Fujitsu

Autonomous Driving Algorithm

CS 4961

Software Requirements Document(SRD)

Ben Galletta

Evan Chiang

Ariel Suarez

Simon Lam

Roberto Rosas

December 1, 2016

Los Angeles, California

Fujitsu

Autonomous Driving Algorithm

Prepared by:

Ben Galletta

Evan Chiang

Ariel Suarez

Simon Lam

Roberto Rosas

Approved By:

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_

Zilong Ye Date

Document Change Log

|  |  |  |
| --- | --- | --- |
| **Update** | **Date Released** | **Changes** |
| Draft | 12/01/2016 | Delivery of the initial Software Requirements Document |
| Delivery | XX/XX/2017 | Final Document Delivery |

# List of To Be Determined (TBD) Items

|  |  |  |  |
| --- | --- | --- | --- |
| Page | **Item** | **Description** | **Status** |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |

Table of Contents

1.0 INTRODUCTION

1.1 Purpose 7

1.2 Scope 7

1.2.1 Document Organization 7

1.2.2 Relationship to Other Documents 8

1.3 Autonomous Driving Algorithm Architecture 8

1.3.1 Context Diagram (Level 0 DFD) 8

1.3.2 Description of Autonomous Driving Algorithm Major Units 8

1.4 Requirement Development Process 9

1.5 References 9

1.5.1 Controlling Documents 9

1.5.2 Applicable Documents 9

1.5.3 Standards 10

1.5.4 Domain Analysis 10

2.0 DETAILED FUNCTIONAL DESCRIPTION OF THE Autonomous Driving Algorithm

2.1 Detailed Functional Description 10

2.1.1 Higher Level DFDs 10

2.1.2 Detailed Functional Description of Major Sub-Units 12

3.0 Autonomous Driving Algorithm REQUIREMENTS

3.1 Autonomous Driving Algorithm FUNCTIONAL REQUIREMENTS 12

3.2 Autonomous Driving Algorithm NON-FUNCTIONAL REQUIREMENTS 15

4.0 Autonomous Driving Algorithm IMPLEMENTATION AND TEST PLAN

4.1 Implementation Notes 15

4.2 Transfer of Knowledge 15

4.3 Test Plan 16

A. ACRONYMS 18

List of Figures

Level 0 DFD 8

Level 1 DFD 11

1. INTRODUCTION

1.1 Purpose

This document aims to provide is four-fold:

1. Completely defines a full set of requirements for the ADA – This is Section 3.0 of the document and it corresponds to a Software Requirements Document (SRD).
2. Completely define the design for the ADA – This will be Section 4.0 of the document which will be fully implemented. This section corresponds to the elements of a Software Design Document (SDD).
3. Define and implement modules for the ADA – This will be Section 5.0 of the document. – This section corresponds to the Software Implementation Document (SID).
4. Completely defines the Test Plan for the ADA– Section 6.0. This section corresponds to a Software Test Plan (STP). The main structure takes the requirements listed in Section 3.0 and adds a new column including the type of test to be included for each of the requirements.

The complete definition of all requirements provides the source requirement inputs for the development of the subsequent supporting software subsystems documents.

1.2 Scope

This documentation was developed as part of the Senior Software Design class CS4961, it includes the Autonomous Driving Algorithm SRD and parts of the SDD and parts of the STP (Sections 1, 2 and 3, and 4).

This document encompasses includes the following:

* All functional and non-functional requirements on the ADA are captured. This includes all the software and hardware requirements necessary to build our algorithm, as well as inner subsystems requirements.
* A complete set of requirements for the ADA in accordance to the CS4962 class. As well as a detailed organization of the requirements represented in the level 1 DFD.
* The functional requirements defined in the ADA Requirements section have been expanded to include the non-functional requirement. The non-functional requirements listed after the software and hardware requirements with no further analysis.

1.2.1 Document Organization

The organization of this document provides a natural 'flow' or allocation of requirements to each succeeding section.

Details regarding the overall document are given in sub-section 1.5 below.

1.2.2 Relationship to Other Documents

The ADA SRD is a complete self-contained document.

1.3 Autonomous Driving Algorithm Architecture

1.3.1 Detailed Context Diagram (DFD Level 0)

The ADA architecture is presented in the Context Diagram (DFD Level 0) given below. A more complete Functional Description is given in Section 2 of this document.

1.3.2 Description and major functions of the Autonomous Driving Algorithm

The algorithm we seek to produce will allow a car with self-driving capabilities to assess situations with no clear lane marks. This includes yellow or white lines in pavement. The project is separated into parts, or modules, to allow us to separate work into specific tasks.

The algorithm will have two components: A Processing Module (PM) and a Response Module (RM). The PM will be responsible for reading and parsing all incoming data from the input sources and send it to the PM. The PM will be responsible for using the necessary data to create responses and thus making driving instructions.

Besides the algorithm, the project will use two open source programs to aid in the design of the algorithm. An open source simulator will be used to test the algorithm; this will be further discussed in the Testing Plan section of the document. Also, a mobile application will be deployed with a working prototype, the dosing for this prototype will be further explained in section 3.1.

1.4 Documentation Development Process

The ADA detailed functional description is documented in Section 2.0. A detailed summary of the functions will be addressed along the higher level DFDs in future sections of the document.

Requirements for the ADA are captured in Section 3.0 of this document. These requirements are described in detail to provide a complete Software Requirement Document(SRD).

Section 4 is the ADA Implementation and Test Plan (TP). This section will discuss the implementation methods and strategies for the project, as well as any ways in which the system will be tested during the design process.

The final section is a list of all acronyms and abbreviations used for complete understanding of this document.

1.5 References

Sample SRDs provided by the professor were used to make this document.

1.5.1 Controlling Documents

There is no document controlling this document.

1.5.2 Applicable Documents

There are no applicable documents

1.5.3 Standards

No Standard have been used in the creation of this document.

1.5.4 Domain Analysis

No previous scholar work has been done in this specific area of our project. We hope to achieve enough results to provide some foundation for future similar projects. Some of these foundations may include:

* A working prototype for future use
* A working algorithm for reference and testing

2.0 DETAILED FUNCTIONAL DESCRIPTION OF THE Autonomous Driving Algorithm

2.1 Detailed Autonomous Driving Algorithm Functional Description.

The major tool used to design the ADA is the Data Flow Diagram, DFD. The rationale behind the selection of DFDs as the preferred design tool was their simplicity and versatility. In the future, more sophisticated tools may be used particularly if a correlation from Design to Requirement to Implementation and Testing is found to be a necessary addition.

2.1.1 Higher Level Data Flow Diagrams.

The ADA major functional design components are shown in the DFDs below.

Autonomous Driving Algorithm Level 1 DFD

2.1.2 Detailed Description of the Autonomous Driving Algorithm Major Sub-Units

The Autonomous Driving Algorithm major functional subunits shown in the DFDs in the previous sub-sections, are described in detail below.

2.1 – Sensor Module:

The Sensor Module (SM) gathers data from the surroundings of the prototype via infrared and ultrasonic sensors mounted on the car. The data received is sent to the CM to be processed accordingly.

2.2 – Control Module:

The Control Module (CM) receives the data from the sensor and eliminates unnecessary data. It then sends it to the AM and receives back driving instructions from the AM. The CM sends the driving instructions to the MM to control the prototype accordingly.

2.3 –Application Module:

The Application Module (AM) is an Android Application that runs on a mounted Android phone on top of the prototype. The AM has the algorithm saved and runs it while the prototype runs. It receives the data from the CM and runs it through the PM. It then receives back driving instructions and send them to the CM via Bluetooth connection.

2.4 – Process Module:

The Process Module (PM) is one of the two modules of the algorithm. It is responsible for reading and parsing the data necessary to make proper driving decisions. The PM sends the data to the RM.

2.5 - Response Module:

The Response Module (RM) is the second module of the algorithm. It will receive the data from the PM and make calculations on how the prototype should proceed.

2.6 - Motor Module:

The Motor Module (MM) is attached to the prototype and controls the steering, acceleration and similar functions of the prototype. The MM receives driving instructions from the CM and executes them in real-time.

2.7 - Simulator Module:

The Simulator Module (SIM) for simulating driving scenarios for testing purposes. The SIM runs the algorithm separately from the prototype and similarly it interacts with the PM and RM.

3.0 Autonomous Driving Algorithm REQUIREMENTS

3.1 Autonomous Driving Algorithm Functional Requirements

This section collects all the ADA Functional Requirements. This section includes the complete set of functional requirements with explanation and rational where the statement of the requirement was deemed insufficient or needing additional background/justification. All requirements relate to the design modules described in Section 2. An effort has been made to standardize the correlation between the design modules and the requirements to make their access and organization more consistent. For example, module 2.1 requirements are labeled 3.1, sub-module 2.1.1 requirements are labeled 3.1.1 and so on. The list of requirements follows.

|  |  |
| --- | --- |
| 3.1 | SM Requirements |
| Requirement No. | Requirement Description |
| 3.1.1 | SM shall run inside the prototype. |
| 3.1.2 | SM shall gather data of nearby obstacles. |
| 3.1.3 | SM shall retrieve a data image from the surroundings of the prototype. |
| 3.1.4 | SM shall send the data gathered to the CM. |

|  |  |
| --- | --- |
| 3.2 | CM Requirements |
| Requirement No. | Requirement Description |
| 3.2.1 | CM shall run inside the prototype. |
| 3.2.2 | CM shall receive data from the SM. |
| 3.2.3 | CM shall send the data to the AM. |
| 3.2.4 | CM shall receive driving instructions from the AM. |
| 3.2.5 | CM shall send driving instructions to the MM. |

|  |  |
| --- | --- |
| 3.3 | AM Requirements |
| Requirement No. | Requirement Description |
| 3.3.1 | AM shall run on an Android mobile phone. |
| 3.3.2 | AM shall run the PM. |
| 3.3.3 | AM shall run the RM. |
| 3.3.4 | AM shall receive data from the CM. |
| 3.3.5 | AM shall send driving instructions to the CM. |

|  |  |
| --- | --- |
| 3.4 | PM Requirements |
| Requirement No. | Requirement Description |
| 3.4.1 | PM shall read the data from AM. |
| 3.4.2 | PM shall parse the data. |
| 3.4.3 | PM shall filter unnecessary data. |
| 3.4.4 | PM shall send the data to the RM. |

|  |  |
| --- | --- |
| 3.5 | RM Requirements |
| Requirement No. | Requirement Description |
| 3.5.1 | RM shall receive data from the PM. |
| 3.5.2 | RM shall use the data in proper calculation. |
| 3.5.3 | RM shall create proper driving instructions |
| 3.5.4 | RM shall send driving instructions to the AM. |

|  |  |
| --- | --- |
| 3.6 | MM Requirements |
| Requirement No. | Requirement Description |
| 3.6.1 | MM shall receive driving instructions from the CM. |
| 3.6.2 | MM shall execute driving instructions in the prototype. |

|  |  |
| --- | --- |
| 3.7 | SIM Requirements |
| Requirement No. | Requirement Description |
| 3.7.1 | SIM shall simulate driving situations. |
| 3.7.2 | SIM shall run the PM. |
| 3.7.3 | SIM shall run the RM. |
| 3.7.4 | SIM shall send driving data to the PM. |
| 3.7.5 | SIM shall receive driving instructions from the RM. |
| 3.7.6 | SIM shall execute the driving instructions. |

3.2 Autonomous Driving Algorithm Non-Functional Requirements

This section contains all the STS Non-Functional Requirements. All non-functional requirements are numbered “NF-n” where “n” indicates the nth requirement.

NF – 1 The ADA requires sensors and control hardware.

NF – 2 The ADA requires a car prototype.

NF – 3 The ADA requires functioning Bluetooth connection.

4.0 Autonomous Driving Algorithm Computer System Implementation and Test Plan

4.1 Autonomous Driving Algorithm Implementation Notes

The ADA project requires simulator testing before the prototype is used. This is due to the fact that the prototype cannot be damaged during runs. Also, the ADA requires the purchase of specific hardware, such as sensors and controllers. The hardware must be built into the prototype.

4.2 Autonomous Driving Algorithm Transfer of Knowledge

All source code for the ADA will be stored into GitHub for future reference or assessment. The documentation for the project will be available in the csns school webpage.

4.3 Autonomous Driving Algorithm Test Plan

4.3.1: Sensor Module (SM):

This module will be tested on practice runs alongside the prototype. The module must be built into the prototype in order for it to work.

4.3.2: Control Module (CM):

This module will be tested for its Bluetooth capabilities separately with the AM. Then, it will be tested on practice runs alongside the prototype. The module must be built into the prototype in order for it to work.

4.3.3: Application Module (AM):

This module will be tested for its Bluetooth capabilities separately with the CM. It will later be tested on practice run alongside the prototype to tests its reading and sending of data with other modules.

4.3.4: Process Module (PM):

The PM, alongside the RM, will be tested initially on the SIM. If the first initial tests are positive, it will then be tested with the prototype.

4.3.5 Response Module (RM)

The RM, alongside the PM, will be tested initially on the SIM. If the first initial tests are positive, it will then be tested with the prototype.

4.3.6 Motor Module (MM)

This module will be tested on practice runs alongside the prototype. The module must be built into the prototype for it to work.

4.3.7 Simulator Module (SIM)

The SIM will be tested separately from the other modules. It will be run on several computers to test its accuracy.

1. ACRONYMS

ADA – Autonomous Driving Algorithm

AM – Application Module

CM – Control Module

MM – Motor Module

PM – Process Module

SM – Sensor Module

SIM – Simulator Module

RM – Response Module