

Project Proposal

Robo Ducks

Project Name	Roboducks
Project Leader	Florentin Gewessler
Document state	In process
Version	V. 0.1

Revisions

Date	Author	Change
November 03, 2011	P. Bauer/T. Stütz	Template
November 06, 2018	Gewessler/Gaisbauer	First Version

Contents

1	Introduction	3
2	Initial Situation	4
3	General Conditions and Constraints	5
4	Project Objectives and System Concepts	6
5	Opportunities and Risks	8
6	Planning	9

1 Introduction

The goal of Roboducks is to participate in the German open Standard Platform League (SPL), where 5 Naos play soccer against other teams. All teams which play there are from Universities. Currently there is not a single team from Austria. That's why it would be great to take part in the German Standard Platform League, which also would be very good for our school's reputation.

2 Initial Situation

Currently there are many teams which participate in the German Open Standard Platform League but none of them are Austrian. Furthermore, like we mentioned earlier, the teams are all from universities and we would like to change that. Like all other teams we will have to develop our own framework because the NAOqi, which the robots have already installed, has too few functions and reacts too slow as we would require. Additionally, in the previous generations of the robotics team were not enough people and had too less resources (robots, money...) to build a robot team which could participate in the German Open Standard Platform League. Also we would like to mention that the previous generation build a system which worked well for one specific task but it was very difficult to add a new function. With our new system it will be very easy to add new features and functions which we will explain in more detail later. A small thing we would like to mention is that this year we will work even more often in the Fabasoft. This is a good thing because we noticed last year that we have worked more efficiently when we were there. Another important fact is that we switched from the programming language Python to C++. C++ has a lot of benefits over Python. For example it is way faster and is in our opinion easier to read.

3 General Conditions and Constraints

In our project we will have to face the other teams in the German Open Standard Platform League. But we have a long way to go till we reach this point. First we have to deal with the hardware and the software conditions which means we have to develop our framework to program and control our robots. Then we have to deal with the "physical" conditions. This means we have to learn our robots how to move, how to kick and even how to see. Actually we have to develop all minor functions and movements from the ground up which will cost a lot of time.

The most constraints we will have to deal with will be hardware constraints. Because the robot has only limited hardware like a weak processor or only 1 GB of RAM. So we have to develop our framework as resource saving as possible to keep up good performance. Also the motors of the joints are rather weak and overheat easily. But there are also some other constraints like we have only limited time for development on our project because of other important things like school. Other constraints are financial nature. At the moment we have through our sponsor Fabasoft 10.000 Euros a year. But as we know from the teams of the universities is the development a very expensive thing. And some of our competitors have five times the financial resources.

4 Project Objectives and System Concepts

Our goal is as mentioned before to be part of the German Open standard platform league. We know this is a tricky goal but we are determined to achieve it. In the end it should be that our robots have different strategies and decide which strategy to use according to their situation. The robots should find their positions reliably. For example our goalkeeper will have certain abilities which he will use, when he sees that a ball is approaching the goal.

Our system concept is based on our framework Duckburg. This framework is composed out of the brain, the engines and the agents.

The Brain is the central part of the Framework, even though being the simplest component upon the three. It works as the junction for the communication traffic of the Engines. The tasks of the Brain are only to provide Engines with the possibility to subscribe to Slots and send Intents to them. Also it is the keeper of the dictionary that contains Slots and Filters. The main focus of the brain is to minimize its logical effort and therefore maximize communication possibilities for Engines.

Engines are the components of Duckburg that do the thinking. But their basic functions are send intents to the brain and they subscribe to slots. An Intent contains a command that should be executed. After an Intent is sent to the Brain it runs through a number of Intent Filters that sort out illegal Intents. Then the Brain forwards the Intent to the correct slot. If the Slot that the Intent is sent to doesn't exist, it is created. When an Intent reaches the Slot it is sent to all Engines that have subscribed to this Slot. An Engine subscribes to a Slot by specifying a method that will be called when an Intent is sent to this Slot. As soon as an Engine subscribes to a Slot this Slot is created if it didn't exist before. The key point of Engines is that they don't really "do" anything but that they "think" about and react to things.

Agents are components of Duckburg that really make things go... literally. They are invoked by their Engines and then perform the task they are made to do. Examples for Agents could be: An Agent:

- ... that walks
- ... that calculates the position of an object
- ... that writes a log entry to a file
- and so on

This way we can develop functions without changing the whole framework.

5 Opportunities and Risks

This project gives us the opportunity to be part of the German Open Standard Platform League. This would be a big thing for our school because the other teams are all from universities. It would also be a good thing for our sponsor the Fabasoft because we would be very present in the media. But this comes with a risk. It is possible that our robots get broken when we be part of game with them and that would cost much money.

In our project we have some risks we have to take account to. For example, like mention earlier, our robots are not that robust and it is possible that some parts of them get broken while we test them on something. This is not a major problem because we have a maintenance contract with Aldebaran but while the affected robot is France we cannot work with it.

6 Planning

In this section we are going to explain our plan, how we are planning to develop our Robo Ducks project. First we will clean up our robots, because over the past few years there was data uploaded which is no use to us anymore. But of course we are going to save that data on a cloud. Then till Christmas we are planning to write the basic Framework, which will enable us to implement further functions specific to the football match. We do not believe that we will be able to take part on the german opens this year. But we will do our best.

Our major milestones:

- Basic Framework (Christmas) 24th December 2018
- Basic Behavior (like detect the ball and kick it...) 21st April 2019
- Working team communication and team play 15th November 2019

Our team roles:

- Team leader: Florentin Gewessler
- Programmers: Jakob Wögerbauer, Lukas Gaisbauer, Georg Sengstbratl, Christoph Knoll

Resources:

- 6 robots
- 1 accesspoint
- lan cables
- plog affords and power supplies
- When will the project end?
- This project will always be continued.
- When will the project start?
- It allready started.
- When will be a first prototype available?
- Hopefully at about easter.
- When does implementation work start?
- In the next few weeks.
- What are the big blocks of work to be done?
- The framework, the behavior, the planning, the testing
- Is this work doable in the given period of time?
- When we are finished with the framework till christmas then yes.
- Do we need any other stuff to make our work (licenses, servers, É)?
- Yes, a cloud to safe the old data from the naos and stroe our projekt things,
We could use GitHub or Microsoft OneDrive.