Requirements Specification

Multiple Autonomous Vehicles in Complex Scenarios

December 13, 2021

Version 1.3

Project Identity

Group E-mail: teamschannel
Homepage: https://tsrt10.gitlab-pages.liu.se/2021/auto_sys/

Orderer: Erik Frisk
Phone: +46 (0)13 - 28 5714
E-mail: erik.frisk@liu.se

Supervisor: Theodor Westny
Phone: +46 (0)13-28 15 81
E-mail: theodor.westny@liu.se

Course Responsible: Daniel Axehill
Phone: +46 (0)13-28 40 42
E-mail: daniel.axehill@liu.se

Project members

Name	LiU-id	Responsibility
Robert Sehlstedt	robse205	Project leader
Thomas Andersson	thoan459	Design, Scrum-master
Johan Forsman	johfo207	Documents
Philip Sjövik	phisj589	Visualisation
Erik Wahledow	eriwa458	Perception
Ludvig Hansson Granström	ludha385	Software
David Grahn	davgr413	Testing
Mohamed Faraj	mohfa185	Planning & Control

CONTENTS

WORDLIST

- Cascar = A repurposed miniature RC-car.
- Qualisys = Motion capture and 3D positioning tracking system.
- ROS = Robot operating system, provides tools and libraries to develop the autonomous vehicles.
- Rviz = A visualisation and simulation tool for ROS.
- Visionen = A research arena located in LiUs facilities to test-drive the vehicles. This room uses Qualisys for position tracking.

1 INTRODUCTION

The research on autonomous vehicles has become a popular topic over recent years, engaging academia and industry alike. Although the development has matured to a degree such that several companies already offer fully autonomous taxi services and freight transport solutions, there still exist numerous challenges in the field. In particular, the interaction with other traffic agents is an especially challenging aspect, due to the stochastic behaviour of human drivers. Modular solutions to this challenge will be explored in this project.

1.1 Parties

The orderer of the project is Erik Frisk. The supervisor is Theodor Westny. The contractors are Robert Sehlstedt, Thomas Andersson, Johan Forsman, Philip Sjövik, Erik Wahledow, Ludvig Hansson Granström, David Grahn and Mohamed Faraj.

1.2 Purpose and goal

The purpose of this project is to investigate the interplay of autonomous ground vehicles in complex traffic situations by developing a reliable system that can handle common simulated traffic situations that arise when multiple ground vehicles interact on public roads.

1.3 Usage

The system is used by connecting the cars via the Visionen-5GHz WiFi to an external computer running Ubuntu 20.04 with ROS1 Noetic installed. The different scenarios should be initiated from the external computer combined with the projectors and Qualisys systems in Visionen.

1.4 Requirements and priorities

This document presents several scenarios with close relation to real life traffic situations. All these situations are interesting to study, but as the project has a limited time budget, we have chosen to prioritise them according to the following priority level definitions.

Priority 1: Obligatory requirements.
Priority 2: Should be finished, but non-obligatory.
Priority 3: Optional requirements.

The objectives to complete are divided into three scenarios, each containing stages to complete. The first scenario builds up to functionality required to complete the latter two and most of the requirements are therefore of category 1 on priority. The second and third objectives to complete are of different size, and difficulty.

2 SYSTEM OVERVIEW

This section presents a brief overview of what the project will aim to achieve, how the system will be structured and under what assumptions it will perform.

2.1 Brief description of the product

A software system with the purpose of autonomously controlling a remote miniature car in complex traffic situations such as common roads, four-way intersections and roundabouts with surrounding traffic. To safely navigate the current environment, the car uses an internal map of its environment, knowledge of relevant traffic laws and by predicting future actions of other agents.

2.2 Product components

2.2.1 *Hardware*

The physical platform, known as the "Cascar" consists of a repurposed RC-car with the internal electronics replaced and augmented with custom made 3D printed parts. The important sensor and computational hardware is

- Raspberry Pi 4 (Main computational unit),
- Arduino (Pro mini) (Sensor data collection and car maneuvering),
- Rplidar A2 (LIDAR),
- Adafruit MPU-6050 (IMU),
- Allegro A1120 Hall effect sensors (wheel speed measurement).

Furthermore, external tracking is provided by Qualisys' position tracking system within the confines of Visionen. Within the space we also used two projectors in the ceiling to enable visualisations on the floor, and one additional projector for the wall.

2.2.2 *Perception module*

This software will process the data from the available sensors to estimate position, velocity and heading of the ego vehicle. Also estimating these states from other vehicles and implementing them into a dynamic map where they will act as obstacles to which the navigation module will plan around. The state estimation will be carried out with an Extended Kalman Filter.

2.2.3 *Planning & Control module*

Software for motion planning using motion primitives and state lattices will be implemented initially as path planning, the path planning might be expanded to trajectory planning to facilitate the timing aspects of autonomous vehicles. Control through the implementation of different methods such as pure pursuit and model predictive control (MPC).

2.2.4 *Visualisation module*

This module contain software for simulating the scenarios in Rviz and hardware in form of two connected projectors that will display things such as the map and other simulated vehicles on the floor in Visionen. A third projector is used to display information about the real time decision making.

2.2.5 *Simulator module*

The simulator module will be used for both testing purposes and in combination with the physical cars in Visionen. Therefore the module should be modular in and of itself, with the ability to add obstacles, different road configurations and simulation of other moving vehicles.

2.3 Dependencies to other systems

Our system is reliant to the accuracy of the Qualisys positioning system in Visionen, however this limitation will be reduced by implementing a Kalman filter. Other than that, the system only depends on Ubuntu machines, ROS and that Visionen-5GHz WiFi work as intended.

2.4 Scope delimitation

The scenarios are first evaluated in a simulation environment by visualisation in Rviz, and later evaluated on hardware in Visionen. We can not assure that the system will work on different hardware and in different scenarios. The system is however modular which eases the difficulty of implementing the system on similar hardware.

2.5 Design philosophy

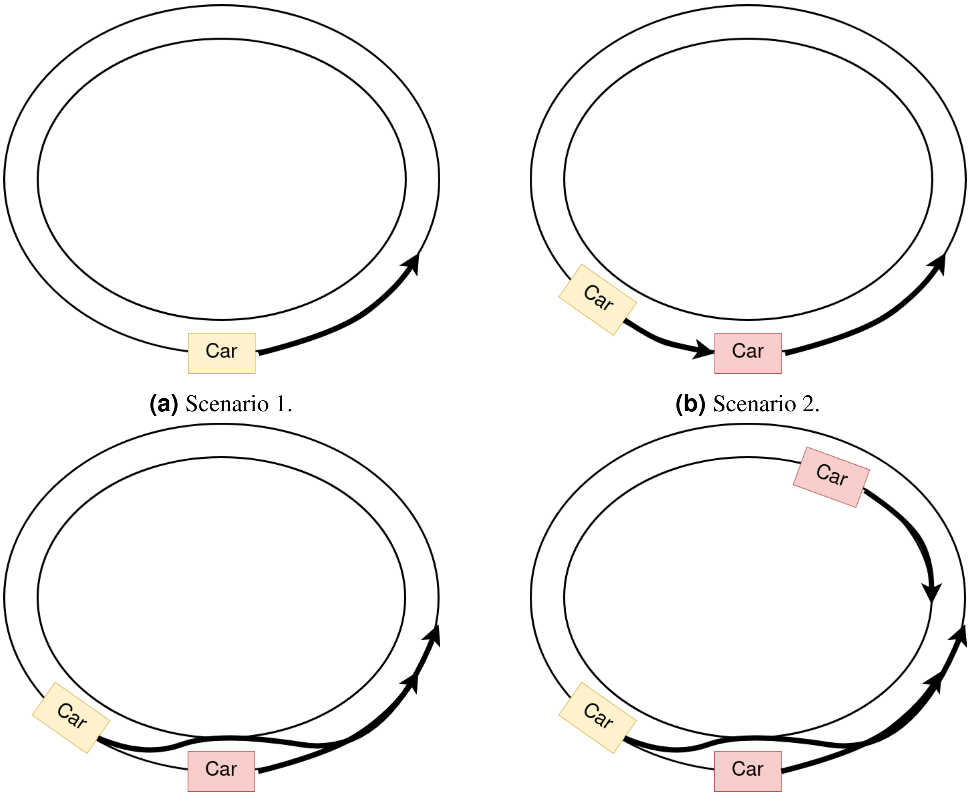
A main philosophy of this project is that we prefer to do a few things well than doing a lot with poor quality. Quality over quantity.

3 OBJECTIVE 1 - COMMON ROAD

The autonomous car may encounter slower moving impeding vehicles and needs a way to pass in a methodical and safe manner. The car will need to make decisions as to when to overtake if suitable. The decisions shall be based upon planned path (is there an exit approaching and is there time to overtake before it) as well as oncoming cars or other obstacles on the road.

3.1 Obligatory requirements

Requirements will be based on if the car can navigate a set of principal scenarios, each with increasing complexity.



(a) Scenario 1. **(b)** Scenario 2. **(c)** Scenario 3. **(d)** Scenario 4.

Figure 1: Scenarios 1-4.

Table 1: Requirements number 1 through 4.

Requirement	Version	Description	Priority
1	1.2	A Cascar can travel around the track with performance specified by technical documentation.	1
2	1.2	A Cascar can adapt its speed to follow a leading Cascar by slowing down.	1
3	1.2	A Cascar can overtake another Cascar by entering the left lane if the leading Cascar moves a fraction slower than itself, specified by the technical documentation.	1
4	1.2	A Cascar can await a suitable opportunity to overtake another Cascar if it detects an approaching Cascar in the left lane.	1

3.2 Non-obligatory requirements

Additional functionality may be implemented to also fulfil these scenarios.

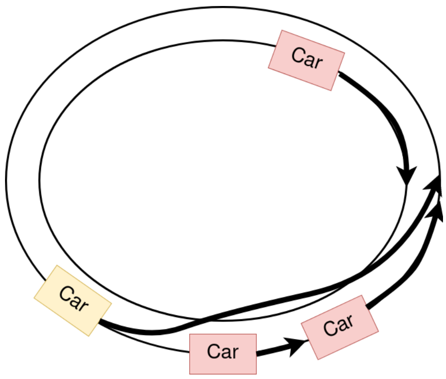


Figure 2: Scenario 5.

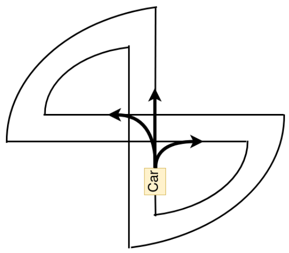
Table 2: Requirement number 5.

Requirement	Version	Description	Priority
5	1.2	A Cascar can perform an overtake of more than one Cascar when there is oncoming traffic in the left lane.	2

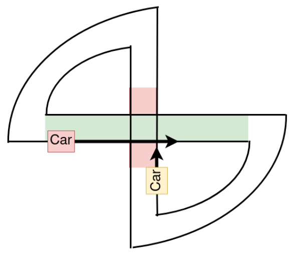
The modular requirements for scenario 5 are the same as for the obligatory requirements for scenario 1-4.

4 OBJECTIVE 2 - INTERSECTION WITH MAJOR ROAD AND YIELD

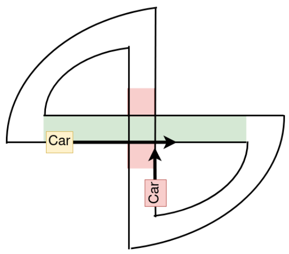
As an extension to the previous problem, the road will be shaped as an eight with an intersection in the middle. In this intersection, the cars that travel along one of the directions have precedence while the cars in the other direction must yield.



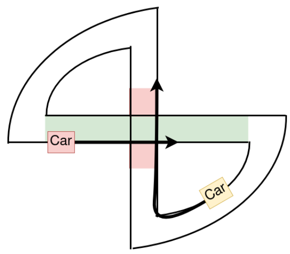
(a) Scenario 6.



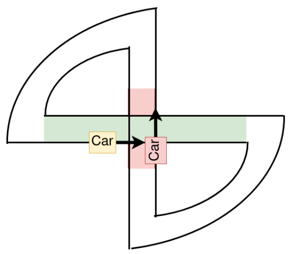
(b) Scenario 7a.



(c) Scenario 7b.



(d) Scenario 8.



(e) Scenario 9.

Figure 3: Scenarios 6-9. Red means yield, green means main route.

Table 3: Requirements number 6 through 9.

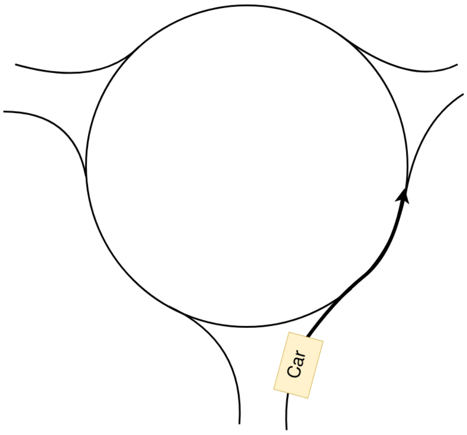
Requirement	Version	Description	Priority
6	1.1	A Cascar can drive around the path including taking left and right turns in the intersection.	3
7	1.1	A Cascar can drive around the path abiding the traffic rules with respect to other cars and the major road / yield intersection.	3
8	1.0	A Cascar can slow down ahead of the intersection if that is better with respect to not having to stop and letting cars on the major road pass.	3
9	1.0	A Cascar on the major road can avoid an accident by reacting in a safe way if another Cascar breaks the yield rules.	3

5 OBJECTIVE 3 - ROUNDABOUT

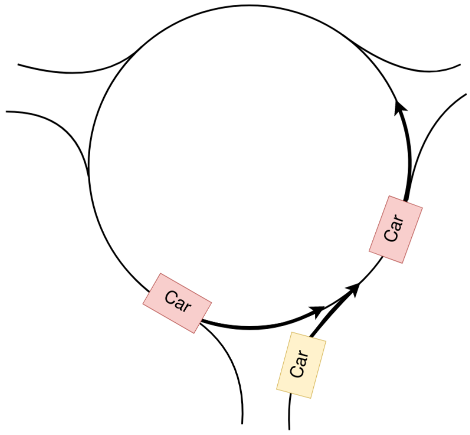
A roundabout is another interesting extension to the first problem. It will be shaped like a circle with one or two lanes and four roads attached to it.

Table 4: Requirements number 10 through 15.

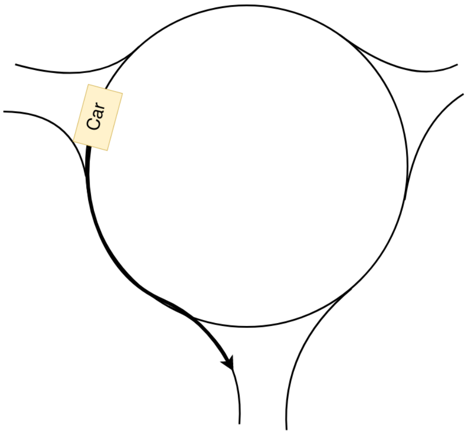
Requirement	Version	Description	Priority
10	1.1	A Cascar can drive around in a single-lane roundabout.	3
11	1.1	A Cascar can enter a single-lane roundabout when the traffic situation allows for it.	3
12	1.1	A Cascar can exit a single-lane roundabout.	3
13	1.1	A Cascar can enter and exit a two-lane roundabout including taking the correct lane given the destination (without surrounding traffic).	3
14	1.1	A Cascar can do everything from requirement number 13 but with surrounding traffic. Including lawful switches and timing when to enter and exit safely.	3
15	1.0	A Cascar can slow down upon entering the roundabout if that is better with respect to not having to stop before entering it.	3



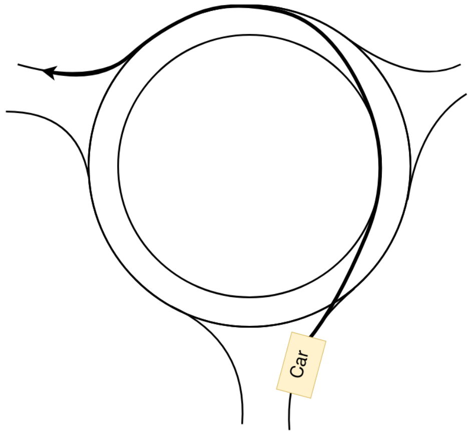
(a) Scenario 10.



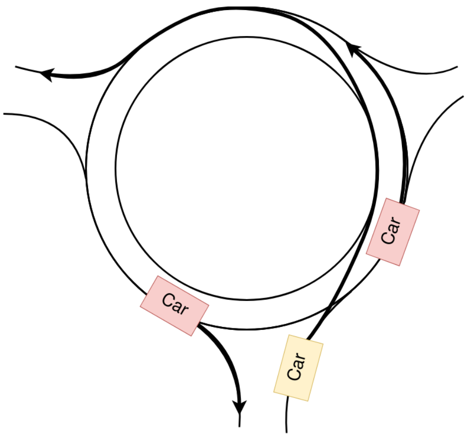
(b) Scenario 11.



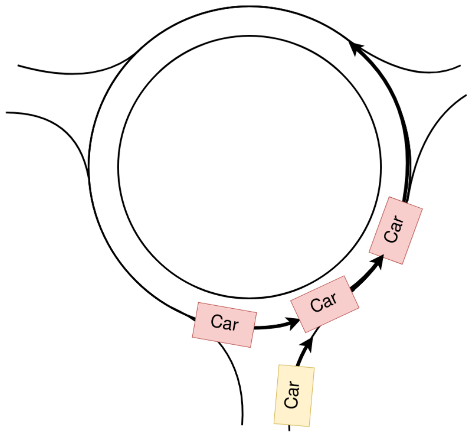
(c) Scenario 12.



(d) Scenario 13.



(e) Scenario 14.



(f) Scenario 15.

Figure 4: Scenarios 11-15.

6 VISUALISATION

Both for development and debugging, but also for marketing, a clear and informative visualisation can be very helpful. The visualisation on the floor in Visionen will be done through Rviz using the two interconnected projectors.

Table 5: Requirement number 16 through 19.

Requirement	Version	Description	Priority
16	1.2	The Cascar's planned and traversed paths can be visualised real-time on the floor in Visionen.	1
17	1.2	The position of simulated Cascars can be visualised in real-time on the floor in Visionen.	1
18	1.2	The road lanes can be visualised on the floor in Visionen.	1
19	1.0	Display high level abstractions such as logical decisions made by the Cascar on standalone projector.	2

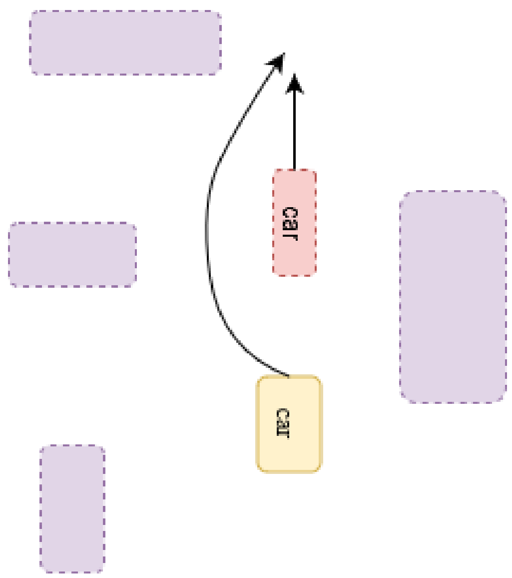


Figure 5: Visualisation on the floor.

7 SIMULATION

The simulations vary depending on what is being tested. In order to test a regulator for instance, Rviz is used together with a route planner. Features related to the perception module however requires prerecorded sensor data.

Table 6: Requirement number 20 through 21.

Requirement	Version	Description	Priority
20	1.2	Newly developed features in the perception module can be simulated and tested using Rviz.	1
21	1.2	Newly developed features in the navigation module can be simulated and tested using Rviz.	1

8 PERCEPTION REQUIREMENTS

The perception module is required to create an understanding about the environment and the cars pose in this environment.

Table 7: Requirement number 22 through 24.

Requirement	Version	Description	Priority
22	1.2	A Kalman filter with incorporated motion model to estimate the position of the Cascar given the sensory data from Qualisys and the wheel speed sensors is implemented, with performance specified in the technical documentation.	1
23	1.2	The Cascar has the ability to detect the position and velocity of other Cascars using Qualisys data. In the simulation environment, the obstacles will be simulated and no detection will be needed.	1
24	1.2	The Cascar has the ability to detect other Cascars while also estimating their future path or trajectory.	2

9 NAVIGATION REQUIREMENTS

The navigation module is required to navigate the environment perceived by the perception module. The path and trajectory planning is mostly focused on local planning as the evaluated scenarios are quite situational.

Table 8: Requirement number 25 through 28.

Requirement	Version	Description	Priority
25	1.2	A local path planner used to switch between lanes while performing an overtake.	1
26	1.2	A controller that allows the vehicle to follow the planned path or trajectory, with performance specified by the technical documentation.	1
27	1.1	A way of fetching information about the type of road (main route or not), regarding traffic rules.	3
28	1.0	A local trajectory planner to navigate around dynamic obstacles.	2