



Team Volkswagen

Michigan State University

VW Car-Net Electric Vehicle Route Planner

Project Plan

Fall 2020

Volkswagen Contacts

Igor Efremov

Shelly Desmet

Team Volkswagen

Andrew Smigielski

Erich Hairston

Joey Kelly

Michael Lin

Zosha Korzecke

Contents

- Executive Summary..... 3
- Functional Specifications 4
- Design Specifications 5
 - Overview 5
 - Visual Design 5
 - Components..... 5
 - Color Scheme..... 6
 - Map 7
 - User Interface 9
 - User Experience 13
- Technical Specifications..... 16
 - System Architecture 16
 - Overview 16
 - System Components..... 16
 - Hardware 16
 - Development Environment / Programming Language..... 17
 - APIs/SDKs 17
- Risk Analysis 18
 - Calculating Fuel Efficiency..... 18
 - Implementing Charging Station Locations to the Route 18
 - Ease of Use of the Application 18
- Schedule 19

Executive Summary

With its American headquarters in Auburn Hills, Michigan, Volkswagen is one of the largest automakers in the world. Volkswagen offers a variety product lines ranging from hatchbacks, SUVs, and crossovers to various sedans and compact cars. Traditionally, all these products are gas vehicles. As Volkswagen is a leader among automakers, they strive to innovate and invest in new technology. Keeping with that trend, this year they have released their first electric vehicle, the ID.4.

Volkswagen's primary goal is to offer attractive, safe, and environmentally sound vehicles which are competitive in an increasingly tough market and set world standards in their respective classes.

Volkswagen is taking these innovations with electric vehicles to market, so a consumer has the best possible options when purchasing a vehicle.

To reach this goal, Volkswagen has employed us, team Volkswagen, to create the "VW Car-Net Electric Vehicle Route Planner". This application aims to educate the consumer about the advantages of electric vehicles to increase the popularity of their new EV product line. The application takes in two addresses from the user and returns a generated route given the constraints of the electric vehicle and conditions of the trip, then displays information about the route to the user. This information includes various cost comparisons between the ID4 and similarly built gas vehicles, travel time, carbon emissions and EV charge time. Using this information, the user will see the vast advantages of electric vehicles compared to their gas counter parts in the form of long-term cost savings, less carbon emissions, and little to no travel time penalty.

Functional Specifications

Electric vehicle technology has grown rapidly in the last few years, yet many people are still unfamiliar with the enormous, long-lasting benefits they could gain by switching from gasoline powered cars. This application shows how EVs can deliver all the needed functionalities of a gas-powered car along with the cost savings, environmental impact mitigation, and the latest automobile technology.

The main emphasis for this project is to persuade non-EV-owners to convert to electric through use of the application. Specific to Volkswagen, the goal is to convince current Volkswagen gasoline powered car owners to switch to their new EV, the ID.4. To do this, user stories have been created to identify the different types of current Volkswagen owners to better understand the key attributes withholding owners from switching.

The underlying reason identified from the user stories on why customers are resistant to converting to EV is simply inadequate knowledge of EVs. One user story describes a customer that owns a VW Atlas who is concerned whether an EV would be practical for their daily commute. With this application, they can see their daily commute would change minimally, or possibly not at all, since the application allows them to customize their ride by adjusting starting battery charge, AC usage, and weather conditions. By customizing their ride, they receive a worry-free route as our application accounts for the effects of these customizations on battery life and includes EV charging stations along the route. Along with this, the application also allows for comparison information between the ID.4 and a wide range of current VW and competitor vehicles including their VW Atlas. It will show the benefits the customer could receive not only from the route taken once, but over time as well.

Design Specifications

Overview

The EV Route Planner UI consists of a single Android front-end interface that will be used by customers in Volkswagen dealerships. This interface will be used on the Samsung Galaxy Tab S7+ and will be available to users on a kiosk within the dealership. To provide an effortless and intuitive experience to all users, the UI prioritizes simplicity in its visual design. The application adheres to the Volkswagen Style Guide to reflect the VW branding standards. There are two main views: the initial splash view and the route planning view. Once the user opens the route planning view from the splash view, they have the ability to customize a potential route, visualize charging locations, and compare the specs of the EV ID.4 and other competing gas vehicles in a concise format.

Visual Design

To follow branding standards, set in place by Volkswagen, the visual design adheres to the VW Technical App Style Guide as described below.

Components

Interface components are be built using the following Material Design Components:

- Buttons
 - Used to represent any buttons on the interface, including the “Get Started” button on the welcome splash screen and any buttons used to navigate through the app.
- Sliders
 - Used to represent route conditions.
- Cards
 - Used to display selection of gas vehicles in a scrollable carousel.
- Text Fields
 - Used to prompt the user to enter starting and ending destination.

Color Scheme

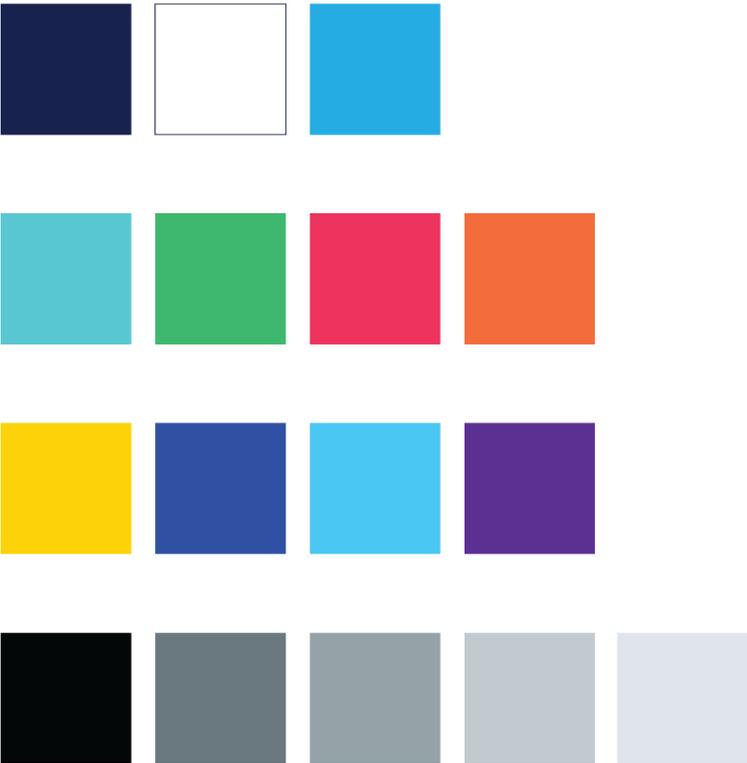


Figure 1 - Color Swatches

Map

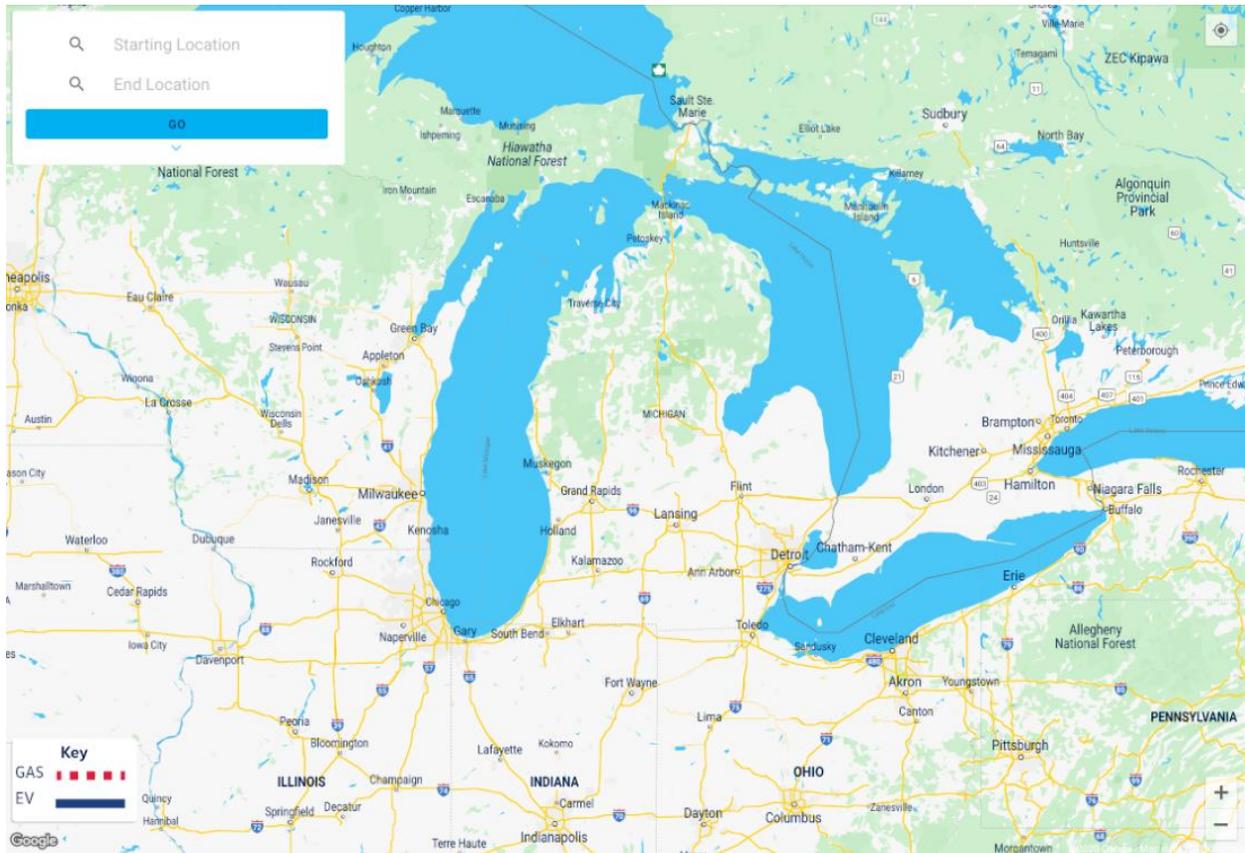


Figure 2 - Custom Google Map Style

Using Google Maps Customization features, the map interface is styled to match the VW branding scheme. As shown in Figure 2 above, the location titles and water formations are displayed according to the color scheme shown in Figure 1.

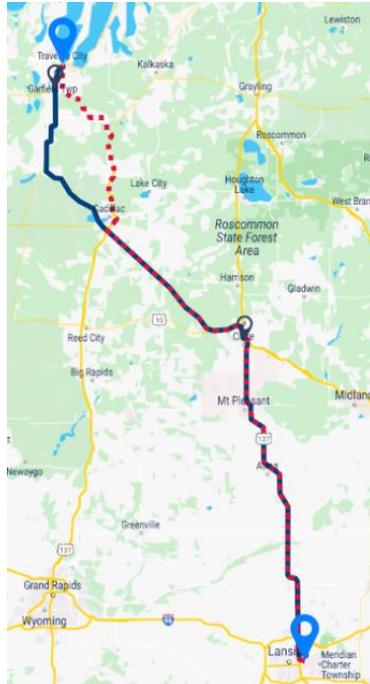


Figure 3 - Route with Charging Stations

Figure 3, above, shows a route with charging stations. A solid blue line is used to depict the optimal route for an electric vehicle and a red dashed line is used to depict the optimal route for a gas vehicle. If the routes are the same, the lines will overlap with the gas route on top. Custom icons are used to depict charging stations and start/end locations.

User Interface

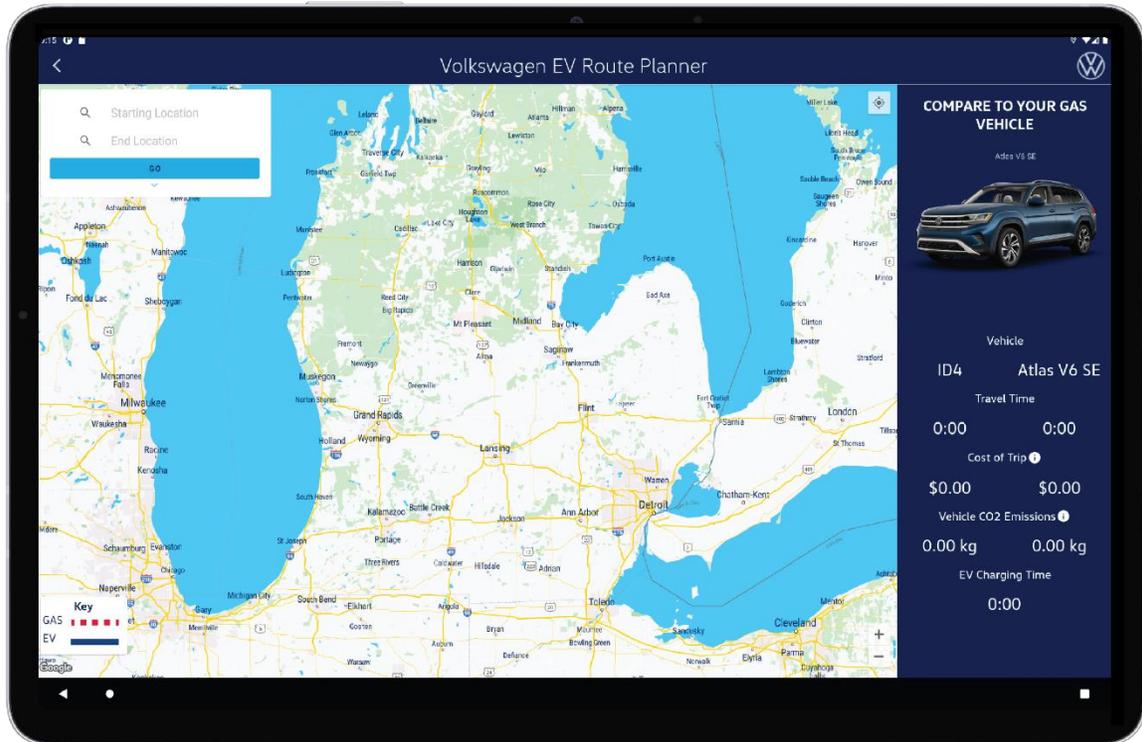


Figure 4 - Initial Map View

Figure 4 shows the view when user first opens the map screen of the application. This prompts the user to input two locations to see charging stations along the most optimal route. After inputting a route, the drop-down menu will allow the user to modify the conditions of the ride. Then, the comparison menu will fill with data allowing the user to see the statistics of the ride.

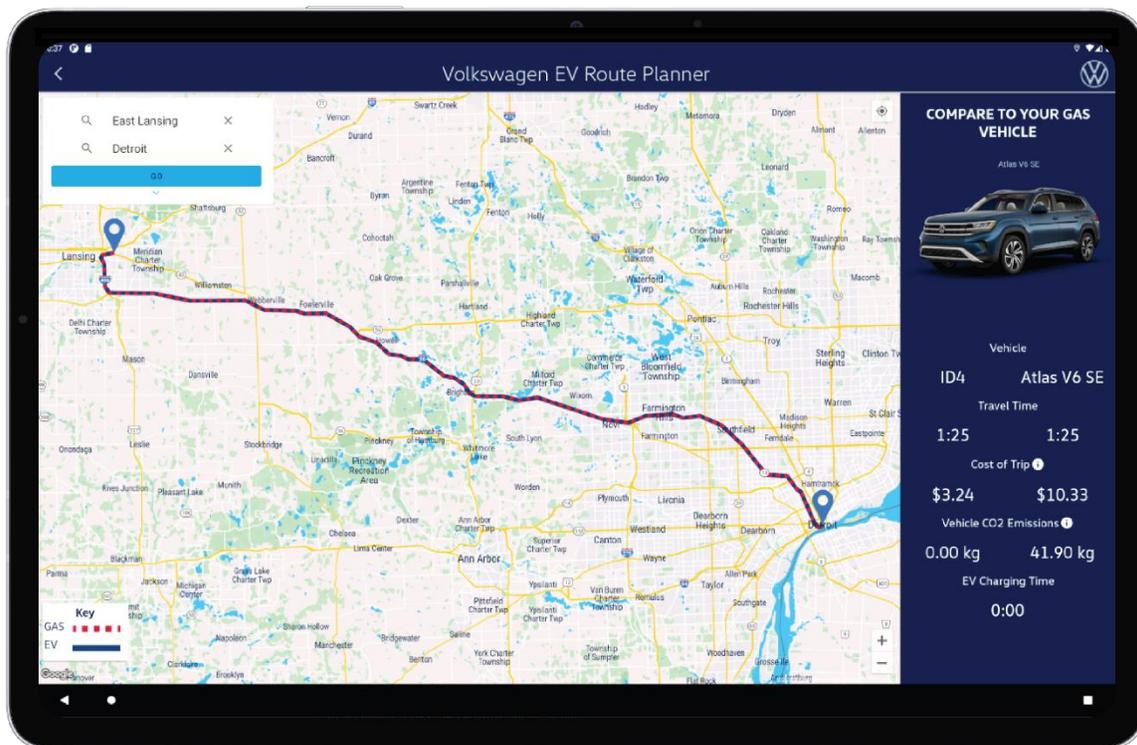


Figure 5 – Map View with Route

Figure 5 shows the map view after the user inputs a starting and ending location followed by pressing the blue “GO” button. The map will center around the route and display optimal tracks for both gas and electric vehicles. The Comparison Menu on the right will update with information from the current route.

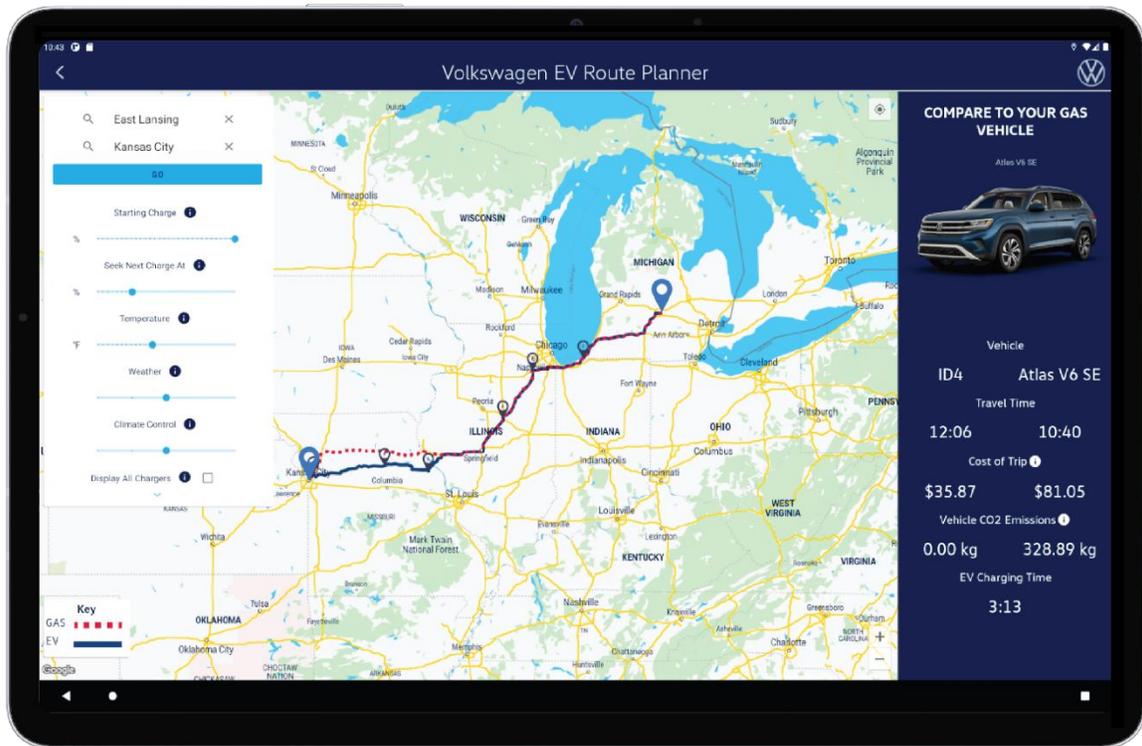


Figure 6 – Map View with Conditions Menu Open

Figure 6 shows the map view with the Conditions Menu open. When the arrow button on the Conditions Menu (the white menu on the left) is pressed, the menu will expand and show optional route customizations as sliders.

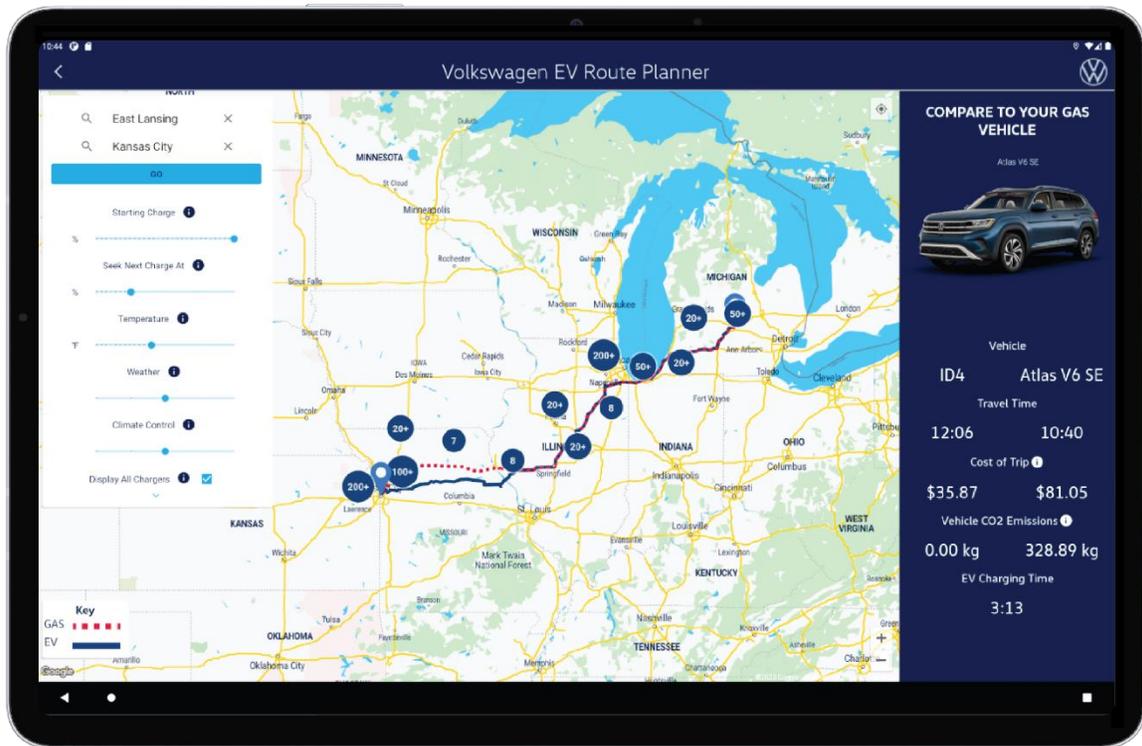


Figure 7 – Map View with All Chargers

Figure 7 shows the UI for the “Display All Chargers” feature. When the “Display All Chargers” box is checked at the bottom of the Conditions Menu, the map will display all chargers within a radius of the route. To consolidate this information, summary bubbles are used to group areas with many chargers together. As the user zooms in on the map, the chargers will regroup and disperse to display this data in the most accurate format for the current view. If any chargers are outliers and do not fit within the radius of the summary bubbles, they will be displayed as individual chargers with the charging station icon.

User Experience

This application has two main use cases. Either the car salesperson in the dealership is showing it to a possible buyer or a possible buyer is using it a kiosk in the dealership lobby. In both cases the application is used to educate the consumer on the advantages of an electric vehicle over gas vehicles. To accomplish this, our app provides many opportunities for user interaction.

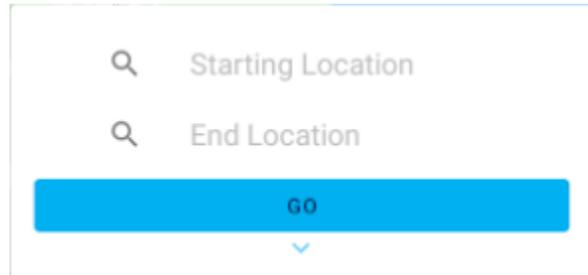


Figure 8 – Search Bars

The search bars (Figure 8) for route calculation exist at the top left corner of the Map View and prompt the user with starting and end locations. Once the user enters two locations and presses the “GO” button, the route will be calculated, and the map will update.

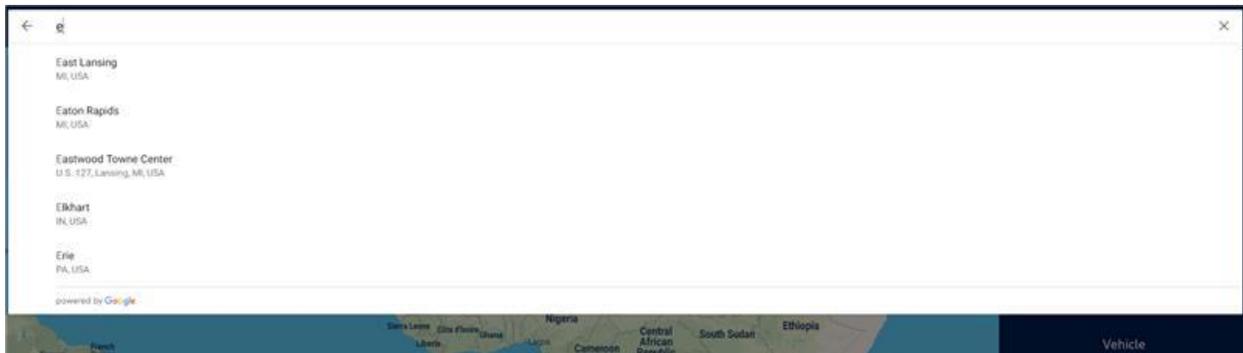


Figure 9 – Location Search Suggestions

As the user searches for locations, possible and nearby location suggestions will appear beneath the search bar as seen in Figure 9 above.

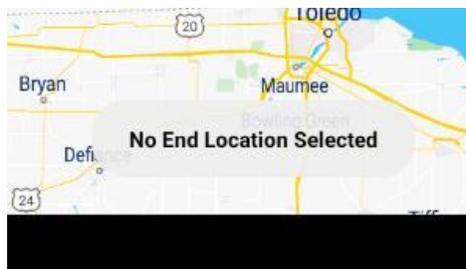


Figure 10 – Search Error Message

If at any time the user enters invalid location parameters (such as missing start or end locations) an error with an appropriate message will display at the bottom of the Map View, as shown in Figure 10.



Figure 11 – Conditions Menu

The drop-down menu (Figure 11) on the left of the map view allows the user to input two addresses and toggle the conditions of the ride. The menu can be expanded and closed with the blue arrow button at the bottom. When the user sets conditions for their ride, they must hit the blue “GO” button to update the route with the new conditions settings.



Figure 12 - Comparison Menu Filled and Default

The Comparison Menu (Figure 12) on the right side of the Map View allows the user to see the comparison of the electric vehicle to a current gas-powered Volkswagen or competitor vehicle. When a user selects a vehicle to compare to, the optimal route of the gas vehicle will be displayed on the map as well as a comparison of its specs to the ID.4. The Comparison Menu will default to an Atlas V6 SE and before a route is calculated, values will be zeroed.



Figure 13 - Vehicle Carousel

The Comparison Menu contains a carousel at the top to scroll through gas-powered Volkswagen competitor vehicles, shown in Figure 13. The user can scroll by pressing on the cars and dragging left or right. The user can select a new car to compare to by clicking a car's image. This will immediately update the comparison menu with the new comparison data.

The map component is an extension of google maps and allows the user to zoom in and out and drag the map to see the route in different ways. The user can re-center the map around their current location at any time by toggling a button to do so.

Technical Specifications

System Architecture

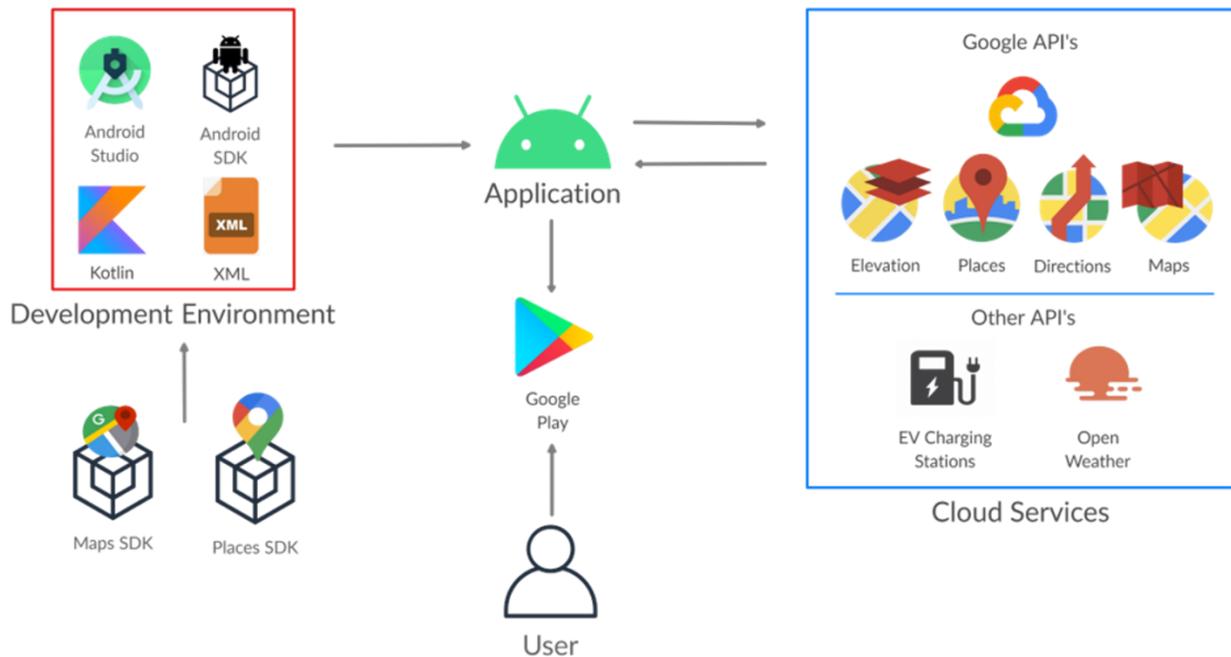


Figure 14 - System Architecture

Overview

The end user will be able to access the app through google play. The application will be delivered as an Android app with the purpose of being run on a Galaxy S7+ Tab. The app is developed in Android Studio running Kotlin and using Androids highest SDK, Android 10.0+. The app is connected to multiple API's to collect data on current road conditions and provide functionality to the map and routing as well as calculations for fuel range.

System Components

Hardware

The application is run solely on android devices with the main purpose of the application to run on Android tablets, specifically the Galaxy S7+ Tab. The tablet will be running Androids highest SDK 10.0+

and will be large enough with the dimensions of 11.22 x 7.28" for the ease of use required for this application.

Development Environment / Programming Language

Android studio is the main tool used for the development of the application. Android studio provides both a very easy to use UI development interface as well as emulators of most of the phones/tablets/devices that Android's SDK runs on. The application is running Kotlin as it is one of the main languages supported by Android studio.

APIs/SDKs

Android SDK

Provides many SDKs required to develop in Android studios, providing services including but not limited to the SDK manager interface which allows easy SDK use, the use and building of the android emulator, and various build and platform tools required to build the application.

Google Maps SDK for Android

Adds the map display and data used by the application based on Google's map data. The SDK handles access to Google map servers and the user interactions on the map like dragging and clicks.

Places SDK

The SDK adds an autocomplete fragment on our application which allows the user to look up addresses easily. The fragment will return a place object with the address of an exact location and information on the place.

Directions API

The API will return a route between two locations in a geocoded JSON File after an HTTPS call. The route will have exact coordinates and a places ID for every coordinate. The API is used to draw the route along the Maps SDK as well as help in calculating fuel cost.

Maps Elevation API

Returns elevation data for any point in the world. This is used to calculate the fuel cost of the car as it goes along its route.

OpenWeather API

Returns the information on the weather from any location. This provides data for fuel calculations.

Department of Energy EV Charger API

Returns EV charger information on the charger's location, plug type, and payment system. This API works in conjunction with the Directions API to provide EV charger waypoints that the Directions API will route to.

Risk Analysis

Calculating Fuel Efficiency

- **Difficulty:** High

- **Description:** There are a multitude of conditions that is needed to be considered when considering a cars fuel efficiency. We need to find a way to use calculate a cars MPG or charge range using the cars weight, elevation change, road type, weather, etc.

- **Mitigation:** The client will send specifications on the cars the application will be comparing. The data we need to collect to predict a cars fuel efficiency is readily available. Most of the APIs have been researched, only a few more might be needed.

Implementing Charging Station Locations to the Route

- **Difficulty:** Moderate

- **Description:** Adding nearby charging station locations to the route will be a challenge. It is possible to add way points along the route calculated by the API, However the EV charger locations along the route will need to be found and the optimal chargers added to the route.

- **Mitigation:** The directions API allows for easy additions of waypoints between two locations and the EV charger's locations can be found with the Department of Energy EV Charger API as well.

Ease of Use of the Application

- **Difficulty:** Low

- **Description:** The application must be easy to use for both the salesmen and consumers who may not be technologically savvy. It must also be accessible to account for consumers with disabilities.

- **Mitigation:** Field testing and working with the client to make sure the application is what they want and is easily operable.

Schedule

Week 1: 8/31 -9/6

- Initial meeting with group members
- Introduction to project and project details

Week 2: 9/7 -9/13

- Initial meeting with client
- Research necessary software and technologies
- Set up a Git repository
- Create initial Android App

Week 3: 9/14 - 9/20

- First Status Report Presentation
- Begin Project Plan
- Delegate team member roles
- Prototype Google Map feature

Week 4: 9/21 - 9/27

- Begin first UI iteration
- Develop Personas and User Stories
- Present UI to the Client to begin second iteration
- Acquire appropriate access to Google APIs and integrate them into the prototype

Week 5: 9/28 - 10/4

- Present Project Plan
- Finalize UI/UX Designs with the Client
- Begin creating UI prototype
- Begin prototype for showing electric vehicle locations on a map

Week 6: 10/5 - 10/11

- Integrate prototypes into one application
- Implement UI for route customization
- Implement backend for route customization

Week 7: 10/12 - 10/18

- Polish up initial application with routing, locations, and user customizations
- Team Status Report Presentations
- Prepare for Alpha Presentation

Week 8: 10/19 - 10/25

- Team Alpha Presentation
- Begin gathering information for vehicle comparison
- Begin developing frontend for vehicle comparison

Week 9: 10/26 - 11/1

- Team Alpha Presentation
- Implement Vehicle comparisons and second routing feature
- Implement conditions sliders
- Polish user interface and assure it adheres to style guide

Week 10: 11/2 - 11/8

- Meet with client and discuss further features to add to the application
- Add splash screen animation
- Begin user testing
- Prepare for Beta presentations

Week 11: 11/9 - 11/15

- Add additional features to application
- Demo the application to the client
- Prepare for Beta presentations

Week 12: 11/16 - 11/22

- Team Beta Presentations
- Begin finalizing all work on the application
- Polish the project plan as needed
- Create a plan for project video

Week 13: 11/23 - 11/29

- Team Beta Presentations
- Finish Team Status Report

Week 14 11/30 - 12/6

- Present Team Status Report
- Finish Project Videos

Week 15: 12/7 - 12/13

- Design Day