Bachelor Project - CAPTAIN
Process

Bachelor - Electronics Engineering
7. Semester project Fall 2017
Group: 17115

Aarhus University, School of Engineering
Supervisor : Michael Alrøe

19. December 2017

CAPTAIN

Nicolai K. Bonde
Studienr. 201405146
AU ID: au512366

Troels Ringbøl Brahe
Studienr. 20095221
AU ID: au283169

Contents

1 Preface

This documents details the development process of the Boat Autopilot "CAPTAIN" bachelor project, developed at Aarhus University School of Engineering. The report was written by Troels Ringbøl Brahe and Nicolai Kjærsgaard Bonde. During development of the project, both were studying Electrical Engineering.

Along with this process report, a project report has been written, as well as documentation, meeting schedule, and additional items in the appendix.

Associate professor Michael Alrøe was the project supervisor.

The project will be handed in December 19th 2017, and will be evaluated at an oral exam on January 17th 2018.

2 Introduction

The process report details the development process for the project. Throughout development, different methods were used to structure the work and increase efficiency.

The ASE model 1, seen in section **??**of this document on page **??**, has been used by the group members in all previous projects, and was used again in this one. Elements of SCRUM and Agile development were used to structure sprints and in making decisions on which module to development next.

This process report gives an outline of how the different strategies were used throughout the project, and where we deviated from them.

3 Group formation

For the bachelor project, it was each person's own responsibility to find a partner or group to be a part of. Additionally, it was required to find a project from the list of projects in the university intranet, which companies had sent to AU. This resulted in students finding someone they perhaps had worked with before or knew from another class, and picking a project from the list they would be interested in working on.

For our group, we had worked together a year earlier on another project, an autonomous car with several sensors. Since we both had interest in autonomous vehicles, and had taken an autonomous robotics course in the spring of 2017, we decided to pursue the Boat autopilot project put up by EIVA.

Despite being Electrical Engineering students, we were and are more interested in software development than circuit design, and since this project turned out to contain mostly software, this seems to have been a good fit for us both.

4 Cooperation agreement

For this project, no official cooperation agreement was made. We had used them on previous projects, and found little to no use for them, even in larger groups.

Instead, we decided to treat the development process like a regular engineering job, working 8-16, breaking this informal rule only when going to other classes, or towards the end of the development cycle, when much of the work was writing and documenting the finished project.

5 Development process

5.1 ASE model

This project was developed using the ASE model, like all our previous projects at the institute. Figure 1 shows the structure of the model.

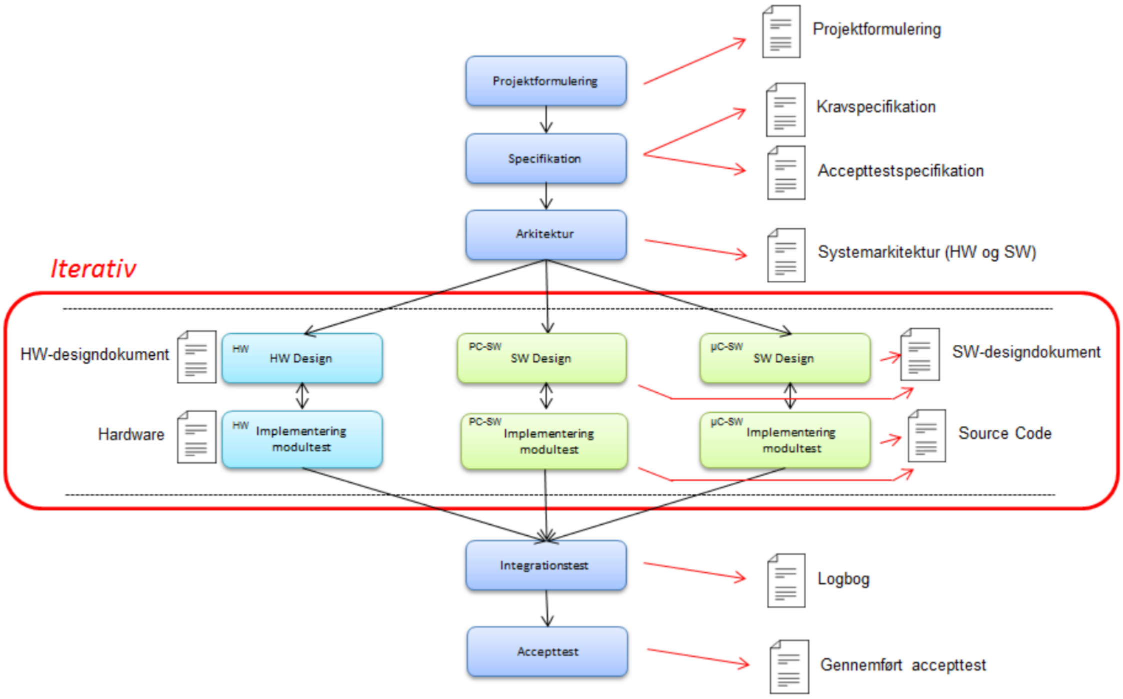

Figure 1: The ASE development model

As seen, the project development proceeds linearly from problem definition to requirements specification, and then architecture.

After this, development begins, and the system is designed and tested iteratively. When all components have been developed and verified, integration tests are run to verify that the components work together.

Finally, the acceptance test (written after the architecture has been completed) is run, and it is verified that the system behaves as expected from the requirements specification.

Using this model is very intuitive after having used it in several projects, and it ensures that due diligence is exercised instead of jumping into development prematurely.

5.2 SCRUM

SCRUM is an agile development process that has seen successful use in a wide variety of organizations for the past twenty years or so.

A modified version of SCRUM was used during development. Since the team consisted of only two members, a full SCRUM approach seemed unnecessarily time-consuming, so only the elements of SCRUM that were deemed relevant were incorporated into the development process, such as Artifacts, and Time boxes.

During certain parts of development it was essential that certain modules were finished before others could be started, which naturally led to several "mini-sprints". A batch of tasks would complete, be reviewed, then be finished, and new tasks were possible to begin, building on top of the finished ones.

In this project, both members of the team were Development Team and Product Owner, as well as SCRUM Master, and meetings were held rather informally, usually after modules finished, and there were problems to discuss and solve.

6 Project management

Management of project development was shared between the two group members.

After the initial stages of problem definition, requirements specification, and architecture, the design tasks were broken down and each member started developing more or less independently until next review/completion cycle.

The team figured out which tasks seemed most relevant continually, and delegated the work based mostly on this to ensure that work could progress at a steady pace.

7 Work distribution

For managing the project we used a scrum like task board. First off the task board was on a physical bulletin board, but later it was moved to the online tool Trello [**trello**].

The task board was divided up into 5 sections; Backlog, doing, review, reviewed, and done. The backlog is the place where all the tasks for the project reside, that is every thing from implementation task, to documentation or research. The doing section is used to signify that a task is currently being worked on, this minimizes the risk of parallel work. When a task in doing is done it is move to the review section, where the other group member looks at what has been done and gives comments. After review the task is moved to the reviewed section, where the original owner of the task decides whether it goes back in the backlog for further work, or it goes to the done section.

Even though the task board is a scrum like board, we did not follow normal scrum. The entire project was more or less one sprint, that means that if one wants to work on a task in the backlog the person can just go a head.

The distribution of task is rather simple, if you find a task interesting or feel like doing that task, then you just put it in doing and its yours. A lot of time tho before a task has prerequisite tasks, that need to be done before it can be completed. That way the less interesting task also gets done. In the beginning of the project we also used the task board, but a lot of the task were done together.

This way of distributing the work task of the project means that, if one want to learn something new by working on a task in the unknown one can do it. But if schedule is tight maybe it makes more sense to work on a task that you are comfortable with, so i might be done a bit quicker.

8 Planning

The planning of the this project has been rather loose. We didn't make any kind of Gantt-diagrams or other kind of schedule. Early on we did use some time trying to figure out what the project would encompass, and we made a MoSCoW, to help prioritize the project and our time.

We have had two major deadlines that we have worked up toward. The first one was for some hardware design, that we never did end up completing, or barely working on, therefore instead some off the shelf components were bought. The second deadline was the deadline for being done with all documentation and writing in general. This deadline also dictated when the product needed to be finished since an acceptance test needed to be completed before everything can be written.

In the end we ended up maybe being lucky that the project got completed, since we did not plan what to work on when. But it might also just be because, we learned to prioritize on the fly throughout the project.

9 Project administration

To administer project artifacts and source code, we have used a git repository. More specifically we have used GitHub[**github**], and their GitHub Desktop application[**github_desktop**]. On GitHub we have 3 repositories; the first one is for the pre bachelor project, The 2 other repositories are for the actual bachelor project. One is called Boat-autopilot and is used for the entire source code of the project. The other repository is called Boat-autopilot-documentation and as the name states holds all of the documentation and other project artifacts.

To keep track of what was done each day of the project, we started out writing logbooks. This worked very well when most of the work was done in collaboration. But when the work shifted to individual work, we slowly drifted from used logbooks, and instead moved over to Trello. Trello keeps track of what is done to all task and when it is done. So we found logbooks to be rather redundant at that point.

For the entire project to get in touch with each other, we have used Facebook messenger [**facebook-messenger**]. Which is just an easy way to communicate, both on the computer or on the phone.

10 Meetings

During this project, there has never been any kind of structure to when and how we have been during meetings. Meetings with out supervisor has been few and far between. Usually meetings were arrange by sending an email asking if a time in the near future would be possible. All of out supervisor meetings have been at our supervisors office. They usually involved showing how far we have got, and asking a few questions.

Internal meetings in the group has been very informal. In the beginning of the project we met almost everyday to work on the project, but the further into the project we got the less ofter we met to work together. Deciding what should be accomplished at the meetings are usually discussed in advanced.

11 Conflict handling

When we started working on this project, we created a requirements specification, for the project that we thought was the one we were supposed to make. It was based on a meeting with EIVA and the problem description that we got in the pre bachelor project. When we sent this requirements specification to EIVA which we worked for at that point, we didn't receive any feedback for week. After this week we ended up calling them and not getting a response the first time. When we finally got a hold of them, they explained that they were very busy at the moment, but that we could arrange a meeting at the "EIVA days" at Aarhus harbour. We then got there in the morning an waited till the afternoon before they had time to meet us. It turned out that there were some major misunderstandings on our part, with regards to what they meant by an autopilot system. This was very frustrating for us, so we ended up talking with our supervisor. We ended up continuing with the project that we specified in the requirements specification, and ended the collaboration with EIVA.

We also had one internal conflict in the middle of the project. This conflict was based around differing perspectives on work planning and managements. We solved the conflict, at a meeting where we ended up discussing it. The discussion ended up in a renewed expectations on how we were to work together.