**Software Requirements Specification**

**for**

**Envorso Car Charging Mobile Application**

**Version 1.0**

**Prepared by Kirsten Boyles**

**CWU**

**January 9th, 2022**

**Table of Contents**

**Table of Contents ii**

**Revision History ii**

**1.** **Introduction 1**

1.1 Purpose 1

1.2 Definitions / Acronyms / Abbreviations 2

**2.** **Overall Description 2**

2.1 Product Perspective 2

2.2 Product Functions 2

2.3 User Classes and Characteristics 2

2.4 Operating Environment 2

2.5 Design and Implementation Constraints 3

2.6 User Documentation 3

2.7 Assumptions and Dependencies 3

**3.** **External Interface Requirements 4**

3.1 User Interfaces 4

3.2 Hardware Interfaces 5

3.3 Software Interfaces 5

3.4 Communications Interfaces 5

**4.** **System Features 5**

4.1 System Features 5

**5.** **Other Nonfunctional Requirements 7**

5.1 Performance Requirements 7

5.2 Safety and Security Requirements 7

5.3 Software Quality Attributes 7

**Revision History**

| **Name** | **Date** | **Reason For Changes** | **Version** |
| --- | --- | --- | --- |
|  |  |  |  |
|  |  |  |  |

# **Introduction**

Electric Vehicles (EV) are cars that have the ability to operate by electricity stored within a battery. As Gasoline is an unsustainable resource and the damages of CO2 has increased, the push for electric cars becoming the norm has escalated in the recent years. As a result, the infrastructure to support these types of vehicles has also increased and several businesses have come into the field to fulfill this need. Tesla is one of the leaders in both supplying the cars and the infrastructure needed to charge them. As Tesla was one of the first in the field, they have their own plugs and their own charging stations that are only able to be used by Tesla customers. Tesla is not the only company in either the EV producing and charger producing field. Ford, Hyundai, Nissan, and most other car companies have started their own line of EV to sell and they all have the same standard charging plug (different from Tesla’s) and also have different brands, prices, and no centralized way to find these charging stations, check prices, and pay for them. The application that will be made as a result of this document will be able to satisfy these needs.

## **Purpose**

This document describes how the mobile application is expected to work and how it will satisfy customers needs.

This System Requirements Specifications is Kirsten Boyles’s version and is meant to be one of the versions that will be used to create a final version as a group at a later date.

The purpose of this document is to identify the needs the customer has and how the application will satisfy them in the delivery of the mobile application (High Priority). The document also includes possible features that can be applied to the mobile application (Optional) at a later date.

The intended party for this document is Envorso employees, fellow project mates, and Central Washington University Computer Science professors.

The use of the mobile application is intended for anyone who owns an Electric Vehicle that has a Port J1772, CHAdeMO, or SAE Combo CCS charging plug.

## **Definitions / Acronyms / Abbreviations**

| Term | Description |
| --- | --- |
| EV | Electric Vehicle |
| CC | Credit Card |
| M | Mandatory – Must be in the prototype |
| O | Optional – Possible improvements post prototype |
| CWU | Central Washington University |

# **Overall Description**

## **Product Perspective**

The product is being created for Envorso Consulting by a team of students from Central Washington University for their Senior Capstone project in Winter Quarter 2022. Their will be a mobile application front end and a data base to store user information.

## **Product Functions**

* User will be able to log in
* User will be able to search for nearby charging stations
* User will be able to see the prices and distance and location of the charging stations
* User will be able to use the app to pay using various different charging stations

## **User Classes and Characteristics**

There will only be one class of users, the owners of the Electric Vehicle that will be the targeted user base and they generally will be well versed in technology and how to use a mobile application.

## **Operating Environment**

The operating environment will be on mobile devices, IOS and Android. In the prototype phase it will be ideal to focus on one operating system first and have a fully fleshed out idea on one operating system and then move over converting the app to the other operating system after the product wanted is achieved.

The system must also be able to interact with other companies applications that are used to pay for their charging stations.

## **Design and Implementation Constraints**

* Interaction between other companies logins and our system will be a large issue and constraint.
* Creating an application for both operating systems will be a constraint.

## **User Documentation**

Brief user tutorial at the beginning, then a help page will be available afterwards for the users to access via tab.

## **Assumptions and Dependencies**

Issues with interfacing with other companies logins and apps. Credit card information.

# **External Interface Requirements**

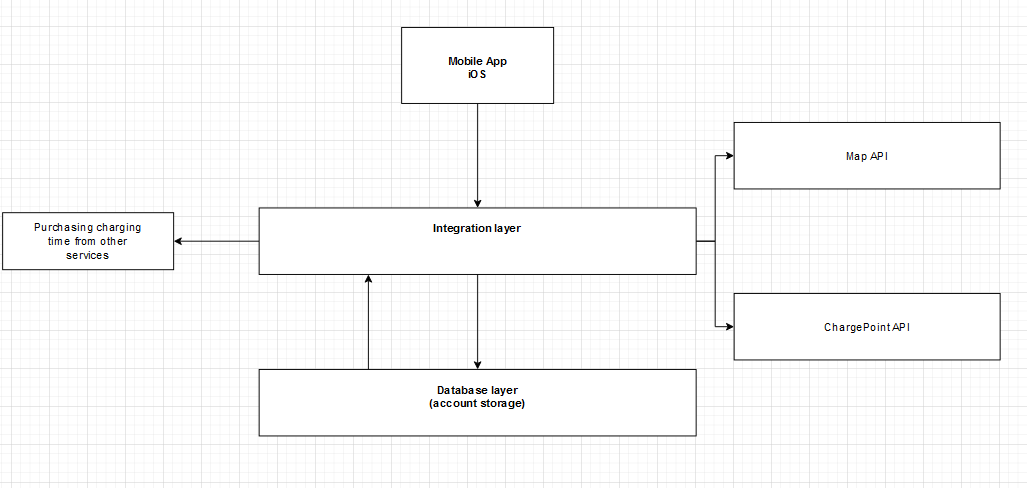
## **User Interfaces**

## **Hardware Interfaces**

Supported devices will be mobile devices (i.e. cellphones). The starting target audience will be IOS users as per the clients wishes. Communication protocols will be via cellular signal and GPS.

## **Software Interfaces**

The system will be connected to a database with update to date information about charging ports. Data will be stored on the external database and the app will communicate with it to find the information that the users is requesting. Will be written in Flutter.



## **Communications Interfaces**

The application will have to communicate with other charging applications or programs in order for the user to properly pay for them. Encryption is needed due to credit card information for the user will be stored and sent to these other applications. Cellphone signal or internet (Wi-Fi) will be required in order for this application to work properly. If disconnection occurs, it will continue the task according to the last known information.

# **System Features**

## **System Features**

H – High priority

L – Low priority

O – Optional (future update)

| Feature | Description | Priority |
| --- | --- | --- |
| Account | The user will have an account that will be tied with their preferences and their user information | H |
| Registering with other services | The use will be able to select which charging services they want to sign up with and the system signs them up using the information they provided when making their account | H |
| Map | A map will display nearby charging stations to user | H |
| Search | Allows the user to search for specific charging stations | O |
| Map filter (or priority) | Allows the user to set a preference for what charger type they want and pricing | L |
| Settings | Allows the user to set their preference and allows them to add and delete subscriptions | H |
| Plan a drive | Have a drive plan made showing all of the charge ports in each area already saved for them | O |
| Save old charges | Allow the user to star or favorite charging ports they like to find again easier in the future | O |
| Report | Function that allows the user to report on the status of the charging station | O |
| Voice | The user will be able to use voice commands to ask input search results | L |

## **Analysis**

| Feature | Analysis |
| --- | --- |
| Account | Will store user preferences and account information for other accounts. Will be stored on a database. Needs encryption and security measures (if time allows) |
| Registering with other services | Creates accounts on other services for users using their information they gave. Possible solution would be a bot put their information in automatically. More investigation needed |
| Map | Google map API most likely |
| Search | Google Map API |
| Map filter (or priority) | Custom according to locations |
| Settings | Change the Database saved settings, may set preferences for map filter here? |
| Plan a drive | Lowest priority, investigate further after map is fully implemented |
| Save old charges | Should just be able to save locations on the google map API |
| Report | A feature that would “review” locations and will be stored on internal database |
| Voice | Text to speech, investigate further, but flutter has options for this |

# **Other Nonfunctional Requirements**

## **Performance Requirements**

Must communicate between different systems and the google maps api in a quick and timely manner. Some loading time is expected for account creation and loading new areas, but bit should be as quick as possible.

## **Safety and Security Requirements**

Concerned with users credit card information and the safety requirements regarding that. Also general account security as it interacts with other accounts and systems. Encryption and data authentication may help with this.

## **Software Quality Attributes**

The product must be able to do the following in testing:

* User must be able to make an account
* User must be able to connect to other charging platforms to pay
* User must be able to use a map to find nearby charging stations
* User must be able to filter the map results