Software Design Document for "Locate a Socket"

1. Introduction

1.1 Purpose

The purpose of this document is to provide a comprehensive design overview for the development of "Locate a Socket," a web application aimed at enhancing the EV charging station discovery experience for drivers through efficient location-based services and secure payment options.

1.2 Scope

"Locate a Socket" will offer users the ability to find nearby EV charging stations, view detailed information about each station (e.g., availability, charging speed, and cost), and pay for charging services directly through the application. The application will cater to EV drivers seeking convenient access to charging facilities on their routes.

1.3 References

Please refer to the previous software requirement specification document. (SRS)

2. System Overview

"Locate a Socket" will leverage a microservices architecture to ensure scalability, reliability, and independence of development and deployment cycles for different parts of the application. Key components will include a User Interface (UI), Backend Services, Payment Gateway Integration, and a Database System.

3. System Architecture

3.1 Microservices Architecture

- User Management Service: Handles user registration, authentication, and profile management. It included a role-based access control (RBAC) to support different end users interacting with the platform.

- Charging Station Service: Utilizes geolocation data to provide users with real-time information on nearby charging stations. Provides detailed information about the station including availability of chargers and charger types.
- Invoice Service: Integrates with payment gateways for processing transactions. And stores all the transactions' details.

3.2 Database Design

An SQL database (e.g., PostgreSQL) will be used for performance in handling a large number of reads and writes on the database. It will store user profiles, charging station details, and transaction records. A detailed schema design for the database was provided in the software requirements specification document. Please refer to the SRS provided.
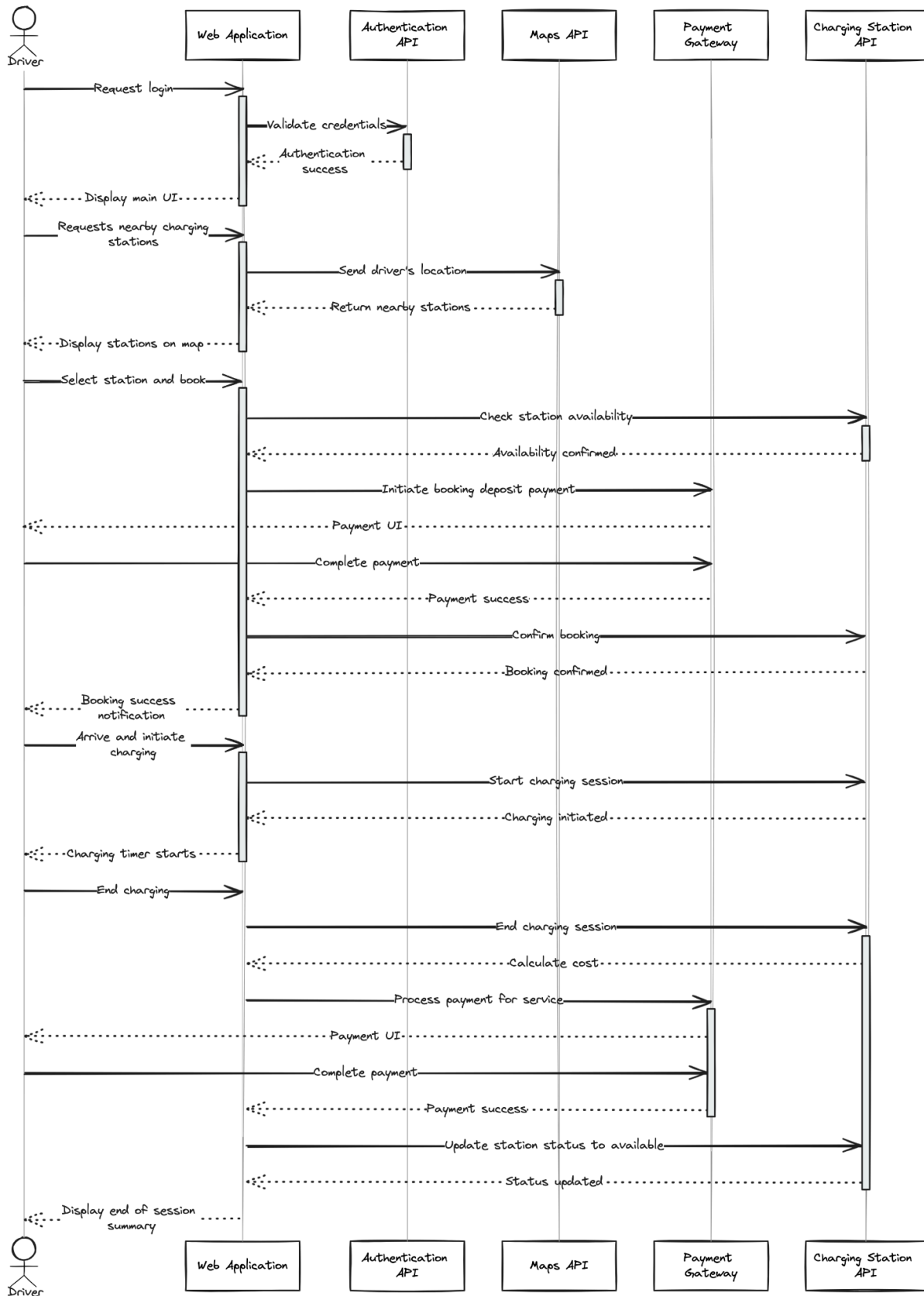
3.3 Security

- Use HTTPS to encrypt data in transit.
- Implement OAuth 2.0, Google, and Apple SSO integration for privacy and feasibility.

4. Detailed Design

A detailed design is discussed in this section covering user flows for every user end on this platform (e.g. drivers, stakeholders, operators) based on the user stories and system functionalities mentioned in the SRS document.
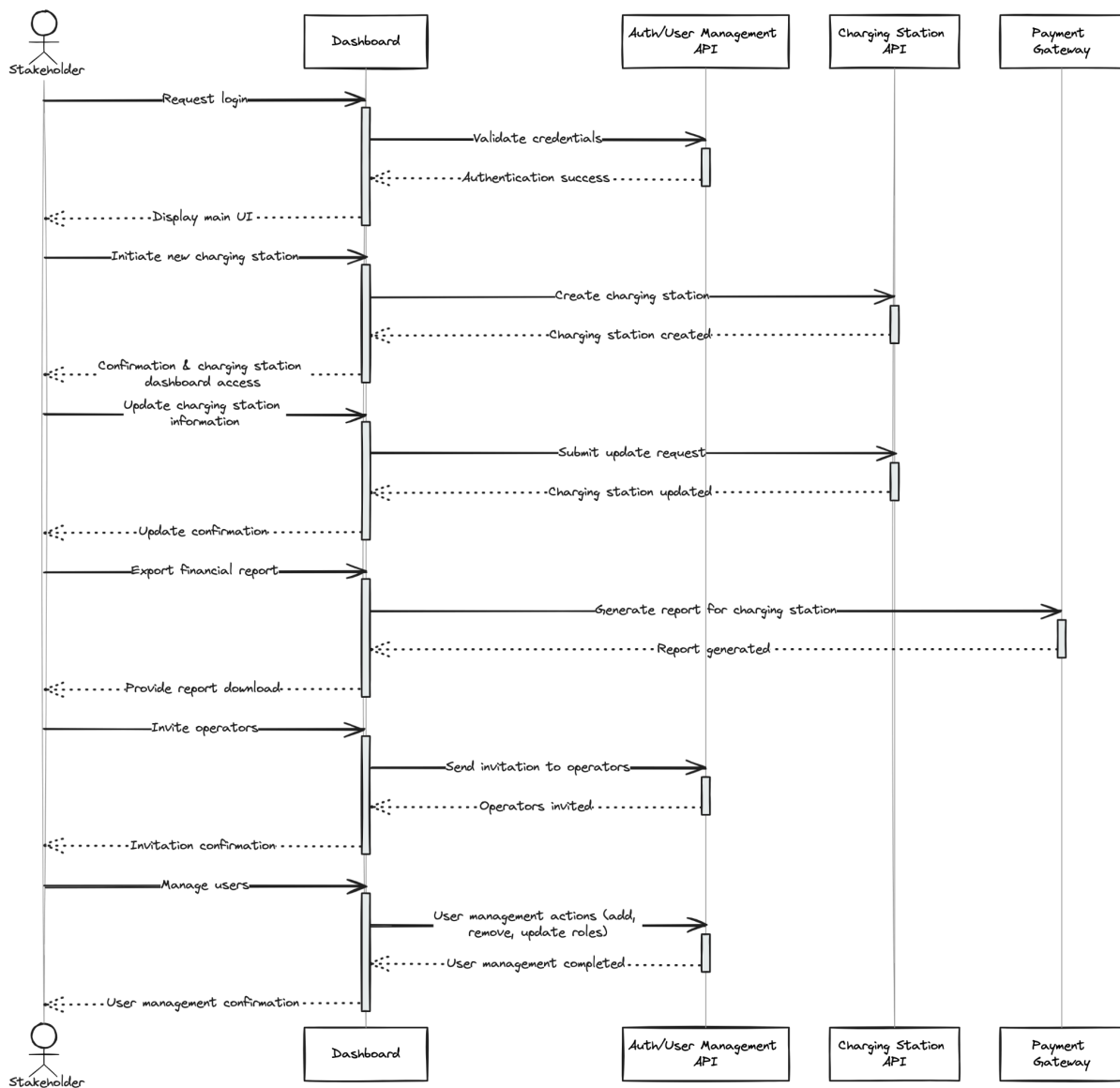
4.1 Drivers User Flow

Regarding the software requirement specification document, here is a sequence diagram explaining the driver's user flow based on the user stories provided. This sequence diagram walks you through each step that a driver needs to take from signing into the application to paying for the charging facility services.

Sequence diagram

Participants: Driver, Web Application, Authentication API, Maps API, Payment Gateway, Charging Station API

Driver → Web Application: Request login
Web Application → Authentication API: Validate credentials
Authentication API --> Web Application: Authentication success
Web Application --> Driver: Display main UI
Driver → Web Application: Requests nearby charging stations
Web Application → Maps API: Send driver's location
Maps API --> Web Application: Return nearby stations
Web Application --> Driver: Display stations on map
Driver → Web Application: Select station and book
Web Application → Charging Station API: Check station availability
Charging Station API --> Web Application: Availability confirmed
Web Application → Payment Gateway: Initiate booking deposit payment
Payment Gateway --> Driver: Payment UI
Driver → Payment Gateway: Complete payment
Payment Gateway --> Web Application: Payment success
Web Application → Charging Station API: Confirm booking
Charging Station API --> Web Application: Booking confirmed
Web Application --> Driver: Booking success notification
Driver → Web Application: Arrive and initiate charging
Web Application → Charging Station API: Start charging session
Charging Station API --> Web Application: Charging initiated
Web Application --> Driver: Charging timer starts
Driver → Web Application: End charging
Web Application → Charging Station API: End charging session
Charging Station API --> Web Application: Calculate cost
Web Application → Payment Gateway: Process payment for service
Payment Gateway --> Driver: Payment UI
Driver → Payment Gateway: Complete payment
Payment Gateway --> Web Application: Payment success
Web Application → Charging Station API: Update station status to available
Charging Station API --> Web Application: Status updated
Web Application --> Driver: Display end of session summary

4.2 Stakeholder User Flow

The following sequence diagram describes the stakeholder role flows. The stakeholder is the facility manager who has to create/update/read/delete access to all the resources in the facility. This user interacts with the platform through a management dashboard as shown below.

4.3 Station Operator User Flow

The next sequence diagram argues the operator role interactions with the locate a socket platform through a limited view of the management dashboard. They have access to limited resources based on the permissions they have been granted by the facility stakeholders. The roles and permissions for this type of user can be dynamic.
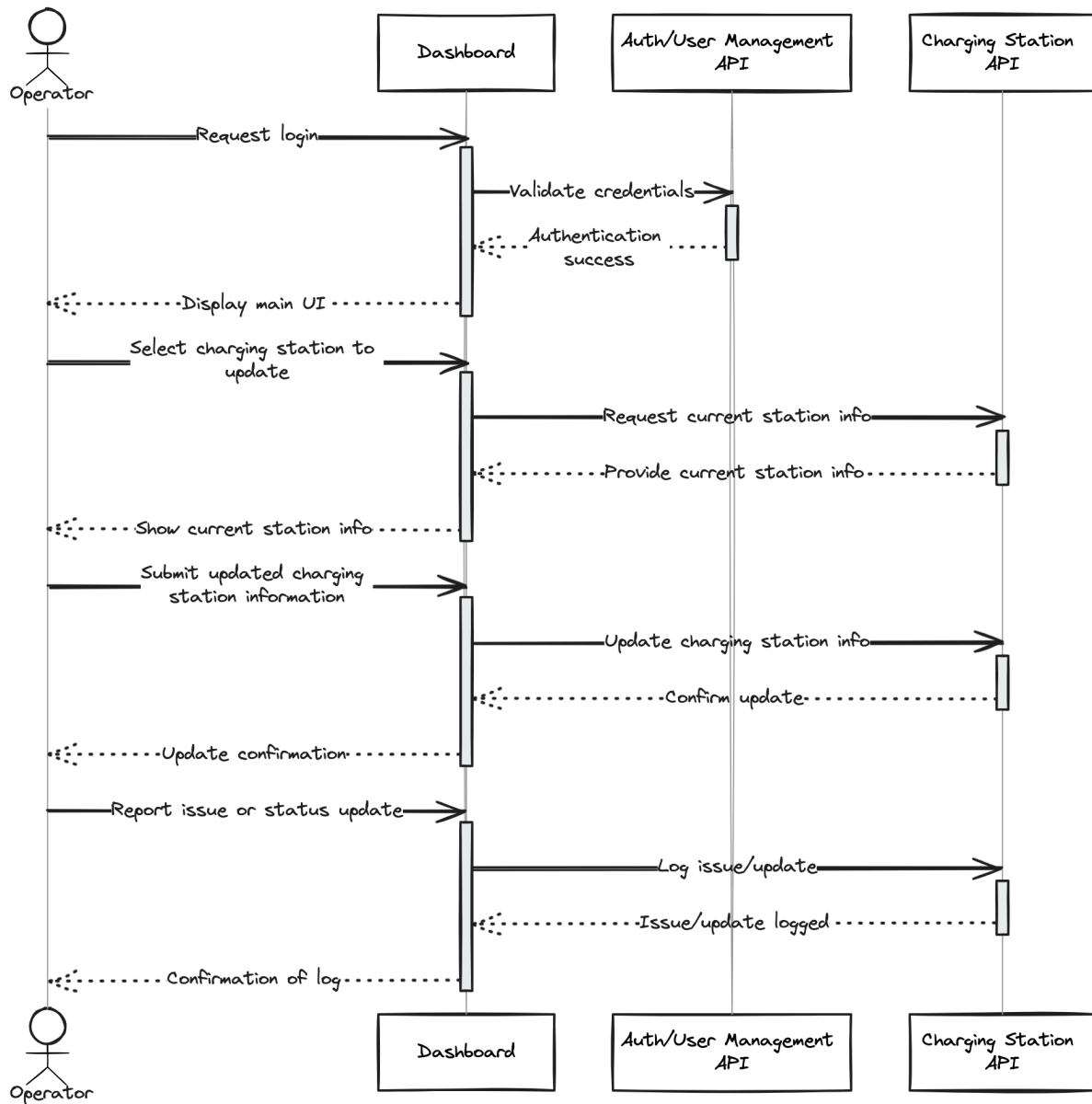
## 5. Integrations and External Services

- Payment Gateways: Integration with services like Stripe or PayPal for handling payments.
- Maps API: Use Google Maps or similar services to display charging station locations. This service will also be used for navigation using traffic data monitoring.

6. User Interface Design

The web application UI will be clean and user-friendly, designed for both mobile and desktop use. Key screens include:

6.1 Home/Map, Navigation, Station Details

The following user interfaces display the web application from the driver's point of view. In the map view driver views nearby charging stations based on his location. Then by clicking on the station card, detailed information such as charging connectors will be displayed to the user. Then the user can launch a navigation to the station taking the driver exactly to the facility's location.
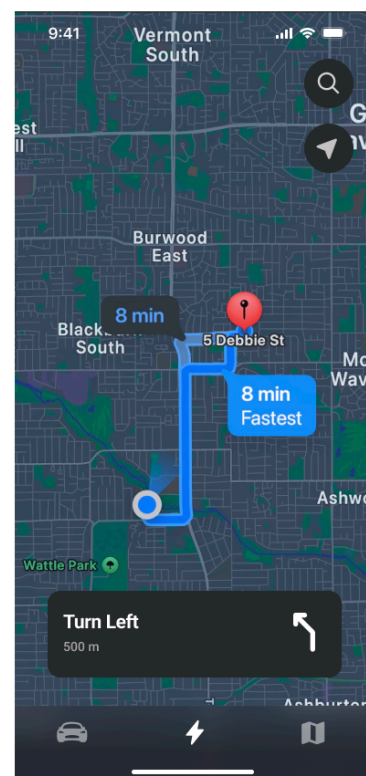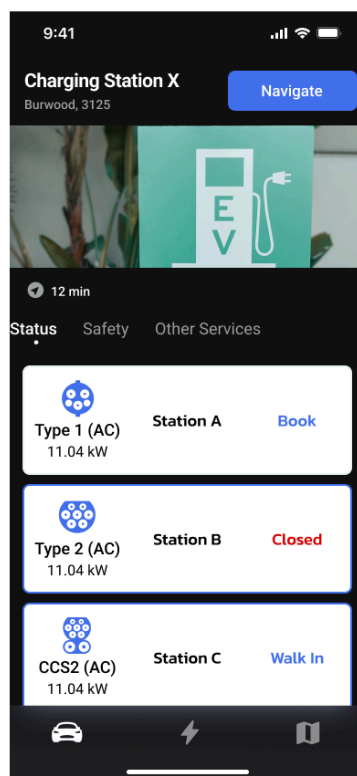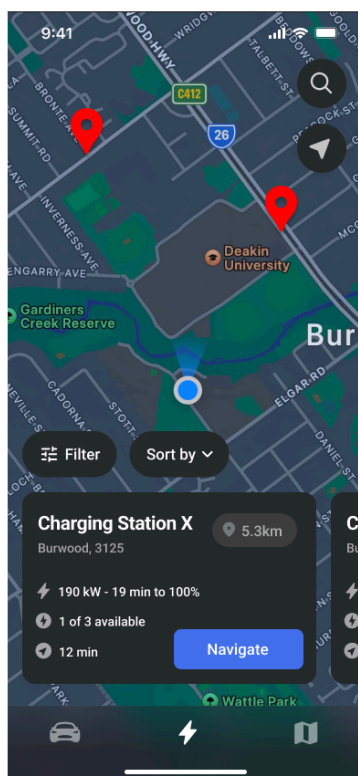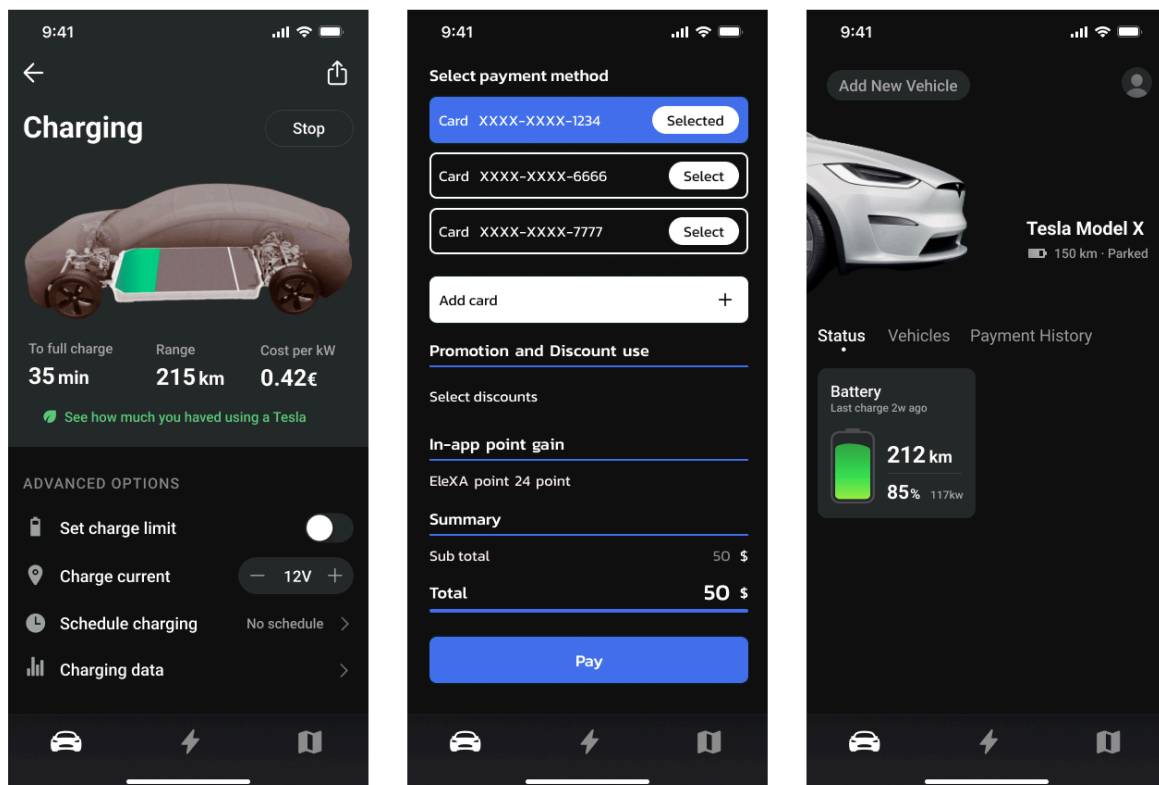
6.3 Charging and Payment Screen

After the user initiates a charging session, they can view the information about the session, At any stage they can pause the charging session and proceed to the payment page

6.4 Profile Management

Where users can edit their profiles and view their transaction history. And manage the vehicles they own.

7. Deployment Plan

The application will be deployed on cloud services like AWS or Azure to ensure scalability and high availability. Continuous integration and continuous deployment (CI/CD) practices will be adopted for smooth and frequent updates. Using cloud providers such as AWS allows us to scale easily regarding our user numbers.