System Requirements Specifications

Carspot for SE 3A04, Tutorial 2

Yasaswi Gopalkrishnan Sharon Platkin

Abhijit Singh Dhoat Joseph Cole Huot

David Eric Hemms Yuchen Liu

March 7th, 2016

Contents

List of Tables

Table 1: Revision History

Revision	Date	Sections Modified	Contributors
Revision 0	February 7th	-	(check work division table)
Revision 1	March 7th	Section 3 - Functional Requirements	Yash,Sharon,David and Cole

Table 3: Work Division

Team Member	Section Completed	Section Reviewed/Edited	Last Date Modified
Abhijit	1,2.6	2	February 4th, 2016
Cole	3,2.1	1	March 5th, 2016
David	4(even),2.4	3	February 4th, 2016
Sharon	3,2.2	1,2.6	March 5th, 2016
Yash	1, 2.3	4	February 4th, 2016
Yuchen	4(odd), 2.5	3	February 4th, 2016

1 Introduction

1.1 Purpose

The purpose of this software requirement specification document is to notify the reader about the android application designed and programed by group 9 in SFWR 3A04 at McMaster University. This document contains information about the: purpose, scope, overall description, functional and non-functional requirements along with a division of labor. The primary audience for this document is individuals who have an understanding of android programming, specifically the members of group 9 and the 3A04 instructional staff. This document is subject to change until the project completion date as group 9 sees fit to best describe the current application they have designed.

1.2 Scope

The android app the team will be making is called 'Carspot'. The application will allow the user to identify a car they spot quickly by entering the characteristics they recognize. It will then search through a database using the minimum characteristics the user provides to narrow it down to a particular car. It will show the car's details and then show you the nearest car dealerships that sell that particular car. It will not show you any more details about the car as the app allows you to redirect to the dealerships website where you can search for something in particular. It also provides directions to the dealerships if necessary. The app does not do any image recognition as the name seems to imply; this is done because it is hard for the user to take a picture of a car that is on the move. The app is also limited to just the Hamilton area for the dealership listing.

The application is designed to provide a better, faster and easier way for people to lookup a car they like and buy that car if they were already in the process of looking for one. It will be very intuitive to use with a very minimal user interface and a walkthrough-like user experience.

1.3 Definitons, Acronyms & Abbreviations

CI: Car Identifier - The algorithm that searches the custom database for a car based on the characteristics specified by the user.

API: Application Program Interface - External Service or Software we communicate with for certain data.

UI: User Interface - How the application looks and feels on Android. It includes everything from the design to the branding.
UX: User Experience - How the user interacts with the application.
Search Form - The set of characteristics that the user goes through to finalize the search criteria

1.4 References

There are no documents that this SRS references but just a suite of resources that the app will use. Hence, the tools the team will use are listed below:

- **Tabris.js:** Native mobile app framework for Javascript (`https://tabrisjs.com/`)
- **Ionic:** Front-end SDK for mobile apps (`http://ionicframework.com/`)
- **React.js:** UI Framework (`https://facebook.github.io/react/`)
- **Github:** Version control and project management. Going to be used to host our project files. (hrefhttps://github.com/)
- **DigitalOcean:** Cloud Hosting. Used to host our databases possibly. (`https://www.digitalocean.com/`)
- **Codeship:** Continous Delivery & Deployment. Will be used to pull code from Github and onto DigitalOcean servers and run tests. (`https://codeship.com/`)
- **Android SDK Docs:** Will be used to figure out how to leverage androids various system components and resources. (`http://developer.android.com/guide/index.html`)

1.5 Overview

This document will include an overall description of the application that will describe the basic functions and features of the application. It will also describe the main constraints, assumptions, and dependencies of the application. Along with business events, nonfunctional and functional requirements that the application fulfills. The software requirement specification document is by far one of the most crucial and useful documents throughout the design process and should be one of the most referenced documents throughout the design.

2 Overall Description

2.1 Product Perspective

The product is a mobile application on the android platform. The application will require use of a database of car features in order to identify cars. Google maps will also need to be accessed to retrieve the locations of nearby car dealerships, and to give directions to the user. The websites of dealerships must be accessed as well to determine if they sell a specific car.

2.2 Product Functions

-The application will receive information about a car and will identify it
-This application will be able to handle different amounts of information about a specific car to identify it by allowing the user to skip over questions
-The application will provide the user with car dealerships information
-The application will allow the user to skip sections they have no answer to
-The application will provide the user with a list of dealerships
-The application will give the option to sort the dealerships in either alphabetical order or by nearest location
-The application will redirect you to the dealership website

2.3 User Characteristics

The users of the product will be the general consumer between the ages of 18 and 60. The user will have at least a high school education and should know how to use a smartphone to a reasonable degree. They are assumed to know how to use the camera and other functions. The target audience is someone who is interested in buying a car in the near future. The user's knowledge about cars is irrelevant, but it will be most useful for those who know only a bit or nothing about cars.

2.4 Constraints

- The application must be fully implemented and operational as specified by the project timeline.
- The application is built to run on Android OS devices, and must be backwards compatible with older versions.

2.5 Assumptions & Dependencies

2.5.1 Factors that may affect the requirements stated in SRS

1. Vehicle market flow.
2. The changing of vehicle shop.
3. The technique spec change for some cars.
4. The knowledge of the user
5. The user's first language (for example: the same car, may have different name in English and French).
6. The currency exchange rate (for some imported cars).
7. The copyright of those vehicle sellers (if they agree we use their logo or other stuff in our application).

2.5.2 Environment Factors (Program Environment)

1. The version of Android.
2. The brand of the phones (Samsung, HTC, Huawei, etc).
3. The interface of the Google Map on the phone.
4. The version of Google Map on the phone.
5. If GPS location available on the device.
6. If the device can provide internet connection.

2.6 Apportioning of Requirements

Innovative/Creative Features

The application has some delayed requirements which may be fulfilled at a later date.

- The application should have a web portal, where users can log in and have similar functionality to the app using their web browsers.
- The application should provide a system in which administrators can view the app as if they were a User, so they can understand what the User sees.

3 Functional Requirements

The following are the experts that will be used in the system. The user will be prompted to pick 3 experts of the 4, and pick one option from each of those 3 experts.

- **Type:** This expert will hold the different categories of cars there are such as Sedan, SUV, Coupe, Hatchback, Minivan or so on.
- **Logo:** This expert will hold the many logos of car makers in the world. The text will also be supplied beneath the logos.
- **Number of Passengers:** This expert will hold small illustrations of the size of the car from a 2 seater, all the way to a 10 passenger vehicle.
- **Origin:** This expert will hold the general geographic origin of a car. American, European (Generic), German (Specific) or Asian. Cars made in each of these area have a similar look or design that carries across all the manufacturers from that area.

1. BE: User wants to identify car
   (a) VP: User
      i. The user will enter relevant attributes from the domains of the experts about the car that they want to identify into the application.
   (b) VP: Developer
      i. The application will implement three distinct experts in the forum.

      ii. An expert, given sufficient information according to its expertise will identify the car in question, and provide an answer to the forum.

2. BE: User wants to swap experts

   (a) VP: User:

      i. The user shall be able to swap experts to cater to known attributes about the car.

   (b) VP: Developer

      i. If current experts incorrectly identify car three times, the application will force one expert to be swapped out.
      ii. The application shall have one expert on standby that can be swapped into the forum when needed.
      iii. Experts shall be chosen before the search attributes are entered into the application.

3. BE: User wants to verify result of search

   (a) VP: User

      i. The user will be able to confirm or deny the result of the car identification in the forum.
      ii. If the result is denied three times, the user will be prompted to re-enter the car attributes.

   (b) VP: Developer

      i. If the result is incorrect, the application will attempt to re-identify the car, knowing that it is not the car that was just shown.

4. BE: User wants to find dealerships that sell the correctly identified car

   (a) VP: User

      i. Nearby dealerships that sell the identified car will be displayed.
      ii. The list of dealerships will be sortable based on a specified category by the user such as alphabetical order or shortest distance.

   (b) VP: Developer

      i. Google maps shall be used to locate nearby dealerships that sell an identified car.

5. BE: User wants to learn more about the app
    (a) VP: User
        i. The user will be able to access a page containing information about the application.
    (b) VP: Developer
        i. The help page must be a page containing information about the application, incuding how to use it and developer information.
6. BE: User wants to review previously identified cars
    (a) VP: User.
        i. The user will be able to access a list of the five most recently confirmed car identifications.
    (b) VP: Developer
        i. A newly confirmed search will push out the fifth most recent confirmed search.
7. BE: User wants to provide feedback on the app
    (a) VP: User
        i. The user shall be able to provide a short 150 word feedback response to the administrator.
        ii. User will also be able to specify whether it is positive or negative feedback.
    (b) VP: Developer
        i. A list of feedback from users will be displayed in a list format. The entries will be anonymous.

4 Non-Functional Requirements

4.1 Look and Feel Requirements

4.1.1 Appearance Requirements

LF1. The application shall be aesthetically pleasant to look at.
LF2. The application shall have an intuitive and easy to use interface.

4.1.2 Style Requirements

LF3. The application shall be designed with a minimal look so it's more trustable and appealing

4.2 Usability and Humanity Requirements

4.2.1 Ease of Use Requirements

UH1. The application shall be easily usable by children as young as 12.
UH2. The application shall be easy for any user to interact with, regardless of technological experience.
UH3. The application interface shall be intuitive and easy to navigate.
UH4. The application shall make users want to use it.

4.2.2 Personalization and Internationalization Requirements

not applicable as the app's scope is limited to Hamilton

4.2.3 Learning Requirements

UH5. Users shall be able to perform the application's basic operation within a few minutes of first interacting with the application.
UH6. The user of the application does not require any special knowledge prior to use.

4.2.4 Understandability and Politeness Requirements

UH7. The application shall use symbols and words easily understandable by the general public.

4.2.5 Accessibility Requirements

UH8. The application shall have the option to increase the size of the text so it's easily visible.

4.3 Performance Requirements

4.3.1 Speed and Latency Requirements

PR1. The application shall respond to the user's car attribute search query within 5 seconds.

4.3.2 Safety-Critical Requirements

not applicable

4.3.3 Precision or Accuracy Requirements

PR2. The application shall provide accurate information about each car and the closest dealership that the car is located at.
PR3. Google Maps/Places API shall take the precise location of the device accessing the application for use in the application.

4.3.4 Reliability and Availability Requirements

PR4. The application shall be available for use while an Internet connection is present.

4.3.5 Capacity Requirements

not applicable

4.3.6 Scalability or Extensibility Requirements

PR5. The database containing cars and their identifiable attributes pertaining to the experts shall be easily scalable to accommodate a larger amount of cars that can be identified.

4.3.7 Longevity Requirements

not applicable

4.3.8 Robustness or Fault-Tolerance Requirements

PR6. The application shall be able to cope with erroneous input to the attributes field, being able to identify that no such car meets the attributes given.

4.4 Operational and Environmental Requirements

4.4.1 Expected Physical Environment

OE1. The application will operate in any environment with connection to the Internet.
OE2. The application shall be used in the city, where users will spot cars that they are interested in and want to identify.

4.5 Requirements for interfacing with Adjacent Systems

OE3. The application shall interface with Google Maps API.

4.6 Productization Requirements

4.7 Release Requirements

4.8 Maintainability & Support Requirements

4.8.1 Maintenance Requirements

MS1. The application shall be updated on an “as-needed” basis.
MS2. The application shall be easily modifiable in the case that further features need to be implemented.

4.8.2 Supportability Requirements

not applicable

4.8.3 Adaptability Requirements

MS3. The application shall be backwards compatible on any Android OS.

4.9 Security Requirements

4.9.1 Access Requirements

SR1. The application shall only allow authorized users to swap experts. Authorized users are those who enter the correct key phrase when prompted, once they initiate the swapping experts process.

4.9.2 Integrity Requirements

SR2. The application shall encrypt all transmitted messages between the experts and the forum.

4.9.3 Privacy Requirements

SR3. At this moment, the app does not need any personal information of the user to be able to perform a search on the car.
SR4. The app shall clearly indicate to the user that their location needs to be accessed in order to provide accurate nearby dealership information.

4.9.4 Audit Requirements

n/a

4.9.5 Immunity Requirements

n/a

4.10 Cultural and Political Requirements

4.10.1 Cultural

CP1. This application shall not use any text, images, or media that will offend anyone.

4.10.2 Political

CP2. This application shall be politically independent.

4.11 Legal Requirements

4.11.1 Compliance

n/a

4.11.2 Standards

LR1. This application shall comply with any relevant standards associated with Android applications.
LR2. The application shall hold no bias towards certain dealerships but show all dealerships as they are listed from the Google Places API.