THE UNIVERSITY OF BUEA REPUBLIC OF CAMEROON

P.O.BOX 63,

Buea, South West Region Cameroon

CEF 440

Internet Programming and Mobile Programming

Department: Computer Engineering

**Faculty Of Engineering and Technology**



**TASK 5**

**Road Sign and Road State Mobile Notification**

**Application**

**UI Design & Implementation**

Submitted to:

Dr. Nkemeni Valery

By:

Group 14

2023/20

**Contents**

[1. Introduction 1](#_Toc168953576)

[2. User Interface (UI) Design 1](#_Toc168953577)

[2.1. Screens 2](#_Toc168953578)

[2.1.1. User Registration and Authentication 2](#_Toc168953579)

[2.1.2. Dashboard 4](#_Toc168953580)

[2.1.3. Map View 5](#_Toc168953581)

[2.1.4. Reports 7](#_Toc168953582)

[2.1.5. Notifications 8](#_Toc168953583)

[2.1.6. Recent Updates 9](#_Toc168953584)

[2.1.7. Settings 10](#_Toc168953585)

[3. Implementation of the UI Design using React Native 12](#_Toc168953586)

[3.1. Libraries and Components: 12](#_Toc168953587)

[4. Conclusion 12](#_Toc168953588)

# Introduction

This report details the design and implementation phase of a mobile application that utilizes smartphone technology to notify drivers of road signs and real-time road state information.

To create a successful mobile app, a design process that involved research, wireframing,and prototyping was followed.

**Research and Analysis**

It began by researching the target audience and their needs. User interviews were conducted with drivers to understand their needs and expectations from our application. Competitive research was carried out to see what other apps were available in the market and what features they offered.

**Wireframing**

After conductng research, wireframes were created to map out the user flow and the app’s layout.The wireframes allowed us to visualize the app’s structure and ensure that the user’s flow was intuitve.

**Prototyping**

With the wireframes finnalized, a prototype of the app was created using Figma, a collaborative design tool. The prototype allowed us to test the user’s flow and user interface design.

Using the requirements gathered and models built during the previous phase, a UI design was made.

# User Interface (UI) Design

The UI design prioritizes clarity, ease-of-use, and minimal distraction for drivers. Key considerations include:

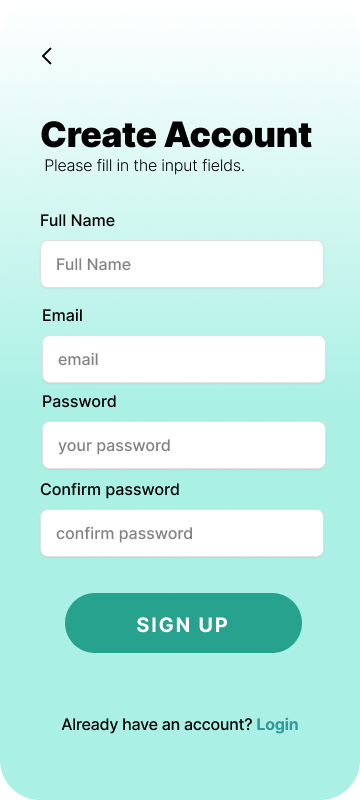
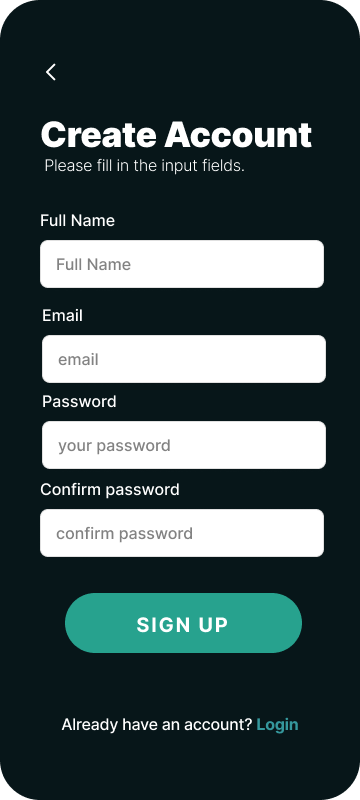
* **User Registration and Authentication**: This allows drivers to access and use the functionalities of the application. It involves signing up (for new users) or logging in (for users who already have an account).
* **Dashboard**: The dashboard would function as a central hub for monitoring the application's health, understanding user behavior, and managing the data it utilizes.
* **Map View:** A primary map view displays the driver's location and surrounding area. Road signs detected by the application are highlighted on the map with clear icons.
* **Reports:** Drivers will be able to make reports on various road signs spotted on the road and road conditions.
* **Settings:** Users will be able to customize their preferences and experience.
* **Notifications:** Interface that alerts drivers of upcoming signs or changes in road state.

## 2.1. Screens

### 2.1.1. User Registration and Authentication

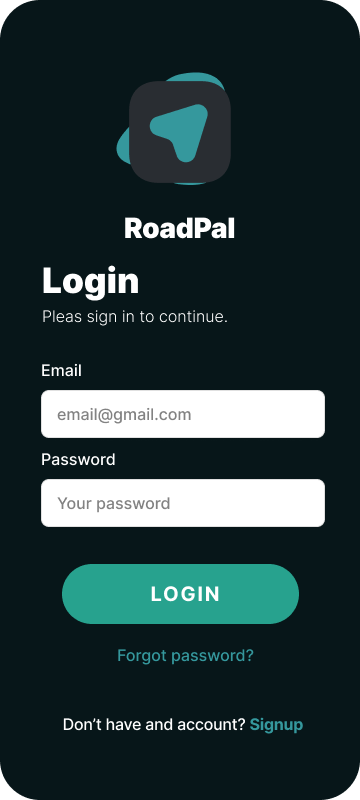
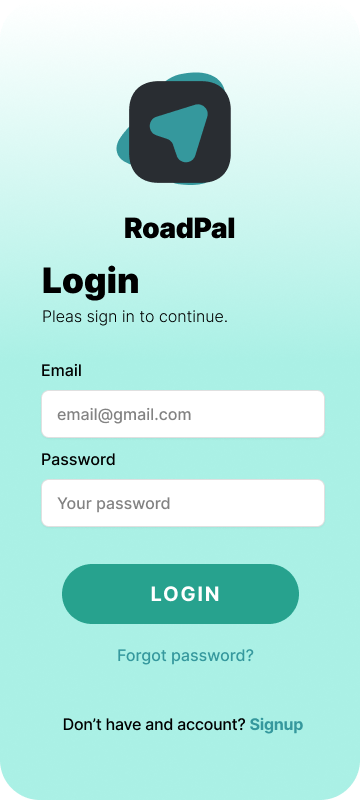
1. **Sign Up**

* Allows new users to create an account with the application.
* Involves collecting user information like email address, username, and password.



1. **Log In**

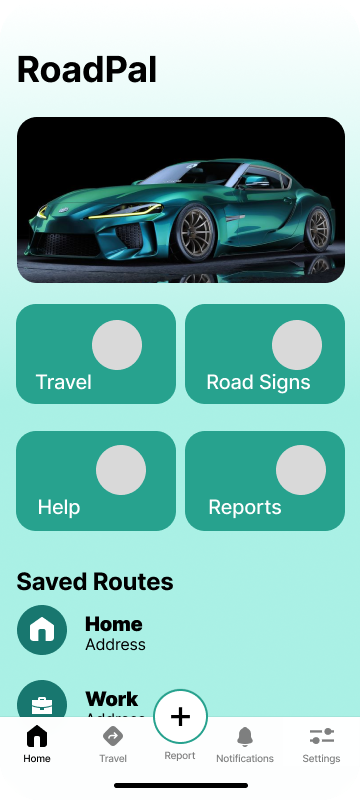
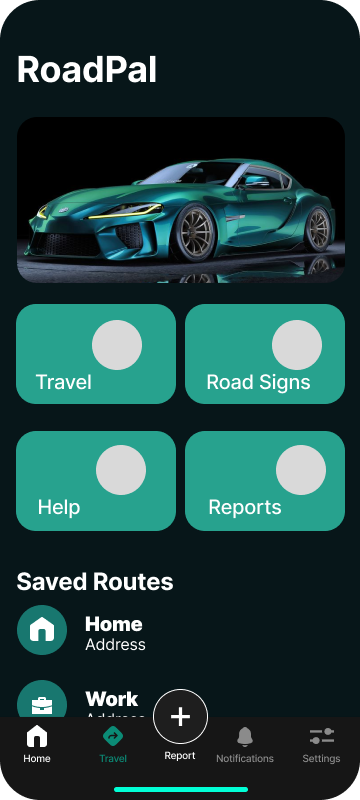
* Enables existing users to access their accounts after signing up.
* Requires users to enter their credentials (username/email and password) to verify their identity.
* Provides a secure way to manage user sessions and access control within the application.



These screens are needed for:

* Personalization: Sign-up allows users to create personalized profiles and preferences within the app.
* Data Storage: Account creation enables users to store and access their data within the app (e.g., saved locations, past interactions).
* Security: Login provides a secure way to authenticate users and control access to sensitive features or information.
* Analytics: User accounts can facilitate data collection and analysis of user behavior to improve the app over time.

### 2.1.2. Dashboard



The dashboard contains:

**Travel:** This functionality allows the user to set his/her travel route.

**Help:** The help screen can provide step-by-step instructions on using the app's main features. This includes explaining how to navigate the map view, access sign information panels, or customize app settings and aid in troubleshooting.

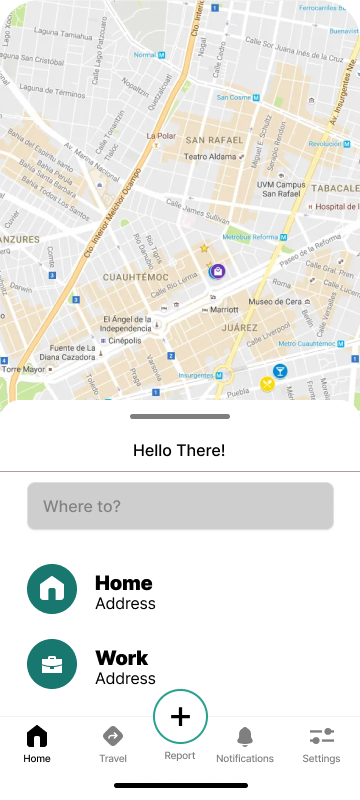
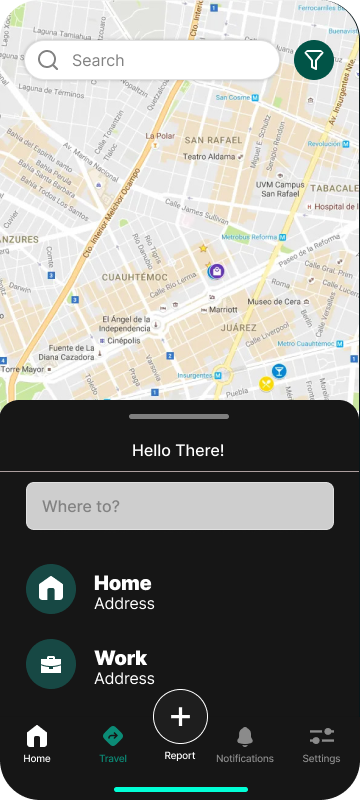
**Road Signs**: A little but expandable collection of road signs and their descriptions for the purpose of educating the various users of the significance of the road signs they might come across.

**Reports:** Allows users to view recent reports made by other users concerning road conditions and signs.

### 2.1.3. Map View

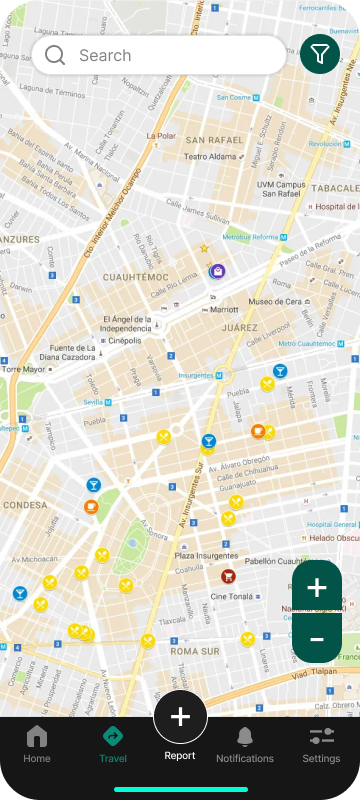
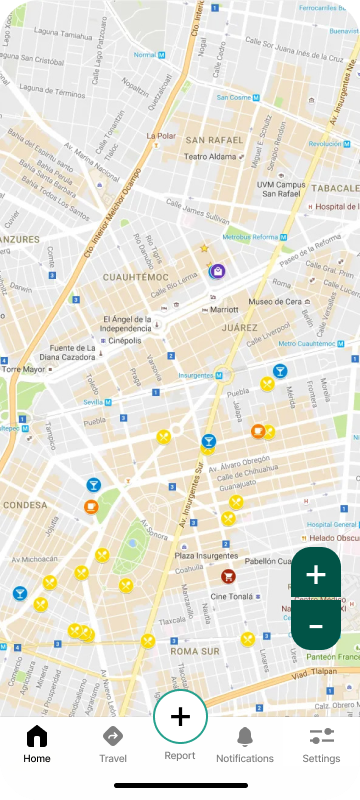
This screen displays:

* The option for the user to set his/her destination
* Then displays the map of the chosen route.
* Also allows the user to store the maps of frequently used routes for offline use and easy access.

This is a full display of the map, once route has been set.

* Highlighting Signs on the Map: The map view uses makers to highlight available road signs on the map.



### 2.1.4. Reports

A screenshot of a phone

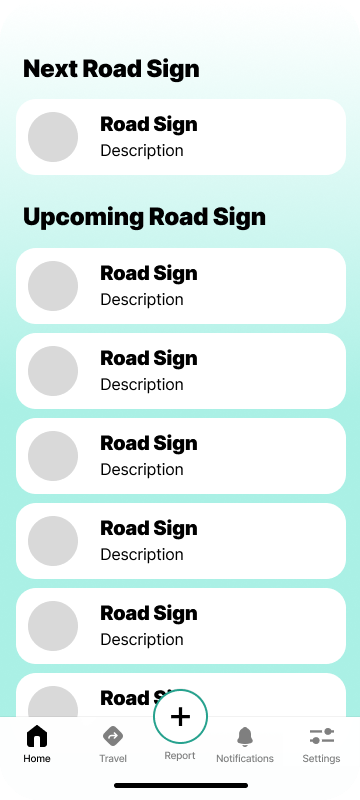
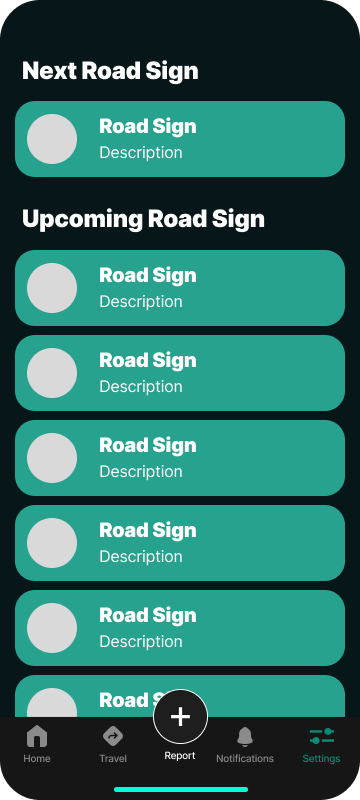
Description automatically generated A screenshot of a phone

Description automatically generated

The report interface permits the users to:

* **Report Sign Detection Errors:** Users can report instances where the app fails to detect a road sign, detects a sign incorrectly, or displays inaccurate information about a detected sign. This allows for continuous improvement of the app's capabilities.
* **Suggest Missing Signs:** Users can report encountering signs that the app doesn't currently recognize in its database. This helps the expansion of the app's sign library and ensures it remains comprehensive for different regions or signage variations.
* **Providing Feedback on Traffic Data:** Users can report discrepancies or inaccuracies they observe in the real-time traffic data displayed on the app. This feedback can help identify issues with the traffic data source or improve the way the app interprets and displays traffic information.

### 2.1.5. Notifications



Displaying Real-time Sign Alerts is the core functionality of the notification screen. It proactively alerts drivers about upcoming road signs by:

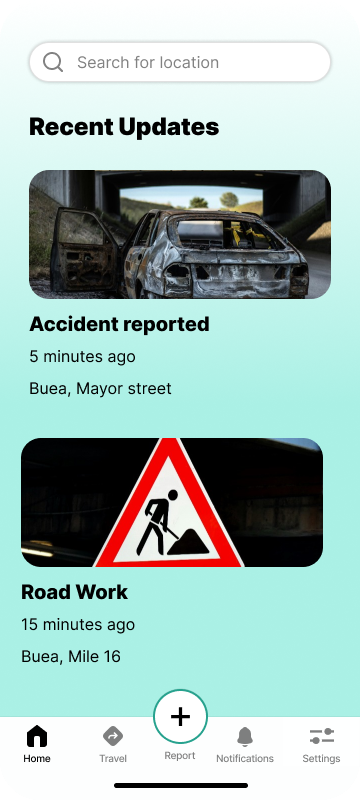
* **Displaying Text Notifications:** Text notifications can appear on the user's phone screen (with appropriate settings enabled) to warn them about upcoming signs.
* **Playing Audio Notifications:** The app can offer optional voice notifications that announce upcoming signs, providing an additional layer of awareness for drivers who might be glancing away from the screen momentarily.

The notification screen can also leverage the app's access to real-time traffic data to:

* **Alert Users about Traffic Congestion:** Notifications can warn drivers about upcoming traffic jams or slowdowns, allowing them to adjust their routes or anticipate delays.
* **Inform Drivers of Accidents**: Real-time accident alerts displayed on the notification screen can help drivers avoid potential hazards and reroute if necessary.
* **Provide Updates on Weather Conditions:** When integrated, the app can display weather alerts on the notification screen, informing drivers about potential road hazards like heavy rain, fog, or icy conditions.

### 2.1.6. Recent Updates

This interface simply provides the user with recent information about reported data and app developments and upgrades.

 A screenshot of a phone

Description automatically generated

### 2.1.7. Settings

Here, users can edit their profile (information about them which the application stores) as well as their preferences.

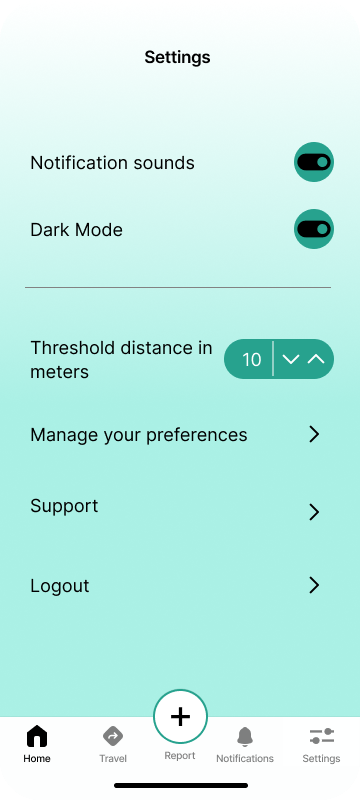
1. **User profile**

A screenshot of a computer

Description automatically generated A screenshot of a login form

Description automatically generated

1. **Preferences**

A screenshot of a phone

Description automatically generated

**Notifications:** Users can control notification preferences, such as enabling/disabling notifications for upcoming signs, changes in road state (traffic, accidents, weather), or setting notification sounds and vibration patterns.

**Themes:** Users might be able to choose from different color themes for the app's interface (e.g., light mode, dark mode) to personalize their experience and improve visibility in different lighting conditions.

**Units:** Users can select their preferred unit system for displaying speed limits (e.g., miles per hour, kilometers per hour).

# Implementation of the UI Design using React Native

React Native provides a powerful toolkit for building cross-platform mobile applications. Here's a breakdown of how we implemented the UI for our road sign notification app:

**Benefits of React Native:**

* React Native allows for building a single codebase for both iOS and Android platforms, saving development time and resources.
* It leverages familiar JavaScript syntax, making development easier for those with web development experience.
* The rich ecosystem of third-party libraries provides readily available components for maps, navigation, and other functionalities.

## 3.1. Libraries and Components:

* **Navigation:** Use a navigation library like react-navigation to manage different screens within the app (e.g., map view, sign information panel).
* **Maps:** Integrate a mapping library like Mapbox or Google Maps to display the user's location and surrounding area.
* **Markers:** Utilize marker components provided by the mapping library to highlight detected road signs on the map.
* **Icons:** Utilize icon libraries like Material Icons or Font Awesome to represent road signs and road state information.
* **Cards and Panels:** Implement components like View and Card from React Native to display sign information and road state overlays.
* **Text and Images:** Use basic components like Text and Image to display information and sign images.

By leveraging React Native's capabilities, we were able to create a user-friendly and informative UI for the road sign notification application.

# Conclusion

This mobile application has the potential to significantly improve driver awareness of road signs and real-time road conditions. By prioritizing a user-friendly interface and leveraging cutting-edge technologies, the application can contribute to safer and more efficient driving experiences.