Group 02

G.U.A.R.D.

Project Plan

**Supervisors:**

Maria Chiara Lucatello

Mayra Soliz



**Team members:**

Axel Granli

Boyan Dai

Erik Laurin

Gabriel Marian Bulai

Joacim Eberlen

Justinas Stirbys

Shaun McMurray

Table of Contents

[1. Introduction 3](#_Toc482897483)

[2. Project Organization 3](#_Toc482897484)

[3. Project Practices 4](#_Toc482897485)

[3.1 Meetings 4](#_Toc482897486)

[3.2 Quality 4](#_Toc482897487)

[3.3 Teamwork 4](#_Toc482897488)

[4. Project Objectives 5](#_Toc482897489)

[4.1 Deliverables 5](#_Toc482897490)

[4.2 Hardware and software resource requirements 5](#_Toc482897491)

[4.3 Work breakdown 6](#_Toc482897492)

[4.4 Schedule 6](#_Toc482897493)

[4.5 Communications between stakeholders 8](#_Toc482897494)

[5. Risk Management 10](#_Toc482897495)

[5.1 Identified Risks 10](#_Toc482897496)

[5.2 Probability and consequences 10](#_Toc482897497)

[5.3 Risk handling 11](#_Toc482897498)

# 1. Introduction

The aim of the project is to create a SmartCar with the ability to be controlled via the mobile application as well as imbued it with a certain level of autonomy. Focus will be put towards enabling the car to track users via their mobile device GPS coordinates. The project will also aim to enable the SmartCar to follow the application’s users once they are in range. Furthermore, the application should also possess camera support and allow the users to stream the video from the car to their mobile device.

# 2. Project Organization

Individual team members will take responsibility for their assigned tasks by acting both as developers and testers aiming towards ensuring a high quality deliverable. Due to the size of the group, the team will also practice collective code ownership. With this practice team members will be able to alter others’ code and take responsibility for the changes made. The goal of this is to use the members’ individual knowledge and abilities improve the quality of the source code as well as reducing the risk related to member absence.

The team will make user stories collectively as well as assigning effort estimations. At each sprint’s initial scrum meeting, the members will pick their user story. The members are encouraged to pick additional user stories in case they finish their initial story before it is due.

Each team member will take upon developer and tester roles. Besides these, several special roles exist. A project manager role will be taken by Erik Laurin. The project manager will act as the leading voice throughout the project meetings. It will be their responsibility to settle disputes amongst team members and to maintain order and productivity throughout the meetings. A test manager role will be taken upon by Joacim Eberlen. The test manager will be responsible for implementing a test environment and educating, if needed, the remaining team on UI tests. The quality manager will be responsible for establishing and monitoring reachable quality thresholds. This role will be taken upon by Justinas Stirbys.

# 3. Project Practices

## 3.1 Meetings

The team utilizes iterative development and will work in two week long iterations. Iterations will begin with an initial sprint plan during which the team will decide which user stories and task to tackle. At this point the team will engage in a discussion surrounding acceptance criteria and what is desired from the tasks. Throughout the iteration the team will aim to have 6 face to face meetings, 3 each week, as it is believed that these are most productive. The face to face meetings will begin with a quick stand-up meeting before moving on to working on the meeting agenda. The team will also utilize Trello as a mean of tracking progress while not on meetings. In particular, it was decided to use percentage indicators for tasks to indicate progress. The last meeting of the sprint will be spent on sprint retrospectives, to find possible improvement that could be incorporated.

## 3.2 Quality

The development team will also use the program SourceMonitor to track and measure code quality. In particular, the team will focus on cyclomatic complexity and block depth. On occasion, if necessary the team will also take into account lines of code. However, this was a late addition to the project and will only be carried out from sprint 4 and onwards. SourceMonitor will be used once at the end of each sprint to examine the source code quality by checking whether or not the established thresholds are met. The threshold for cyclomatic complexity was designated as 15, the threshold for block depth as 7. In addition, to SourceMonitor the team will also use code review to determine quality. The team aims to have code review sessions each week or once a sprint the very least. With code review sessions the team aims to familiarize with each other’s code and possibly improve by utilizing shared code ownership. At this point lines of code will be taken into account if the class is rather large and lacks cohesion. If classes over the threshold are identified, they will undergo refactoring to improve maintainability.

## 3.3 Teamwork

During the face to face meetings the team will also have time to program. If any member has difficulties with their assigned tasks for if they are rather large, the members are free to ask for aid. At which point aid will be provided in the form of pair programing. This is done in order to get fresh eyes on the task on to hopefully help the developers improve. Additionally, the team members use group programming, because it is believed that this will cause an increase in productivity. When producing documentation a similar practice to pair programing will be used, to hopefully improve the quality and readability of the project’s documentation.

# 4. Project Objectives

## 4.1 Deliverables

Mandatory deliverables:

* Mobile application
* Collision prevention
* Camera support

Optional deliverables:

* Tracking a user
* Following a user
* Obstacle avoidance
* Self-Navigation

## 4.2 Hardware and software resource requirements

Hardware:

* Arduino Mega 2560
* Raspberry Pi 3 Model B
* SR04 Ultrasonic Module x 6
* Lidar Lite V3
* SG90 (Servo)
* Odometer x 2
* Gyroscope
* Bluetooth module
* Battery pack x 2
* GPS module
* Camera
* Adafruit LSM303 compass
* Voltage Divider
* Heat sink
* AA Rechargeable battery x 4
* Power bank

Software:

* WebStorm
* PyCharm
* Arduino IDE
* Android Studio
* Fritzing
* SourceMonitor

## 4.3 Work breakdown

Mobile application:

1. Provide basic movement control (forward, right, left, reverse)
2. Show live video stream from car
3. Initialize the tracking
4. Initialize the following

Collision prevention:

1. The sensors will continuously scan the car’s surroundings and stop the car to prevent a collision.
2. The user will be notified in the mobile application of the reason why the car stopped.

Camera support:

1. The camera on the car allows live streaming to the mobile application.

Tracking a user:

1. The SmartCar tracks the user by comparing its position to the position of the phone (GPS/Glonass coordinates).
2. The SmartCar travels to the user’s location.

Following a user:

1. The SmartCar follows the traveler after this has been initiated in the mobile application.
2. The SmartCar will send notifications if problems are encountered (for example the traveler goes out of range, battery life drops to critically low or the GPS/Bluetooth signal becomes too poor to send sufficient signals.)

Obstacle avoidance:

1. The SmartCar should be able to avoid obstacles in its path while following and tracking the traveller.
2. Sensors such as lidar and ultrasonic will provide an estimated distance to the obstacles. The camera will provide obstacle recognition which will aid the obstacles avoidance.

## 4.4 Schedule

Sprint 1:

* Android application
  + Digital analog controller
* Improved collision prevention
* Improved manual controls

Activities:

* Creation of user stories and backlog definition
* Standup meetings
* Pair-Programming sessions
* Sprint Retrospectives
* Set up a test environment
* Documentation/Code Review

Sprint 2:

* Incorporate the Raspberry Pi 3 into our system
* Streaming video to android application with over laying controls

Activities:

* Creation of user stories
* Standup meetings
* Pair-Programming sessions
* Sprint Retrospectives
* Testing
* Documentation/Code Review

Sprint 3:

* Mapping immediate surroundings
* Finalized hardware prototype

Activities:

* Standup meetings
* Pair-Programming sessions
* Sprint Retrospectives
* Testing
* Documentation/Code Review
* Code Refactoring

Sprint 4:

* Object avoidance
* GPS following

Activities:

* Standup meetings
* Pair-Programming sessions
* Sprint Retrospectives
* Testing
* Documentation/Code Review
* Code Refactoring

Prototype demo:

* Live showcase of features

Activities:

* Demo preparation
* Hardware maintenance

Sprint 5:

* Object recognition

Activities:

* Standup meetings
* Pair-Programming sessions
* Sprint Retrospectives
* Testing
* Documentation/Code Review
* Code Refactoring

Sprint 6:

* Focal point following

Activities:

* Standup meetings
* Pair-Programming sessions
* Sprint Retrospectives
* Testing
* Documentation/Code Review
* Code Refactoring

Sprint 7:

* Self-navigation

Activities:

* Standup meetings
* Pair-Programming sessions
* Sprint Retrospectives
* Testing
* Documentation/Code Review
* Code Refactoring

Final presentation

* Live showcase of features
* PowerPoint presentation (if necessary)
* Video (if necessary to demo the functions)

Activities:

* Presentation preparation
* Hardware maintenance
* Finalize documentation

**Resource plan**

Our source code and documentation will be found at the following GitHub repository:

[DIT524-V17](https://github.com/DIT524-V17)/[group-2.](https://github.com/DIT524-V17/group-2) [Link](https://github.com/DIT524-V17/group-2)

## 4.5 Communications **between** stakeholders

The group will meet at least three times in person in one of the study room at Patricia building or Science Park.  Additionally, a scrum meeting could be held at one of the team member’s residence if everyone is willing. The team will meet on Mondays 10:00-12:00, Wednesdays 10:00-12:00 and Fridays 10:00 - 15:00. An additional meeting time is on Thursdays 10:00 - at most 17:00. Although this is an extra meeting and will only be held if required i.e. the team is falling behind with the work load.

The team will use Slack for basic communication by text. Trello will be used for planning and assigning user stories to the team members. A discord server for talking about problems we encounter between meetings will also be set up. Although discord is not a mandatory communication platform and not all team members are required to participate. It is used for providing a quick reply via vocally and/or text and is aimed at team members helping each other with coding problems encountered between meetings.

Several ground rules have been established that are geared towards improving and maintaining the communication quality. Note that the following the rules will result in us consulting the teacher assistants about the problem.

**Ground Rules:**

* Missing three meetings without providing a rational reason.
* Missing the delivery deadline two times in a row.
* Be reachable every weekday

# 5. Risk Management

## 5.1 Identified Risks

Several risks that may occur throughout the course of the project have been identified. These risks are of varying types, such as planning and resource management, technical as well as communicatory.

The identified risks relating to the relationships of stakeholders include***disagreement amongst stakeholders.*** This risk focus on the possible disagreements that may occur between the project’s stakeholders, such as team member disputes on which feature takes priority or even disagreement on the features necessity. Another identified risk of this type includes ***team members not being committed to the project****,* meaning that possibly some team members may experience distractions taking away their focus from the project and leading them to accomplish less than expected.

Risks relating to planning and resource management that have been identified are ***hardware unavailability, limited hardware quality* and *developers lacking key skills****.* Quality hardware can have a significant impact on embedded system projects, therefore unavailable and quality lacking hardware would have a negative effect on the final product by restricting the development team to settle for subpar utilities. For example, sensors lacking quality would not produce as reliable readings as possible, thus downgrading the SmartCar’s accuracy and providing less customer value. Furthermore, the most significant risk of this type would be the team’s lack of knowledge and skills pertaining to hardware and its handling, which could cause an array of problems leading to lower quality product and customer value.

Lastly, technical risks include ***unfamiliar tools***and ***unfamiliar system software****.* The team opted to use “Arduino” for movement control of the SmartCar and “Android Studio” for the construction of the mobile application. These two would be the aforementioned unfamiliar tools, since the team currently has a narrow understanding of “Arduino” and “Android Studio” capabilities. Moreover, the team decided to use unfamiliar software systems to develop their project. These systems include PyCharm, for dealing with Raspberry Pi via python, and Android Studios, for the development of the mobile app.

## 5.2 Probability and consequences

The team discussed the probability and consequences of the before identified risks throughout the project planning. Unfortunately, none of the identified risk had low probability of occurrence. It was decided that ***users not committed to the project, hardware unavailability and limited hardware quality***had a medium probability of occurrence i.e. between 25% and 50%. The rest of the identified risks, ***unfamiliar tools,* *unfamiliar system software, developers lack key skills***and***stakeholders disagree*** had a high probability to occur, which is over 50%.

User commitment decreasing was considered to have moderate probability of occurrence based on the fact that several team members currently being employed and thus cannot spend as much time with the project as the others. It was also based on the length of the project and basic human needs, such as rest and relaxation, which leads to fatigue. Although the consequences are not catastrophic they are definitely moderate. This risk can lead to lower productivity and a lower overall velocity of the team as the project progresses. This could cause easily avoidable errors as well as lower the value of the deliverable. Imposed hardware limitations were also assigned a moderate probability, due to the team already being denied some desired components, although not all. The consequences of these risks have been evaluated to moderate, since the team would compromise for lower quality components requiring additional time to be spent combating this drawback. This in term would lead to reduced accuracy of the SmartCar and reduced movement capabilities, thus lowering the delivered customer value.

Throughout the project plan discussion it was agreed that disagreements will take place and that it is inevitable, therefore a high probability was assigned to this risk. However, it was thought that the consequences stemming from this risk would be tolerable. It would result in some more time spent during the meetings and work deviation phase and possibly lower the team’s synergy. Moreover, unfamiliar tools and software system were also assigned a high probability to occur. This was based on the team’s current familiarity with the tools and systems. It was mutually agreed that there is not enough of time to familiarize with them before the first sprints. These risks shall cause a portion of time allocated to learning the software at the start of the development phase. However as time progresses, knowledge of these tools will be acquired diminishing the effects of the risks, therefore it is believed that these risks will produce tolerable consequences. Finally, developers lacking key skills were assigned high probability based on the team’s knowledge and skills, in particular the lack of them when it comes to hardware. However, the team has access to several teaching assistants specifically for this purpose as well as numerous guides online; therefore this risk was deemed to possess consequences of tolerable nature.

## 5.3 Risk handling

Risks assigned with a high probability of happening are not avoidable, for example there will always be disagreements amongst the stakeholders and there is not enough time left to familiarize enough with the tools used before the sprints begin, although the effects of these risks can be diminished. The team attempted to obtain as much knowledge as possible about these tools before the developments phase began. The team will also continue using these tools throughout the project and thus improve with their handling significantly. This will diminish the severity of the risks as the project progresses. Moreover, hardware can be purchased; therefore the risks relating to hardware unavailability are partially avoidable. Additionally, the team has accesses to two teaching assistants and to some additional help relating to hardware, both of which can be used to compensate for team members lacking key skills. User commitment faulting can possibly be avoided. This risk depends specifically on the team members and their approach towards the project. There is not much that can be done if the reason for members not committing enough is work related. People have responsibilities and commitments and it would be wrong to expect them to abandon or postpone them for the project benefit. However, the team will have a discussion with the problem member if it becomes apparent that they are not committing due to laziness. Currently, the agreed approach of handling risk is use the resources provided to us, for example lectures, TAs and online guides, to solve technical and resource related risk or challenges. The team will attempt to focus on communication and improved planning as well as have civil discourse for problems relating to relationship and planning challenges. It is agreed that if aforementioned risks cannot be solved within the team, the teaching assistance will be asked for guidance.