.
- Customizable notification settings (e.g., mute alerts during nighttime).

Implementation Approach:

The app will integrate the Google Maps SDK for precise location tracking and route analysis. Firebase Cloud Messaging (FCM) will handle push notifications, while geofencing logic will ensure alerts are contextually relevant. Users will be able to adjust notification preferences in the app settings.

Real-Time Traffic and Weather Alerts

The app must provide live updates on traffic congestion, accidents, and weather-related road hazards (e.g., floods, fog). These alerts should be displayed on a map overlay for visual clarity.

Key Features:

- Crowdsourced incident reports from other users.
- Integration with weather APIs for real-time hazard alerts.

- Visual representation of hazards on an interactive map.

Implementation Approach:

Traffic data will be sourced from user reports and, where available, the Google Traffic API. Weather alerts will be fetched from OpenWeatherMap. The app will use Mapbox GL to overlay hazards on the map, with color-coded icons for quick recognition.

Nearby Petrol Stations

Users should be able to locate nearby petrol stations along their route, with details such as distance, operating hours, and fuel availability.

Key Features:

- Display of petrol stations on the map.
- Filtering options (e.g., 24-hour stations).
- Offline access to frequently visited stations.

Implementation Approach:

The Google Places API will provide data on petrol stations, which will be cached for offline use. The app will highlight stations along the user's route and allow filtering based on preferences.

User Feedback Mechanism

Users must be able to report incorrect sign information, road hazards, or app issues. These reports should be reviewed by moderators to ensure data accuracy.

Key Features:

- In-app forms for submitting feedback or hazard reports.
- Photo and GPS attachment for verification.
- Moderation dashboard for administrators.

Implementation Approach:

Firebase Storage will handle photo uploads, while Fire store will store report data. A moderation dashboard built with Firebase Admin SDK will allow authorities to validate reports before they are broadcast to other users.

Non-Functional Requirements

Usability

The app must be intuitive and accessible to users with varying levels of technical literacy, including older drivers and those in rural areas.

Key Features:

- Simple, clean interface with large buttons and clear icons.
- Voice-guided navigation and tutorials for first-time users.
- Support for low-resolution devices.

Implementation Approach:

The UI will follow Material Design 3 guidelines, with accessibility features like text-to-speech. User testing will be conducted with diverse groups to refine the interface.

Performance

The app must respond quickly to user inputs, even on low-end devices or slow internet connections.

Key Features:

- Load sign data and alerts in under 2 seconds.
- Smooth navigation and map rendering.

Implementation Approach:

Frontend optimizations include lazy-loading images and using Flutter's Skia engine for animations. Backend optimizations involve Firestore indexing and CDN caching for static assets.

Scalability

The system must handle increasing numbers of users and data without degradation in performance.

Key Features:

- Support for 10,000+ concurrent users.

- Efficient data storage and retrieval.

Implementation Approach:

Firebase Autoscaling will manage database loads, while Cloud Functions will handle resource-intensive tasks like image processing.

Offline Mode

Core features, such as the road sign library and basic navigation, must remain functional without an internet connection.

Key Features:

- Access to cached sign data and routes.
- Automatic sync when connectivity is restored.

Implementation Approach:

SQLite will store offline data, and Firestore's offline persistence will sync user reports once back online.

Multilingual Support

The app must support both French and English, with an easy language-toggle feature.

Key Features:

- Dynamic language switching.
- Localized content for regional dialects.

Implementation Approach:

Flutter's intl package will manage translations stored in JSON files, with language preferences saved to Firebase.

Security

User data must be protected, and privacy concerns addressed, especially for location tracking.

Key Features:

- Encrypted data transmission and storage.
- Anonymized public reports.

Implementation Approach:

TLS/SSL will secure APIs, and Firebase Auth will handle authentication. GPS data in reports will be generalized to protect privacy.

Each requirement is designed to address real-world challenges faced by Cameroonian drivers while ensuring scalability, usability, and security.

5. Software Requirements Specification (SRS) and validation with stakeholders.

1. System Overview: The Road Safety Mobile App aims to enhance commuter awareness and improve road safety by providing a visual guide to road signs and real-time updates on road conditions, including traffic, weather, and roadblock alerts. The application is intended for student drivers, commuters, and general users.

2.1. Functional Requirements:

Users can browse a categorized library of road signs with descriptions.
The app sends real-time traffic alerts based on user location.
Users receive pop-up notifications about weather and road conditions.
Instructors or drivers can provide feedback on road sign knowledge.
Users can customize notification preferences.
The app supports search and filtering of road signs.

2.2. Non-functional Requirements:

The app must be compatible with Android and iOS platforms.
Notifications should be delivered in under 10 seconds after detection.
The app must have a user-friendly interface and responsive design.
Minimal battery and data consumption.
High availability and uptime (>99%).

3. User Stories / Use Cases:

As a driver, I want to receive real-time traffic alerts so I can avoid delays.

As a learner, I want to study road signs so I can pass my driving test.

As a commuter, I want to get updates on weather and blocked roads so I can plan my trip better.

4. Assumptions and Constraints:

Assumes users have internet-enabled smartphones.

Government-provided Road data might be limited or unavailable.

Users' location data will be used for alerts (with permission).

5. Validation with Stakeholders:

The draft SRS was shared with a sample of student drivers, instructors, and general road users.

Stakeholder feedback included:

>>Strong preference for visual road sign guides.

>>Desire for real-time alerts.

>>Requests for offline access and language options.

Adjustments made:

Added offline support for sign library.

Included multilingual support as a "Should Have" item.

Simplified notification settings based on user feedback.

Conclusion

This Requirement Analysis provides a clear foundation for the Road Sign Recognition Apps design and implementation. It ensures that all features are prioritized, feasible, and validated with stakeholder approval and a complete SRS.

We are ready to proceed to system design in Task 4 Final validation involved reviewing and confirming that all key stakeholder concerns were addressed. The document is now locked in for development.

