Bilkent University
Department of Computer Engineering

Senior Design Project
T2511
E-Way

Analysis and Requirement Report

22103511, Utku Yüksel, utku.yuksel@ug.bilkent.edu.tr
22103680, Furkan Özek, furkan.ozek@ug.bilkent.edu.tr
22102898, Halis Vefa Türkyılmaz, vefa.turkyilmaz@ug.bilkent.edu.tr
22102800, Aziz Üzümcü, aziz.uzumcu@ug.bilkent.edu.tr

Supervisor: Salih Özgür Öğüz
Course Instructors: Mert Bıçakçı

December 19, 2025

This report is submitted to the Department of Computer Engineering of Bilkent University in partial fulfilment of the requirements of the Senior Design Project course CS491/2.

Contents

Analysis and Requirement Report

1 Introduction

With the increasing global shift towards sustainable transportation, the adoption of Electric Vehicles (EVs) has gained significant momentum in Türkiye. While the number of EVs and charging stations is rising rapidly, drivers continue to face critical challenges such as range anxiety, unpredictable charging station availability and long waiting times due to congestion. Despite its expansion, the current infrastructure lacks a uniform method for predicting station occupancy in the near future or in real time, which frequently results in EV users making ineffective trip plans.

E-Way is an intelligent route planning system designed to address these challenges by predicting charging station occupancy and optimizing travel routes accordingly. The primary objective of the project is to collect and analyze historical data from public sources to estimate station availability using artificial intelligence models. Unlike traditional navigation tools, E-Way incorporates these AI based predictions into route generation allowing users to avoid congested stations and minimize total travel time.

The system is designed not only for efficiency but also for user comfort. It provides detailed station information including socket types and power levels and highlights nearby amenities such as cafés, markets and rest zones based on user preferences. By combining historical data analysis, machine learning and user centralized design, E-Way aims to make EV travel smarter and more predictable.

2 Current System

Currently, Electric Vehicle (EV) drivers in Türkiye rely on a fragmented ecosystem of applications to plan their journeys and find charging stations. There is no single unified system that combines route planning with comprehensive, real time or predictive station data across all network operators. The existing solutions can be categorized into three main groups each with significant limitations regarding the problem E-Way aims to solve.

The most common tools are general navigation applications like Google Maps. While these platforms offer basic EV routing and station location services, they lack detailed, Turkey specific data such as socket types, real time power levels and historical occupancy analysis. Crucially, they do not provide future occupancy predictions meaning a user cannot know if a station will be congested by the time they arrive.

The second category consists of operator specific mobile applications (e.g., Eşarj, ZES). These apps provide accurate real time data but are limited exclusively to their own charging networks. A driver planning a long distance trip often needs to switch between multiple applications to find available stations from different providers which is inefficient and distracting while driving. Furthermore, these apps typically do not offer cross operator route optimization or AI based demand prediction.

The third category includes crowdsourced platforms like PlugShare. While useful for finding station locations and reading user reviews, these platforms rely heavily on user inputs which can lead to outdated or inaccurate information. They generally lack the integrated routing capabilities and predictive analytics required for reliable long distance travel planning.

In summary, the current system forces EV drivers to manually synthesize information from multiple sources leading to uncertain travel times. There is a clear lack of an integrated solution that offers predictive, data driven route optimization tailored to the specific charging infrastructure of Türkiye.

3 Proposed System

3.1 Overview

E-Way is proposed as a comprehensive intelligent route planning system specifically tailored for Electric Vehicle (EV) users in Türkiye. The primary goal of the system is to reduce range anxiety and optimize travel efficiency by predicting charging station occupancy and generating smart routes based on these predictions.

The proposed solution operates by continuously collecting historical and real time data from the EPDK EV Charging Network and other public sources [1]. This data is stored in a centralized cloud database to form a rich dataset containing hourly, daily and seasonal usage trends. An Artificial Intelligence (AI) model within the system utilizes this historical dataset to predict station availability, congestion levels and potential wait times for the user's expected arrival time.

When a user initiates a navigation request, the system calculates multiple route alternatives. Unlike standard navigation tools, E-Way integrates these AI driven occupancy predictions into its routing algorithm. This ensures that the recommended path not only considers distance and traffic but also avoids charging stations predicted to be busy or crowded.

The system architecture is modular and consists of five major layers: Client, Communication, Backend, Logic, and Storage. The Client Layer is a cross platform mobile application (developed in Flutter) that serves as the user interface. The Backend Layer (FastAPI) acts as the central coordinator, managing data flow between the user, the AI logic and external APIs. The Logic Layer houses the machine learning models responsible for occupancy prediction and intelligent data processing.

Furthermore, E-Way emphasizes user comfort by integrating contextual data. Users can customize their routing preferences based on charging socket types (AC/DC), network operators and nearby amenities such as cafés, markets or shopping malls. The system dynamically highlights these facilities during route planning, offering a personalized and comfortable travel experience.

3.2 Functional Requirements

This section defines the specific behaviors and functions of the E-Way system. These requirements describe the interactions between the users and the system, as well as the internal processes required to fulfill those interactions.

3.2.1 User Functionalities

These requirements cover the features directly accessible to EV drivers via the mobile application.

- **Account Management:** Users shall be able to create an account and log in securely using email/password or third party providers (Google).
- **Vehicle Profile:** Users should be able to input and save their vehicle information including battery capacity and consumption rates to ensure accurate route calculations.
- **Personalization:** Users shall be able to specify and save preferences such as preferred charging networks (e.g., Eşarj, ZES), socket types (e.g., AC Type-2, DC CCS) and preferred nearby amenities (e.g., cafés, shopping centers).
- **Route Planning:** Users should be able to generate routes by entering a start point and a destination. They can choose from multiple route alternatives based on criteria like shortest time, lowest cost or comfort based.
- **Map Interaction:** Users shall be able to view the recommended route, charging stations and their status on an interactive map or list view.
- **Station Details & Filtering:** Users should be able to view detailed information for each station (power, socket types, operator, predicted occupancy) and filter stations based on city, operator or power rating.
- **Notifications:** Users should receive alerts for critical events such as when predicted congestion exceeds a threshold or battery levels are insufficient, and shall have the option to enable/disable these notifications.

3.2.2 System Functionalities

These requirements define the backend processes, data handling and AI operations performed by the system to support user requests.

- **Data Integration:** The system shall fetch station data (metadata, socket info) from the official EPDK Charging Station API and update the database accordingly.
- **External API Connection:** The system connects to mapping and routing APIs (e.g., Google Maps) and amenities APIs (e.g., Google Places) to retrieve route data and nearby facility information.

- **AI Based Occupancy Prediction:** The system shall retrieve historical occupancy data and utilize the AI model to predict the expected occupancy percentage of a station for the user's estimated time of arrival.
- **Smart Labeling:** The system shall categorize stations based on predicted occupancy levels: Available (0-30%), Moderate (30-70%) and Busy (70-100%).
- **Route Optimization:** The system shall integrate AI predictions into the routing logic to actively avoid occupied stations and suggest alternatives with lower waiting times.
- **Error Handling:** The system shall provide meaningful error messages in case of API failures and automatically retry essential data collection processes.
- **Security & Validation:** The system shall validate all requests to ensure data privacy and authorized access to user profiles.

3.2.3 Admin Functionalities

These requirements cover the capabilities provided to system administrators for maintenance and monitoring.

- **System Monitoring:** Administrators should be able to view system statistics including the total number of stations, size of occupancy history data and registered user counts.
- **Model Retraining:** Administrators shall have the ability to trigger a manual retraining process for the AI prediction model to incorporate new data and improve performance.

3.3 Non-functional Requirements

This section specifies the quality attributes, performance goals and constraints under which the E-Way system shall operate.

3.3.1 Usability

- **Intuitive Interface:** The mobile application shall provide a user friendly interface designed specifically for drivers. Navigation, station viewing and route selection shall require minimal interaction to ensure safety.
- **Visual Clarity:** Visual elements such as occupancy indicators (color coded), maps and icons shall be legible at a glance.
- **Accessibility:** The system should follow standard accessibility guidelines to ensure usability for a wide range of users.

3.3.2 Reliability

- **Data Consistency:** The system shall reliably present charging station data. In cases of temporary external API failures (e.g., EPDK API delays), the system should display the last known valid data rather than an empty state.

- **Service Availability:** Critical backend services, such as the routing engine and user account management, shall ensure high availability.
- **Graceful Degradation:** The application shall handle unexpected errors (e.g., connectivity loss) gracefully without crashing, informing the user appropriately.

3.3.3 Performance

- **Query Efficiency:** Database queries such as filtering stations or retrieving historical occupancy data shall return results efficiently. This requires proper indexing strategies in the MongoDB database.
- **Real Time Interaction:** The system shall support near real time updates for recalculating routes or refreshing station status while a trip is in progress.

3.3.4 Supportability

- **Modularity:** The system architecture shall be modular (Client, Backend, Logic, Storage layers) to allow different teams to work on specific components without affecting the whole system.
- **Maintainable Codebase:** The backend shall adhere to clean API design principles (RESTful endpoints, standard JSON schemas) to facilitate debugging, testing and future feature integration.

3.3.5 Scalability

- **Horizontal Scaling:** The backend architecture (FastAPI) and database (MongoDB Atlas) shall support an increasing number of current users and growing historical datasets without a significant performance decrease.
- **Extensibility:** The system design shall allow for the easy addition of new data sources (e.g., new charging networks or amenities APIs) with minimal architectural changes.
- **AI Model Adaptation:** The AI prediction module shall be capable of being retrained on larger datasets as the historical occupancy records grow over time.

3.4 Pseudo Requirements

1. **Mobile Development Framework:** The mobile application shall be developed using Flutter to ensure cross platform compatibility (Android and iOS) from a single codebase.
2. **Backend Language & Framework:** The backend services shall be implemented in Python using the FastAPI framework due to its high performance and support for asynchronous processing [2].
3. **Database System:** MongoDB Atlas (Cloud) shall be used as the primary database for storing station data, user profiles and historical occupancy records.

4. **Architectural Style:** The system shall follow a modular, service oriented architecture with a RESTful API Gateway for client server communication.
5. **External Data Sources:** The system shall integrate with the EPDK EV Charging Network API for station data and Google Maps Platform (Maps, Routes, Places APIs) for navigation and amenity data.
6. **Documentation Standards:** Software requirements shall be documented according to IEEE 830-1998 and system models shall be designed using UML 2.5.1 standards.
7. **API Specification:** REST APIs shall be designed and documented in accordance with the OpenAPI Specification 3.0.
8. **Data Privacy Regulations:** The storage and processing of user data shall comply with KVKK (Law on Protection of Personal Data) regulations.

3.5 System Models

3.5.1 Scenarios

Scenario 1: New User Registration and Vehicle Setup

- **Participating Actor:** EV Driver
- **Precondition:** User has downloaded the app and has internet connectivity.
- **Flow of Events:**
    1. The user opens the E-Way mobile application.
    2. The user selects "Sign Up" and enters their email and password (or uses Google Login).
    3. The system creates the account and prompts for vehicle details.
    4. The user enters vehicle information (e.g., battery capacity, average consumption).
    5. The user sets personal preferences such as preferred socket type (e.g., DC CCS) and amenities (e.g., Café).
    6. The user saves the profile.
- **Post-condition:** User profile is stored in the database with vehicle and preference data.

Scenario 2: Smart Route Planning with Occupancy Prediction

- **Participating Actor:** EV Driver
- **Precondition:** The user is logged in and has enabled GPS.
- **Flow of Events:**
    1. The user enters a destination address on the home screen.
    2. The system retrieves user preferences (e.g., comfort-based routing).
    3. The backend calculates route alternatives, fetches station data, and generates AI-based occupancy predictions using historical data, which are created according to the estimated arrival time at the station.

4. The system displays route options, highlighting stations labeled as "Available", "Moderate", or "Busy".
5. The user selects the recommended route, which avoids a high-occupancy station.
6. The system displays the route on the map with charging stops and nearby amenities.

- **Post-condition:** Navigation starts with an optimized route.

Scenario 3: Viewing Station Details and Filtering

- **Participating Actor:** EV Driver
- **Precondition:** User is viewing the map interface.
- **Flow of Events:**
    1. The user wants to find a specific type of charger nearby.
    2. The user opens the filter menu and selects "DC Fast Charger" and "ZES" operator.
    3. The system updates the map to show only matching stations.
    4. The user clicks on a specific station icon.
    5. The system displays a detailed card showing power (kW), socket types, predicted occupancy percentage and nearby amenities.
- **Post-condition:** User has the necessary information to decide whether to drive to that station.

Scenario 4: Admin Model Retraining

- **Participating Actor:** System Administrator
- **Precondition:** Admin is logged into the web dashboard.
- **Flow of Events:**
    1. The admin views the system statistics and notices a significant increase in new historical data.
    2. The admin navigates to the "AI Model Management" section.
    3. The admin clicks the "Trigger Retraining" button.
    4. The system initiates the training pipeline using the latest dataset.
    5. The system notifies the admin upon successful completion.
- **Post-condition:** The AI occupancy prediction model is updated with the latest data patterns.

Scenario 5: User Feedback About the Charger

- **Participating Actor:** EV Driver
- **Precondition:** User’s location indicates that (s)he is leaving a charger on the route.
- **Flow of Events:**
    1. The user is asked if (s)he wants to give information about the charger by a pop-up form.
    2. If the user wants, he/she enters the number of empty slots, the time (s)he passed while waiting and general satisfaction about the station.
    3. The user saves the feedback and continues his/her route.
    4. The system saves the data that the user provided.

- **Post-condition:** The AI occupancy prediction model is updated with the latest data patterns.

3.5.2 Use-Case Model

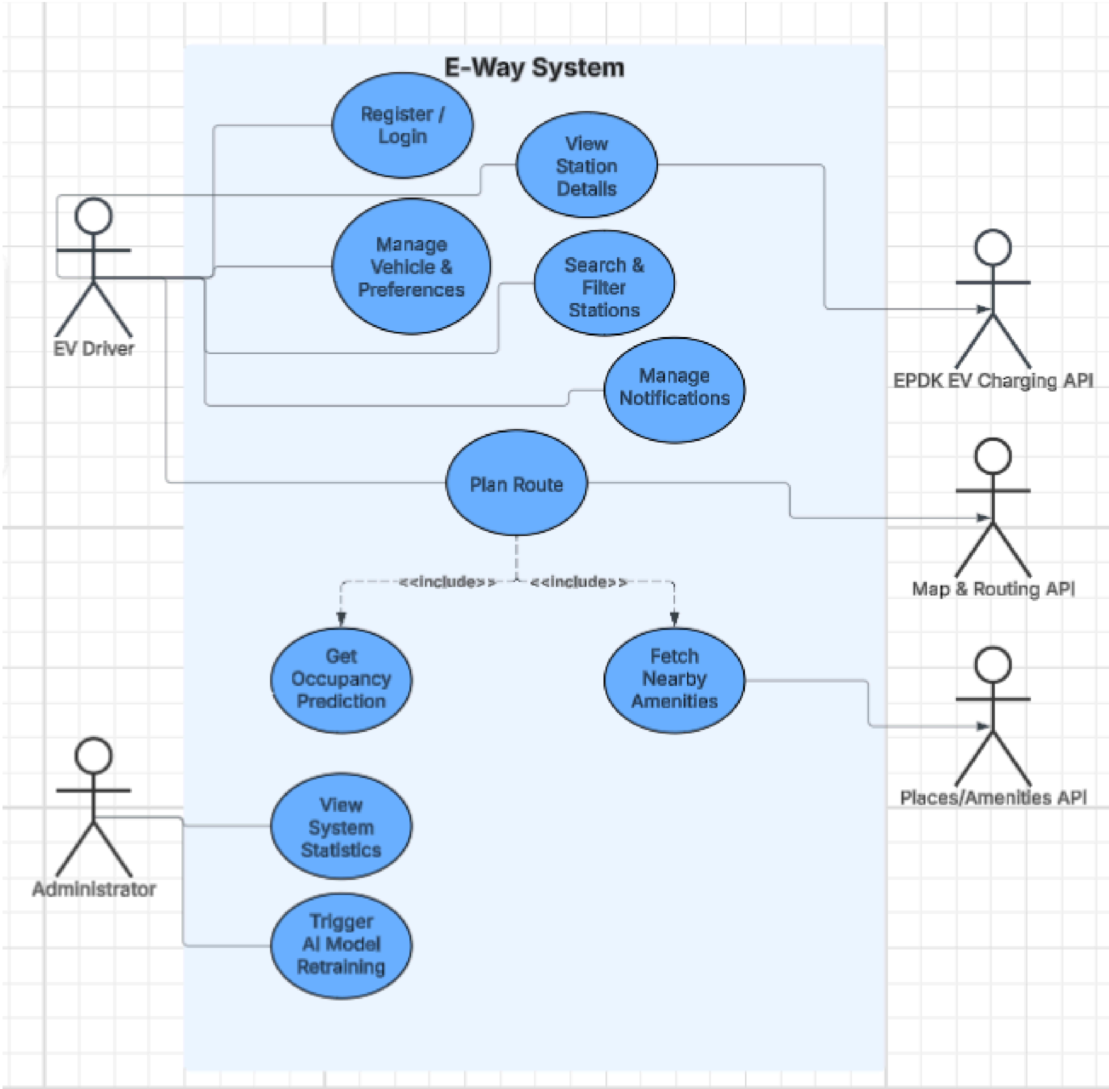


Figure 1. Use Case Diagram of E-Way

3.5.3 Object and Class Model

The Object and Class Model, visualized in Figure 2, illustrates the static structure of the E-Way system, defining the data entities, their relationships and the system's control logic. The model is designed using Object Oriented Programming (OOP) principles to ensure modularity, extensibility and data integrity.

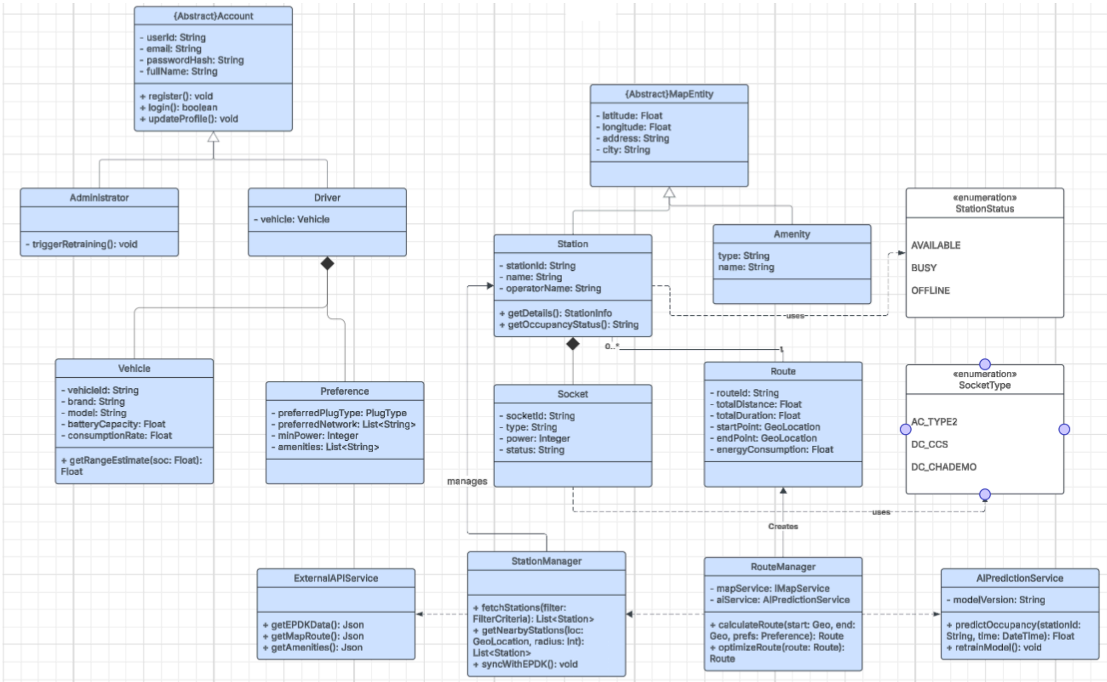


Figure 2. Class Diagram of E-Way

3.5.4 Dynamic Models

3.5.4.1 Sequence Diagram

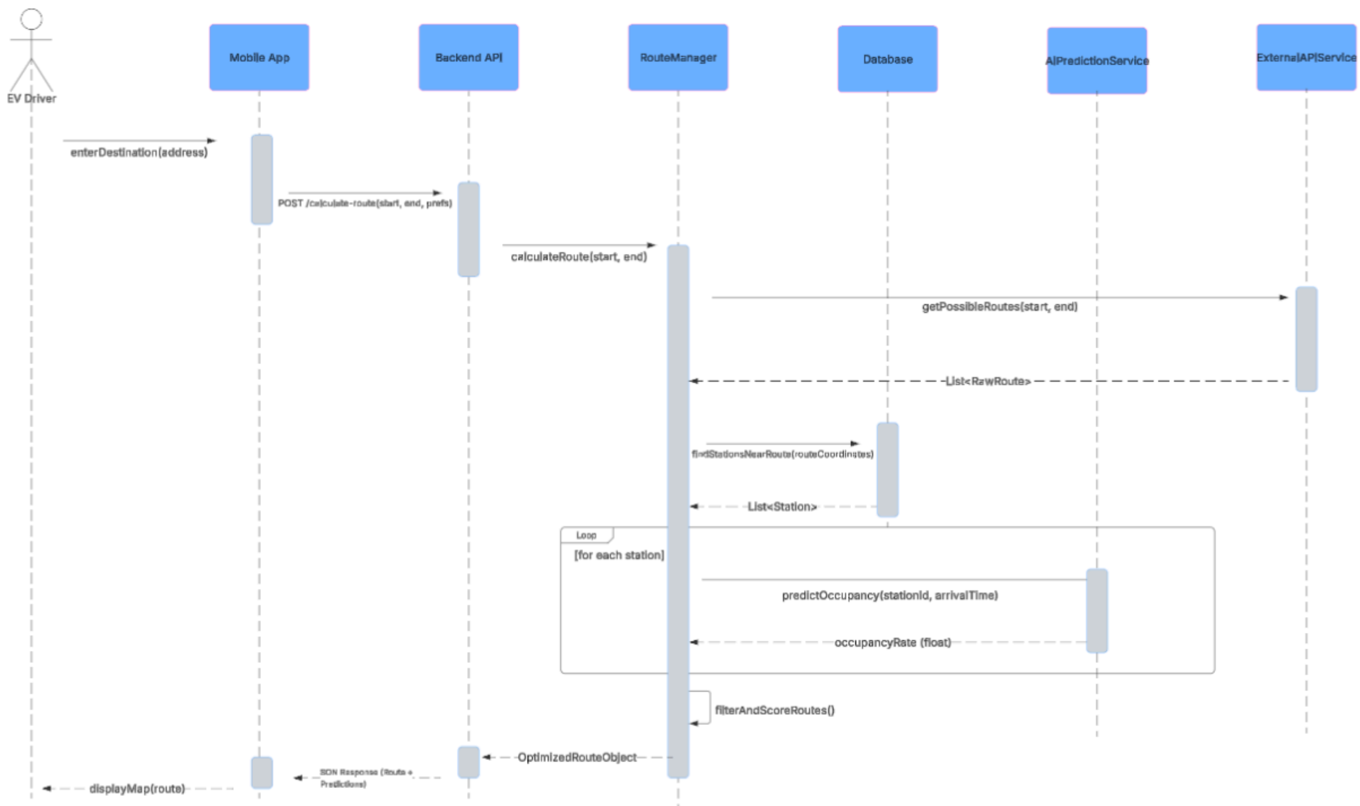


Figure 3. Sequence Diagram of E-Way

Figure 3 depicts the dynamic interaction for the "Smart Route Planning" scenario. The process begins when the EV Driver initiates a request via the Mobile App. The Backend API (FastAPI) acts as the controller, forwarding the request to the RouteManager.

The RouteManager first retrieves raw route data from the ExternalAPIService. Crucially, the system iterates through each potential charging station along the route. For each station, the AIPredictionService is invoked to estimate the occupancy rate based on the user's estimated time of arrival. This predictive analysis allows the system to filter out congested stations before presenting the optimized route options to the user. Finally, the optimized route object is returned to the client application for display.

3.5.4.2 Activity Diagram

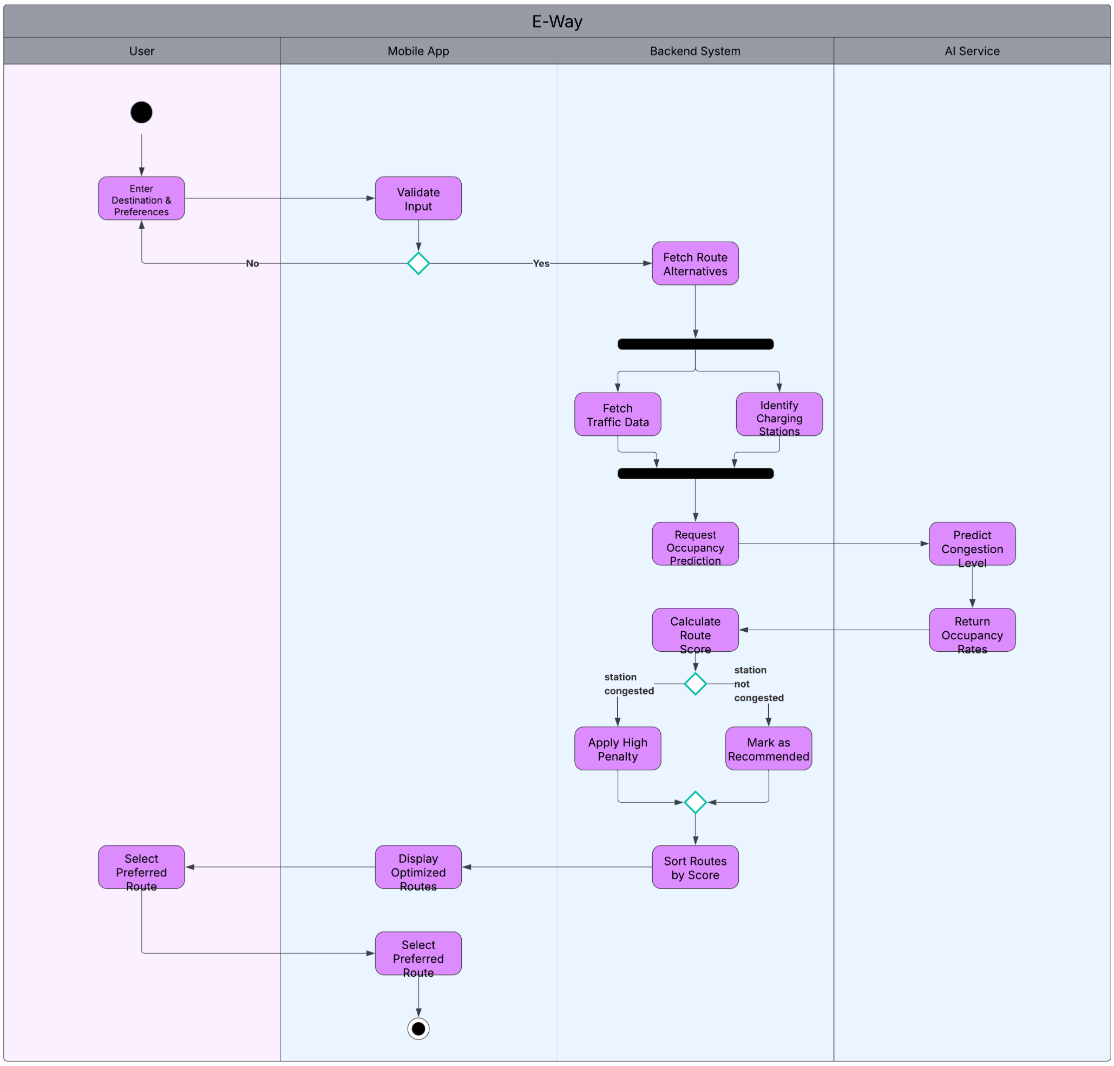
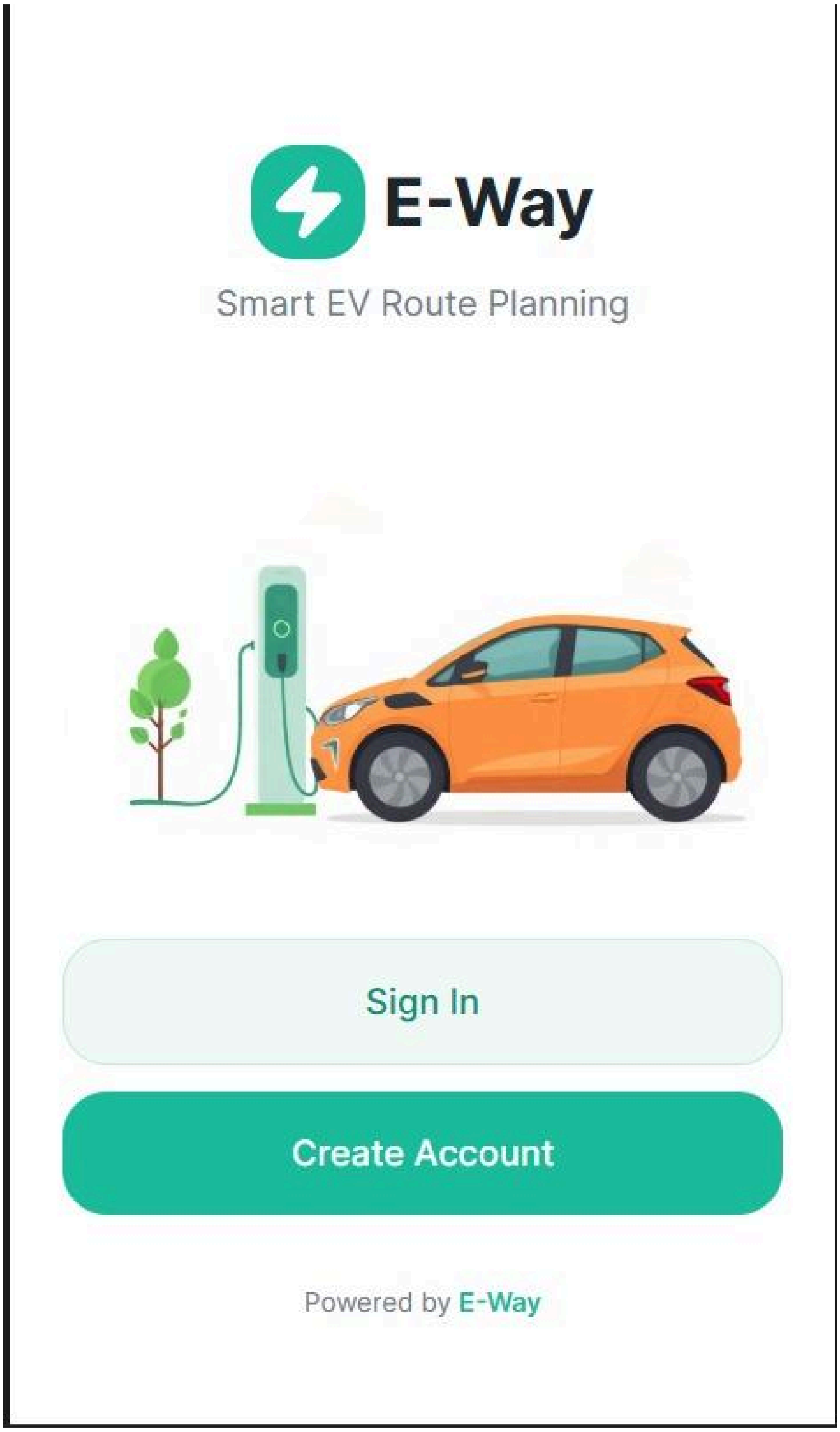


Figure 4. Activity Diagram of E-Way

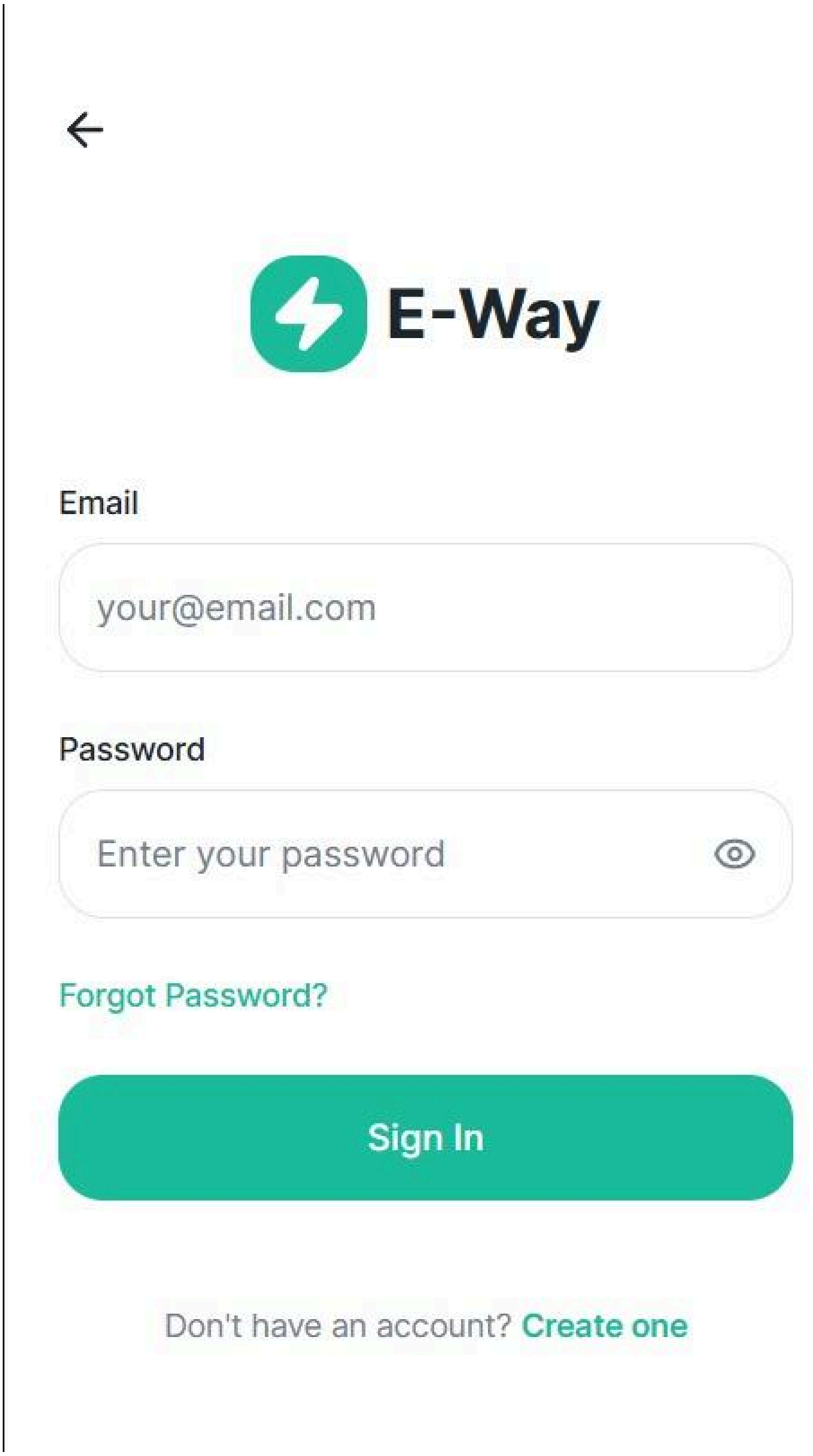
Figure 4 illustrates the workflow of the intelligent route planning process. The flow is partitioned into four swimlanes: User, Mobile App, Backend System and AI Service. The process initiates when the user inputs destination and preferences. The system executes parallel tasks to fetch route geometries and identifying charging stations. Crucially, the AI Service is invoked to predict station congestion levels. Based on these predictions, the Backend System calculates a suitability score for each route, penalizing congested paths. Finally, the optimized route options are presented to the user for selection.

3.5.5 User Interface

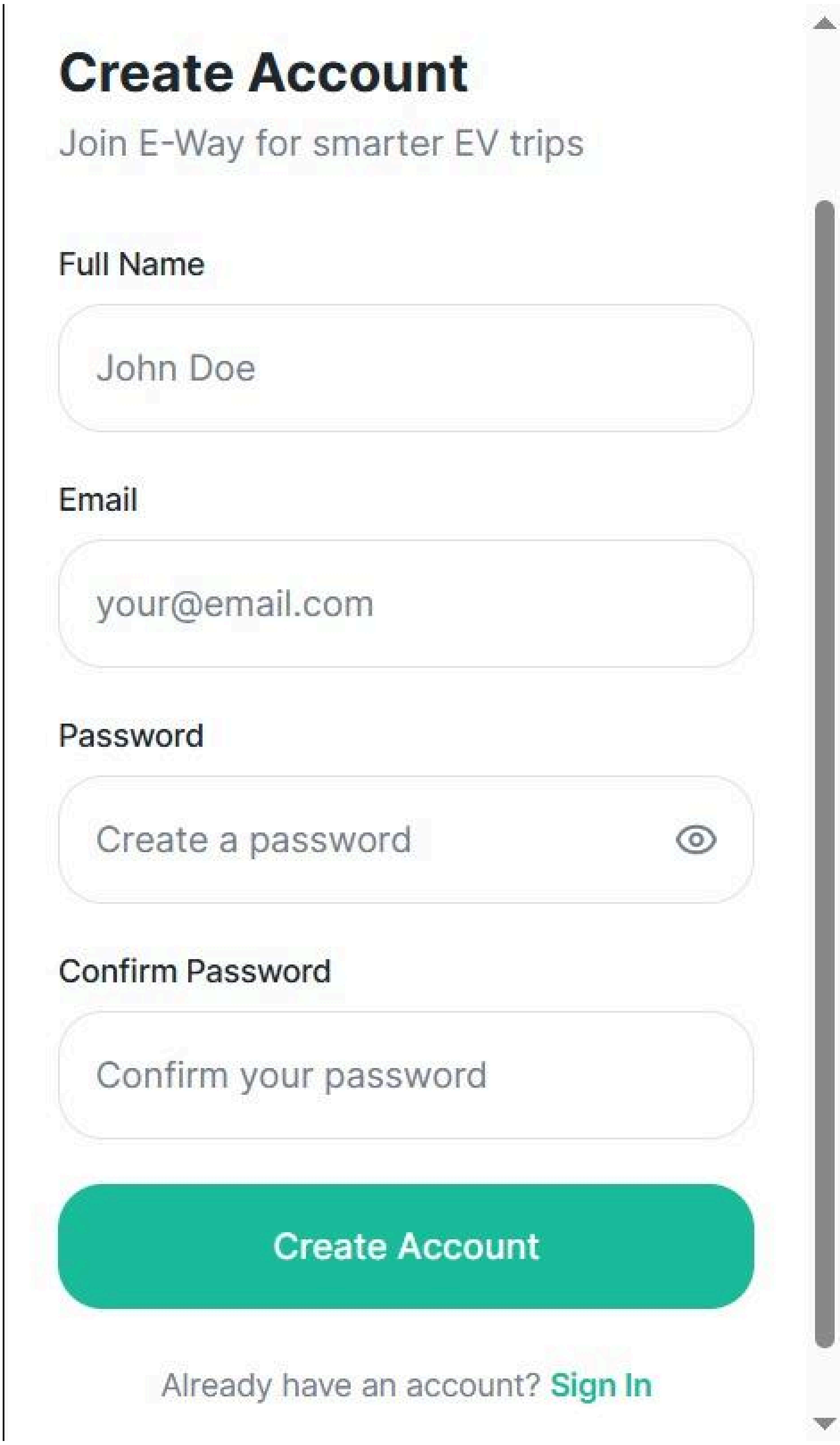
3.5.5.1 Welcome Page (/welcome)



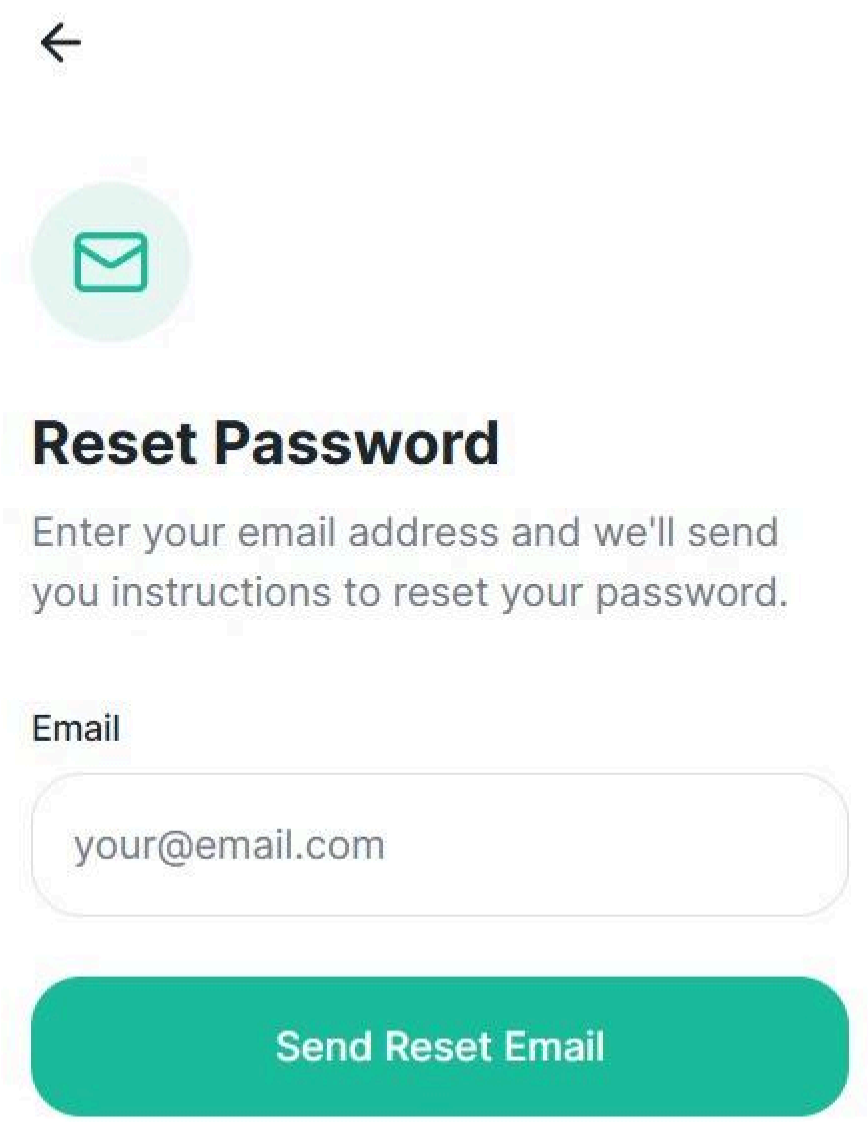
3.5.5.2 Login Page (/login)



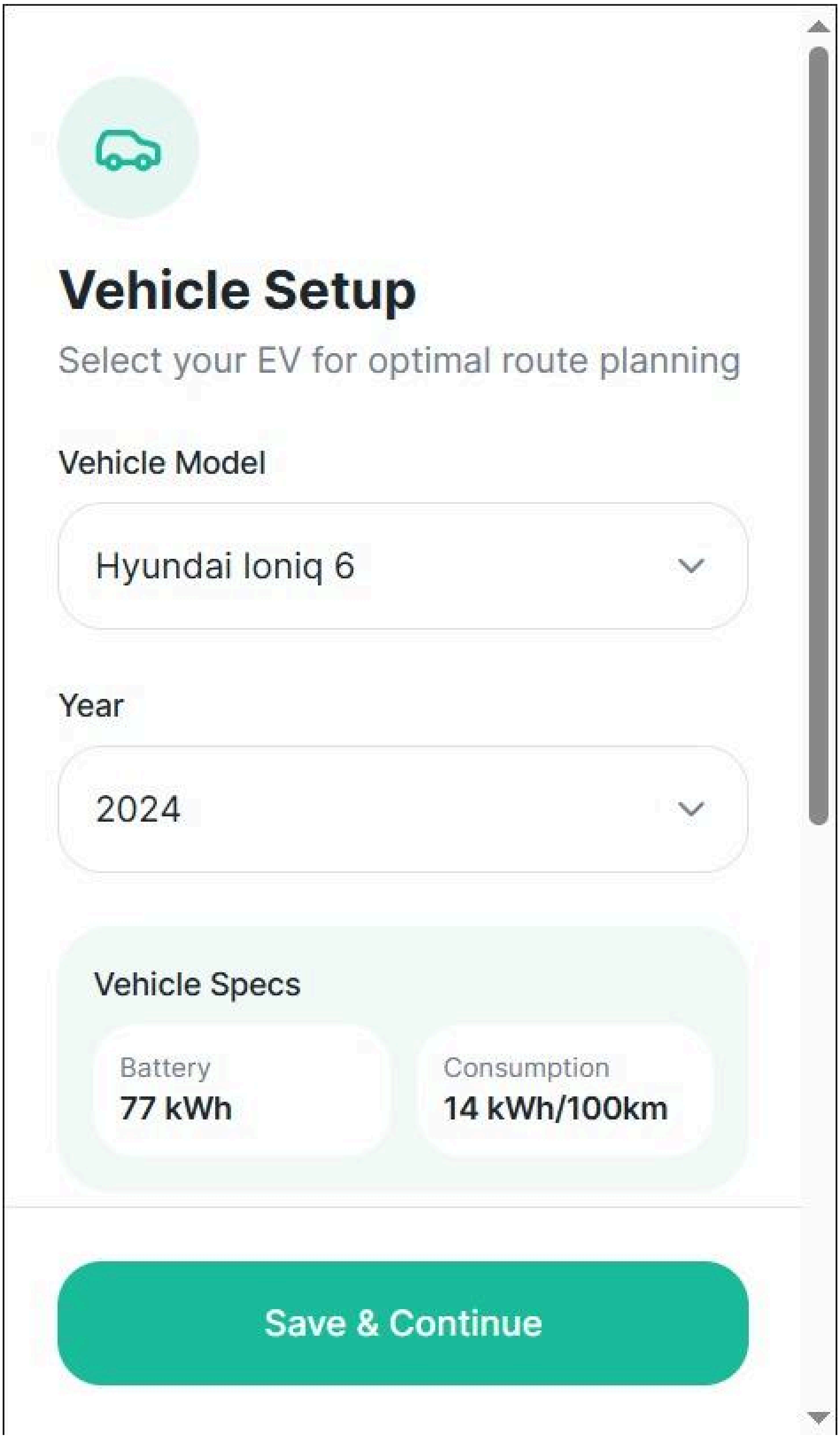
3.5.5.3 Register Page (/register)



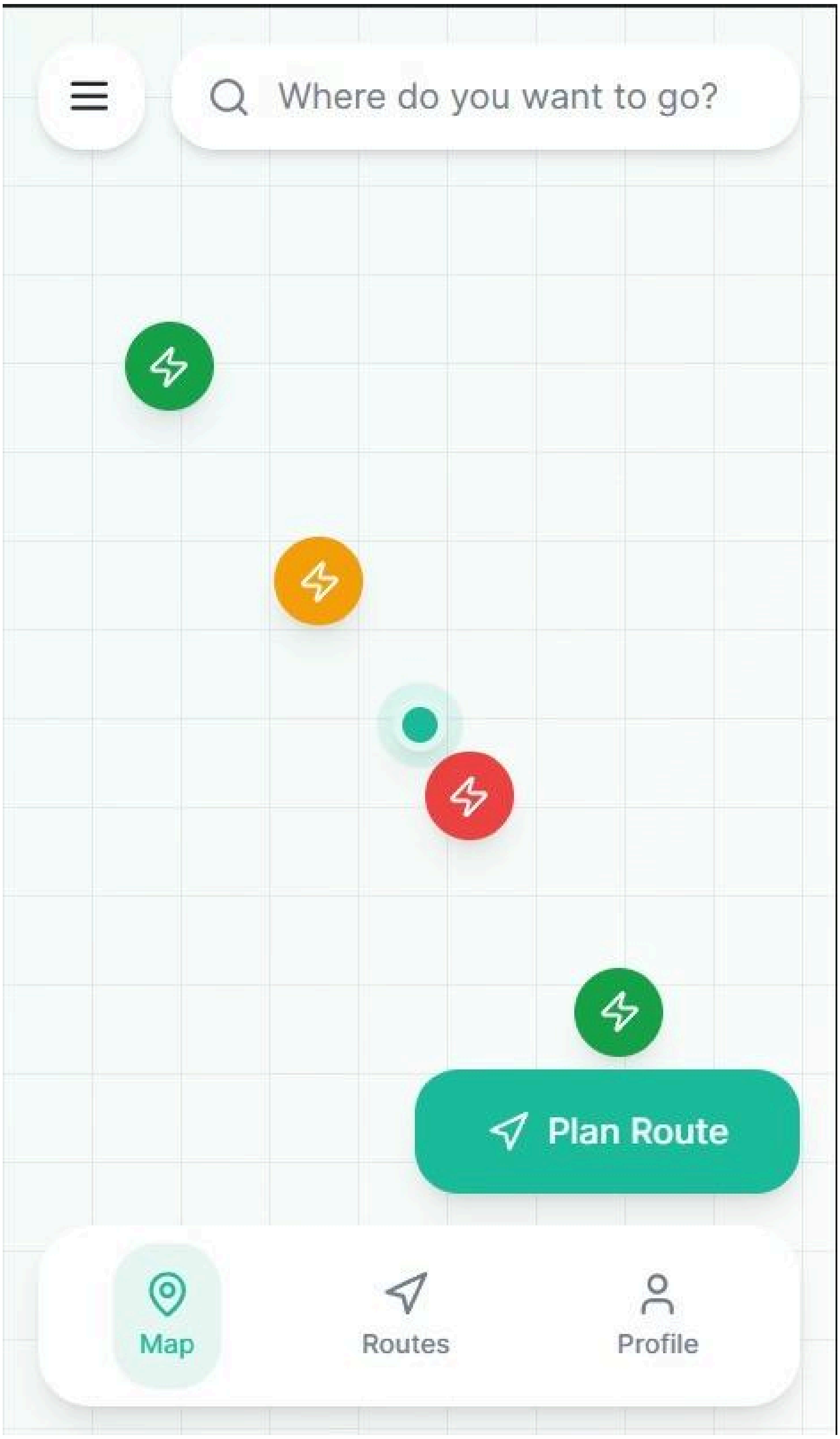
3.5.5.4 Forgot Password Page (/forgot-password)



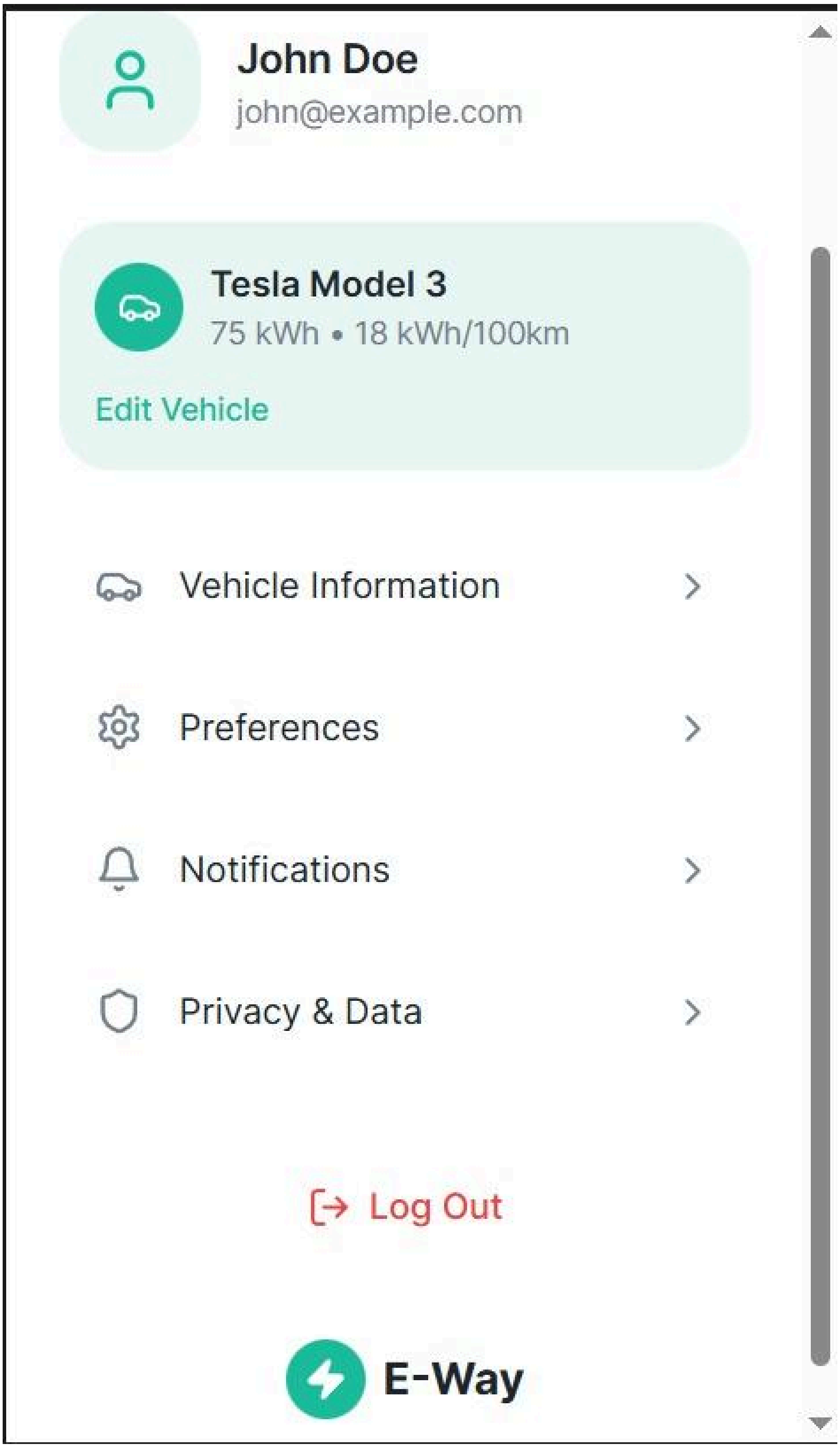
3.5.5.6 Vehicle Setup Page (/vehicle-setup)



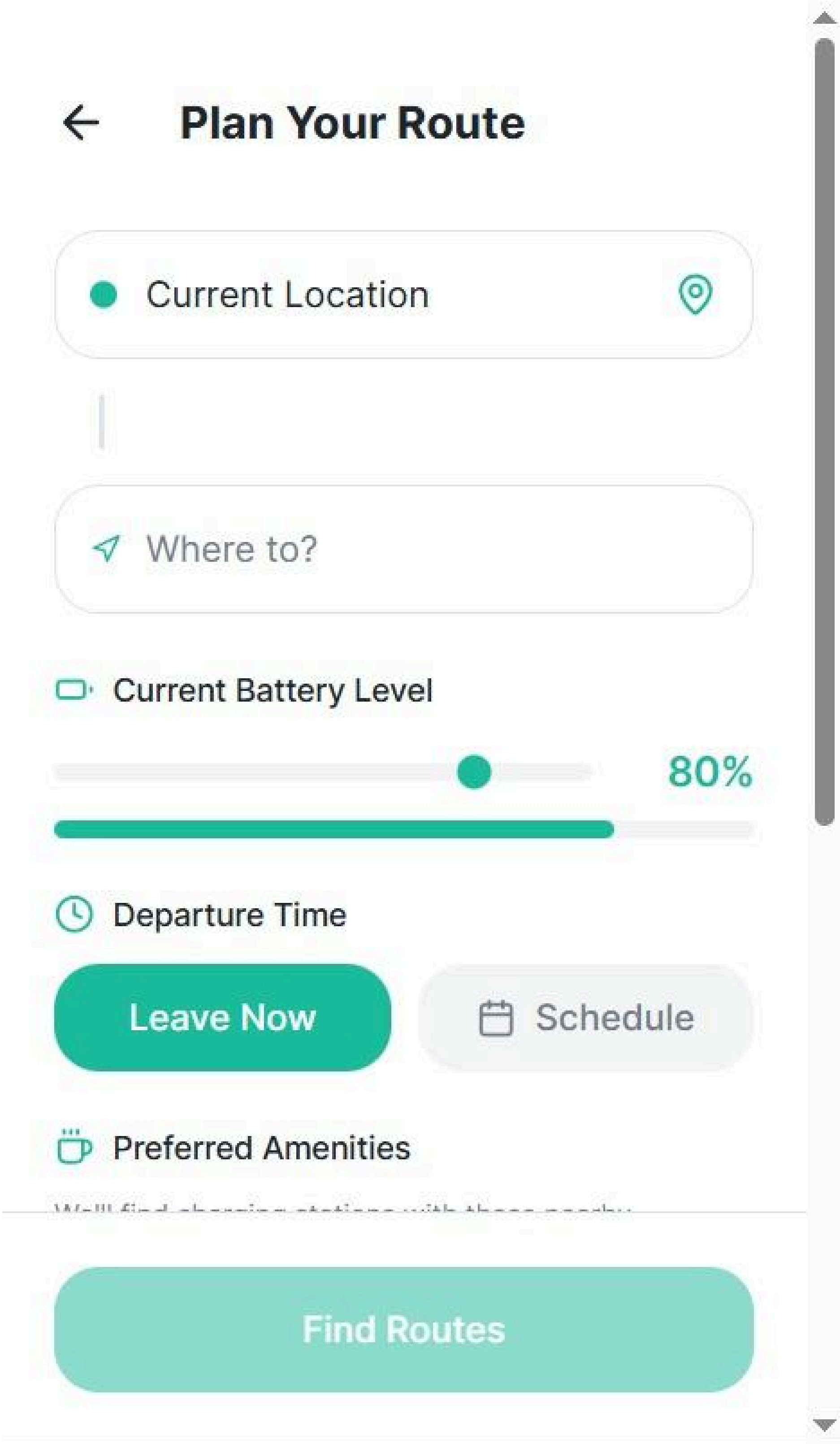
3.5.5.6 Home Page (/home)

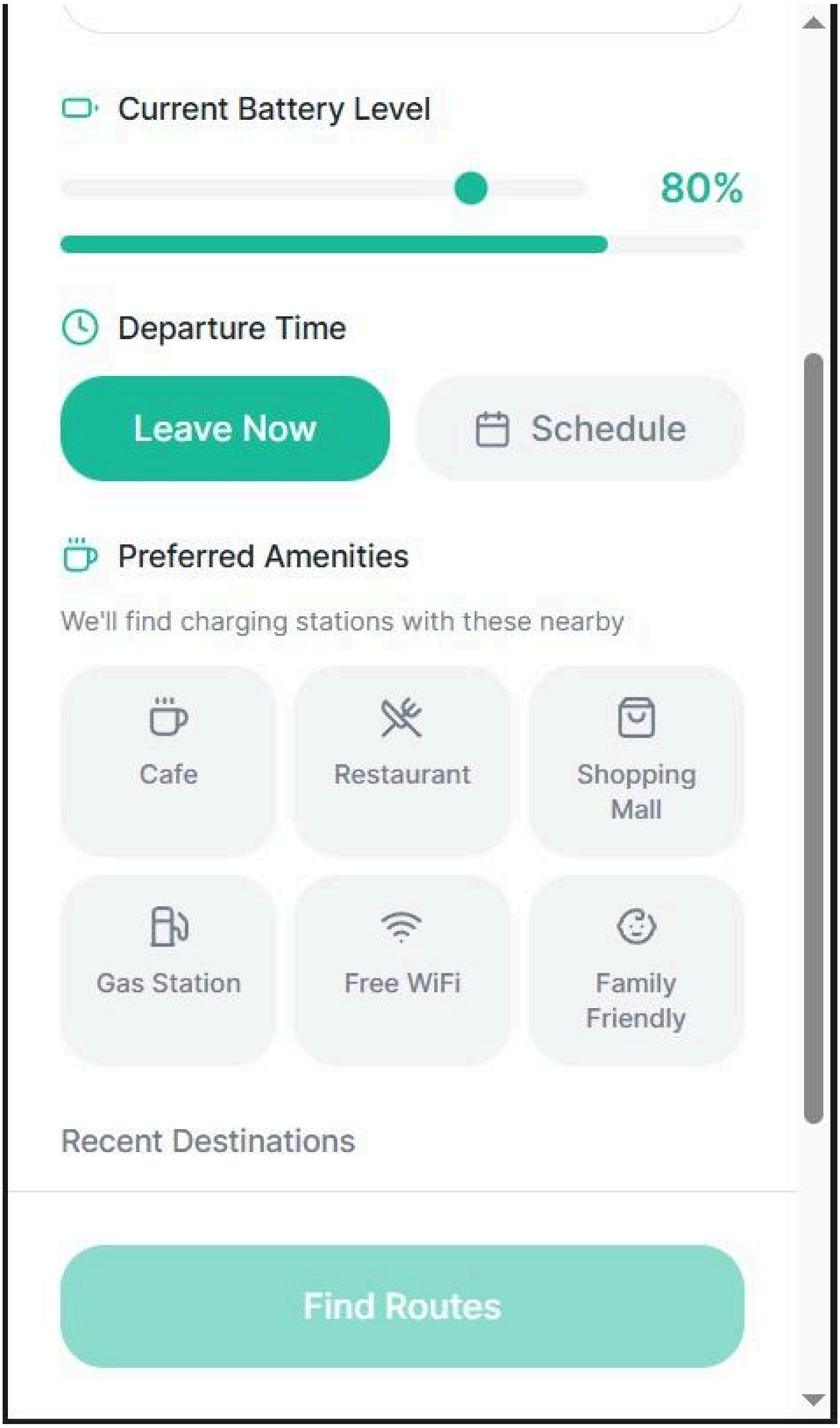


3.5.5.7 Profile Page (/profile)

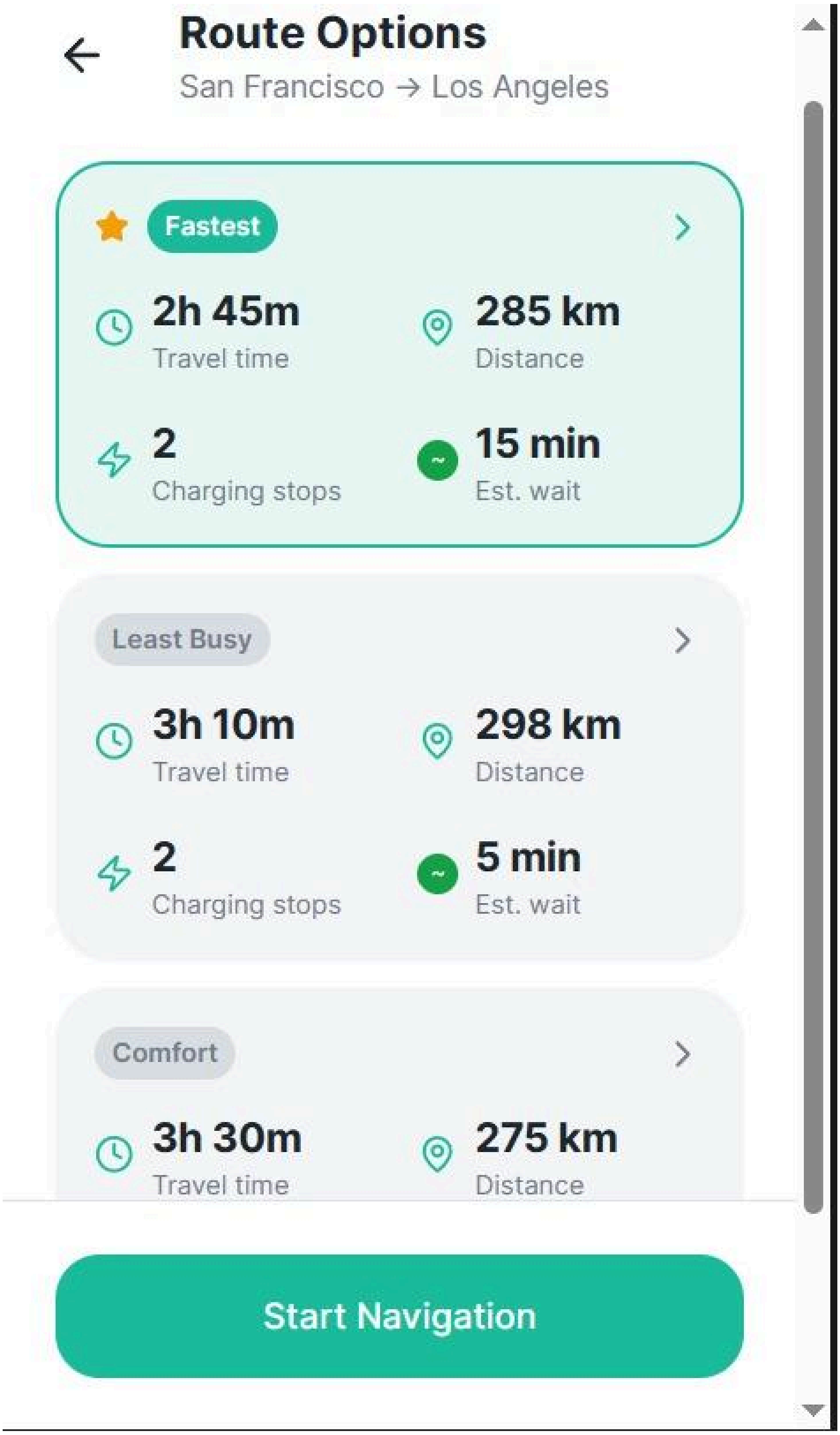


3.5.5.8 Route Input Page (/route-input)

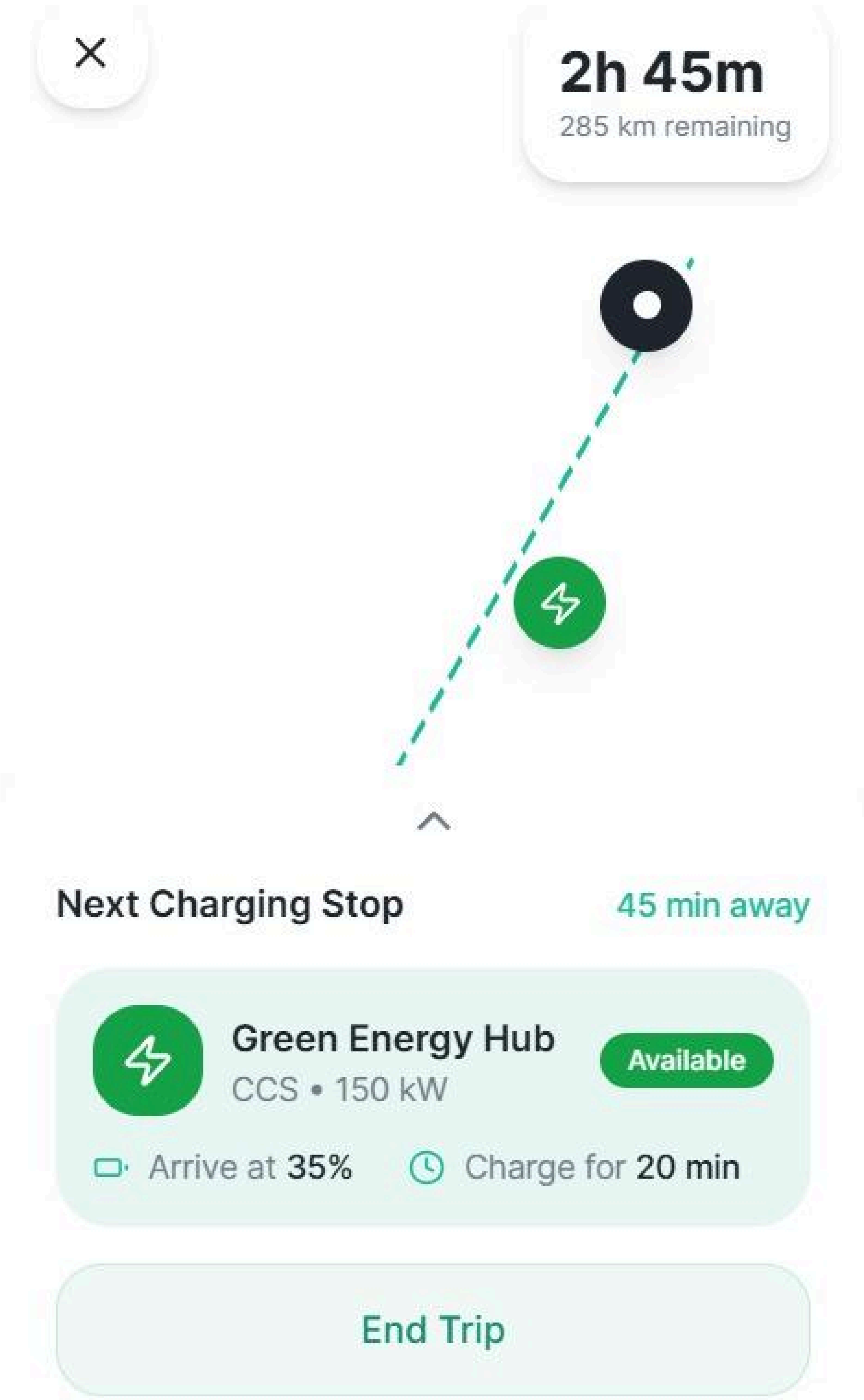



Current Battery Level
80%
Departure Time
Leave Now
Schedule
Preferred Amenities
We'll find charging stations with these nearby
Cafe
Restaurant
Shopping Mall
Gas Station
Free WiFi
Family Friendly
Recent Destinations
Find Routes

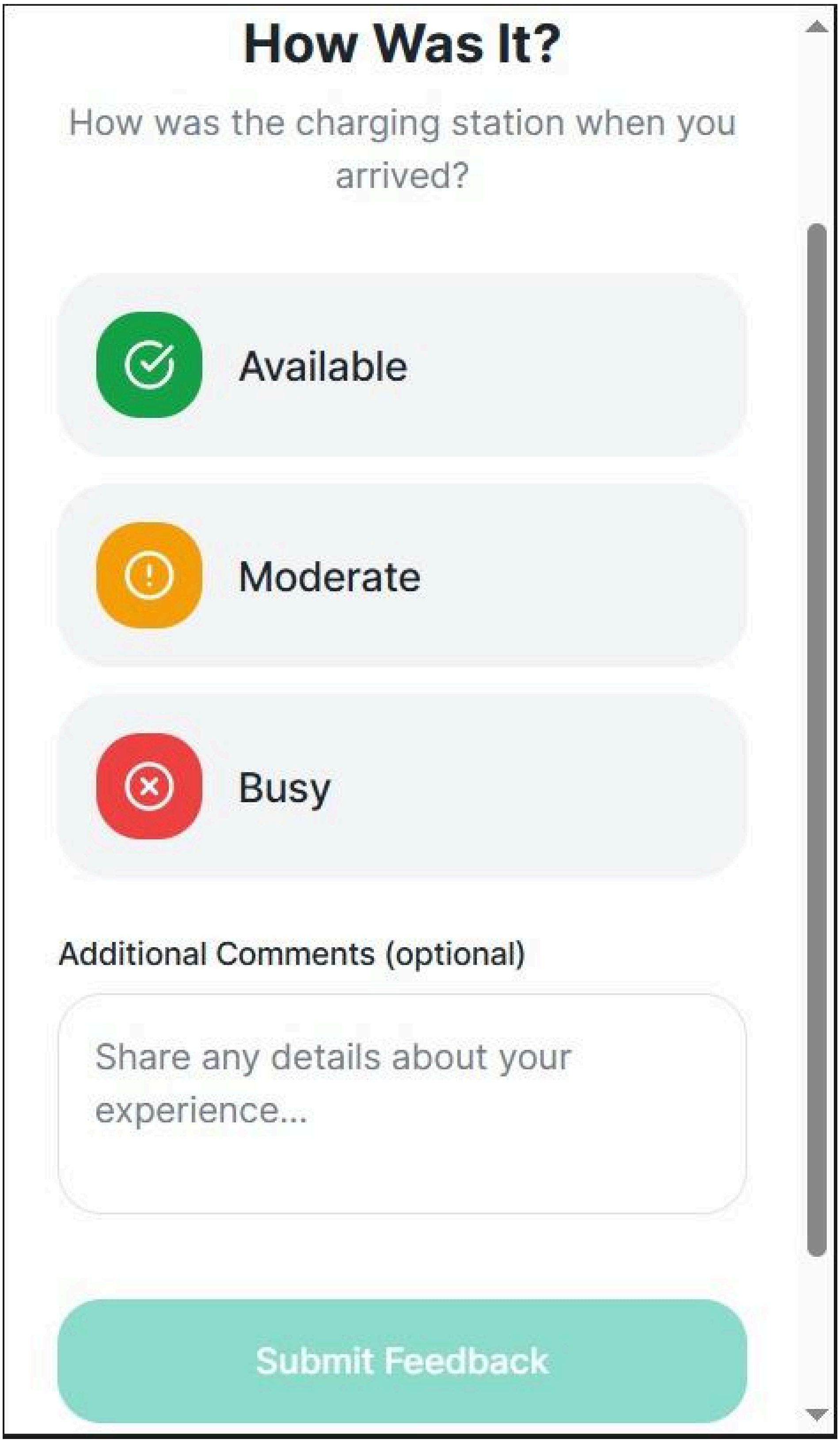
3.5.5.9 Alternative Routes Page (/route-alternatives)



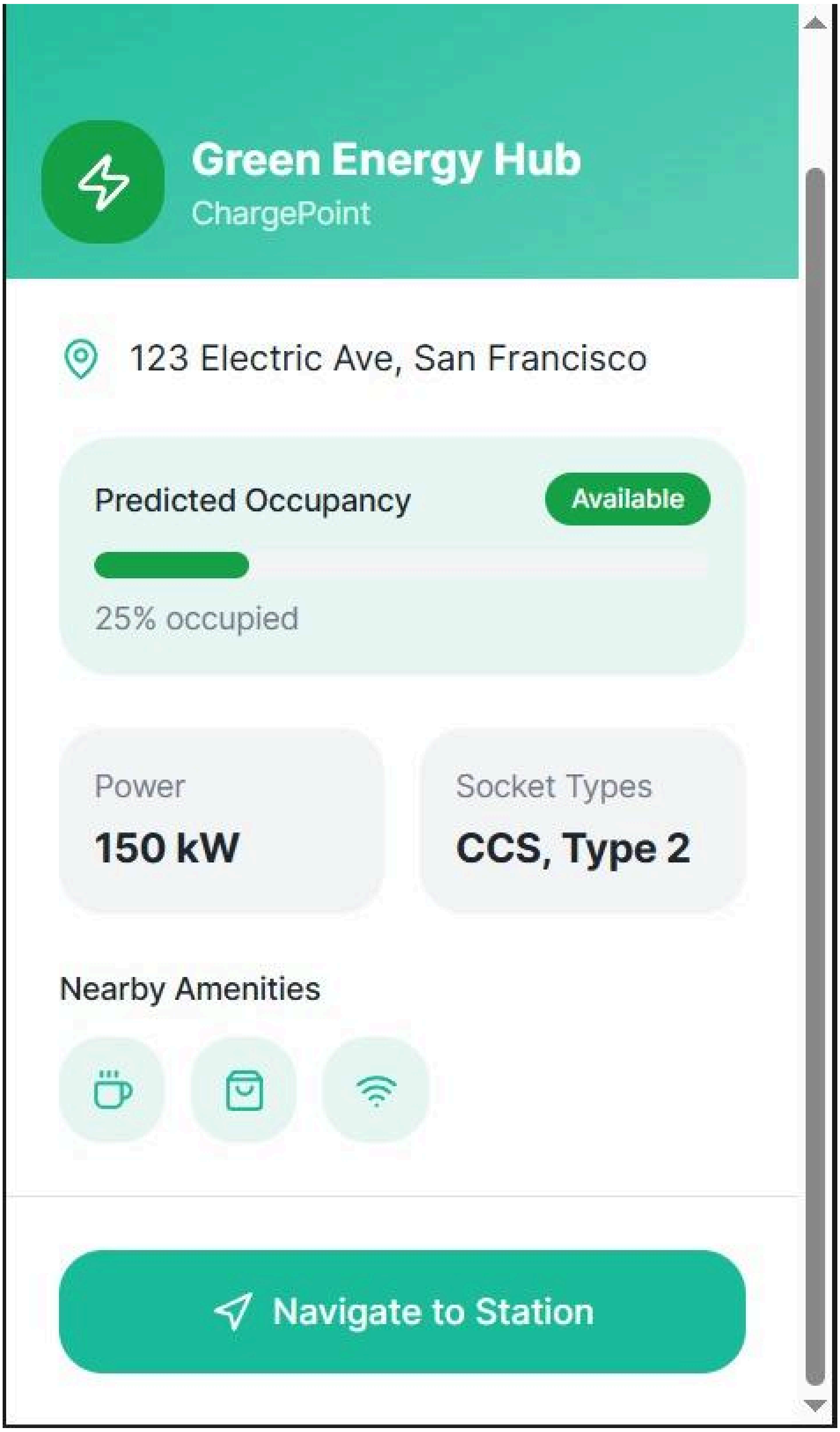
3.5.5.10 Navigation Page (/navigation)



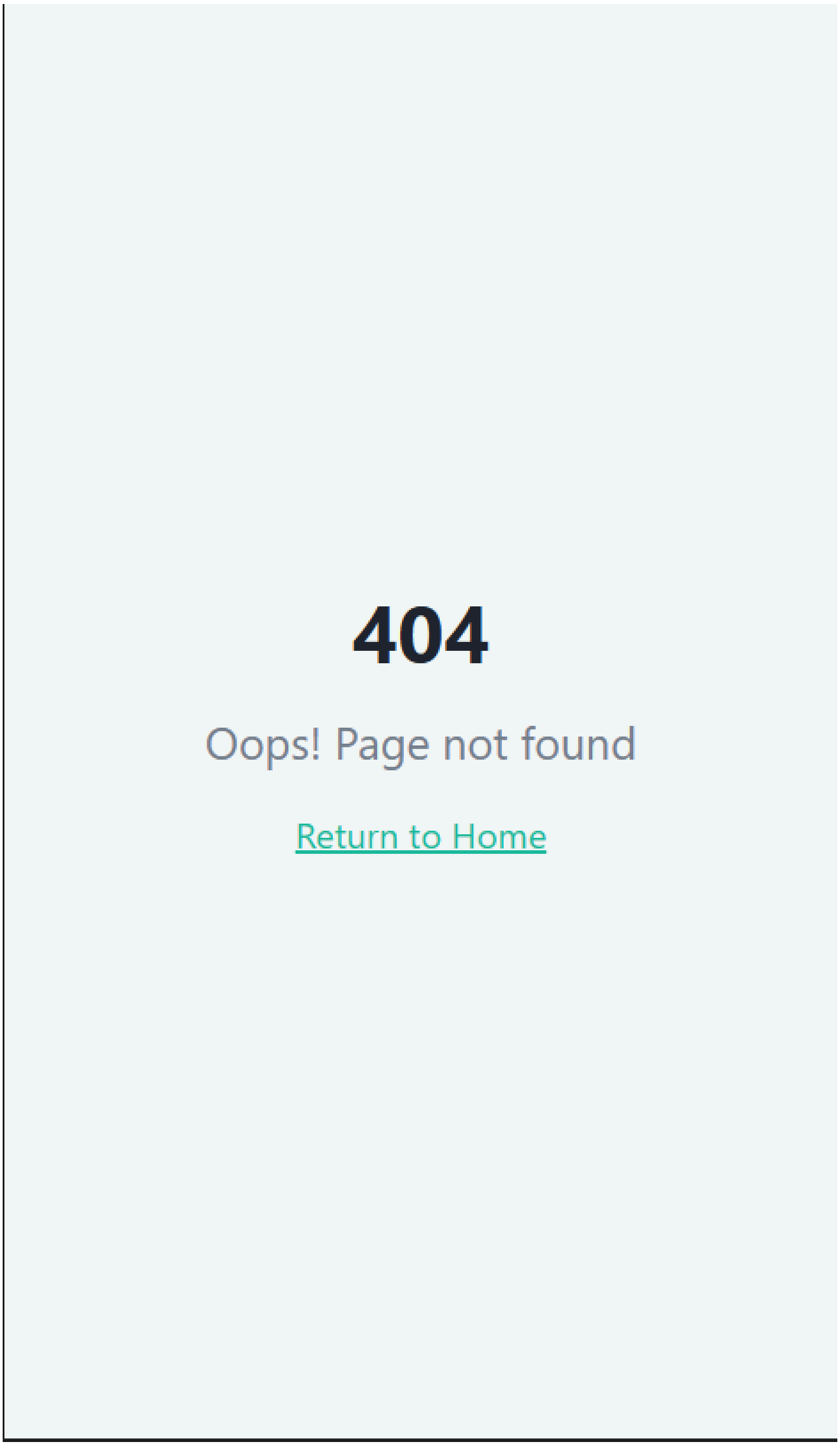
3.5.5.11 Feedback Page (/feedback)



3.5.5.12 Station Details Page (/station/:id)



3.5.5.13 Page Not Found Page (/page-not-found)



4 Other Analysis Elements

4.1 Consideration of Various Factors in Engineering Design

This section analyzes the impact of the E-Way project on various engineering, social, and global factors. The design choices have been made considering these constraints to ensure a safe, sustainable and ethically responsible system.

4.1.1 Constraints

The system relies heavily on third party services, specifically the EPDK EV Charging Stations API for station data and Google Maps Platform for routing and amenities. Any downtime or changes in these external APIs directly impact system functionality. Additionally, since real time data is not fully supported by all operators, the system is constrained to use historical data for predictions.

4.1.1.1 Public Health

The E-Way project indirectly contributes to public health by promoting the adoption of Electric Vehicles (EVs) which reduce air pollution compared to internal combustion engine vehicles. However, the system itself does not have a direct medical or clinical impact. The primary consideration here is to reduce the stress and fatigue of drivers by providing reliable planning tools.

4.1.1.2 Public Safety

Since E-Way is a mobile application intended for use by drivers, public safety is a critical constraint. The user interface is designed to minimize distraction, adhering to driver safety guidelines. Complex interactions are disabled or limited while the vehicle is in motion to prevent accidents. Furthermore, route recommendations prioritize safe and main roads over hazardous shortcuts.

4.1.1.3 Public Welfare

The system significantly enhances public welfare by optimizing the transportation infrastructure. By distributing the charging demand more evenly across stations, E-Way helps prevent congestion at popular hubs, saving time for citizens and increasing the overall efficiency of the national energy network.

4.1.1.4 Global Factors

While the current dataset is focused on Türkiye (EPDK data), the system architecture is designed to be globally scalable. The dependency on global standards like REST APIs and standard map data allows the application to be adapted to other regions with minimal architectural changes. However, data privacy regulations (like GDPR in Europe) must be strictly followed when expanding globally.

4.1.1.5 Cultural Factors

The transition to electric mobility requires a shift in driving culture. E-Way addresses the digital literacy barrier by offering a simple and intuitive interface ensuring that users from all demographics can easily access charging information without needing advanced technical skills.

4.1.1.6 Social Factors

The project encourages a social community among EV drivers. By reducing the fear of running out of charge, it encourages more people to switch to EVs, creating a positive social trend towards sustainability. Additionally, the system respects social equity by providing neutral, data-driven recommendations without favoring specific operators unfairly.

4.1.1.7 Environmental Factors

Environmental sustainability is the core philosophy of E-Way. The system encourages energy efficient travel by optimizing routes. On the technical side, backend algorithms are optimized to minimize computational power and cloud resource consumption, thereby reducing the digital carbon footprint of the project itself.

4.1.1.8 Economic Factors

The project operates under economic constraints related to cloud infrastructure and API costs (e.g., Google Maps pricing). To ensure economic viability, the system employs caching strategies and optimized database queries. For the end user, the system provides economic benefits by suggesting routes that may be shorter or by highlighting stations with better pricing (if data is available), thus reducing travel costs.

Table 1. Effects of Various Factors

Factor	Effect Level (0-10)	Effect Summary
Public Health	2	Indirect benefit via reduced pollution; low direct impact.
Public Safety	9	High critical impact; driver distraction must be minimized to prevent accidents.
Public Welfare	7	Improves transport efficiency and reduces time waste for citizens.
Global Factors	5	Scalable architecture, but currently localized data dependencies.
Cultural Factors	4	Addresses digital literacy;

		requires user adaptation to EV planning habits.
Social Factors	6	Encourages sustainable living and equitable access to information.
Environmental	8	Strong alignment with sustainability goals and energy optimization.
Economic	7	Direct impact on development budget (API costs) and user savings.

4.1.2 Standards

The development of E-Way adheres to internationally recognized engineering and software standards to ensure quality and interoperability.

- **IEEE 830-1998 (Software Requirements Specifications):** Used for documenting the functional and nonfunctional requirements of the project to ensure clarity and verifiability [3].
- **UML 2.5.1 (Unified Modeling Language):** Used for creating system diagrams (Use Case, Class, Sequence, Activity) to visualize the architecture and design [4].
- **KVKK & GDPR:** The project complies with the Law on Protection of Personal Data (KVKK) and general GDPR principles regarding the anonymization and secure storage of personal user data [5].
- **OpenAPI Specification 3.0:** Adopted for designing and documenting the RESTful APIs between the backend (FastAPI) and the client (Flutter), ensuring standardized communication [6].
- **IEEE Code of Ethics:** The project follows ethical guidelines prioritizing public health, safety and welfare, ensuring unbiased algorithmic recommendations.

4.2 Risks and Alternatives

The development and operation of E-Way involve several risks related to external data dependencies, prediction accuracy, and resource constraints. Identifying these risks early allows for the preparation of mitigation strategies to ensure system stability.

One of the primary risks is the dependency on external APIs, specifically the EPDK EV Charging Network and Google Maps Platform. Any downtime, rate limiting or structural change in these APIs could disrupt the service. To mitigate this, the system incorporates caching mechanisms and is designed to support alternative map providers (e.g., OpenStreetMap) if necessary.

Another significant risk is the absence of real-time occupancy data from the official source (EPDK currently provides metadata but not live status). The system relies on AI-based predictions using historical patterns. If prediction accuracy is low, user trust may diminish. As a B-plan, the application will transparently label data as "Estimated" and potentially integrate crowdsourced status updates from users in future iterations.

Table 2. Risks and Mitigation Strategies

Risk Name	Likelihood	Effect on Project	B-Plan Summary
External API Failure (EPDK/Google Maps)	Medium	High (Loss of routing & station data)	Implement aggressive caching for station data. Switch to OpenStreetMap for visualization if Google Maps becomes unavailable or too costly.
Low AI Prediction Accuracy	Medium	Medium (User dissatisfaction)	Allow users to provide feedback/manual status updates. Fallback to displaying only static station capacity without occupancy estimates.
Exceeding Cloud/API Quotas (Cost Overrun)	High	High (Service interruption)	Optimize database queries and API calls. Use free-tier compatible open-source alternatives (e.g., Leaflet map) if budget limitations are reached.
Data Privacy / Security Breach	Low	Very High (Legal/Reputational damage)	Strict adherence to KVKK. Minimal data collection policy. Hashing sensitive data and enforcing rigorous authentication protocols.
Mobile Device Performance Issues	Low	Medium (Poor UX on older devices)	Offload all heavy computations (AI, Routing) to the backend. Keep the mobile client lightweight by optimizing Flutter rendering.

4.3 Project Plan

The project plan is organized into six major work packages (WPs) covering the entire development lifecycle from analysis to final deployment. The timeline spans the academic year (October 2025 - May 2026).

Table 3. List of work packages

WP#	Work package title	Leader	Members involved
WP1	Analysis and Documentation	Halis Vefa Türkyılmaz	All Members
WP2	Backend & Database Architecture	Furkan Özek	Aziz Üzümcü, Halis Vefa Türkyılmaz
WP3	Mobile Client Development	Aziz Üzümcü	Furkan Özek, Hüseyin Utku Yüksel
WP4	AI & Prediction Module	Hüseyin Utku Yüksel	Aziz Üzümcü, Halis Vefa Türkyılmaz
WP5	Integration and Testing	Furkan Özek	All Members
WP6	Project Demo & Final Deliverables	Aziz Üzümcü	All Members

Table 4. Detailed description of work packages

WP 1: *Analysis and Documentation*			
Start date: *15.10.2025* **End date:** *02.05.2026*			
Leader:	*Halis Vefa Türkyılmaz*	**Members involved:**	*Furkan Özek, Hüseyin Utku Yüksel, Aziz Üzümcü*
Objectives: *Defining the project scope, analyzing requirements, and preparing the initial documentation (Specification, Analysis Report).*			
Tasks: ***Task 1.1 Requirement elicitation and competitive analysis*** **:** *Identify the core needs of EV drivers and analyze existing solutions like Google Maps or operator apps to define E-Way's unique value proposition and functional requirements.* ***Task 1.2 Designing system architecture and UML diagrams*** **:** *Create detailed architectural blueprints including Use Case, Class, Sequence and Activity diagrams to guide the development process and ensure a modular system design.*			

Task 1.3 Preparing the Analysis and Requirement Report **:** *Document all functional and nonfunctional requirements, constraints, risk analyses and system models into a formal report for stakeholder approval and submission.*			
Deliverables ***D1.1:*** Project Information Form. ***D1.2:*** *Project Specification Document.* ***D1.3:*** *Innovation Expert Evaluation form.* ***D1.4:*** *Analysis and Requirement Report.* ***D1.5:*** *Detailed Design Report.* ***D1.6:*** *Final Report.*			
WP 2: *Backend & Database Architecture*			
Start date: *15.10.2025* **End date:** *02.05.2026*			
Leader:	*Furkan Özek*	**Members involved:**	*Aziz Üzümcü, Halis Vefa Türkyılmaz*
Objectives: *Setting up the server, database, and API infrastructure.*			
Tasks: ***Task 2.1 Configuring MongoDB Atlas and designing schemas*** **:** *Set up the cloud database environment and define optimized data schemas for users, stations and routes to ensure efficient data storage and retrieval.* ***Task 2.2 Implementing FastAPI backend and REST endpoints*** **:** *Develop the core server side application logic and create secure REST API endpoints to handle client requests, user authentication and data processing.* ***Task 2.3 Integrating EPDK and Google Maps APIs*** **:** *Implement services to connect with external APIs for fetching official charging station data and retrieving map and routing information required for the system.*			
Deliverables ***D2.1:*** *Functional Backend API.* ***D2.2:*** Database Schema Design.			

WP 3: *Mobile Client Development*			
Start date: *15.10.2025* **End date:** *02.05.2026*			
Leader:	Aziz Üzümcü	**Members involved:**	Furkan Özek, Hüseyin Utku Yüksel
Objectives: *Developing the cross platform mobile application using Flutter.*			
Tasks: ***Task 3.1 UI/UX design and prototyping*** **:** *Create high accurate visual designs and user flow mockups using tools like Figma to visualize the application interface and ensure a user friendly experience before coding.* ***Task 3.2 Implementing Authentication, Map View and Filtering*** **:** *Develop the foundational mobile screens that allow users to securely log in, interact with the main map interface and filter charging stations based on their preferences.* ***Task 3.3 Integrating Route Planning UI and Navigation features*** **:** *Build the user interface for entering destinations, displaying optimized route alternatives and visualizing the AI based occupancy predictions on the map.*			
Deliverables			

D3.1: *UI Mockups.* ***D3.2:*** *Beta Mobile Application (APK/IPA).*			
WP 4: *AI & Prediction Module*			
Start date: *15.10.2025* **End date:** *02.05.2026*			
Leader:	Hüseyin Utku Yüksel	**Members involved:**	Aziz Üzümcü, Halis Vefa Türkyılmaz
Objectives: *Developing the occupancy prediction model.*			
Tasks: ***Task 4.1 Data preprocessing and cleaning*** **:** *Process the collected historical station usage data to remove inconsistencies, handle missing values and format it suitably for training machine learning algorithms.* ***Task 4.2 Training machine learning models*** **:** *Train and evaluate AI models to accurately predict future charging station occupancy based on temporal and location based patterns.* ***Task 4.3 Exposing the model via an API service*** **:** *Wrap the trained model in an API service or microservice so that the backend system can request and receive real time occupancy predictions during route calculation.*			
Deliverables ***D4.1:*** *Trained Prediction Model.*			

WP 5: *Integration and Testing*			
Start date: *10.01.2026* **End date:** *02.05.2026*			
Leader:	Furkan Özek	**Members involved:**	*Halis Vefa Türkyılmaz, Hüseyin Utku Yüksel, Aziz Üzümcü*
Objectives: *Merging all modules and ensuring system stability.*			
Tasks: ***Task 5.1 Connecting Mobile App with Backend APIs*** **:** *Link the Flutter frontend with the FastAPI backend to enable real data exchange, ensuring all features function correctly in a live environment.* ***Task 5.2 User acceptance testing and bug fixing*** **:** *Conduct end to end testing scenarios with team members to simulate real world usage, ensuring the system meets all requirements and resolving any remaining bugs.*			
Deliverables ***D5.1:*** *Final Integrated System.*			
WP 6: *Project Demo & Final Deliverables*			
Start date: *10.04.2026* **End date:** *02.05.2026*			
Leader:	*Aziz Üzümcü*	**Members involved:**	*Halis Vefa Türkyılmaz, Hüseyin Utku Yüksel, Furkan Özek*
Objectives: *Preparing for the CS Fair and final presentation.*			
Tasks: ***Task 6.1 Preparing the project poster and demo video*** **:** *Design a professional poster and create a video demonstration that effectively*			

showcases the project's key features and technical achievements for the CS Fair. ***Task 6.2 Writing the Final Report and User Manual*** **:** *Compile the comprehensive final report detailing the entire development process and create a user manual to guide end users on how to use the application.* ***Task 6.3 Final Presentation rehearsal*** **:** *Practice the final project presentation to ensure a smooth, coherent and professional delivery to the jury and supervisors.*
Deliverables ***D6.1:*** *Final Report.* ***D6.2:*** *Project Poster & Demo.*

4.4 Ensuring Proper Teamwork

Effective teamwork and clear communication are essential for the success of the E-Way project. To ensure equal contribution and maintain a steady workflow, the team employs a set of industry-standard collaboration tools and methodologies.

- **Communication:** We utilize Discord and WhatsApp for daily instant messaging and quick problem-solving.
- **Version Control:** GitHub is used for source code management. We have adopted a "Feature Branch" workflow where each new feature is developed in a separate branch and merged into the main branch only after a code review by another team member. This ensures code quality and minimizes integration conflicts.
- **Task Management:** We handle task distribution and tracking primarily through our meetings and a shared task list on Google Drive (Sheets). During each meeting, pending tasks are reviewed and new responsibilities are assigned to team members based on the project's current needs. This shared document acts as a central source of truth, tracking the owner, deadline and status of each task. This approach allows for flexible, real-time updates and ensures accountability without the overhead of complex management software.
- **Documentation:** All project documentation, meeting notes and API specifications are maintained on a shared Google Drive folder and within the repository's README files to ensure everyone has access to the latest information.

4.5 Ethics and Professional Responsibilities

The E-Way project strictly adheres to professional engineering codes of ethics, prioritizing user privacy, data security and public welfare. Based on the analysis of the project specification, the following ethical responsibilities are identified and addressed:

- **User Privacy & Data Protection (KVKK/GDPR):** The system collects sensitive personal data including real time location, travel history and vehicle details. To comply with KVKK (Personal Data Protection Law), all sensitive data is encrypted both in transit and at rest. Users are explicitly asked for permission before any data collection and they retain the right to delete their data at any time.

- **Location Data Sensitivity:** Since the core functionality relies on geolocation, we ensure that location data is used strictly for routing and occupancy prediction purposes. This data is never shared with third parties for commercial gain without the explicit permission of the user.
- **Unbiased AI Recommendations:** The occupancy prediction model is trained on historical data. We acknowledge the risk of algorithmic bias (e.g., favoring certain operators or regions). To mitigate this, we continuously evaluate the model to ensure it provides neutral, data driven recommendations that do not unfairly prioritize specific commercial charging networks.
- **Transparency:** The system is designed to be transparent. When a route recommendation is based on an AI prediction rather than real time data, this distinction is clearly communicated to the user to maintain trust and prevent misleading guidance.
- **Safety First:** As a driver focused application, the user interface is ethically designed to minimize distraction. Features that require complex interaction are disabled or simplified while driving to ensure road safety.

4.6 Planning for New Knowledge and Learning Strategies

Developing E-Way requires expertise in diverse technical domains, including mobile development, backend engineering and artificial intelligence. Since the team members have varying levels of experience in these areas, we have devised a strategic learning plan.

- **Mobile Development (Flutter):** While the team has general programming knowledge, developing a production ready mobile app with **Flutter** is a new challenge. We plan to bridge this gap by following the official Flutter documentation and completing advanced courses on state management and Google Maps integration on platforms like Udemy and YouTube.
- **Backend & API Architecture (FastAPI):** To build a high performance backend, the team needs to master FastAPI and asynchronous programming in Python. We will study the official FastAPI documentation and best practices for designing RESTful APIs. Additionally, we will learn MongoDB for handling geospatial data queries efficiently.
- **Artificial Intelligence (Time Series Forecasting):** The core innovation of E-Way lies in its prediction module. We need to acquire deep knowledge in time series forecasting using models like LSTM (Long Short-Term Memory) or Regression. We plan to review academic papers on traffic and demand prediction and utilize practical tutorials for libraries such as TensorFlow or PyTorch.
- **Domain Knowledge (EV Infrastructure):** Beyond coding, we must understand the EV ecosystem in Türkiye. We will analyze reports from EPDK and study the technical specifications of EV charging standards (AC vs. DC, CCS, Type-2) to ensure our data model accurately reflects reality.

5 Glossary

EV: Electric Vehicle

SoC: State of Charge

API: Application Programming Interface

UI: User Interface

ML: Machine Learning

AI: Artificial Intelligence

DB: Database

ETA: Estimated Time of Arrival

POI: Point of Interest

GIS: Geographic Information System

LSTM: Long Short-Term Memory (Neural Network Model)

6 References

[1] Enerji Piyasası Düzenleme Kurumu (EPDK), "Şarj Hizmeti Yönetmeliği," Resmi Gazete, Sayı: 31797, Apr. 2022.

[2] Tiangolo, "FastAPI Framework," [Online]. Available: https://fastapi.tiangolo.com/

[3] IEEE Computer Society, "IEEE Recommended Practice for Software Requirements Specifications," IEEE Std 830-1998, Oct. 1998.

[4] Object Management Group (OMG), "OMG Unified Modeling Language (OMG UML)," Version 2.5.1, Dec. 2017. [Online]. Available: https://www.omg.org/spec/UML/2.5.1

[5] T.C. Resmi Gazete, "Kişisel Verilerin Korunması Kanunu (Kanun No. 6698)," Apr. 2016.

[6] Google Developers, "Flutter Documentation - Build apps for any screen," [Online]. Available: https://flutter.dev/docs

[7] M. Schneider, A. Stenger, and D. Goeke, "The electric vehicle routing problem with time windows and recharging stations," Transportation Science, vol. 48, no. 4, pp. 500–520, 2014.