# Requirement analysis and validation for a road sign and road state notification system

Reported by;

KANG JEVIS
EBANG BLESSING
FOUYOU CHINJEH BRAIN
VISEMIH JANICE

INSTRUCTOR: DR. NKEMENI VALERY

institution: Biaka University Institute Of Buea

**Table of content**

## 1) Review and analyse the requirements gathered;

This step critically evaluates the data gathered during requirement elicitation to ensure that the foundation of the application is robust, well-understood, and feasible to implement, especially considering the Cameroonian context.

**Overview of Spreadsheet Data**
The spreadsheet contains raw survey responses primarily focused on:

- Road condition challenges
- App usage behaviour
- Notification preferences
- Willingness to report hazards
- Privacy and technical concerns

Total usable responses: 45

**Key Insights from Survey Data**

**A. Common Driving Issues**

| Issue | % of Respondents Affected |
|---|---|
| Bad roads | 78% |
| Police stops | 62% |
| Potholes | 59% |
| Wrong signage | 43% |

Interpretation: The app should prioritize updates on potholes, poor roads, and police checkpoints. These are high-friction pain points.

**B. Road App Usage**

| | |
|---|---|
| Use Google Maps/Waze | 25% |
| Do not use navigation apps | 75% |

Implication: There is a gap in localized and accessible navigation solutions. The new app should offer a lightweight, offline-friendly alternative.

**C. Notification Preferences**

| Notification Type | % Preference |
|---|---|
| Voice Alerts | 65% |
| Pop-up Alerts | 48% |
| Vibration | 8% |

Design Implication: Voice alerts are essential and must support local languages (English,French, Pidgin). Vibration is least effective.

**D. Desired Alerts**

| | |
|---|---|
| Accident Reports | 82% |
| Potholes | 77% |
| Police Checkpoints | 64% |

Prioritization: Accident, pothole, and police alerts are the most critical features to implement in the MVP.

**E. Crowdsourcing Willingness**

| | |
|---|---|
| Will report hazards | 72% |
| Want anonymity | 59% |

Feature Design: Include a one-tap reporting interface and ensure anonymity to maximize adoption.

**F. User Concerns**

Privacy66%
Data Usage 49%
Battery Life 44%

Mitigation Strategy:

- Use anonymized GPS data.
- Implement offline caching and Wi-Fi syncing.
- Add a battery-saver mode.

## A. Completeness of Requirements

The gathered requirements do cover **major features and needs of the system**. Here's what is included and what's missing:

| Category | Included in Requirements | Remarks |
|---|---|---|
| Road Sign & State Alerts | Yes | Alerts based on user location and event severity |
| Real-Time Data | Yes | From traffic cameras, sensors, crowdsourced users |
| User Preferences | Yes | Users can customize notification type, area, and frequency |
| Navigation System Integration | Yes | Meant to allow in-app guidance or sync with existing apps |

## Analysis of Missing Requirements and Their Feasibility

| Missing Requirement | Feasibility | Implementation Options |
|---|---|---|
| **User Management (Login/Register)** | Yes | Easily implemented using Firebase Auth, JWT (for REST APIs), or social login (Google, Facebook). Can be optional for guest access. |
| **Moderation of Reports** (spam/fake reports) | Yes | Build in a report flagging system + admin panel for reviewing reports. Add logic to limit abuse (e.g., max daily reports, CAPTCHA). |
| **Admin Dashboard** (for updating road sign DB, reviewing reports) | Yes | Can be built as a web app using React/Angular + backend APIs. Admin roles can control who sees/modifies road or report data. |

| | | |
|---|---|---|
| **Offline Support / Caching** | Partially | Full offline maps/alerts may be heavy. But basic caching of recent signs, local hazards, and notifications is doable via SQLite or Room DB on Android. |
| **In-App Map View of Signs** | Yes | Use OpenStreetMap or Mapbox SDK to display road signs and hazard pins in-app. Allows full customization and works offline with preloaded tiles. |
| **Notification Severity Filtering** | Yes | Let users choose notification levels (e.g., "All alerts", "Critical only", "Road closures only"). Stored in user settings/preferences. |
| **Feedback Mechanism** (users report app issues) | Yes | Simple form or in-app feedback tool. Store in DB or send via email to dev/admin. |
| **Localization (French/English)** | Yes | Cameroon is bilingual. Android/iOS support multi-language apps with string resource files. |
| **Notification History / Log** | Yes | Store past alerts locally. Use SQLite/Room on Android or CoreData on iOS. Useful when users miss a notification. |

## 2**. Clarity of Requirements;**

we'll now examine the questions/issues that need clarification,in order to elimnate inconsistences and ambiguities, especially to tailor the app to **Cameroon-specific standards, traffic laws, and technical constraints**. Below is a detailed assessment with action-oriented questions and clarity refinements.

---

# Clarity of Requirements — Assessment and Refinement

---

### 1. Supported Road Signs – What types? Cameroon-specific?

**Issues:**

- Cameroon follows a mix of **Vienna Convention**-style road signs but also includes **local variations**.
- No national public API or online catalog of official road signs.

**Clarification Required:**

- What categories of road signs will the app support? (Regulatory, Warning, Guide, Information, etc.)

- Will the signs include **images and local descriptions in French and English**?

**Recommendation:**

- Source road signs from **Cameroon’s Ministry of Transport** or licensed driver manuals.
- Build a **local database of signs** (with icon, meaning, location, and usage context).

---

## 2. Integration with Navigation Systems – What method?

**Issues:**

- Google Maps and Apple Maps **do not allow direct overlay of persistent custom alerts** without their Enterprise APIs.
- **Cameroon map data** might not be as detailed as developed countries.

**Clarification Required:**

- Do we aim to:
  - Overlay alerts/signs within external navigation apps (e.g., Waze)?
  - Or embed a **custom map view inside the app** with OpenStreetMap/Mapbox?

**Recommendation:**

- Use **OpenStreetMap or Mapbox SDK** to build an **in-app navigation/map** system with full control.
- Consider **linking to external apps** (Google Maps/Waze) when giving directions, but not for alert overlays.

---

## 3. Crowdsourced Reports – What kind of reports? What validation?

**Issues:**

- Without moderation, users could spam or misuse the reporting tool.
- Users may not know how to classify hazards accurately.

**Clarification Required:**

- What types of hazards can users report? (e.g., potholes, accidents, roadblocks, floods, missing signs)
- Should users attach **photos, location, and category**?
- Will there be a **moderation or reputation system**?

**Recommendation:**

- Limit initial report types to 4–5 categories.
- Require GPS location + optional photo upload.
- Use a **flag/report abuse** system for spam detection.
- Build a **lightweight admin panel** for manual moderation.

---

## 4. Data Sources – Traffic, Weather, Road States

**Issues:**

- Cameroon lacks a central open traffic API like Google’s Waze feed.
- Weather APIs (like OpenWeatherMap) have **low-resolution local data**.

**Clarification Required:**

- What is the intended data source for:
    - **Traffic congestion**
    - **Weather and road hazards**
    - **Roadworks/closures**

**Recommendation:**

Use:

- **Crowdsourced data** from users for road hazards.
- **OpenWeatherMap API** for weather alerts.
- Partner with **local road authorities** for traffic updates, if possible.
- Integrate **SMS-based alerts** from road authorities as a backup.

---

## 5. User Preferences – How granular should filtering be?

**Issues:**

- "Customizable notifications" is vague.

**Clarification Required:**

- Should users filter:
    - By **region/route**?
    - By **type of alert** (e.g., weather only)?
    - By **time of day** (e.g., only during commute)?

**Recommendation:**

- Implement basic preferences first:
  - Alert types (Road sign info, Hazards, Closures, Weather)
  - Geographical filtering (e.g., within 50km radius)

---

## 6. Language and Localization – Is bilingual support required?

**Issues:**

- Cameroon is officially bilingual (French and English), but local dialects exist.
- Legal/official road data might only be in French.

**Clarification Required:**

- Should the app support **both English and French UIs and notifications**?

**Recommendation:**

- Implement **full bilingual support**.
- Store road sign descriptions and messages in both languages.
- Allow language switching in settings.

---

## 3) Prioritizing requirements based on importance and feasibility in cameroon

| Module | Feasibility | Notes |
|---|---|---|
| GPS Tracking | ✅ High | Fully supported on devices |
| Road Sign Database | ⚠️ Medium | Needs manual or sourced dataset |
| Real-Time Traffic/Hazards | ⚠️ Moderate | Crowdsource or partner with agencies |
| Weather Data | ✅ High | OpenWeatherMap API sufficient |
| Crowdsourced Reporting | ✅ High | Simple and scalable |
| Push Notifications | ✅ High | Use Firebase |
| Voice Alerts | ✅ High | Android/iOS native TTS |
| Navigation Integration | ⚠️ Moderate | Use custom maps like OSM/Mapbox |
| User Preferences | ✅ High | Local/backend settings possible |
| French/English Language | ✅ High | Use i18n for full bilingual UX |
| Backend Data Storage | ✅ High | Firebase or cloud backend |

## 4. Classify Requirements;

for the *Road Sign and Road State Mobile Notification Application*, specifically organizing them into:

- **Functional Requirements**: What the system **should do**
- **Non-Functional Requirements**: How the system **should behave**

This classification will help guide architecture, development, and testing strategies.

---

# REQUIREMENT CLASSIFICATION

---

### Functional Requirements (FR)

These define the **core functionality** of the system — what features it must provide:

| ID | Functional Requirement | Description |
|---|---|---|
| FR1 | **User registration and login (optional)** | Allow users to create accounts (optional if anonymous usage is allowed). |
| FR2 | **GPS-based location tracking** | App tracks user location to determine proximity to road signs or hazards. |
| FR3 | **Road sign identification and display** | Based on location, the app displays road signs with their meanings. |
| FR4 | **Real-time road condition alerts** | Show updates on traffic, accidents, weather hazards, and roadblocks. |
| FR5 | **Crowdsourced reporting module** | Users can report hazards, accidents, or missing signs. |
| FR6 | **Notification system** | Push or in-app alerts for road signs and hazards. |
| FR7 | **Customizable alerts** | Users choose which types of signs or hazards to be notified about. |
| FR8 | **Offline access to road sign database** | Cached signs can be viewed when offline. |
| FR9 | **Multilingual support** | Provide content in both English and French. |
| FR10 | **Integration with navigation systems** | overlay information onto existing apps (like Waze). |
| FR11 | **Admin backend (moderation)** | Allow moderators/admins to manage user reports and content. |
| FR12 | **User feedback system** | Let users report inaccurate data or rate the app. |

---

### Non-Functional Requirements (NFR)

These define **system qualities —** performance, usability, security, etc.

| ID | Non-Functional Requirement | Description |
|---|---|---|
| NFR1 | **Performance** | App should respond to user actions within 2 seconds. |
| NFR2 | **Availability** | System must be available 24/7, with minimal downtime. |
| NFR3 | **Scalability** | System must support scaling to thousands of users across regions. |
| NFR4 | **Security** | GPS data and user reports must be protected; implement basic encryption and secure APIs. |
| NFR5 | **Privacy compliance** | App must comply with local data protection regulations in Cameroon (if any) and best practices. |
| NFR6 | **Maintainability** | Modular code design for easy updates and future feature additions. |
| NFR7 | **Accessibility** | App should be usable by visually impaired users (voice support, large text options). |
| NFR8 | **Localization** | App must support text and audio in both French and English. |
| NFR9 | **Low bandwidth optimization** | App should function efficiently on slow or intermittent internet connections. |
| NFR10 | **Battery efficiency** | GPS and background processes must be optimized to preserve battery. |

**After this assessment we proceed to developing the Software Requirements Specification (SRS).**

**And then validation**