Car Maintenance

Software Requirements Specification

CarMaintenanceApp.doc
Draft 1.0

January 28, 2013

Georgia Gwinnett College

Revisions

Version	Primary Author(s)	Description of Version	Date Completed
Version 1.0	Rachael Wright, Jerome Cox, Marcus Michalske, and Blair Sanders	Requirements.	02/04/13

Contents

1 Introduction

This document will cover the required functionality for the Car Maintenance Application, and describe the requested interfaces.

1.1 Purpose

The purpose of this document is to describe the requirements of the Car Maintenance Application described herein. It is intended for the persons writing the technical specification, and the developers in order to describe the requirements of the application.

1.2 Scope

The application will inform users based on their mileage what maintenance needs to be performed. The default setting for the Car Maintenance Application will be based off the *waynesgarage.com* car maintenance schedule. The user will be able to override any of the default settings, if they so choose, based off their vehicle. The application will calculate the MPG and report on necessary maintenance each time the user inputs miles or updating any data to the application. The application will track and keep records of past maintenance per car, will alert user when a deadline approaching or service milage. The application will not be responsible for interfacing with the car's computer, liable for incidental damages to the car due to faulty maintenance instructions, nor irregular maintenance.

1.3 Definitions, Acronyms, and Abbreviations

The common and uncommon uses of definitions, acronyms, and abbreviations that is presented thereforth for the "Car Maintenance Application".

- MPG - Miles per Gallon; The amount of miles that is driven per the US gallon of gasoline.

1.4 References

- "Car Maintenance Schedule." *Wayne's Garage. http://www.waynesgarage.com/ maint.html.*

1.5 Overview

The rest of this document will elaborate on the functionality, the interfaces, and the attributes of the car maintenance application. Once completed, this android app will help the user keep track of when they need to maintain their car.

2 Overall Description

This section will go in depth of the background information needed to run the Car Maintenance Application.

2.1 Product Perspective

The product is independent and self-contained. It will be built using the Android SDK which provides a framework, but all business logic will rely on the system itself.

2.1.1 System Interfaces

The CMA system interfaces that are identified and the related functionality of the product.

- Touch screen: Used to interact with in the application and make selections.
- Touch intellisense keyboard: A virtual keyboard that is used in the absence of a hardwired keyboard.
- Keyboard: A hardwired on board keyboard used for typing in values to the application.
- Internet connection: A connection is need for any updates or information from the a server if such a service is provided.
- Memory storage: In addition to the memory allocated to the running of the program, additional hard drive memory will be needed for storage of persistent data.

2.1.2 User Interfaces

Home Page must contain the ability to choose between recording new maintenance (whether or not the app reported there was maintenance required) and updating the miles driven. It should also have alerts on what maintenance will be required in less than 500 miles.

There should be a separate screen where the user can input maintenance. All maintenance types should be on this screen, and it should be sorted by when maintenance is next due with the soonest being first. Once the user selects a maintenance type, they should be able to input the number of miles they have driven at that time.

If the above screen is not multi-purposed to display and record maintenance, another will be required to show when maintenance is required on the vehicle. The display should include what date maintenance is next required, when it was last performed, the mileage when you performed maintenance, how many miles you’ve driven since performing maintenance, and how many you have until maintenance is required, and at what mile amount maintenance is required. We suggest a slider where the origin is the last maintained date/miles, the , the end is the date/

mileage maintenance is required.

The Miles-Per-Gallon Calculator should be the method used for inputting miles. It should have a place to input miles, and gallons. There should also be instructions on how to get the MPG of your car, ex. Completely fill the gas tank. It should display the miles driven, the number of gallons used, and the calculation. If possible, the user should be able to update the miles driven.

3.1.1 Hardware Interfaces

- Android QWERTY Keyboard
- Mobile wireless adaptor
- Android touch screen
- Physical memory storage of the mobile device

The app will run on Android Devices.

3.1.2 Software Interfaces

3.1.3 Communications Interfaces

The application must be able to connect to a 3G or 4G network

3.1.4 Memory Constraints

Users need to have at least 100 MB of free space on their phone or SD card

3.1.5 Operations

Specify any normal and special operations required by the user, including:

- *periods of interactive operations and periods of unattended operations*
- *data processing support functions*
- *backup and recovery operations*
- *etc.*

3.1.6 Site Adaptation Requirements

Define requirements for any data or initialization sequences that are specific to a given site, mission, or operational mode. Specify features that should be modified to adapt the software to a particular installation.

3.2 Product Functions

The software will calculate the MPG. It will store the last time a user performed maintenance on the car, and it will alert the user when it is time to maintain the car, based on the miles driven, and the date.

3.3 User Characteristics

The target user has a smartphone and is assumed to have basic familiarity with the mobile device, but is not expected to know anything about car maintenance. All users will have a driver's license which means they will be over 16 years of age, able to read, and able to see.

3.4 Constraints

<Describe any other items that will constrain the design options, including

- *reliability requirements*
- *criticality of the application*
- *safety and security considerations regulatory policies*
- *hardware limitations*
- *interfaces to other applications*
- *parallel operation*
- *audit functions*
- *control functions*
- *higher-order language requirements*
- *signal handshake protocols*
- *etc.>*

3.5 Assumptions and Dependencies

List factors that affect the requirements. These factors are not design constraints, but areas where future changes might drive change in the requirements.

3.6 Apportioning of Requirements

Method of calculating miles per gallon?

4 Specific Requirements

<This section should describe all software requirements at a sufficient level of detail for designers to design a system satisfying the requirements and testers to verity that the system satisfies requirements.
The remainder of this sample document is organized according to A.5 Template of SRS Section 3 Organized by Feature shown in the Annex of Std 830-1993. For alternative organizational schemes by system mode, user class, object, stimulus, functional hierarchy, and combinations,

see the standard.>

4.1 External Interface Requirements

<Provide a detailed description of all inputs into and outputs from the software. This section should complement the interface descriptions under section 2.1 and should not repeat information there. Include both content and format as follows:

- *name of item*
- *description of purpose*
- *source of input or destination of output*
- *valid range, accuracy, and/or tolerance*
- *units of measure*
- *timing*
- *relationships to other inputs/outputs*
- *screen formats/organization*
- *window formats/organization*
- *data formats*
- *command formats*
- *end messages*

These requirements may be organized in the following subsections.>

4.1.1 User Interfaces

Several user interfaces will be required, one for inputting maintenance done, one for calculating MPG, and one for time until next maintenance.

4.1.2 Hardware Interfaces

The app will run on Android devices, and should be built to run on the minimum hardware requirements for a phone running Gingerbread.

4.1.3 Software Interfaces

4.1.4 Communications Interfaces

There is no requirement at this time for any communication with any other existing external system.

4.2 Software Product Features

4.2.1 MPG Calculator

Miles-per-Gallon Calculator

Purpose

This calculator rewards the user for frequently putting in their miles, and calculates the number of Miles-per-Gallon based on the Gallons used, and miles driven. It should come with instructions on how to calculate your miles-per-gallon. For example, “Fill the gas tank to full, and divide the amount of gas added by the miles driven.”

Stimulus/Response Sequence

The user inputs their miles, and the system responds with the calculation of the miles-per-gallon used.

Associated Functional Requirements

Miles-Per-Gallon Used

Introduction

The MPG-o-meter calculates the miles-per-gallon, which will help with the maintenance report.

Inputs

The user enters the current total miles as well as the gallons of gas added to the car.

Processing

The application will calculate miles-per-gallon by subtracting the previous miles from the current miles and then dividing the result by the gallons of gas added to the car.

MPG = (current miles - previous miles) / Gallons Added

Outputs

The miles-per-gallon value would be given to the user.

4.2.2 Maintenance Report

Repeat subsections at this level and below for each feature.

Purpose

Based on what the user inputs for their mileage, they will be reminded when they need certain maintenance done to their car based off the miles the put on their car since they last maintained it.

Stimulus/Response Sequence

The user will input the miles they have on their car or an approximation of how many miles it has. Users will also need to input when they last had maintenance done on their car. The system will respond by storing the information and when the mileage or date gets to a point where the car needs maintenance the system will respond by alerting the user they need maintenance.

Associated Functional Requirements

Report

Introduction

Reporting system to help alert users when it’s time to get maintenance done based off the mileage

Inputs

miles used (gathered from MPG calculator)

Processing

Application will use the miles or date to calculate when the next maintenance needs to be done.

Outputs

An interface will be shown to the user including when they need maintenance done on their car.

Change Your Oil

Introduction

This feature will alert the user when the oil needs to be changed. The default parameter is 5000 miles or 6 months, whichever the user arrives at first.

Inputs

The number of miles driven (gathered from the MPG Calculator)

Processing

The application has the number of miles since the last oil change, and the date stored, as well as the number of miles. It subtracts the number of miles of the next oil change from the current miles. It also compares the date, and the date oil change needed.

Outputs

The number of miles/date until the next oil change.

4.3 Performance Requirements

Specify static and dynamic numerical requirements placed on the software or on human interaction with the software.
Static numerical requirements may include the number of terminals to be supported, the number of simultaneous users to be supported, and the amount and type of information to be handled.
Dynamic numerical requirements may include the number of transactions and tasks and the amount of data to be processed within certain time period for both normal and peak workload conditions.

4.4 Design Constraints

Specify requirements imposed by standards, hardware limitations, etc.

4.5 Software System Attributes

The following items provide a partial list of system attributes that can serve as requirements that should be objectively verified.
Other possible options include scalability, portability, robustness, recoverability, etc.

4.5.1 Reliability

During any 14 day period, the Car Maintenance Application shall not be down for longer than 60 minutes in case of software failure. Also, all data entered in for the user's car will be stored for that specific phone in case the phone shuts off.

4.5.2 Availability

The Car Maintenance Application will be accessible through google play and available to all users.

4.5.3 Security

Specify the factors that will protect the software from accidental or malicious access, misuse, or modification. These factors may include:

- *cryptography*
- *activity logging*
- *restrictions on intermodule communications*
- *data integrity checks*

The Car maintenance App will not have any security measures

4.5.4 Maintainability

The car maintenance application will be designed and developed for long term maintainability

4.6 Logical Database Requirements

The application will store the data when the user maintains the car, and when the car should regularly receive maintenance. It should also store the previous number of miles.
The application will store default dates and miles for each type of maintenance referenced in the appendix. It will also hold data of when the car was last maintained, and the mileage when it was maintained. It will also store a proposed date when the car next requires maintenance. It will also include one data point for the car's previous value for miles driven.

When the car should regularly receive maintenance will be rarely updated, possibly never. It should contain heavy validation, because the user will be entering the values.

When the car will need maintenance, as well as when it last received maintenance, will be queried on a regular basis. These queries will also be updated by the application on a monthly basis.

4.7 Other Requirements

Appendix A: Car Maintenance Default Schedule

SERVICE	DUE BY MILEAGE	DUE BY TIME
Tune up - Major	30,000 - 60,000	
Lube Oil and Filter Service	3,000	4 month
Brake Inspection	15,000 - 20,000	1 yr
Automatic Transmission service	30,000	2 yr
Manual Transmission service	50,000	3 yr
Coolant change	50,000	2 yr
Wheel Bearing pack	50,000	
Valve Adjust (if applicable)	30,000	
Timing Belt (if applicable)	See Owner's Manual	8 yr
Cooling System Hoses	100,000	4-5 yr
Accessory Belts	40,000 - 60,000	4 yr
Idle air control service or throttle cleaning	30,000	2 yr
Brake system Flush non Anti-lock Brakes		3 yr
Brake system Flush with Anti-lock Brakes		2 yr
Power steering fluid flush	60,000	
Differential/Transfer case Service	50,000	3 yr
Engine Thermostat	30,000	3 yr
Tire Rotation	5,000 - 10,000	8 months