

Webproject II

Process Report

Isabell Helen Johnsen, Rina Yamada,

Elisabet Therese Bruce Høyer

May 31, 2020

Contents

1	Introduction	2
2	Work environment	2
2.1	Weekly Meetings	2
2.2	Rules	3
2.3	Group Members	3
2.4	Working Plan	3
3	Technology Used in the Project	4
3.1	React environment	5
4	Contributions	5
5	Reflections	6
5.1	Planning	6
5.2	React	7
6	Conclusion	8

1 Introduction

In the course IMT2671 Webproject II, a project assignment involving the design and development of a solution that would meet the product owners' needs was given. The first meeting with product owners was held on January 24th, giving us 17 weeks to complete the task.

In this process report, we are first going to clarify our work environment and working process under the development of the application and then finalize the report with reflections on our process as a whole.

2 Work environment

The work environment was as follows: a shared git repository was used to share the project, and gitkraken was used for management of the work process. The gitkraken addon glo was also used as a to-do list. A Microsoft teams' group was set up for communication outside of group meetings and to share files that were not in the repository, such as the group rules, the pro's and con list for different frameworks, and the development plan. After the university was put on lockdown due to Covid-19, MS teams was also used for online meetings. The project and process report were written using Overleaf.

2.1 Weekly Meetings

At the beginning of the project, we set up two weekly meetings: on Tuesdays and Fridays, with Friday meetings ending with the writing of a quick summary of what has been done that week to be used in the process and progress report. After the Covid-19 lockdown, we had two weeks with three online meetings on Monday, Wednesday and Friday and then eventually went back to two online meetings a week; typically Tuesdays and Fridays. But these days could change from week to week. We also had extra meetings when necessary.

2.2 Rules

On the first meeting, these internal rules were set:

- Don't be late. If you will be late or cannot attend a meeting let us know asap.
- Ask for help when necessary – don't struggle on a question alone.
- Be open about your own personal schedule/availability.
- Participate in planning and decision making.
- If a member does not succeed in delivering what was assigned to them in time (unless problems happened and they informed the group) they have one week to do so. If after one week they still haven't done so it will be noted and after 3 times Carlos will be contacted.

Initially we set a role list alongside the internal rules which looked like this:

Elisabet: Leader, database

Isabell: React lead

Rina: Design lead

Serena: Git/Github Master

The full group agreement can be seen in the attached file gr1-project-agreement-imt2671-2020.pdf.

2.3 Group Members

We initially formed a group consisted of four members, which later became three. After Serena left the group and as things progressed the roles above were not upheld very strictly, and all members have participated in every aspect of the assignment.

2.4 Working Plan

Using a Gantt Chart Template (see figure 1), we created a development plan a few weeks into the project. For the first few sections of the development, we kept within

the frame of the development plan, but as we came to week 13 we were no longer following the initial schedule. The development and implementation of the features took more time than we anticipated, so the database development did not start until week 16 in April, nearly a month behind the schedule. The development and implementation of the database also happened in tandem with further development of features, so in reality the "implement functionalities" section and "implement database" section were both going on at the same time during the last 4 weeks of the project.

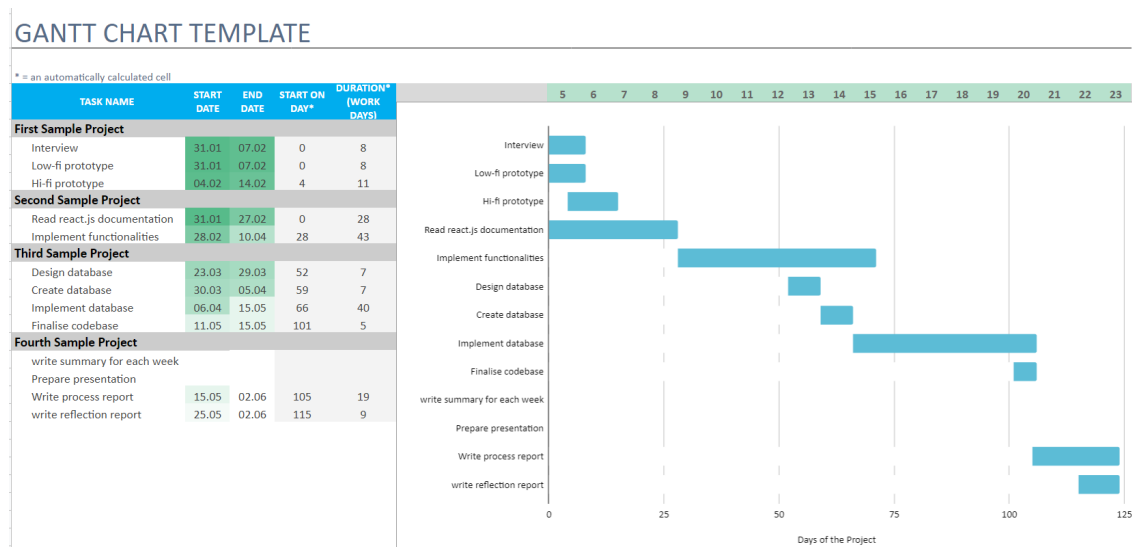


Figure 1: *The gantt chart template holding the development plan created in January. The numbers on the top right half of the chart are the week numbers.*

In this development plan, there was also set off 3 weeks at the end to focus on the process and project report. However, the reports were begun significantly earlier, typically in the form of shorter notes to create a quick draft.

As a result, the final 3 weeks were spent working on front-end, back-end and the reports at the same time.

3 Technology Used in the Project

After the first lecture of the course, we made a list over pros and cons of react, vue, angular and polymer and discussed what might work optimally for the project.

This list can be found in the attached file "gr1-frameworks-comparison-imt2671-2020.pdf".

We came to the conclusion of using ReactJS. One reason for this was the quantity of documentation and tutorials that existed for React. This made it possible to start right ahead with the task. We also had previous knowledge of this framework from fellow students, who often recommended ReactJS. Outside of this particular project, it is a good framework to be familiar with because of its status in the job market and it's seemingly bright future. The design and plan for the set-up of the application process for both students and companies also lent itself very well to a single page application format, which is something React, alongside with React Router, is well suited for.

3.1 React environment

The react environment was created using the command 'npm create-react-project folder-name' in the git bash console. This command set up the basic folder structure and created a template for a quick start. In order for this to work everyone also needed to have the newest version of Node.js downloaded on their computer. The react library React Router was added to the project and used to navigate between "pages" by changing which components were displayed depending on the url.

The back-end was set up using php and AJAX in cooperation with assignment 3 for IMT3851 Programming for Web II.

4 Contributions

During the planning of features and creation of lo-fi prototypes everyone participated equally. The visual design itself was largely planned by Rina. Isabell created several of the graphics present on the page such as the way-in logo and the arrow images. The github repository and the react environment was set-up by Elisabet, including the initial installation and set-up of react router.

As pages were added, whoever created the page added the necessary React Router

connections for it. For the development section of things, we worked simultaneously on several features so that everyone had a chance to learn. When we came together during the group meetings, we used whichever solution seemed best or was the most complete to continue building on. Due to this, the features on the /int_overview page (projectOverview.js and its children), except for the form, were largely a group effort.

Elisabet achieved displaying the list of internships/ bachelor theses with the data stored in the database as well as creating the function that displays all the information on the right side of the page when one of the list element was clicked by a user (the features on the /int_overview page (projectOverview.js and its children) as mentioned above). She also created CSS files and fixed practical and aesthetic issues by using a grid layout. Footer with onClick functionalities was created by her as well.

Isabell created a log-in function as well as developing a coordinator page where it displays a list of companies with data stored in the database. Alongside this the Header itself was also developed by her.

Rina was responsible for the development of the application forms. She made it possible to send data that users fill in to the relevant tables in "way_in_db" together with one's chosen internships/ bachelor projects. Some part of the database structure and tables were also changed when the necessity arose. She also found fonts used for the application.

5 Reflections

There are several things we realized what we should have done or that we could have done better during this project.

5.1 Planning

As we mentioned earlier, the working process of developing the application did not follow the initial development plan. We created a lo-fi and hi-fi prototype within the

planned schedule, however it was not enough time set aside for the implementation. Due to the fact that we were using tools we were not familiar with, such as React, implementing and developing features took longer than expected. In the end we did not get to implement several planned features. The guidance meetings with Carlos and Gioele also were not used to its full potential, and members often struggled with a feature alone instead of bringing these problems to the meetings for help and guidance.

As one of our group member left at the beginning of March, it meant that we lost 14 hours of working hours per week. We should have had a meeting then to plan the process again and discuss which features should be prioritized to develop.

A more carefully planned process and asking for help more often could have prevented us working almost twice as much as the initially planned working hours for several weeks.

5.2 React

We were aware of that file structure and its hierarchy was crucial for React. Due to this we tried to carefully plan the react hierarchy before starting development, however the hierarchy had to be changed several times for certain features to work during the development process.

Being behind of the planned schedule also caused some poor behavior when writing code. For example, the code was not being commented clearly and it caused confusions when a different member had to work on that same piece of code. It also led to extra work at the end as we had to go through the code and make sure everything is commented well. In addition, we should have spent time refining the code each time we finished developing a feature. This would have prevented having to clean up the code after the whole product was finished, which took a lot of time.

Due to lack of familiarity with React.js it was not utilised as it could have. Among other things the component state could have been used more and planned around

in better detail. Using a state container library, such as Redux, would have reduced some of the workload which would have given us more time to spend elsewhere.

6 Conclusion

In conclusion, we have achieved to develop and implement some of the planned features, yet there are a number of features remaining. A gap between our skills and the estimation of time needed in order to complete tasks often caused a delay. Taking time to occasionally look at our progress and our goals, and reevaluate our development plan can help prevent this in the future.