







**SP-6 Blue – Time Mileage Location Tracker –  
RouteLink Tracker**

**Requirements Document**

**CS 4850 – Section 02 – Spring 2024 – January 19, 2024  
Professor Sharon Perry**

 <p><b>Armando Orti</b></p>	 <p><b>Jacob Carrington</b></p>	 <p><b>David Huu Nguyen</b></p>
 <p><b>Andrew Millsap</b></p>	 <p><b>Ethan Strickland</b></p>	 <p><b>KSU Spirit</b></p>

# Table of Contents

1.0 Introduction .....	3
1.1 Overview .....	3
1.2 Project Goals .....	3
1.3 Definitions and Acronyms .....	3
1.4 Assumptions.....	3
2.0 Design Constraints .....	3
2.1 Environment .....	3
2.2 User Characteristics .....	3
2.3 System.....	4
3.0 Functional Requirements.....	4
3.1 Create an account .....	4
3.2 Sign-In to the App .....	4
3.3 Reset Password .....	4
3.4 Two-Factor Authentication .....	4
3.5 Start and Stop Trips.....	4
3.5 Save Trip Information.....	5
3.6 Generate Trip History Report.....	5
3.7 Generate Directions .....	5
3.8 Display Visual Map .....	5
3.9 Temporary Data Backup.....	5
3.10 Display Ads .....	5
3.11 Premium Account .....	6
4.0 Non-Functional Requirements.....	6
4.1 Security .....	6
4.1.1 - User passwords .....	6
4.1.2 - User emails.....	6
4.1.3 - Permissions .....	6
4.2 Usability .....	6
4.3 Accessibility.....	6
5.0 External Interface Requirements .....	6
5.1 Hardware Interface Requirements.....	6
5.2 Software Interface Requirements .....	6
5.3 Communication Interface Requirements .....	7
APPENDICES.....	7
Appendix A – Figma App Skeleton .....	7

## 1.0 Introduction

### 1.1 Overview

Over this document you will see the unfolding of our time mileage tracking app by the various details about the app such as different functional and non-functional requirements, assumptions, and external requirements. This document should help you understand that we are not building just another simple map app but instead we are striving to create a map app that will replace the user's current app used for navigation. This will be accomplished by the various new features being offered to them such as GPS tracking, trip history, user friendly UI, and the ability to export trip data for tax reasons to just name a few.

### 1.2 Project Goals

- Create a functional app that can help users keep track of their travel history between various locations.
- Provide a map display and direction generation within the app to allow users to only require one app for all for their travelling needs.
- Track information about trips such as location and distance so that the user can use this information during tax season to get write-offs for driving between work locations.

### 1.3 Definitions and Acronyms

Here are the definitions or acronyms are used throughout this documentation:

- API “application programming interface”
- DB or Database “organized collection of information or data that is normally stored electronically in a computer system”

### 1.4 Assumptions

Our app will be very basic to not only use but also easy to understand if we assume the following:

- All devices that use the app will have location services enabled.
- All devices that use the app will have the necessary GPS hardware for location tracking.
- All devices that use the app will either have iOS or Android operating systems.

## 2.0 Design Constraints

### 2.1 Environment

- Languages: JavaScript
- Frameworks: React Native
- Database: Firebase

### 2.2 User Characteristics

- This app is designed for the general consumer
- The interface of the app must be intuitive and easy to learn
- Functionality is clearly labelled and easily understood

## 2.3 System

- The created app must be functional on both Android and iOS devices

## 3.0 Functional Requirements

### 3.1 Create an account

#### MUST HAVE

- Allow a user to create an account using a Username, Password, and Email Address
- This account data will be stored within the database of the system

### 3.2 Sign-In to the App

#### MUST HAVE

- Allow a user to sign into an account they have created using a combination of Username or Email and password
- A successful sign-in will allow the user to access all information and features their account has permissions to access

### 3.3 Reset Password

#### MUST HAVE

- Allow a user to reset the password to their account
- User will be prompted to enter the Email or Username associated with their account
- An email will be sent to the Email associated with the provided account
- The email will link to a password reset page located on the website associated with the application (unless there is a better way to do this)
- The Password entry in the database associated with the account will be changed after the reset password request is submitted

### 3.4 Two-Factor Authentication

#### MODERATELY IMPORTANT

- Allow a user to enable the Two-Factor Authentication feature
- This feature will send an email to the user's email with a randomly generated code
- If the user inputs the correct code, they will be successfully signed in

### 3.5 Start and Stop Trips

#### MUST HAVE

- A user must be able to begin and end a trip when they desire
- When a trip is started, the app will take note of the time and begin tracking the location of the user
- While a trip is being tracked, information such as location and time will be stored in the database
- When a trip is ended, tracking will stop, and a final time will be added to the trip in the database

### 3.5 Save Trip Information

#### MUST HAVE

- Allow a user to store data taken from trips tracked by the app
- The app will track the duration of the trip, the distance travelled during the trip, and the date/time of the trip
- Each trip will be stored independently in the database, linked directly to the account that started the trip

### 3.6 Generate Trip History Report

#### MUST HAVE

- Allow a user to generate a report consisting of data regarding their trips over a certain period
- Each report will contain the individual trip data and summary data such as total distance travelled, time spent driving, average speed, etc.
- The amount of time outlined in the report can be selected from certain defined presets (One day, one week, two weeks, one month, three months, six months, one year).
- Additionally, if a user's account is premium, the user will be allowed to customize the report, selecting the specific date/time boundaries for which they want their report.

### 3.7 Generate Directions

#### MODERATELY IMPORTANT

- When a user enters the app, allow them to generate directions from one location to another
- Starting location will default to the current position of the User, though the starting point can be changed
- The directions will be displayed to the user once generated

### 3.8 Display Visual Map

#### MODERATELY IMPORTANT

- When a user is on the app home screen, a map will be displayed with the user's current location
- After directions are created, two points will be displayed on the map
- The path between the directions will also be highlighted

### 3.9 Temporary Data Backup

#### LOW IMPORTANCE

- Temporarily store trip data locally on the device
- In the case of the database being down, this local cache will be stored temporarily and uploaded once the database is back up

### 3.10 Display Ads

#### MUST HAVE

- While the user is in the app, display banner ads provided by Google AdSense
- Track ad clicks and views in database (maybe)

### 3.11 Premium Account

#### MODERATELY IMPORTANT

- Users will be able to purchase a premium version of the app
- This version will remove ads from being displayed to the user
- Certain features will be unlocked when a user is using a premium version of the app

## 4.0 Non-Functional Requirements

### 4.1 Security

#### 4.1.1 - User passwords

- Store user's passwords in database with a hashed version of the password
- When a user is entering their password, blur the password unless they click the button to unblur the password. This behavior can be seen both during logging in and or changing passwords in settings

#### 4.1.2 - User emails

- Obscure email when displaying to the user (i.e. instead of JohnDoe123@gmail.com, display J\*\*\*123@gmail.com)

#### 4.1.3 - Permissions

- Limit data access with permissions (Users can only see their trips, while admins can see the entire database)

### 4.2 Usability

- Users will be able to use the app with minimal to no training
- Users will be able to easily and quickly generate a report of their past trips

### 4.3 Accessibility

- Users will be given an option of different visual themes according to their preferences (dark, light)
- While travelling, directions will be read aloud to the users (The speed of the speech can also be adjusted)

## 5.0 External Interface Requirements

### 5.1 Hardware Interface Requirements

- Compatibility with Apple CarPlay and Android Auto

### 5.2 Software Interface Requirements

- Google Maps Platform
- Maps API for Dynamic Visual Map
- Routes API for generating directions between locations within the app
- Connection to Firebase database

### 5.3 Communication Interface Requirements

- User must have some form of internet connection to interact with the maps and database

## APPENDICES

### Appendix A – Figma App Skeleton

[Figma Mockup for App Skeleton](#)