VIA University College

Semester Project: Heterogeneous System

Process Report – The Happy Pig Company Time Logging System

Supervisors:

Jakob Knop Rasmussen

Jan Munch Pedersen

Students:

Gais El-AAsi – 279910

Marcel Notenboom – 279963

Number of characters incl. spaces, images and footnotes 27,186 characters

Third Semester: Software Engineering
VIA University College – Horsens Campus 20th of December 2019

Table of content

List of Figures and Tables

1. Introduction

The project was introduced to us at the begging of the semester on the 4th of September, where we were provided with the information on what is expected from this project, milestones and formal requirements. On the 16th of September after forming after deliberation for a few different ideas, we decided that for this project we will be working with a time logging system.

Next, different milestones followed up, including defining the project description, preparing the basic requirement of the future software and proving that the product is viable. The Software Development of Distributed Systems (SDJ3) class provided us with valuable knowledge needed to meet those milestones in regards to system architecture.

We decided that the best software development framework for us would be Kanban (Radigan, 2019) that allowed a good structured and organized way of developing the project, at the same time, less pushing and complicated than SCRUM (Rehkopf, 2019).

During the Semester, we would work for the project on the assigned days – Wednesdays, though to keep up with the milestones, and ensure that our wishes in regards to the project are satisfied, more work needed to be put in. Therefore, many times we used other days as well to work on the project.

Being a heterogeneous system, .NET development classes (DNP) provided us with great resources of understanding the .NET platform, and how to utilize it to our advantage in the semester project. Many tools and technologies ending up to be crucial for the functioning of the system.

Supervision was carried on during the entire period, many times teachers assisting us in our semester project even during their other classes (DNP and SDJ3), this helped us stay on track and ensure that we are working our best for this project.

2. Group Description

For the new project, all the members of the group have worked together in the previous semesters. This fact helped us find a common language much easier, and therefore many of the hops encountered in the previous project were successfully avoided.

2.1. Group Contract

Even if in the last semester we have all worked together, this semester the Group Contract was completely reworked. The contract was redone, to ensure that it will become more liberal, to encourage and engage all the members of the group into productive communication. It still provided the necessary structure, rules, and work-ethics to ensure that there was a guideline to follow for the members. The group contract was very much inspired by Valve's *Handbook for New Employees* (Valve Corporation, 2012). The group contract can be found in Appendix A.

2.2. Project Experience

The fact that all members have had experience in working in this kind of projects as well as the fact that we have had experience working with each other, helped us establish common unspoken work rules. Moreover, it helped us gain a common understanding of what outcome we expect from this project, how much work is needed to be put into the project and how dedicated should everyone be.

2.3. Group Roles

As stated in the *Group Contract's Section 3 Management, Article 3.3* there are no official roles for any member of the group, though, in case of different roles naturally emerging during the project, the roles were more than welcomed as long as it contributed to the well-being of the project. The roles were not fixed or/and imposed, in addition, they could morph through the project. With that in mind there were certain responsibilities that the group members took over to ensure that the project was developing as intended:

- Levi was responsible for ensuring that the team stays on track with what are the formal requirements of the project. He was as well a motivator and support member of the team many times ensuring that there are no obstacles (not necessarily related to the project) that could harm the progress of the semester project;

- Marcel had the responsibility with quality control, and critique thinking, his assessment of the overall work many times helped us spot issues that would have occurred in the future. He was in charge of the overall direction of the design and implementation;
- Gais was responsible with the management of the work and team, many times acting as a proxy for the team when communication with supervisors, as well as a role of coordination and work distribution;

In general, we have had a great collaboration for this project, with occasional obstacles that were easily solved by communicating and expressing our concerns.

2.4. Member dismissal

As stated in the Group Contract *Section 4 Conflict resolution and member dismissal Article 4.1* a member cannot be dismissed by the group, only in case when the member himself/herself has the decision to leave. In our case, this was the situation when during the period after the Project Description was approved, Levi had to inform the team that he will not be able to continue the work with us because of personal reasons.

After that, the expectation for the project outcome was reworked with the remaining members, as well as the work amount, which was redistributed to ensure that all the milestones were hit.

3. Project Initiation

After we were introduced to the semester project, the first milestone to hit was to decide on a project topic. For finding out what we wanted to work with for the next months, it was decided that every member was allowed to make two proposals. With those in mind, 3 of them were democratically picked and chosen to be elaborated. The proposals can be found in Appendix B.

After a meeting with the supervisors, another voting was made, the decision being for a time logging system. The arguments for this proposal were:

- Can easily be implemented in a distributed system with three tiers;

- Did not have much unnecessary overhead that did not contribute to the development of the semester project or learning process;
- Was realistic – meaning that it was achievable – which can be a great motivator during the semester period;

4. Project Description

As our first main milestone for the project, we had to make a Project Description. The project description had the role to provide a more structured way of viewing what we wanted to do. Some of the details that previously were not worked out were put in place, and after finishing it, we could see a clearer image of what is the problem that we wanted to solve, what we will not focus on, what tools we will use, the schedule, the risks, etc.

The first draft of the Project Description had some issues in the Time Schedule chapter, as we were both overestimating and underestimating some of the milestones that needed to be achieved. Other than that, the project description has been a guiding tool through the entire project development. It helped us stay focused on what was needed to be done. Project Description can be found in Appendix C.

5. Project Execution

During the project execution period, this semester we took the decision to use Kanban Software Development Framework as it fitted better our team and style of work.

5.1. Kanban

Kanban is a Software Development Framework, very similar to SCRUM but with a few differences that make it great for small teams that prefer to work in a less structured way.

5.1.1. Reasoning

The main reasons that made us try Kanban as our framework are:

- A rather small group – 3 people initially, 2 during most of the time;
- Goal-based tasks – the tasks did not have a particular size (hours, points, etc.) rather they were determined by a clear goal;

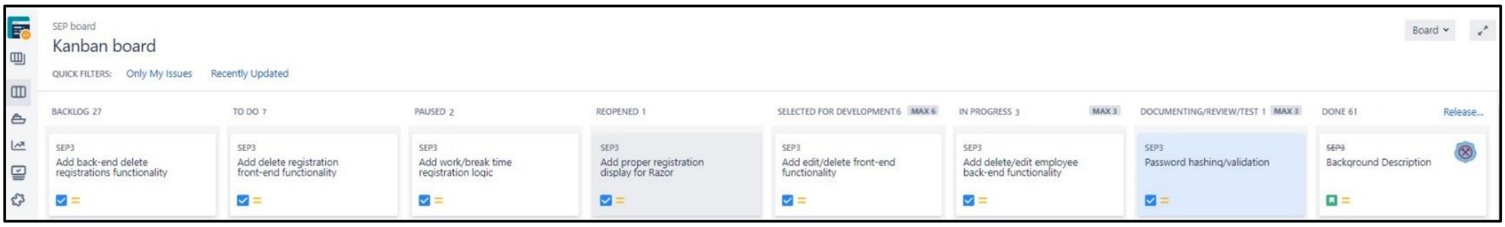
- No roles – there were not any kind of roles needed (ex. SCRUM Master, etc.), the focus was that everyone was contributing to the management of the team, through proper management of themselves;
- Less strict iterations – an iteration ended when a substantial amount of value was added to the system;
- Less amount of work in the management and more into work;

5.1.2. Kanban board

To ease the planning and usage of the Kanban framework *Jira Software* from Atlassian was used as the main Kanban dashboard. The board was made of 8 sections as follows:

- Backlog;
- To do;
- Paused;
- Reopened;
- Selected for development – maximum 6 tasks;
- In progress – maximum 3 tasks;
- Documenting/Review/Testing – maximum 3 tasks;
- Done;

Figure 1 - Kanban Board



SEP board

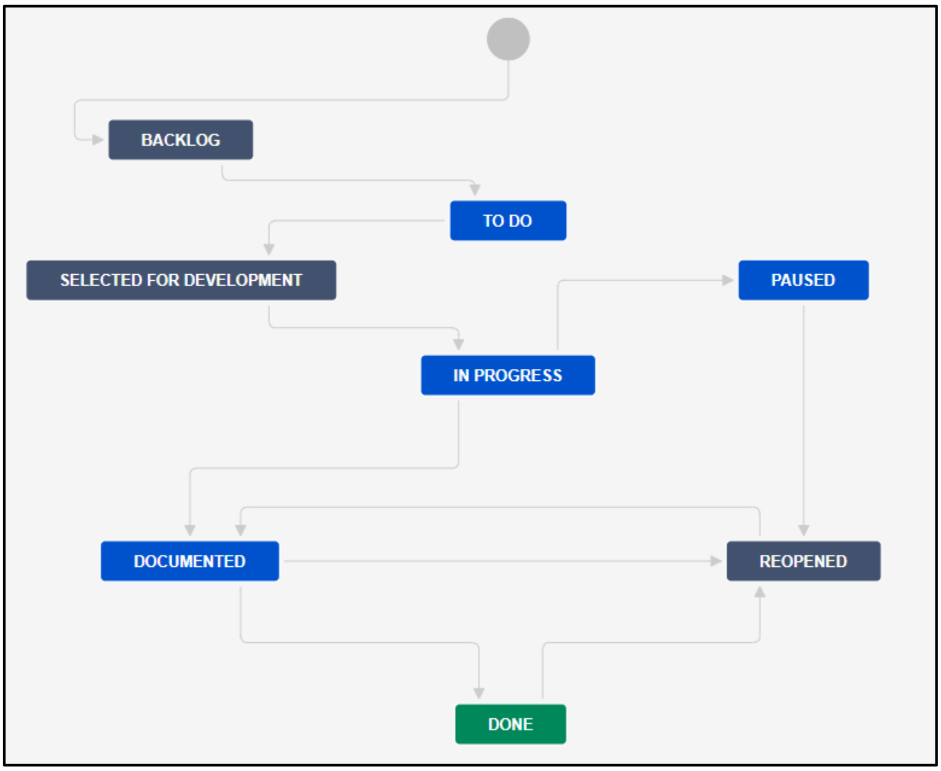
Kanban board

QUICK FILTERS: Only My Issues Recently Updated

BACKLOG 27	TO DO 7	PAUSED 2	REOPENED 1	SELECTED FOR DEVELOPMENT 6 MAX 6	IN PROGRESS 3 MAX 3	DOCUMENTING/REVIEW/TEST 1 MAX 3	DONE 61 Release...
SEP3 Add back-end delete registrations functionality	SEP3 Add delete registration front-end functionality	SEP3 Add work/break time registration logic	SEP3 Add proper registration display for Razor	SEP3 Add edit/delete front-end functionality	SEP3 Add delete/edit employee back-end functionality	SEP3 Password hashing/validation	~~SEP3~~ Background Description

A certain workflow was established, to ensure that a proper flow of the tasks is in place as presented in the figure below.

Figure 2 - Task flow



Jira Software helped us a lot in better organizing our work, with a little setup at the beginning of the project, we used it throughout the entire period.

5.1.3. Backlog planning

A planning session was made every time a low number of tasks (2 or less) were left in the backlog of the Kanban board. The meeting was either online or physical (depending on the situation) where we will take next inline user stories and would split them into more manageable tasks that were added to the backlog.

Sometimes, when a task turned to be more complex than expected, it would be split into smaller subtasks, so that it can be more achievable in a shorter period, to make it easier to track the progress and test.

5.1.4. Iterations

An iteration was considered complete when all of the tasks that were obtained from a single user story were completed. It was important after each iteration to "stop" and assess our progress, look retrospectively and try to draw some critique.

Because the iterations were not time-bounded, but rather task bounded, we did not have a fixed schedule for them, and would usually have the meeting on the next day, giving us some time to individually reflects on the "goods" and "bads" of the iteration.

5.1.5. WBS and WIP

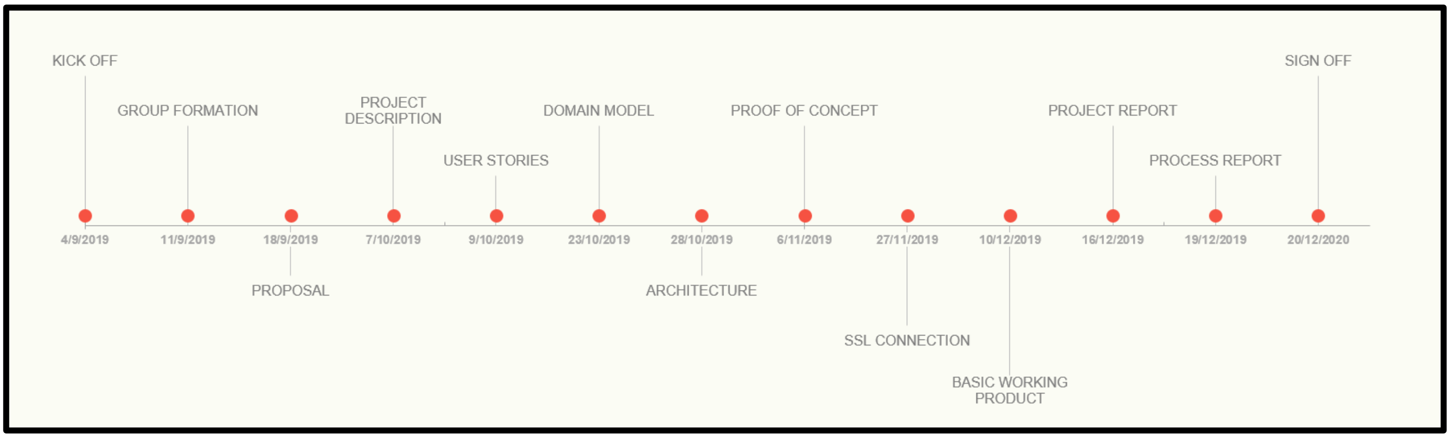
Work breakdown structure principle helped us manage even difficult tasks, by splitting them into smaller and smaller pieces and solving them one by one. Even if, one could say that it should be natural to do so, sometimes having a structure for splitting tasks is a great tool for handling hard to finish tasks.

Work in progress principle limited the number of tasks that we were allowed to work at the same time (In progress column from the Kanban) to 3 tasks at the time. This helped us to guarantee that each task is completed before the next one is started.

5.1.6. Time Schedule

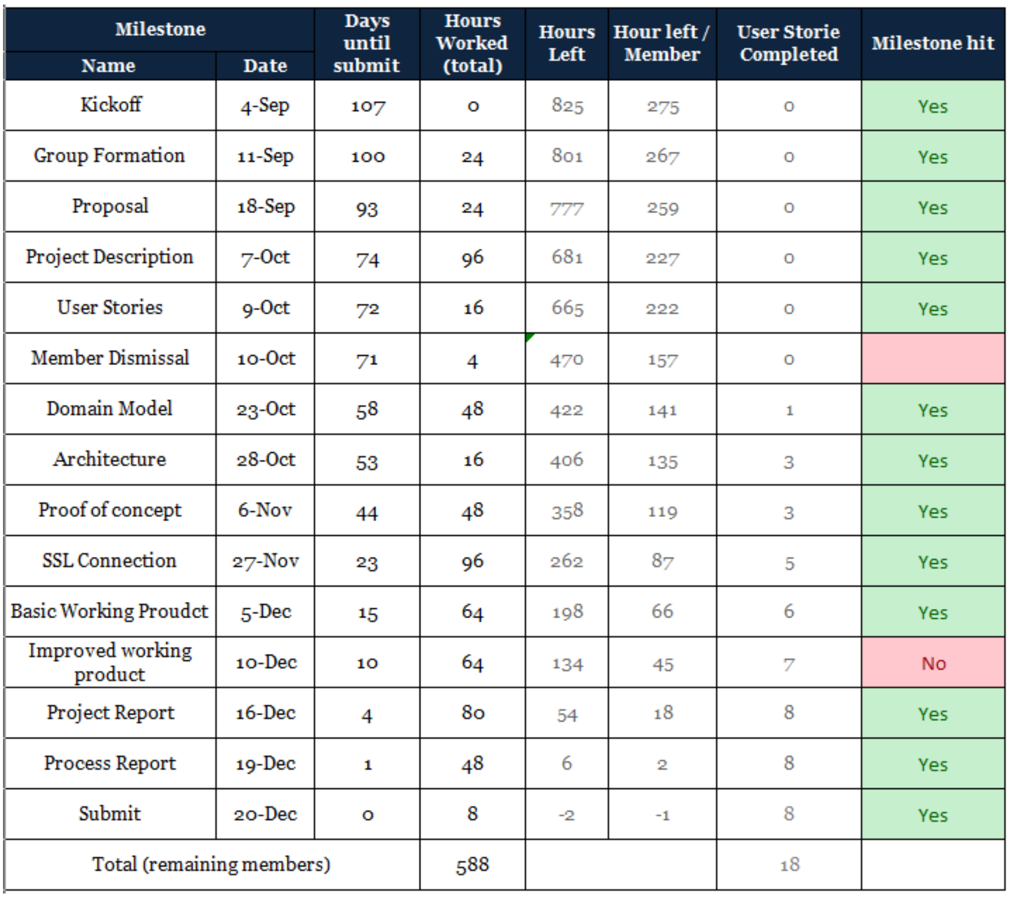
To ensure that we stay on track a time schedule was made with the main milestones that were made in the Project Description along with others developed during the project.

Figure 3 - Schedule



Along with the milestones, a time tracking table/chart was made, *to keep an eye,* on how the work was progressing. The hours were updated at each milestone and it offered a good overview of how much work is needed, how much time is left, what is the next milestone, etc. The Excel file can be found in Appendix D.

Figure 4 - Time reference table



Milestone		Days until submit	Hours Worked (total)	Hours Left	Hour left / Member	User Storie Completed	Milestone hit
Name	Date						
Kickoff	4-Sep	107	0	825	275	0	Yes
Group Formation	11-Sep	100	24	801	267	0	Yes
Proposal	18-Sep	93	24	777	259	0	Yes
Project Description	7-Oct	74	96	681	227	0	Yes
User Stories	9-Oct	72	16	665	222	0	Yes
Member Dismissal	10-Oct	71	4	470	157	0	
Domain Model	23-Oct	58	48	422	141	1	Yes
Architecture	28-Oct	53	16	406	135	3	Yes
Proof of concept	6-Nov	44	48	358	119	3	Yes
SSL Connection	27-Nov	23	96	262	87	5	Yes
Basic Working Proudct	5-Dec	15	64	198	66	6	Yes
Improved working product	10-Dec	10	64	134	45	7	No
Project Report	16-Dec	4	80	54	18	8	Yes
Process Report	19-Dec	1	48	6	2	8	Yes
Submit	20-Dec	0	8	-2	-1	8	Yes
Total (remaining members)			588			18	

We were keeping track of how many days are left until the submission date to ensure that know we are on the timeline. Having milestones helped us ensure that we end up with the base requirements (for the project, not software) completed. Hours left was giving an approximation of how much work is needed to be put as well as how much was done (motivational).

The *Member Dismissal* milestone, of course, was not planned and later introduced in the table, as it was important because we needed to rework the hours, to ensure that we are working what it is expecting from us.

5.1.7. Critique of Kanban Framework

Working with Kanban Framework was interesting and quite different from what we were used to with SCRUM. Because mostly we worked only two members it felt nice not to have all the responsibilities that involve doing SCRUM and it felt a bit more natural.

On the other hand, because last project working with SCRUM, felt like a *sprint* we were afraid that not having a constant push would make us *slack*. But it turned out to be the other way around. Because there were not clear achievements for each iteration, we constantly had the sensation that we are falling behind. Because of that sensation, we constantly were rushing, which was quite exhausting, burning us up in the end.

All in all, we feel like the best option for a development framework would be a combination of those two (SCUM and Kanban) which we will experiment in the next project.

5.2. Unified Process

The unified process helped us ensure that we added a fully developed feature to the software. Working in an iterative way, provide many benefits (easier to adjust to the needs, harder to lose focus, easier to predict) and helps stay agile in the software development process

An iteration would go through all the stages (Elaboration, Construction, Transition) providing a finished, tested and documented feature in the end.

5.3. Version Control

Even if last semester we touched upon using version control, this semester we tried to use it from the beginning and until the end. It helped us a lot to be able to share everything between us without having constantly send files and compare versions. At the same time, it was very useful when adding new features that would break something to be able to go back and understand what has broken.

All in all, it was a very useful tool, that provided great support for our software development process. One problem was when we tried using *branching*. Because it is a rather small project, using branching was an *overkill* meaning that it caused more problems than solved. But still, it was great to gain experience of the system behind.

5.4. Critique of the project

There are a few points that are not good as we wished in regards to this project, point that we will try to keep in mind and not repeat in the next projects. Because of the learning process, it is a priority to mention them to ensure that they are documented and can be learned from.

5.4.1. SOLID Principles

Because a lot of importance was paid to the architecture of the software, ensuring that it respects the requirements of a distributed system, we paid less attention to the SOLID principles. As a consequence, the system is much more fragile, closed and immobile than we would have wanted it to be, at the end affecting the software's possibility for scalability and expansion. It was our mistake, not paying more attention to the principles when designing the software and rather focusing on the overall architecture which in the end is negatively affected by us violating the SOLID Principles

5.4.2. User Stories

In the beginning, phase, when working on the user stories, we expected to complete at least 2/3 of them. But as we progressed through the project, we realized that we will never be able to reach that amount. In addition, a member of the group had to leave, which meant that even less would be done. We wanted this project to be more round, in the end, providing at least a glance of a solution that could actually be used. But due to the time needed to analyze and design, as well as implement the skeleton, we fell short.

5.5. Group Reflections

All in all, we enjoyed this project better compared to the last semester, because of the fact that it felt much closer to a *real project* due to its nature (Heterogeneous System). Working with different technology options for the problems, made our critical and engineering thinking to improve (compared to when one has only one option). Deciding on what to use in order to solve a certain problem, and how to organize everything so that it made sense was, enjoyable.

Moreover, this semester not having a class that was solely responsible with how the semester project should be made (Software Engineering Classes of the last semesters) gave us a freer hand on deciding how to work, which felt a bit, *unsafe* through the project, but was a great learning tool for us.

6. Personal Reflections

In the next chapter personal reflection on the project and process around it will be presented.

6.1. Gais El-AAsi

The third semester was a good one. I particularly loved the semester project *Heterogeneous System* as it was quite a step forward form the last semester. Having the possibility to work with different technologies, and try to combine then in a way that the result makes sense in a meaningful way was of great pleasure.

This semester we reduced the group size from five to three to ensure that everyone has the same expectations from the outcome of this project, as well as is willing to contribute the same amount of energy. Because of that, and the fact that we previously worked together we had quite a good synergy.

Unfortunately, one member had to leave which meant a bit of restructuring of the workload, expected deadline, but most importantly the expectations of the outcome. We quickly realize that we were already doing our best, and cutting 1/3 of the men's power, would have influenced the result of the project.

Nevertheless, I consider than each member of the group did their best, and lived up to the expectations of this project. Even if a bit below what I wanted, we are happy with the outcome of the project, as well as the process of getting in there.

One thing that bothered me throughout the entire semester, was that I constantly felt like we have *no time*. This can be attributed to the fact that we were short one member, or that the projects became larger or both of the above.

6.2. Marcel Notenboom

In this semester the project group had been changed significantly, we have gained a new member since the first semester, but also reduced the group size to three as opposed to a five-member group in the second semester. The reduction in the size of the group has been very beneficial in terms of smaller project scope and ease of communication and planning.

Despite this change in group structure, the chemistry between the group members and the vision of the group was still as solid, also because we already knew each other from previous semesters which made the start of the project smoother. In addition, we did not need to test the waters with each other and as a result, the issues that we encountered were quickly and civilized resolved.

The group did have one significant problem that needed to be solved this semester mid-project which we did not expect and have not experienced before. This was that we lost a group member reducing our already small group size of three into two. As a result, this meant that the work that was originally planned and split into three now needed to be restructured and split between two people. Luckily, Gais who has been acting as the project lead planned in a talk to resolve this issue and apart from a bigger workload per person this did not leave us with a lasting problem.

At the beginning of the project, we decided to not have structured meetings and instead adopt a more fluid project structure. Because of this fluid-structure, we decided to use an online instant messaging *#Slack* platform for near-constant communication. This made sure that even though we did not have weekly physical meetings every member was still in contact with each other and knowledgeable of where the project stands.

For the project overview, we used a digital Kanban software which a project member had sourced a free student version for. This helped the group to keep track of which features and tasks where already done, still needed to be done or the ones which were being worked on already. Filling out the tasks on this Kanban board was difficult at first because we made each task too big and undefined, later we figured out that we could use the requirements of the project and break them down in small testable sections of work. When we structured the Kanban in this matter we could easily make, test and then implement a section that a group member made.

In order to facilitate the fact that the group needed to work on the same project without physically being together, we made a git-repository which worked out well during the project although we spend more time then we wanted on figuring out how the branches worked since they gave us some problems when we wanted to merge them together with the main branch.

A factor that significantly helped us in this project, in my opinion, was the mid-semester deadline to show a working skeleton of the project. This forced us to focus our attention on the backbone of the project on which the rest was based and therefore also was the very time-consuming part of this project. Having this done in the early stages of the project made it much easier to make the different features and enabled the group to work simultaneously.

7. Supervision

During this project, we have had two supervisors, Jakob and Jan that provided guidance during the entire period. Both teachers, provided great directions and advice for the semester project with matters as organization and structure, as well as the technologies to be used.

During the semester (classes) many times they have been helpful enough to even offer guidance during their non-semester project classes (DNP/SDJ3). We constantly relied on their support when we were in a bottleneck or did not know how a certain technology works.

Even so, during this semester, the provided guidance was more *on-demand* meaning that we needed to ask for it, rather than expect being taken care of, which was a bit uncertain in the beginning.

All in all, I think that supervision provided great help for our software development, in all the stages.

8. Conclusion

After contemplating the semester project, and the development process there are a few points that should be highlighted before making a general conclusion. This point is related to group work, time management, frameworks and technology, and the overall feeling of the semester project.

- **Software Development Framework** – either it is SCRUM, or Kanban or any other framework it is important to figure out what works for the group/project and use one to aid the team. Even if sometimes a taboo, realizing that everyone needs help for self-structuring as well as work-structuring is powerful enlightenment which can help overcome many obstacles on the road;
- **Version Control –** it should be though and used from the first semester. It helps at every point of the software development process and cannot be dismissed in any way;
- **Packet Manager –** even if it is not too much related to the .NET platform as it has one build in. When it comes to working with Java, using a packet manager (Maven in our case) was of great help to utilize different packets and libraries useful for the project development. By using it there was less headache with Java versions, with finding and manually adding packets, and so on;
- **Understanding that Semester Project is a learning tool –** as in the previous two semesters, it was of great focus that we already had the understanding of the use of the semester project. It is not mean to develop great solutions that will change people's lives and *make the world a better place* it is a learning process, for the students, and the better we understanding this fact, much easier is to work with it, gaining a lot of motivation from the learning process;

All in all, the semester project was a great experience, where we got to gain new knowledge about architecture, technologies, languages. We had the possibility not only to learn about them but to utilize them in a project that offered the possibility of combining and adding everything together in the hope that it will make sense in the end. We consider that this project semester offered the greatest knowledge rewards so far, and doing it has improved us as software engineers and programmers.

9. Source of information

- Larman, C., 2004. *Applying UML and Patterns: An Introduction to Object-Orientated Analysis and Design and Iterative Development.* Third ed. s.l.:Addison Wesley Professional.
- Radigan, D., 2019. *www.atlassian.com.* [Online]
  Available at: https://www.atlassian.com/agile/kanban
  [Accessed 2019].
- Rehkopf, M., 2019. *Agile Coach: Kanban vs Scrum.* [Online]
  Available at: https://www.atlassian.com/agile/kanban/kanban-vs-scrum
  [Accessed 2019].
- Valve Corporation, 2012. *Handbook for New Employees.* 1st ed. Bellevue: Valve Press .
- VIA University College, 2019. *Semester Project: Heterogeneus System (IT-SEP3).* [Online]
  Available at:
  https://studienet.via.dk/sites/uddannelse/ict/Horsens/studymaterial/Pages/Course-Descriptions.aspx
  [Accessed September 2019].

Appendices