**1 Introduction**

**1.1 Purpose**

The purpose of this document is to present a detailed description of the CSE 360 Auto project.It will explain the system and what it is intended to do. It will also explain the interface and features of the system. This document is for the user as well as the developers

**1.2 Scope**

The All-in-One Car Software System is a simple user interface using various inputs to simulate a car driving experience. Using several user interface elements to change conditions in the vehicle, the software will simulate speed and acceleration and generate live statistics of the current user's drive.

As part of the All-in-One system, the software is also able to simulate radio, phone and mapping functionality. For the radio, the user has the ability to change stations, adjust volume and select saved stations. The phone function allows the user to make calls and change speaker and microphone volumes. Finally, the map function allows for tracking of the start, end and current locations of several pre-defined routes.

Since the software is merely a simulation and will not put into a physical car, all normal car inputs (i.e. pedal, brake) will be mock interface elements on the screen using keyboard and mouse as input.

The software will be written in JAVA, and as such will be platform independent.

**1.3 Definitions, acronyms and abbreviations**

The following list summarizes the definitions used in this document.

**User ID:** Refers to a four digit PIN used to login a user.

**Profile:** Refers to the statistics data generated by a user during a driving session.

**Route:** A pre-defined line upon which the car will drive with a fixed start and end distance.

**User:** The current driver logged with a valid User ID.

**1.4 References**

* IEEE Recommended Practice for Software Requirements Specifications, IEEE Std 830-1998 (1998)

**1.5 Overview**

The next part of this document is the overall description of the product going into detail about the specific requirements for each part of the software’s implementation and user interaction.

**2 Overall description**

**2.1 Product perspective**

This will include all of the requirements for the whole project rather than just the first increments requirements. This is a standalone software and is not designed to be integrated with an actual car. This program is just a representation of car functions and is not intended to supplement a car's functions.

**2.1.1 Inputs**

The Auto software will have many inputs for the user, beginning with the most important the startup input. The user will also be able to choose between inputting the speed and acceleration. The user will also have interaction with the radio phone and map.

* Startup input

1.User pin input Must Have

1.1 Must be 4 digits Should Have

1.2 Must be a valid pin and match a user profile Should Have

2. On button - Button on gui that must be pressed to turn on the car Must Have

2.1 The car must have fuel to start Should Have

2.2 The car must be off to start Should Have

3. Off button - Button on the gui that must be pressed to turn off the car Must Have

3.1 The car must be stopped to be turned off Should Have

3.2 The car must be turned on to turn off Should Have

1. Speed input

4. Acceleration - Input must be in miles per hour per hour Must Have

4.1 Must be a valid acceleration not exceeding maximum Should Have

4.2 Acceleration will be set to zero if max speed is reached Should Have

5. Speed - Input must me in miles per hour Must Have

5.1 The speed must not exceed the maximum speed Should Have

6. Deceleration - Input must be in miles per hour per hour Must Have

6.1 Must be a valid deceleration (a negative value) Should Have

6.2 Program cannot reach a negative speed Should Have

* Radio input

7. Radio - The car radio must be interactive Must Have

7.1 The volume will be adjusted using the up and down buttons Must Have

7.2 The stations can be changed using the up and down buttons Must Have

7.3 AM and FM can be selected Must Have

7.4 The maximum volume of 100 cannot be exceeded Should Have

7.5 The minimum volume of 0 cannot be exceeded Should Have

* Phone input

8. Phone - The phone should also be interactive Must Have

8.1 The volume of the speaker will be adjusted using the up and down buttons Must Have

8.2 The volume of the microphone will be adjusted using the up and down buttons Must Have

8.3 A list of phone numbers that the user can select to call Must Have

8.4 A keypad where the user can enter numbers to call manually Must Have

8.5 A call button Must Have

8.6 An end call button Must Have

8.7 The volumes cannot exceed a maximum of 100 Should Have

8.8 The volumes also cannot exceed a minimum of 0 Should Have

8.9 The phone number entered must be a valid phone number Must Have

**2.1.2 Outputs**

The software will have the following output fields:

1. Login

1.1 A display for the input

1.2 Pass/Fail message for login attempt

1. Persistent Banner

2.1 Current fuel level

2.2 Current fuel consumption

2.3 Current state (park/drive)

2.4 Current speed

2.5 Odometer

2.6 Trip distance

2.7 Session time

1. Radio

3.1 Current radio band

3.2 Current station title

3.3 List of available radio stations

3.4 Volume slider for radio volume

1. Phone

4.1 Contact list

4.2 Current number being dialed

4.3 Key pad

4.4 Volume sliders for both the speaker and the microphone

5. Map

5.1 Route on map

5.2 Current position on map

6. Stored session info

6.1 Average speed

6.2 Average acceleration

6.3 Maximum speed reached

6.4 Phone calls made

6.5 Phone call duration

6.6 Time spent listening to radio

6.7 Trip duration

6.8 Trip length

**2.2 Product functions**

These are the requirements for the functions of the software and what they intend to accomplish. Most of these are not yet implemented due to the nature of our process model. The requirements are also designated as “Must have”, “Should have” and “Nice to have” to demonstrate their importance in implementation.

**2.2.1 Functional Requirements**

1. The software must be able to recognize users by a unique pin Must Have

1.1 Software must allow pins to be 4 digits long Must Have

1.2 Software must not allow users to have the same pins Must Have

1.3 Software must allow pins and user info to be stored in a file or files Must Have

2. Software must have the ability to let users create their own unique data Must Have

2.1 Software must record driver name Must Have

2.2 Software must record average speed Must Have

2.3 Software must record average acceleration Must Have

2.4 Software must record max speed reached during session Must Have

2.5 Software must record fuel level Must Have

2.6 Software must record the date of trips Must Have

2.7 Software must record trip duration Must Have

2.8 Software must record phone calls Must Have

2.9 Software must record phone call duration Must Have

2.10 Software must record the time period where the radio was used Must Have

2.11 Software must record the time period where the phone was used Must Have

2.12 Software must record a complete list of drivers Must Have

3. Software must have the ability to calculate and/or display Must Have

3.1 The current trip distance Must Have

3.2 The current speed Must Have

3.3 The current acceleration/deceleration Must Have

3.4 Current fuel level Must Have

3.5 Current fuel consumption based upon acceleration Must Have

3.6 Current radio station and volume Must Have

3.7 Current microphone and speaker volume for phone Must Have

3.8 Current radio band Must Have

3.9 A phone contact list displaying numbers Must Have

3.10 A radio station list Must Have

3.11 Current location on a map Nice to Have

3.12 Whether the car is in motion or not on the map Nice to Have

4. Software must have the ability to accelerate and decelerate based on user input Must Have

4.1 The software should be able to handle two different kinds of user input Should Have

4.1.1 The software should be able to accelerate at a rate specified by user Should Have

4.1.2 The software should be able to accelerate up to a speed set by user Should Have

4.2 The software must be able to decelerate at a constant rate when the user wants to Must Have

5. The software must be able to make phone calls Must Have

5.1 The software should be able to call numbers from a contact list Should Have

5.2 The software should be able to call numbers entered by the user Should Have

6. The software must have radio functionality Must Have

6.1 The software should be able to seek through different radio stations Should Have

6.1.1 The software should be able to only play certain stations in certain areas

7. Software should be able to reproduce the stored user information

**2.2.2 Non-Functional Requirements**

1. Software must be able to run across multiple platforms Must Have
2. Software should have a useful good looking UI Should Have
3. Software must have a “User guide” Must Have
4. Software should handle exceptions and errors Nice to Have

4.1 Software should handle errors and exceptions with useful messages Nice to Have

5. Software should be reliable Nice to Have

**2.3 User characteristics**

The users are expected to have basic knowledge of the software and how it works.

**2.4 Constraints**

The system is designed to provide the users with a nice GUI and it should be runnable on the average PC. The software should not just run on one operating system but it may not be runnable on all conceivable operating systems.

**2.5 Assumptions and dependencies**

There are no assumptions or dependencies for this product

**2.6 Apportioning of requirements**

Refer to section 2.2

**3 Specific requirements**

This is covered elsewhere in the report and will not be repeated here.