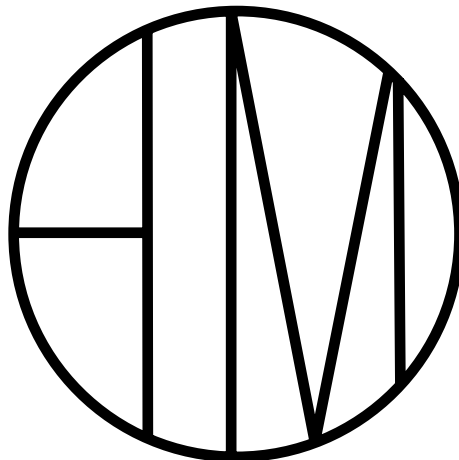


Project Plan LiU Racetrack 2018

LiU Racetrack 2018
Author: Oskar Karlsson
2018-10-08
Version 1.0



TRUKKMANIA

Status

Reviewed	Kim Larsson	2018-09-28
Approved	Oskar Ljungqvist	2018-10-08

Project identity

2018 HT, LiU Racetrack 2018

Department of Electrical Engineering - Linköping University

Name	Role	Email
Tomas Busk (TB)	Responsible for LEGO Truck	tombu079@student.liu.se
Mergim Dushku (MD)	Responsible for Migration to ROS	merdu044@student.liu.se
David Forsberg (DF)	Responsible for Migration to Visionen	davfo471@student.liu.se
Marcus Hall (MH)	Responsible for Simulation Enviroment	marha585@student.liu.se
Oskar Karlsson (OK)	Document Manager	oskka906@student.liu.se
Kim Larsson (KL)	Software Architect	kimla207@student.liu.se
Jacob Mourad (JM)	Project Leader	jacmo228@student.liu.se
Sara Nilsson (SN)	Test Manager	sarni339@student.liu.se

Orderer: Oskar Ljungqvist, Linköping University, +46 70 577 18 68,
oskar.ljungqvist@liu.se

Course responsible: Daniel Axehill, Linköping University, +46 13 28 40 42,
daniel.axehill@liu.se

Advisor: Olov Holmer, Linköping University, +46 13 28 16 17, olov.holmer@liu.se

Contents

1 Overview of the Project	1
1.1 Background	1
1.2 Purpose and goal	1
1.3 Deliveries	2
1.4 Limitations	2
2 Phase plan	3
2.1 Before project starts	3
2.2 During project	3
2.3 After project ends	3
3 Organization plan	4
3.1 Overview	4
3.2 Project group	4
3.2.1 Project leader	4
3.2.2 Document responsible	5
3.2.3 Software architect	5
3.2.4 Test responsible	5
3.2.5 Design responsible	5
3.2.6 Simulator responsible	5
3.2.7 LEGO-truck responsible	5
3.2.8 Responsibility regarding the migration to ROS	6
3.2.9 Responsibility regarding the migration to Visionen	6
4 Documentation plan	6
5 Development methodology	8
6 Report plan	8
7 Meeting plan	9
8 Resource plan	9
8.1 Personnel	9
8.2 Material	9
8.3 Work areas	10
9 Milestones and tollgates	10
9.1 Milestones	10

9.2 Tollgates	11
10 Activities	12
10.1 General and documentation	12
10.2 Parking functions	14
10.3 ROS	14
10.4 Simulator	15
10.5 LEGO truck	15
11 Time plan	16
12 Risk analysis	16
13 Quality plan	17
13.1 Reviews	17
13.2 Test plan	17
14 Project's ending	18

Document history

Version	Date	Changes	Performed by	Reviewed by
0.1	2018-09-21	First draft	Project Group	OK
0.2	2018-09-28	Revised after comments from orderer	Kim Larsson	KL
1.0	2018-10-08	Approved by orderer.		

1 Overview of the Project

This project is a part of the course *Automatic Control - Project Course, CDIO* which is held at Linköping University at the department of Electrical Engineering, ISY. This section describes the background and purpose of the project as well as the limitations and deliveries to the customer.

1.1 Background

The LiU Racetrack project has been around since 2011 and has been a the platform for various student projects and thesis projects. The work done on the platform has resulted in functionality such as controllers for trajectory tracking and path following, overtaking of obstacles, motion planners and a simulator among else. All of this has been developed for several remotely controlled (RC) vehicles. Autonomous vehicles is an exciting field of research and there is a large amount of work to be done in this area.

1.2 Purpose and goal

The purpose for the project is to create a robust system to be used for research and in education in advanced control courses at Linköping University. The goals for this year are to deliver a module-based system implemented in ROS with a performance at least as good as the original system and to deliver an automatic LEGO truck with a dolly-steered trailer. The LEGO truck shall have the ability to execute advanced parking maneuvers such as parallel parking and reversing up to a loading bay. The existing parts in the original system will be evaluated based on their performance to conclude which of them performs good and which need to be revised. The project group will distribute several roles and responsibility areas and will create an activity list along with a time plan. By the end of the project the project group will have gained experience in working together in a larger group in a larger project following the directives from the *LIPS project model* and from a customer.

1.3 Deliveries

Date	Delivery	Version
2018-09-14	Requirement Specification	First Version
2018-09-21	Project Plan	First Version
2018-09-24	Time Plan	First Version
2018-09-26	Requirement Specification	Final Version
2018-09-26	Project Plan	Final Version
2018-09-26	Time Plan	Final Version
2018-09-26	Design Specification	First Version
2018-09-26	Verbal Presentation of the system	
2018-09-28	Test Plan	First Version
2018-10-03	Design Specification	Final Version
2018-10-03	Test Plan	Final Version
2018-11-16	Simulation for truck with trailer finished	
2018-11-16	A path following controller for the LEGO truck with trailer should be able to stabilize the system around an eight-shaped reference path in Visionen.	
2018-11-16	System ready to move to Lilla Arenan	
2018-12-03	Test Protocol	First Version
2018-12-03	User Manual	First Version
2018-12-07	All functionality delivered	
2018-12-07	Test Protocol	Final Version
2018-12-07	User Manual	Final Version
2018-12-07	Presentation showing fulfillment of all requirements	
2018-12-07	Technical Documentation	First Version
2018-12-14	Technical Documentation	Final Version
2018-12-14	Afterstudy	Final Version
2018-12-14	Poster Presentation	
2018-12-14	Webpage finished	
2018-12-14	Film to Publish	

Table 1: Deliveries throughout the project.

1.4 Limitations

The most significant limitation to the project is the amount of time the group members can put in the project as well as the amount of time the advisor can be utilized.

This year there are several projects wanting to use the room Visionen which limits the access to this room.

2 Phase plan

The project will be divided into three phases according to the LIPS project model: before the project starts, during the project and after the project ends. This section describes these phases and which activities that is associated with them.

2.1 Before project starts

This phase marks the forming of the project group and defining the task to be completed. Activities associated with this phase are the distribution of roles and responsibilities, definition of requirements specification and the creation of this project plan together with the time plan.

2.2 During project

This is the phase also called as the *development phase* where most of the activities defined in the project plan are exercised. A design specification is constructed along with a system architecture and is followed by the implementation of the design. A test plan is defined and status reports are delivered weekly to the orderer and the customer.

2.3 After project ends

In this phase the product is to be delivered to the customer and an acceptance test is made in order to verify the fulfillment of the requirements. A technical documentation, user manual and an after study will be created by the project group that summarizes the project. In order to advertise the final product the project group should also launch a web page containing all relevant documentation along with a advertising movie. The group shall also create a poster that is to be displayed at the conference at the end of the course.

3 Organization plan

This section describes the organization of the project and describes the roles, both in the project group and external, and the connections between them.

3.1 Overview

The project consist of several roles both within the project group and outside of it. Figure 1 presents an overview of the roles and the connections between them where the color green refers to roles within the project group and blue refers to external roles.

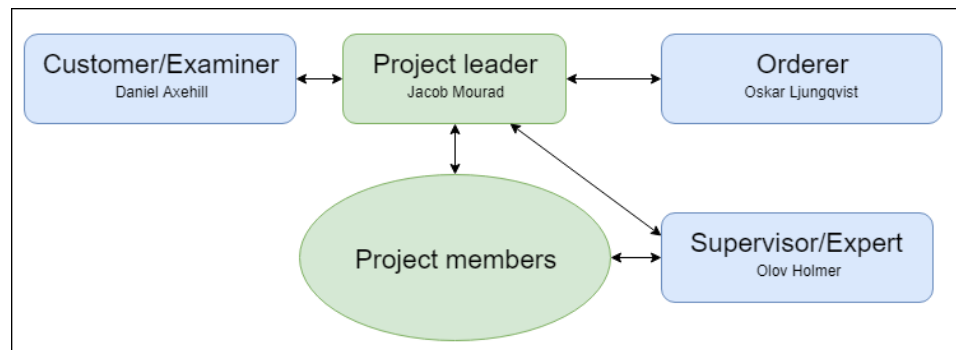


Figure 1: Project organization. Green is project group roles, blue is external.

3.2 Project group

This section describes the different roles within the group and what their respective responsibilities and commitments are.

3.2.1 Project leader

The person assigned to this post is responsible for making sure that all the goals of the project are achieved. The project leader should also structure up the work and encourage the other project members to work effectively together.

3.2.2 Document responsible

The document responsible is responsible of making sure all the documents are structured correctly, contain the correct information and also making sure all documents are delivered on time.

3.2.3 Software architect

The software architect is responsible for making sure that the code follows the Google code standard, which can be found at [1], as well as making sure that the code is well structured and has been documented correctly.

3.2.4 Test responsible

The test responsible is responsible for that the system is properly tested throughout the project. A test plan and test protocol developed early in the project process will help with the testing.

3.2.5 Design responsible

Responsible for the overall design of the different systems and making sure they function as a unit.

3.2.6 Simulator responsible

This person is responsible for the development of a simulation environment in Gazebo and Rviz.

3.2.7 LEGO-truck responsible

Making sure the LEGO-truck is implemented correctly and fulfills the specified requirements.

3.2.8 Responsibility regarding the migration to ROS

This person is responsible for the modulation and migration of the original system to ROS.

3.2.9 Responsibility regarding the migration to Visionen

This person is responsible for the migration of the positioning system from the project room to Visionen.

4 Documentation plan

The documents in the project will be written using \LaTeX via the web-tool *Overleaf*. The documents will follow the LIPS project model, the LIPS book can be found at [2]. Table 2 states all documents that are to be written during the project. The target groups are: Project group (PG), Customer (C), Course responsible (CR) and Orderer (O).

Document	Description	Target group (s)
Requirements Specification	Specifies the requirements that shall be fulfilled at the final delivery of the project.	PG, C, O
Project Plan	Defines the methods used by the project group in order to accomplish the tasks given to them. It gives an overview of the organization, milestones and deliveries for the project.	PG, O
Time Plan	Specifies how the activities shall be distributed between the project members throughout the project and how much time to put into each activity.	PG, O
Design Specification	Describing how the different parts of the project shall work and how they shall be implemented.	PG, O
Test Plan	Describing the tests used to verify that the requirements are fulfilled.	PG, O
Test Protocol	A list of the requirements that shall be tested and whether they are fulfilled or not.	O
User Manual	Describes how the system shall be used and how to solve common issues that might occur during use.	C
Technical Documentation	Describes how the system works in detail once implemented as well as some improvements that can be made in future projects.	O
After Study	Describes how the project has gone, what went well, what could be done better and how it could be used in a future project.	O, CR
Poster	A short summarizing poster of the project giving an overview of the project and its features.	C
Advertising Movie	A video of the system in action that demonstrates its features.	C
Web page	Web page that presents the project with the relevant documents.	C
Meeting protocol	Notes on what has been said during project meetings.	PG
Status Report	Status of the project that shall be delivered every week.	PG, C, O
Time Report	Report of activities and time spent that shall be delivered every week.	PG, O

Table 2: Documents to be produced during the project.

5 Development methodology

The project will be conducted using the *Kanban methodology*. Kanban is a lean methodology that focuses on minimizing multitasking and active concurrent tasks. The main focus of Kanban is the *Kanban board* where all the projects tasks are listed as *cards*. Figure 2 shows an example of a Kanban board with a set of cards, marked with yellow. Initially the cards are located in the *backlog* and every time a new task is started the corresponding card is marked with the ID of the person responsible for the task and moved to *in progress*. The Kanban methodology restricts the amount of cards that each person can be assigned and the amount of *active cards*.

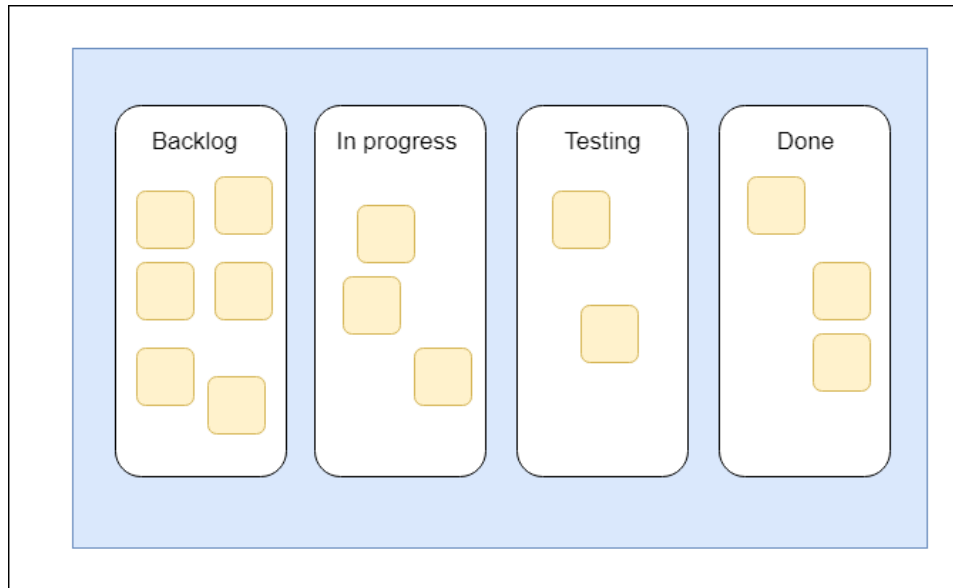


Figure 2: Example of Kanban board.

6 Report plan

The group is to deliver weekly reports to the orderer containing time reports from each project group member and a summarizing status report of the past weeks work. These status reports are handed over to the orderer by the group leader. Each member of the project group is responsible for updating the time plan with worked hours on the different activities. Besides the weekly status reports the project group

will have, at least, weekly meetings planing the activities of the upcoming week and evaluating the past work. Meeting protocols will be written during these weekly meetings in order to keep notes of the discussions and decisions taken. The meeting protocols will be written and stored in the project groups Google Drive.

7 Meeting plan

As mention earlier the group is to have weekly meetings where planing of upcoming work and evaluating of past work will be made. These meetings will be held on Mondays 13:15-15:00. Prior to each meeting an agenda will be available in the Google Drive and each member is responsible for reading through it in order to prepare for the meeting. The project group may spontaneously decide to have additional meetings if the need arises, this shall be published in the groups Slack and meeting protocols shall be written.

8 Resource plan

The resource plan states the resources such as work hours, available hardware and work areas and describes the planned usage of these resources.

8.1 Personnel

The project group consist of eight members whereas each member is expected to put in 240 work hours on the project. This implies a total of 14 weeks from project's start to end where every member is to put in about 20 hours/week. One exception is the exam- and re-exam periods during weeks 43-44 where the members are not expected to work the same amount of time.

The project group is also entitled to a total of 40 hours of tutoring from the supervisor and orderer as long as it is planned in good time.

8.2 Material

Below is a list of the hardware included in the project.

- RC-controlled race car

- RC-controlled semi-truck with additional dolly
- LEGO truck equipped with a dolly-steered trailer and a LEGO EV3 unit
- IR-cameras used in the positioning system
- Linux-computer that will run the ROS-system
- Windows computer that will run the positioning system
- Raspberry Pi used for communication with the semi-truck
- Projector for visualization of trajectories on the race track
- Miniature Grand Prix race track for driving the vehicles on

8.3 Work areas

The project group will have access to two work areas, one project room where the racetrack is located and one group room in connection to the Visionen arena. These work areas are project specific and is not shared with others outside of the project. The group will also have access to the small Visionen as well as limited access to the big Visionen.

9 Milestones and tollgates

This section describes each of the project's milestones and tollgates and what is expected to be accomplished by these. Milestones are internal tollgates made by the project group. Tollgates are decision points where the work up till that point is evaluated and it is decided upon if it is beneficial for the project to continue or not, this decision is made by the orderer.

9.1 Milestones

Below are the milestones defined by the project group and their respective date.

ID	Description	Date
1	State observer working for the LEGO truck.	2018-10-12
2	ROS nodes created for existing code.	2018-10-16
3	Model of car, LEGO truck and truck with semi-trailer finished in the simulator.	2018-10-21
4	Trajectory following for the LEGO truck is working.	2018-10-21
5	Parking feature working for the car.	2018-11-02
6	Simulator working for the car.	2018-11-07
7	Simulator working for the LEGO truck.	2018-11-09
8	Simulator working for the truck with semi-trailer.	2018-11-14
9	System up and running in lilla Visionen.	2018-11-30

Table 3: Milestones

9.2 Tollgates

The tollgates associated with this project is provided by the orderer and is described in Table 4.

ID	Description	Date
2	Delivery of requirements specification, project plan including time plan, a first draft of the design specification and a verbal presentation of the system.	2018-09-26
3	Delivery of design specification and test plan.	2018-10-03
4	Delivery of simulator for truck with semi-trailer and a path following controller for the LEGO truck that can stabilize the system, both backward and forward, around an eight-shaped reference path. The whole system shall also be ready for migration to Visionen.	2018-11-16
5	Delivery of the final project together with a user manual. A test protocol shall also be delivered along with a presentation of the system proving the fulfillment of the requirements of priority 1.	2018-12-07
6	Delivery of technical documentation, web page, poster and advertising movie. An after study is also handed in and the project is terminated.	2018-12-14

Table 4: Tollgates

10 Activities

This section describes all the activities defined for the project. All activities are estimated to take a total of 1920 hours to complete, including planned buffer time.

10.1 General and documentation

Table 5 describes the planned general activities including the work related to the documents in the project.

Number	Activity	Description	Estimated time (hours)
1	Requirement specification	Establish a requirement specification for the project.	80
2	System outline	Learn how the old system works and establish a system outline for the project.	80
3	Project plan	Establish a project plan for the project.	40
4	Time plan and time reporting	Establish a time plan for the project.	25
5	Test plan	Establish a test plan for the project.	40
6	Design specification	Establish a design specification for the project.	100
7	Technical documentation	Establish a technical documentation for the project.	80
8	Test protocol	Establish a test protocol for the project.	10
9	User manual	Establish a user manual for the project.	20
10	Preparation for final presentation	Preparation of the presentation for the project conference	20
11	Meetings	Planned meetings with the project group as well as with advisor and orderer.	192
12	Web page	Online presentation and reference for the project after it is completed.	20
13	Poster	A poster giving an easy to understand view of the project.	15
14	Video	A short video showing the features of the project.	20
15	Testing	Testing to make sure the requirements are fulfilled.	50
16	Deliveries	Time for the presentation of the project during the deliveries.	24
17	After study	Evaluation of the project as well as its future potential.	16
18	Buffer time	Excess time for activities exceeding their estimated time or for new activities.	168
		Partial sum	1000

Table 5: Activities for general and documentation

10.2 Parking functions

Table 6 describes the activities regarding the parking functions for the car and the truck with a semi-trailer.

Number	Activity	Description	Estimated time (hours)
19	Evaluate current motion planner for application in parking function	Explore how the current motion planner can be used for the parking function	20
20	Implement parking function for the car	Implement parking function for the car.	60
		Partial sum	80

Table 6: Activities for the parking feature

10.3 ROS

Table 7 describes the activities regarding the modularization of the system and implementation of ROS-nodes.

Number	Activity	Description	Estimated time (hours)
21	Define ROS nodes from the old system	Create nodes of the existing code.	80
22	Define and document interfaces between ROS nodes	Define and document interface between nodes.	20
23	Implement defined ROS nodes	Implement defined ROS nodes in the system.	80
24	Evaluate performance against old system	Evaluate the performance of the new system so that it at least is as good as the old system.	25
		Partial sum	205

Table 7: Activities for ROS implementation

10.4 Simulator

Table 8 describes the activities regarding the simulator.

Number	Activity	Description	Estimated time (hours)
25	Learn Gazebo	Learn how Gazebo works	25
26	Learn ROS and Rviz	Learn how Rviz and ROS works	70
27	Visualize the system in Rviz	Implement visualization of ROS in Rviz.	40
28	Create models for the car, LEGO truck and truck with semi-trailer	Create models for the car, LEGO truck and the truck with the semi-trailer.	30
29	Develop simulator using Gazebo	Develop functionality in the simulator.	100
		Partial sum	265

Table 8: Activities for simulator

10.5 LEGO truck

Table 9 describes the activities regarding LEGO truck, including those related to its parking functions.

Number	Activity	Description	Estimated time (hours)
30	Install ROS-node on EV3	Install ROS-node on truck EV3 unit	10
31	Preliminary study of planning algorithm	Do pre-study of planning algorithm for the LEGO truck.	40
32	Implement planning algorithm	Implement planning algorithm for the LEGO truck.	80
33	Preliminary study of trajectory following controller	Do a pre-study of trajectory following controller.	30
34	Implement path following controller	Implement path following controller for both forward and backward movement.	100
35	Implement parking function for parallel parking	Implement parking function for parallel parking	40
36	Implement parking function for reversing towards loading bay	Implement parking function for reversing towards loading bay	20
37	Develop state observer	Develop state observer for the LEGO truck	50
		Partial sum	370

Table 9: Activities for LEGO truck

11 Time plan

The time plan made for the project can be found in Appendix A. This time plan will every week be updated and modified as each member of the project group register how much time they spend on each activity.

12 Risk analysis

The main risks in this project are the ones that causes a delay in deliveries such as damaged hardware, illness, physical injuries and arguments among group members. Even though the members will do their best in order to not cause any harm to

themselves there is still a risk for accidents to happen that will delay the project. By choosing to follow the Kanban methodology the project group may decrease delay since the methodology limits the amount of tasks that a person can be assigned. In order to minimize the delay the following actions will be taken:

- If a group member is injured or becomes sick for a longer period of time the rest of the group will contact the orderer and discuss the lost work hours and eventually re-negotiate the requirements. The group will then continue to divide the lost member's work among them, preferably to members who has some knowledge of said work.
- If hardware gets damaged the group will stop using the hardware and contact the orderer as soon as possible in order to get the hardware fixed without causing it further damage.
- Arguments within the group are difficult to avoid and may require input from the orderer. The project group will try to solve the argument by ex. voting on design arguments or letting the member responsible for the corresponding part of the system decide. If the conflict can't be solved by these methods the orderer is contacted.

13 Quality plan

This section describes how the project group will work in order to keep a high quality of the produced code.

13.1 Reviews

When working on separate parts of the code these will be worked upon in a separate branch. When connecting this to the master branch it will be reviewed by a different project member before merging them to ensure it fulfills the code standard.

13.2 Test plan

To ensure the quality of the performed activities everything will be thoroughly tested according to the defined test plan.

14 Project's ending

The project will end when all deliveries are fulfilled, at tollgate 6. All documentation together with the poster and the promotional video will be published on the website. The project group will conduct an after-study where the group members evaluate their newly found experience and the outcome of the project.

References

- [1] Google code standard for C++, Available at:
<https://google.github.io/styleguide/cppguide.html>
- [2] Tomas Svensson and Christian Krysaner. *Project model LIPS*. Studentlitteratur, 2011.