Willy The Robot Project Plan

Jeroen van 't Hul
S1139163

Thomas Zwaanswijk
s1089273

Tom van den Noort
s1101124

February 5, 2019

Supervisor Mischa Mol Windesheim Zwolle

Contents

1 Introduction

This project plan describes a project managed by three students of Windesheim University of Applied Sciences. Prior to this group, other teams worked on Willy the robot as well. As such, the already existing code will have to be edited and expanded upon. This project will continue and build upon their results as much as possible.

2 Motivation

This chapter describes the motivation for the project. It describes why it started, what the current problem is and the set goal to reach within the allotted time.

2.1 Context

Windesheim University wants to have a robot which can drive autonomously and interact with guests to provide them with information on open days. Several project groups have been working on this goal prior to this project, creating and/or completing various modules that work together to realize the autonomy and interactivity of the robot. The goal that this group wants to reach will be described below and continues from what the previous group(s) have left behind.

2.2 Problem

The main problem of the robot in its current state is the autonomous driving capabilities. The robot is able to drive autonomously to a certain degree, but it lacks a fluent and safe navigation system. Next to this the overall impression of the robot is messy. The cables are not properly managed, sensors and controllers are loosely attached. Mechanically the undercarriage is not stable and one wheel rubs against the frame. These and other flaws pose a potential safety risk to the robot and others nearby.

2.3 Goal

Out of the problems stated above, autonomous driving has been chosen to be the core focus of this project. The challenge is to improve upon the driving system(s) to make sure the robot will be able to drive autonomously without posing a threat to itself, the environment or others. Currently, the scope is limited to T5 in Windesheim, but the ultimate end-goal is that Willy can be used anywhere without issue. Additionally, the documentation of the project itself needs to be corrected and maintained, giving future groups the ability to pick up and continue the project without any delay or misunderstandings.

2.4 History

This is not the first project group to work on Willy. At this point in time, there have been 4 other groups to work on the robot. That means that there is a lot of pre-existing code and functionality, all of which must be considered and either re-worked or worked with. Another curiosity is that Willy started off as a completely different product. Originally, it started as a garbage collection robot, but somewhere during development of Willy the company responsible for Willy handed it over to Windesheim, who chose to turn Willy into a robot greeter to be used during public events. As such, the choice was made to focus on autonomous driving, with a sub-goal to reduce the clutter present on Willy and improve the way the sensors are mounted.

2.5 Parties, Roles and Stakeholders

Currently, the stakeholders are as follows:

Name	**Role**	**Contact Info**
Mischa Mol	Product Owner/Coach	mc.mol@windehseim.nl
Thomas Zwaanswijk	Project member	thomas.zwaanswijk@windesheim.nl
Tom van der Noort	Project member	tom.vanden.noort@windesheimflevoland.nl
Jeroen van ‘t Hul	Project leader	jeroen.van.t.hul@windesheim.nl

3 Project

In this chapter the project will be described in further detail, such as the scope and the deliverables.

3.1 Scope

Every task that will take the project group closer to the target goal can be considered something that is in scope of this project. This is something that can be considered a grey area in practice. It is not always that easy to know how relevant a certain task is, therefore the team needs to be alert that a certain task does not go out of scope. To help with this, the relevant goals of this project will be described below.

- Safety for the robot and its surroundings
- Improved autonomous driving
- Mechanical improvements to the structure of the robot
- Electrical improvements to the components of the robot
- Well-maintained, simple documentation

Any tasks that do not contribute towards one of these goals is considered outside of the scope of the project to the discretion of the group.

3.2 Deliverables

- Project plan
- Technical design document
- Research report into current navigational abilities
- Final report
- Updated code & Documents

4 Methodology

For this project, an agile methodology is very convenient. There are of course multiple options when it comes to working in an agile manner, but as the entire group already has experience with SCRUM, that will be the method used. The sprint duration will be two weeks, with daily stand-ups being held every day.

4.0.1 Daily Stand-up

Every morning as soon as every member is present a stand-up will be held. During this session every member tells the others what tasks they worked on yesterday, what tasks are going to be done today, as well as any anticipated challenges. The purpose of this meeting is to ensure every member of the group knows what the others are doing, and allow them to offer accurate advice on current issues.

4.0.2 Sprint review

A sprint review will be held at the end of every sprint, which means that this will be a bi-weekly event. This should include a demonstration or presentation about the events that occurred in the last two weeks. The product owner should be invited to every review, so that feedback from the product owner can be incorporated into the next sprint planning.

4.0.3 Sprint Retrospective

The retrospective offers the group a chance to consider what went well and what went poorly each sprint. Therefore, this should be held after the review, but before the planning where possible. The retrospective should be kept short and simple as much as possible, since after it the planning needs to be held. The retrospective will be done through Trello, creating lists of what went well, what went poorly, and what helped along the way. It ends when all cards have been discussed and any issues brought up have been resolved.

4.0.4 Sprint Planning

After every review, a sprint planning is held. The goal of this is to select the tasks which have priority and move them to the appropriate sprint backlog. Any work that has not been finished is moved back to the backlog, and re-evaluated. Any sprint planning beyond the first should start after the retrospective is done, preferably around noon, and finish at 4 o'clock. After every feature has been discussed and supplied with story points, the planning is finished.

4.1 Daily & Weekly Activities

Certain activities are held on a regular basis. These include the agile standards, weekly meetings and reports, as well as daily gatherings.

4.1.1 Daily Stand-Up

As has been mentioned before, a stand-up will be held every day. They are described in this section 4.0.1

4.1.2 Weekly Meeting With Coach/Product Owner

Especially at the beginning, it is important to meet with the coach Mischa Mol quite often, as a lot of work has already been done by previous teams. In order to ensure that the main goal of the project is not missed, every week progress and difficulties can be discussed. This allows feedback from the coach in addition to that given during the sprint review 4.0.2.

4.2 Quality Management

In order to ensure quality in both code and documentation, several measures have to be taken.

One of these is having to have at least one person review any code and documentation before it can be merged into a development branch. This means that errors are more likely to be caught, since they are generally more obvious to people who did not write a section.

Another method that will be used is code standards. These can be found in appendix A.1.

Code testing must be done by at least one person, who also then writes down the test results. This allows for better transparency in what went wrong and why, resulting in easier bug-fixing.

4.3 Tools

Not every tool will be supported in this project, as a lot of decisions have already been made by previous teams.

- Whatsapp - Communication within the team
- GitHub - VC remote and reviews
- Trello - Planning boards and reviews
- Google Calendar - Keeping track of meetings and planned absences
- Robotic Operating System - The system used to control Willy, and what must be programmed in
- UMLet - Free and open source tool for drawing diagrams, works well with GitHub for version control
- Git - version control
- MikTex - LaTeXdistribution for Windows
- LaTeXeditor of choice - Note that only one member of the team has LaTeXexperience, and only in TexWorks
- Google Drive - file sharing and places for keeping quick notes
- Skylab - control of visuals of robot

4.4 Agreements

- All code must follow the Google Style where it is relevant to do so.
- Branches can only be merged into development after at least one other member of the group has performed a code review and marked it as passing on Git.
- Code version control is done through GitHub
- Any work submitted must follow the Definition Of Done, or DoD.
- General work days are from 9 to 4 where possible, with exception on mondays where PV subjects are concerned
- The target amount of hours per week is 32 hours, regardless of their distribution throughout the week.
- In case of late arrival, should there be unavoidable circumstances it can be ignored
- Every time a group member is late due to their own fault, they must buy coffee for all group members.
- After every third time, they must buy a treat of some description for the entire project group as well.
- Communication is primarily done through Whatsapp or in person when available.

5 Schedule

The following table describes the schedule of the project.

Week	**Activities**
1: 28 January 2019	Introduction
2: 4 February 2019	Start sprint 0
3: 11 February 2019	Start sprint 1
4: 18 February 2019	Holiday
5: 25 February 2019	
6: 4 March 2019	Start sprint 2
7: 11 March 2019	
8: 18 March 2019	Start sprint 3
9: 25 March 2019	
10: 1 April 2019	Start sprint 4
11: 8 April 2019	
12: 15 April 2019	Start sprint 5
13: 22 April 2019	
14: 29 April 2019	Start sprint 6
15: 6 May 2019	
16: 13 May 2019	Start sprint 7
17: 20 May 2019	
18: 27 May 2019	Start sprint 8
19: 3 June 2019	
20: 10 June 2019	Project deadline

5.1 Global Sprint Planning

Sprint 0
Introduction of the projects, team building, contacting stakeholders, project plan, contact with previous groups
Sprint 1
Getting to grips with Willy and ROS (Robotic Operating System)
Sprint 2
Functional design, mechanical design
Sprint 3
begin updating controls on Willy
Spring 4
work on Willy autonomous driving
Sprint 5
Continue autonomous driving
Sprint 6
Continue autonomous driving
Sprint 7
Testing
Sprint 8
Bugfixing + preparation for Winnovation

Any spare time will be dedicated to improving other features, but the scope of those is dependent of whether or not any time is left, and the interest of the group itself. Additionally, the way Willy is currently built leaves a lot to be desired, so in any spare time improvements in the mounting of hardware and cable-management will also be made.

A Conventions

A.1 Code Conventions

Code conventions allow for more readable code. By having every member follow the same standard, it prevents confusion as to naming schemes. The choice has been made to follow the Google C++ style conventions, found at [?].

A.2 Document Conventions

Conventions for LaTeXdocuments are as follows:

- Since \\ is used to end the current line, it is reasonable to do so in the source file as well, but use it only when needed.
- If referencing sources, create a BibTex entry in the .bib file (check this link for examples).
- When referencing an internal section or paragraph, use `\label{}`, with a short abbreviation to note the type of section referenced. E.g. `\label{sec::conventions}`.
- Citations are done in IEEE format, done automatically by LaTeX.
- Put a label under every image or figure for ease of reference.
- Start a new sentence on a new line.
- One root file that contains all sections as a final document.
- Section sources should be in the proper folder, `./sections`
- Images and figures should be placed in `./images`
- The root file (typesetting target) and the `.bib` file should be the only files in the root directory.
- Document names are not capitalized, and a hyphen is used rather than a space.

A.3 Version Control Conventions

The following conventions are to be used when working with Git/GitHub.

- Branch names should follow this format: `subject/name-of-feature`, e.g. `doc/project-plan` or `code/motor-control`.
- `master` and `development` should be protected from pushes. Pull requests need to be reviewed by colleagues that did not work on functionality or branch as much as possible.
- `master` should only ever be updated by a pull request from `development`
- Commit messages should be clear in purpose and necessity.
- Commits should be kept neutral, no personal attacks or callouts.
- Code review must be kept equally neutral and impersonal, focus on the content, not the person who made it.

B Definition Of Done

- Planning is up to date
- Relevant documentation must be up to date. Technical Design must be checked by at least 1 reviewer.
- Functionality must be clearly documented, using class diagrams or other UML diagrams where appropriate.
- If changes have been made to the configuration or other parts of the code, update those sections.
- Documentation and code follow conventions
- All text must be in English, and grammar/spelling errors have been fixed.
- All code is uploaded to the proper branches, approved by at least 1 reviewer.
- Project must build without errors, and be compatible with pre-existing code.
- All functions must be commented in javadoc style.
- Any to-do's in code and documentation have been addressed properly.