The University of Edinburgh

System Design Project

Process Report

Group 8

s1409854 - Filipe Wang
s1436084 - Vesko Stefanov
s1438974 - Anthony Gallagher
s1402967 - Andra Zaharia
s1421803 - Alex Wilson
s1453013 - Lazar Lazarov
s1418923 - Danial Khastalani

February 27, 2017

1 Introduction

This document is the Process Report for the Group SD-8P (Group 8). It includes an overview of how the group is structured, how it communicated and how it formulated plans for this particular group project. Throughout the report we will use the terms team to refer to Team D (composed of Group 7 and 8), group (to refer to either Group 7 or 8) and subgroup (to refer to the smaller groups that were formed around certain tasks within each group).

1.1 Tasks

The first order of business was to split this colossal task of getting a robot to play football into a smaller subset of tasks. As a team, there was a discussion to determine what different tasks there were and, following that discussion, five main tasks were created. The group then proceeded to split the tasks by simply asking who wanted to do which task. The teams originally assigned to each task were as follow:

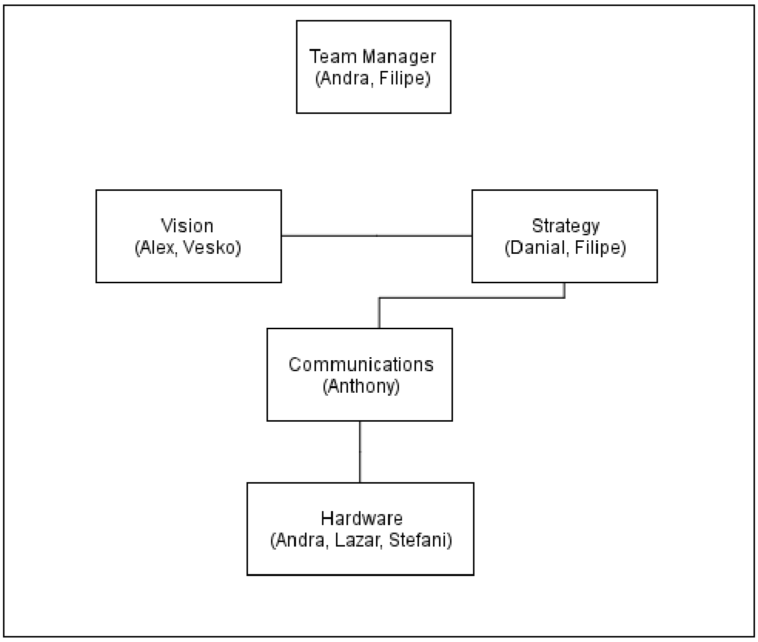

Figure 1: Diagram with Tasks Assigned

Team Manager Andra took the role of lead Team Manager while Filipe would help in anything that was needed. The Team Managers would be responsible for ensuring appropriate communication with the mentor. Examples of other task that were part of the Team Managers responsibilities are:

organizing weekly meetings, making sure everyone is doing an equal amount of work and keeping track of everyone's work.

Vision Alex and Vesko both volunteered to do the Vision part of the assignment. The Vision team would be responsible for building a world model using the camera feed and passing that to the Strategy team. Examples of tasks that the Vision team had to accomplish for are: calibrating the camera feed from the pitch rooms, manipulating the images coming from the camera feed and analysing the processed data from the camera feed.

Strategy Danial and Filipe decided to do the Strategy part of the assignment. The Strategy team would be responsible for building a strategy that the robot can use during friendlies and the final day. Examples of tasks that the Strategy team would have to do are: communicating with the different teams to understand how each part of the project is going on, developing a strategy for the robot and coordinating with the other group to further progress both robots.

Communications Anthony chose to do the Communications part of the assignment. The Communications team would be responsible for transferring messages between the computer and the Arduino using the RF stick. This task involved many smaller task such as: testing the RF stick, implementing a protocol to send/receive messages and ensuring the messages are being processed correctly on the robot.

Hardware Andra, Lazar and Stefani decided to do the Hardware part of the assignment. The Hardware team would be responsible to design and build the robot. They would also have to: implement the different instructions that the robot would have to follow, think of a kicker/grabber mechanism and find the needed parts to build the robot.

There have been no movements between teams but Stefani is no longer taking SDP so the hardware team has been reduced to two members. At many times during the project the different teams also help each other in their respective parts (so no singular team is working only with themselves).

2 Communication Channels

Communication is key when undertaking a project of this size. At the first meeting it was decided that we would meet at least once a week to discuss how everyone is coming along in their roles. We set up the communication channels as such:

Facebook Messenger Most of our information would be sent through the Facebook Messenger. It is by far the most convenient channel of communication and the one that people check the most (so any messages sent here would be hard to miss). We also communicate with the other group through this channel of communication.

Slack Any formal communications between subgroups, questions to the mentors or talking to our own mentor would be done through Slack. Slack is the only platform where all the participants/organizers of SDP are, so any questions regarding SDP would be posted there.

Asana Asana is a web application designed to help a team track their work. We used Asana to monitor and track various goals. Each subgroup had their own agenda leading up to the different deadlines and each member could check Asana to see what another subgroup was up to.

Team Gatherings Generally whenever a team had to work on certain tasks they would meet together. In these gatherings they would work together in their daily goals and would share information between the present members of the team. There were also weekly meetings where the team would meet with the mentor.

GitHub The team's GitHub repository also serves as a channel of communication since people write a message describing what they did when they commit to the repository. This way, by checking the commit history, one can have an idea on what the different teams were doing on a particular day.

3 Task Allocation

The majority of the tasks allocated were done by the individual subgroups and not as a whole group. The individual subgroups mostly came up with daily tasks as they saw fit (this will change after the second friendly as from then on we will start working with Group 7 and we are planning to set goals together).

4 Progress Tracking

Asana We tracked the progress of the team with Asana. We used its functionality to create tasks and assign them to ourselves or one another; add due dates; split the tasks across parts of the project; and mark tasks as completed. As completed tasks would still be visible, we are able to track previous progress and plan our future tasks. We decided to split the project, in Asana, into Communications, Hardware, Strategy, Vision and Team Meetings (for the Team Managers).

Planning At the onset of SDP, we held a meeting where we identified the subtasks for each part of the project. We used those subtasks to create a Work Breakdown Structure diagram (refer to Figure 2). The structure would be later edited with new or modified subtasks.

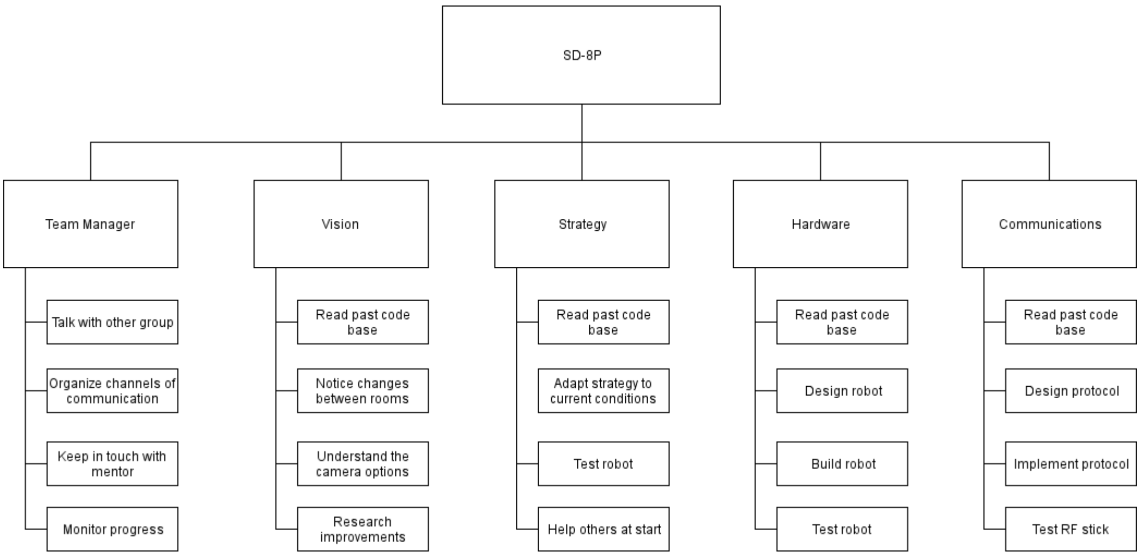

Figure 2: Work Breakdown Structure diagram.

In the beginning of a working session, people who work together set goals to complete. At the end of the session, they would assess their progress and discuss a plan for their next meeting. This gave us a clear idea of which tasks have been completed in a given section and which tasks to tackle next. We update Asana and the Work Breakdown Structure to show the current progress.

Comparing We spend some time assessing the functionality of other teams' robots. This gives us a realistic idea of where we stand among all teams and the opportunity to learn new techniques.

Milestones We created a Gantt chart (refer to Appendix C) to depict the constraints between the subtasks and estimate the completion time for our milestones. Thus, we can easily separate parallel and serial subtasks and identify slack time. Our aim has been to stick to the Gantt chart schedule and reassess it when there are substantial variations. While each subtask is a minor milestone, we have been working towards getting optimal performance for each of the matches.

5 Risks

Code loss/Git Pieces of code can be lost due to deletion or modification. They can also be corrupted due to a recent change. We use Git because it allows us to go back in time to restore the previous state of our lost/corrupt code or just observe the differences to manually fix a issue.

Teammate absence/User Guide Some members of our team may be absent for long periods of time due to unforeseen circumstances. Their absence may cause a halt of the progress of the production process if an error has occurred. To safeguard against this, we have created a user guide (i.e. README) that would provide a brief description of the developed system and instructions for solving common errors. On the day of the match we have at least one additional member present to fill in for an absent operator or handler. Our project managers track everyone's progress and are able to perform the task of any missing teammate.

Issues on match day Our main battery pack may discharge faster than expected on match day. We keep a second battery pack charged for emergencies on match day.

Late Delivery In the case of late delivery of one of system's main parts, we would use the last known functional version of that part. We look to minimise the impact late delivery would cause on the other teammates and the part they develop.

Task specific risk assessment Each group member, periodically, has recorded a personal risk assessment and contingency plan in the group's log by answering some pre-defined questions (refer to the appendices for excerpts of the group's log). Members from different subgroups mentioned different risks. A member of the hardware team, for example, considered some potential risks like the speed of the robot being too slow or cases where the robot does not respond to any message. They then developed a contingency plan around that.

The log provides a way for each member to assess the risks associated with their tasks and also allows others to check on the rest of the group members.

6 Budget

Like every group, Group 8 had a budget of 100 to spend. Here is how the budget was spent thus far:

- Blue LEGO 16x16 plate (x4), £7.32
- Black Technic Axle 10 Threaded, £9.28
- Helix 75 Rubber Bands, £2.46

7 Team Plan

There were some initial communications between the two groups from Team D to try and determine how we should approach the project as a whole. It was first decided that each group should act independently (while sharing information where appropriate) until the second friendly match, where thereafter we would reformulate our plans and strategy for the future matches. From the initial communications both groups settled on using the Python codebase from previous year. After the first friendly match, both vision subgroups from each group decided that it would better to work together to work on improving the current vision system.

8 Conclusion

In face of such a large task the group has kept itself on track in an organized, and relaxed, fashion. Expectations were high for the first friendly, but due to several different factors the robot did not perform as expected. The second friendly allowed us to see our progress in comparison to other teams and, although we did not win any matches, we did see significant improvement in the robot's performance.

Appendix A - Periodic Questions

Andra's Log

What is your role? Do you share it with someone?
My role as part of the hardware team (consisting of two people) was to aid with the robot construction (and reconstruction) process.

What was your plan in the beginning? (Describe as a process of consecutive steps because the plan can be used for a Gantt chart)
Our main goal during the first week was to build a robot that is both modular and sturdy, that has at least a functional kicker built into it. But to do so we had to:
- firstly decide which wheels to use, which meant collaborating with the software team as well, in order to pick the code base that they would like to work with
- decide on the motors to be used and whether they would need gearing or not
- considering the spatial layout of the Lego blocks so that the design would be modular
- beginning to build the robot
- fitting the main motors for the wheels
- understanding how the circuitry works, so that we can consider the layout on the "long-term"
- designing a kicker
- fitting the kicker and a motor for it
- fitting the circuitry
- checking that everything works
- implementing some basic communication commands for the Arduino (such as go, turn, stop, reverse, kick)

What could go wrong with your plan?
Apart from technical difficulties, such as cables or connections not working, other things that could have gone wrong have to do with:
- gearing not providing enough torque or speed; changing the gearing could imply a change in the robot layout as a whole
- not appreciating the spatial layout available correctly and end up building a robot that is too large or bulky; this would mean that the robot would most likely need to be disassembled and reassembled, which is rather time consuming
- different motors providing different powers to each wheel; the consequence of this being that the robot cannot move in a straight line if the motors are both operating at the same power
- the weight of the robot; if the robot is too heavy then more power is required in order to get it moving
- the required power in order to run the kicker - higher powers would drain the battery levels faster
- the strength of the kicker (or rather the lack of strength of the kicker)

How would you have fixed each of the things that could go wrong?

Appendix B - Normal Log

Vesko's Log

- 20 Jan - Ran the vision system software from both implementations - visionwrapper.py for Craig the Robot (CtR)
- 22 Jan - Read Gui Features in OpenCV
- 24 Jan - Read 'Preprocessing' folder in CtR
    - 24 Jan - Read Open CV - Camera calibration
    - 24 Jan - Read OpenCV Background Subtraction
    - 24 Jan - Read Open CV Changing Colorspaces
        - --> Object Tracking/extracting a colored object/
        - (see later chapters for removing noise)
    - 24 Jan - Read Open CV Histograms - 2: Histogram Equalization
        - --> Is there a reason to use adaptive histogram equalisation in preprocessing
- 25 Jan - Read Equipment Guide's Fast and Inexpensive Color Image Segmentation for Interactive Robots
    - The software system is composed of four main parts:
        - a novel implementation of a threshold classifier,
        - a merging system to form regions through connected components,
        - a separation and sorting system that gathers various region features, and
        - a top down merging heuristic to approximate perceptual grouping.
    - A key to the efficiency of our approach is a new method for accomplishing color space thresholding that enables a pixel to be classified into one or more of up to 32 colors using only two logical AND operations.
    - *Much of the information in an RGB image varies along the intensity axis, which is roughly the bisecting ray of the three color axes.*
    - 2.2 Contains representation of YUV conversion
    - 2.3-2.5 Two stages for implementing connected regions
- 25 Jan - Read Equipment Guide's Fast and Accurate Vision-Based Pattern Detection and Identification
    - the document, supposedly, presents the derivation of an efficient and highly accurate detection algorithm for many different patch-based patterns
    - p.4 col.2 Using vectors to establish the plate's dots location
- 26 Jan - Watched Working in Groups or Teams by Jane Hillston
The following text is an adapted summary of the video presentation
    - We are advised to use *agile methods,* which
        - recognise that the initial search for route to solution may be suboptimal
        - aims to reduce the (time) penalties incurred by changes of route to solution
        - involves continuously evolving requirements, frequent small releases and continuous testing
    - We are advised to use *pair programming*, which means that 2 people work in a group. One writes the code and the other thinks about how the code works and considers improving the code's strategy. Thus, while writing the code, its weaknesses are revealed and fixed.
    - The need and methods of communication will change during the project

Appendix C - Sample Gantt Chart