Software Requirements Specification (SRS) for SmartCar Project

*Baseline version 0.2*

*Issued on November 7, 2014*

Issued by :

Roberto

Ryan

Ryant

Steven

Issued for : Sutrisno

# 

# Change History

|  |  |  |  |
| --- | --- | --- | --- |
| **Version** | **Date** | **Author** | **Changes** |
| 0.1 | October 27, 2014 | Group | Initial Version |
| 0.2 | November 7, 2014 | Group | SRS |

# Document Approval

The following Software Requirements Specification has been accepted and approved by the following :

|  |  |  |  |
| --- | --- | --- | --- |
| **Name** | **Title** | **Date** | **Signature** |
| Ryant | Project Manager |  |  |
| Sutrisno | Lecturer |  |  |
|  |  |  |  |

# Table of Contents

Contents

[Change History i](#_Toc403665221)

[Document Approval i](#_Toc403665222)

[Table of Contents ii](#_Toc403665223)

[List of Figures iv](#_Toc403665224)

[Chapter 1 1](#_Toc403665225)

[INRODUCTION 1](#_Toc403665226)

[1.1 Purpose 1](#_Toc403665227)

[1.2 Scope 1](#_Toc403665228)

[1.3 Definitions, Acronyms, and Abbreviations 1](#_Toc403665229)

[1.4 References 1](#_Toc403665230)

[1.5 Overview 2](#_Toc403665231)

[Chapter 2 3](#_Toc403665232)

[GENERAL DESCRIPTION 3](#_Toc403665233)

[2.1 Product Perspective 3](#_Toc403665234)

[2.2 Product Functions 3](#_Toc403665235)

[2.3 User Characteristics 3](#_Toc403665236)

[2.4 General Constraints 3](#_Toc403665237)

[ Smart car simulation program will start automatically after the user execute the smart car simulation program. 3](#_Toc403665238)

[2.5 Assumptions and Dependencies 3](#_Toc403665239)

[Chapter 3 4](#_Toc403665240)

[SPECIFIC REQUIREMENTS 4](#_Toc403665241)

[3.1 External Interface Requirements 4](#_Toc403665242)

[3.1.1 User interface 4](#_Toc403665243)

[3.1.2. Hardware interfaces 4](#_Toc403665244)

[3.1.3. Software interfaces 4](#_Toc403665245)

[3.2 Functional Requirements 4](#_Toc403665246)

[3.3 Use Cases 5](#_Toc403665247)

[3.4 Classes/Objects 6](#_Toc403665248)

[3.5 Non-Functional Requirements 6](#_Toc403665249)

[3.6 Design Constraints 7](#_Toc403665250)

[Chapter 4 8](#_Toc403665251)

[ANALYSIS MODELS 8](#_Toc403665252)

[4.1 Sequence Diagrams 8](#_Toc403665253)

[4.1.1 Run Program 8](#_Toc403665254)

[4.1.2 User Feedback 8](#_Toc403665255)

[4.1.3 Maintain Program 9](#_Toc403665256)

[4.1.4 Client Decision 9](#_Toc403665257)

[4.2 State Diagrams 10](#_Toc403665258)

[Chapter 5 11](#_Toc403665259)

[CHANGE MANAGEMENT PROCESS 11](#_Toc403665260)

[5.1 Process 11](#_Toc403665261)

[5.2 Actor 11](#_Toc403665262)

[5.3 Approval 11](#_Toc403665263)

[Appendices 12](#_Toc403665264)

[Appendix 1 12](#_Toc403665265)

[Appendix 2 12](#_Toc403665266)

[Appendix 3 12](#_Toc403665267)

# List of Figures

**Use Case.**….……………………..........................……………………………............................5

**Class Diagram**..…………………………………………………………………………………..6

**Sequence Diagram 1**…..……………………..........................……………………………........8

**Sequence Diagram 2**…..………………………………………………………………………....8

**Sequence Diagram 3**…...……………………..........................…………………………….......9

**Sequence Diagram 4**...…………………………………………………………………………...9

**State Diagram**...…………………………………………………………………………………10

**Process Diagram**…...……………………………………………………………………………11

# Chapter 1

# INRODUCTION

## 1.1 Purpose

The purpose of this project is to simulate how a Smart Car works in a real life situation for example how the car response to an obstacle, how was the tire condition, how the car sensor worked, etc.

.

## 1.2 Scope

The scope in this project are how the car react to something in front of the car with the sensor that are available in the car or to know what are the condition on the car surroundings.

## 1.3 Definitions, Acronyms, and Abbreviations

* SRS: Software Requirement Specifications
* GUI: Graphical User Interface.
* JRE: Java Runtime Environment.
* Use Case: Methodology used in system analysis to identify, clarify, and organize system requirements.
* Class Diagram: An illustration of the relationships and source code dependencies among classes in the Unified Modeling Language (UML).
* Sequence Diagram: Show the interactions between objects in the sequential order that those interactions occur.
* State Diagram: Diagram used in computer science to describe the behavior of a system considering all the possible states of an object when an event occurs

# 1.4 References

* Meyer, Bertrand. Object-Oriented Software Construction 2nd Edition. California : ISE. Inc, 1997.
* Farrel, Joyce. Java Programmming 6th Edition. Boston : Cengage Learning, 2012.

# 1.5 Overview

The first section of the SRS introducing what are our project that is a SmartCar application, Second section of the SRS discussing about our project description, Third section of the SRS discussing specific requirement about our project. Fourth section of the SRS discussing about our Analysis model and the last section discussing about change management process.

# Chapter 2

# GENERAL DESCRIPTION

## 2.1 Product Perspective

The Client and users will be given and the client and user can access the program and can see the smart car simulation.

## 2.2 Product Functions

The smart car simulation program will run The program will run a smart car simulation with some details like speed, temperature, light intensity, tire pressure, day and light status.

## 2.3 User Characteristics

General Users / Guest: They will be in a position to permit access to the users in the Internet and acknowledge their account status.

Administrators: They are the core users and are able to add new users to the system and permit them to access the Internet resources. They can also view in real time what a user is performing right now. They can also get the overall report of the user sessions.

## 2.4 General Constraints

* Smart car simulation program will start automatically after the user execute the smart car simulation program.
* Only Administrator and group members can control and change the smart car simulation program.

## 2.5 Assumptions and Dependencies

* GUI is not the top priority
* Using International System of Units
* Car will move based on accelerated linear motion rule.
* As the speed goes up, then the temperature goes up
* If the car run too long and the tire hold the car weight, then the pressure on the tire will decrease.
* Car will slow down if the tire pressure less than 10% of its initial position
* Car will turn, if the car close with a obstacle or on certain speed
* Track always straight
* Lights will on at night.
* If there are many obstacle, obstacle will be filled at the same row
* Car will stop if the tire pressure is empty
* Car light intensity is 100 candela

# Chapter 3

# SPECIFIC REQUIREMENTS

## 3.1 External Interface Requirements

## 3.1.1 User interface

Simulator Screen: The screen show how the simulation works (car, track and obstacle).

Status Screen: The screen show the status of the and the track (speed, auto lock speed, temperature, light intensity, tire pressure, day and light status).

## 3.1.2. Hardware interfaces

The user will need a laptop / netbook / desktop computer that runs windows operating systems and have installed JRE in their laptop / netbook /desktop.

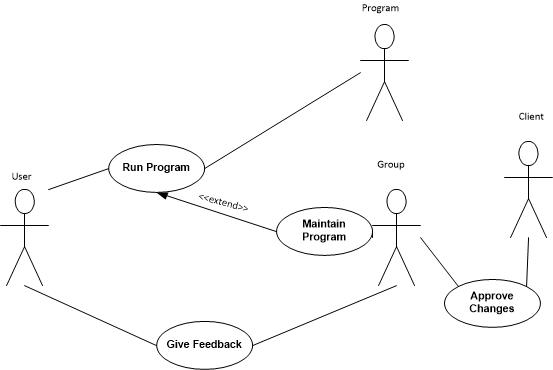
## 3.1.3. Software interfaces

The user must have installed the smart car simulation software.

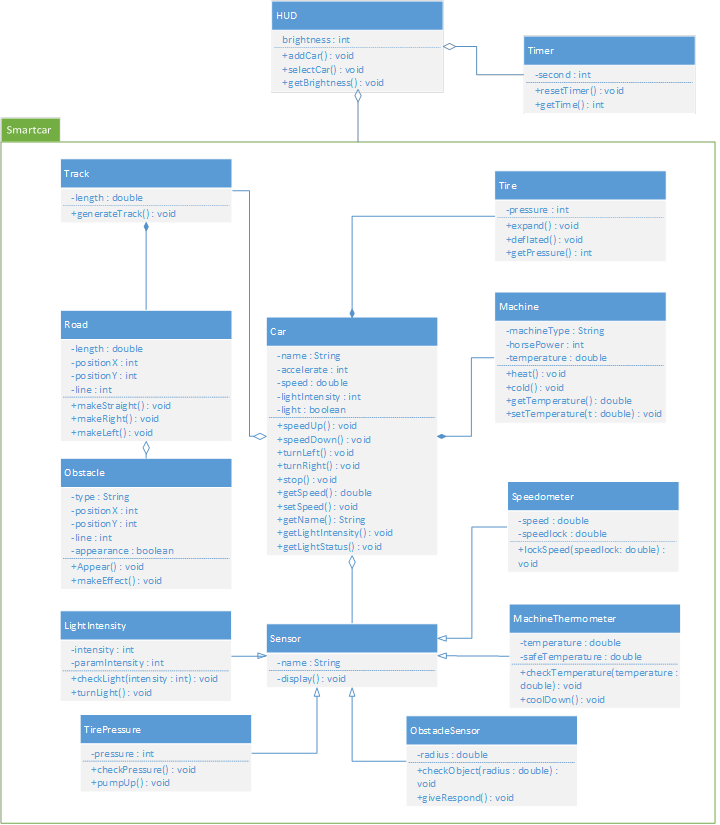
## 3.2 Functional Requirements

* The smart car simulation program runs without error.
* The smart car simulation program do simulate responsiveness of its sensor.

## 3.3 Use Cases

**

## 3.4 Classes/Objects

**

## 3.5 Non-Functional Requirements

* Using International System of Units
* Car will move based on accelerated linear motion rule.
* Car light intensity in candela.

## 3.6 Design Constraints

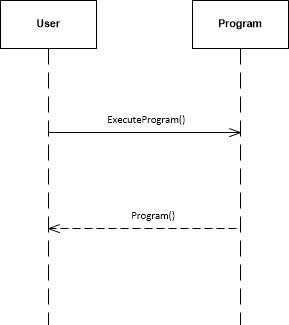
* 2D Graphics.
* OOP Methodology
* Java Programming Language
* Using Graphical User Interface

# Chapter 4

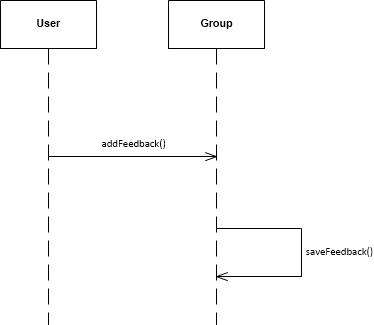
# ANALYSIS MODELS

## 4.1 Sequence Diagrams

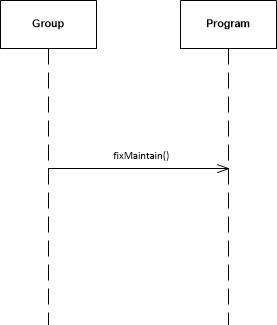
## 4.1.1 Run Program



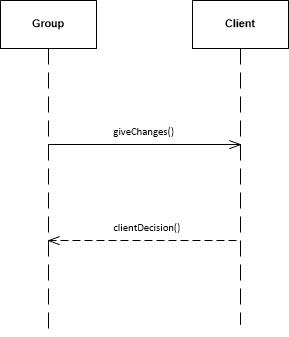
## 4.1.2 User Feedback



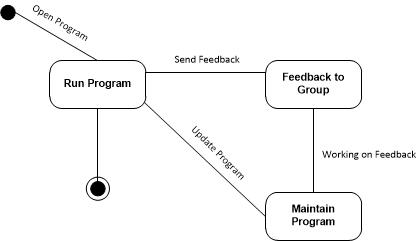
## 4.1.3 Maintain Program



## 4.1.4 Client Decision



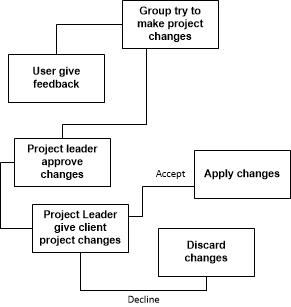
## 4.2 State Diagrams



# Chapter 5

# CHANGE MANAGEMENT PROCESS

## 5.1 Process



## 5.2 Actor

The one who responsible to change the SmartCar program are only group member.

## 5.3 Approval

Project changes will be approve by project leader first if the changes are error-free and sufficient, then the project leader will give the changes to the client. If the client approve than the project can change. If the client approve than the changes will be discarded

# Appendices

## Appendix 1

Meyer, Bertrand. Object-Oriented Software Construction 2nd Edition. California : ISE. Inc, 1997.

## Appendix 2

Farrel, Joyce. Java Programmming 6th Edition. Boston : Cengage Learning, 2012.

## Appendix 3

Hurby, P. (2013). Download Visio Stencil and Template for UML 2.5. Retrieved November 13, 2014, from http://www.softwarestencils.com/uml/