Project Management

Thor Valentin Aakjr Olesen, Jacob Mullit Mniche,
Dennis Thinh Tan Nguyen, William Diedrichsen Marstrand

December 16, 2015

1 Group Collaboration

1.1 Organising the Project

This is a short description of the various group structures we have used to organise our group throughout the BDSA course at ITU. To establish a common ground and similar expectations for the project, we reused a group contract from prior work. The contract provided a relaxed and open minded work environment. However, we still felt a need for a clearer distinction between group work and private life, which lead to an official definition of meeting hours. In other words, each group member specified in which time span he is available and vice versa. This allows us to avoid trespassing on group member's spare time.
Initially, we kept the meetings informal without a Mediator, but with a Note Taker. A Mediator would only get elected upon sudden conflicts that required a more planned approach.
Even though consensus was reached on the group contract, we still lacked a dedicated strategy for the weekly assignments. Tasks were randomly delegated to group members without any direct understanding of the actual required workload of the task. Consequently, an unbalanced work distribution sometimes occurred among team members, which ultimately led to otherwise avoidable group conflicts.
To solve this problem, we iteratively improved our workflow by experimenting with different techniques, e.g. Scrum. We could not implement a pure Scrum implementation due to limitations like time constraints, but features like the task board proved especially valuable. The task board helped us visualise the current tasks and quality control already solved tasks. On the task board each task starts in the **Backlog** area. Then we move the most vital tasks to the **Current Sprint** area. From here each group member is assigned to a task. When a task is completed, it is moved to the **Review** area. Then it is evaluated by other group members. The task is either approved and moved to the textbfDone area or failed and moved back to the **Current Sprint** area. As a result, all tasks are reviewed and continuously kept track off.

1.2 Division of Tasks

Whenever new tasks were brought up at group meetings, they were initially divided into parts and subcomponents that could be iteratively delegated to each member of the group. By way of example, the requirement analysis document sections were marked as tasks, which could either be worked on individually or in groups. The group members have chosen tasks independent from time consumption and scope as one does not know this until having worked with the given task.

As a baseline, each member was expected to work on a given task single-handedly. However, if a distributed task was exceedingly great, it was either divided in more subtasks or assigned to additional members. We did try to analyze if a given task had a tremendous impact on the whole project and was thus regarded as a task that demanded everyone's attention.. By way of example, when working on the Design Goals in the system design document, the whole group was required to work on this section before they could proceed with the document.

The distribution of work/tasks for each part of the project can be seen on figure 1, figur 2 and figur 3. Notice how each task is divided into sections that are delegated to each member. Each member had some main tasks that they were responsible for (which can be seen on the high percentage). However, one may also notice that other members do have a smaller percentage on some given tasks. This is because the given task was either reviewed or too demanding. As a result, other members that had finished their originally assigned tasks would place their resources onto other tasks.

RAD	Introduction	Current System	Proposed System	Scenarios	Use Case	Object Model	Dynamic Model	Glossary
Dennis	10%	0%	20%	50%	15%	0%	40%	25%
Jacob	0%	100%	20%	50%	15%	30%	25%	25%
Thor	85%	0%	30%	0%	15%	30%	10%	25%
William	5%	0%	30%	0%	55%	40%	25$	25%

Figure 1: RAD Work Distribution

SDD	System Purpose	Design Goals	Subsystem Decomposition	Persistent Data	Access Control	Global Flow	Hardware
Dennis	0%	25%	5%	95%	5%	95%	0%
Jacob	0%	25%	0%	0%	0%	0%	100%
Thor	0%	25%	10%	5%	90%	5%	0%
William	100%	25%	85%	0%	5%	0%	0%

Figure 2: SDD Work Distribution

Code Skeleton	UserManagement	ExportM	ProtocolM	StorageM	WebAPI	StudyM	PaperM
Dennis	50%	5%	5%	5%	60%	25%	0%
Jacob	0%	0%	0%	0%	0%	50%	50%
Thor	20%	0%	60%	90%	0%	25%	0%
William	30%	95%	35%	5%	40%	0%	50%

Figure 3: Code Skeleton Work Distribution

1.3 Cooperation Tactics and Tools

The group work was coordinated primarily by using tools for planning, version control and communication. We generally applied a loose version of scrum to manage the project and become familiar with the SCRUM methodology. In practice we tried to keep each other updated every time we met (partial stand up) and would try to give each other an overview of three things accordingly; what did we do last, what are we planning to do today and finally whether anything is blocking this purpose. If anything was blocking a team member from continuing his work, the SCRUM facilitator would try to find the required help to solve this. We also established official meeting hours and contact periods to separate study related activities from social life. This was done to cope with the otherwise stressful environment that team members felt due to the heavy work load in this semester. Communication tools such as Facebook and Messenger were used to keep in contact and inform the group about practical information. Finally, the collaboration tool "Trello" was used to keep track of everything. Note that all members are assumed to stay updated about changes on both Facebook, Git and Trello. **Trello** is a collaboration tool we use to organize the project into so called "boards". In one glance, Trello tells you what's being worked on, who's working on what, and where something is in a process. The board represent a combination of the different phases used in SCRUM and the Waterfall Model. We have a Backlog board that corresponds to the Product backlog containing all possible features and requirements in the system. Secondly, the Sprint board contains a backlog with a work that must be addressed during the next sprint (usually one for each week). When team members finish a task it is tested and reviewed by another member in the other boards and finally put in the Done board.

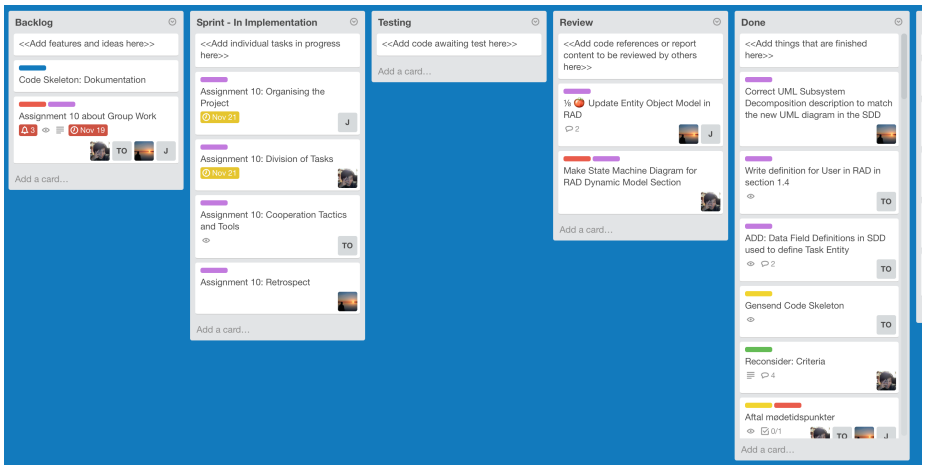

Figure 4: Trello Board

1.4 Retrospect

When thinking of the group work in retrospect, it has become clear that certain things could have been done differently, which would probably have improved the communication, cooperation and efficiency in the group. The first thing which should be mentioned is the attempt to follow the SCRUM method, which only was accomplished partially, since one of the core features (the stand up meetings) was not followed consistently by the group. If these meetings had been upheld, it would presumably have strengthened the communication. As a result, this could potentially have lead to fewer misunderstandings and less miscommunication during the work.
During the beginning of the group work, Trello was not used properly, which meant that the group work did not become as structured as it could have been. By using the Trello board probably it would also have made the first issue with SCRUM easier to handle, since it would have been possible to structure the SCRUM using this tool. Also, the use of a Trello board would have made the planning and distribution of tasks a lot easier.
Better communication could also have been accomplished by the use of a Trello board combined with a more structured use of the Facebook group, e.g. by setting up some guide lines on how and what to write. By scheduling strict deadlines and communicating more about them, some of the unfortunate mistakes with missing content, which happened during hand ins could have been avoided. Further, a better set of rules for VCS when writing the documents in LaTex could have prevented some critical compile errors.
In the beginning, the working hours were very flexible and mostly decided based on people's job schedule. This lead to occasionally late working sessions and meetings where only parts of the group could attend. Thus, it had an impact on the stress level in the group, which was why a decision was made to make a schema containing the office hours where people could be contacted and why a planning a fixed meeting schedule for the week was made. This should have been done a lot earlier in the work process, since this initiative created a better working environment for most of the group members.
Besides the group work, it has become clear that a better use of TAs and application domain specialists throughout the course would have been rewarding, because this would have resulted in a better understanding of the application domain from the beginning, which would have meant less resubmissions and a better foundation for future work. Some of these challenges mentioned above might originate from a delegation of tasks happening too fast, and thus the group did not always take the time to talk about the theory and establish a solid and common knowledge before beginning the work. In this way, the approach became much more practical with a "fail faster" mentality, that also had its pros because a lot of practical experience was achieved quickly. However, some resources could have been saved by using slightly more time on the theory before trying to solve the tasks. ======= When thinking of the group work in retrospect it has become clear that certain things could have been done differently which would probably have improved the communication, cooperation,

and efficiency in the group. The first thing which should be mentioned is the attempt to follow the SCRUM method, which only was accomplished partially since one of the core features (the stand-up meetings) was not done by the group. If these meetings had been upheld it would probably have strengthened the communication which could potentially have lead to fewer misunderstandings and less miscommunication during the work.

During the beginning of the group work Trello was not used probably, which meant that the group work did not become as structured as it could have been. By using the Trello board probably, it would also have made the first issue with SCRUM easier to handle since it would have been possible to structure the SCRUM using this tool. Additionally, the use of a Trello board would have made the planning and distribution of tasks a lot easier.

Better communication could also have been accomplished by the use of a Trello board combined with a more structured use of the project's Facebook group e.g. by setting up some guidelines for how and what to write. By scheduling strict deadlines and communicating more about them, some of the unfortunate mistakes with missing content, which happened during hand ins could have been avoided. Furthermore, a better set of rules for VCS when writing the documents in LaTex could have prevented some critical compile errors.

In the beginning, the working hours were very flexible and mostly decided based on people's job schedule. This lead to occasionally late working sessions and meetings where only parts of the group could attend. Moreover, it had an impact on the stress level in the group, which was why a decision was made to make a schema containing the office hours where people could be contacted and why a planning a fixed meeting schedule for the week was made. This should have been done a lot earlier in the work process since this initiative created a better working environment for most of the group members.

Besides the group work it has become clear that a better use of TAs and application domain specialists throughout the course would have been rewarding, because this would have resulted in a better understanding of the application domain from the beginning which would have meant fewer resubmissions and a better foundation for future work. Some of these challenges mentioned above might originate from a delegation of tasks which was too fast, and thus the group did not always take the time to talk about the theory and establish a solid and common knowledge before beginning the work. In this way, the approach became much more practical with a "fail faster" mentality which also had its pros because a lot of practical experience was achieved quickly, but maybe some resources could have been saved by using slightly more time on the theory before trying to solve the tasks.

2 Individual Reflection

2.1 Individual Reflection of the Work Thor

The thing that struck me the most, was the changed structure of the project. We have been used to work on projects with predefined tasks and requirements. As opposed to this project where we rely on our own observations and efforts. Initially, I did not understand that we were supposed to treat our teaching assistants and lecturer as people from within the application domain (e.g. consultants). I was rather skeptical about this approach but now understand how important it to be able to analyze customer needs in a given domain and derive a solution from these requirements. It may feel like a conjured environment, since we do not keep contact with actual customers but the similarity is close and has helped challenge my work principles.

In regards to the group work, we started out with a flexible meeting schedule, which worked most of the time but I personally prefer a more planned approach. Sometimes we did not manage to review the work of all team members, which ultimately resulted in assignments with missing or inconsistent content. Consequently, we ended up using Trello to achieve a better overview along with a time schedule that all team members agreed with. This helped improve the cooperation and coordination of the project dramatically.

As in the prior projects, we used version control (Git) allowing us to work simultaneously on the mandatory assignments and code. This worked very well but required a strategy on how to use Git in order to avoid merge conflicts. Thus, we thoroughly went through on how to use branches and specific naming conventions when working on separate parts of an assignment or code project. I think we should have done this from the start, instead of having many merge conflicts requiring refactoring. This along with proper coding conventions minimize the overall time spend refactoring and makes what we produce more consistent and correct from the start. Consequently, I personally wrote a document about naming conventions and branches in Git, which has been read and understood by all team members. I had a bad experience with the lack of these agreements in the First Year Project were people would have different opinions about documentation, placement of brackets and version control. This ultimately resulted in many extra hours of work close to the deadline, which could have been avoided.

The work load was equally distributed among team members and I did not find any personal issues working together. This is based on previous experience working with the same people. Thus, we already know our different personalities and have thoroughly established the different skills that we have to complement each other. The only new conflict we had in this specific project was based on disagreements about time and priorities. In other words, some members were fine working on the project all days of the week while some members preferred

a clearer separation between work related activities and social life. We had to solve this by making an actual document showing which times of the week each member is available or does not want to be disturbed. This will probably affect my choice of group for the next project, since I think a general agreement of work hours and time spent on the project is important to establish a common goal and work ethic.

2.2 Individual Reflection William

The major issues I have encountered during the project was in regards to:

1. Time
2. Understanding the application domain
3. Working with LaTex using Version Control
4. Partly Unstructured Work and lacking Communication
5. Hand-In of Assignments

In regards to working on LaTex, I found that the communication and the rules for using git was too vague. This resulted in many merge conflicts and lost data, because the guidelines were not clear enough. The work was done without enough structured planning in the beginning, which resulted in people not being sure about who was assigned to which specific assignment. Instead people had a more general idea about which parts of the assignments they were working on. This made it hard to know exactly who was responsible for what, and so it was hard to know, who to go to when having/spotting a problem. About the communication, the lack of this (also in the beginning) resulted in missing hand ins or hand ins with missing sections or wrong/duplicated UML diagrams. Also the documentation of the development could have been better if versioning had been done in the beginning of the project. New realeses of the program and the different design documents should have been made frequently during the work to reflect design choices and evolution of the program better.
Another major struggle has been understanding the application domain. In the beginning the application domain seemed very confusing, but not enough effort were made to generate questions for the domain expert to explain the domain. This was a huge mistake, which had a seriouse impact on the project during the implementation, because the development of the program went in a wrong direction. This resulted in a wrong implementation, which had to be chaged quickly 2 days before the handin deadline. Because of this no functioning implementation of the program was ready for handin.

2.3 Challenges in the Cooperation and Coordination

Some troubles were encountered in accordance to when and how much to work. This was because there were conflicting opinions about how or even if it should

be acceptable to declare oneself for unreachable because one would like to separate the study from the social life.

2.4 Individual Reflection Dennis

3 Challenges while working on my part of the project

One of the greatest challenges I encountered regarding the project was to have an understanding on the application domain but also an understanding on how the requirement analysis document and system design document was to be written. This resulted in some sections being incorrectly written and was consequently required to be rewritten.

Final edition
As written previously the greatest challenge that I encountered throughout the project, was indeed the understanding of the application domain. This resulted in the program not being implemented as required. However, after I had a personal meeting with Steven and Paolo I was then truly able to understand the application domain which I shared with the group. Yet, this was unfortunate too late. However, I did manage to create and implement a study configuration UI that was to reflect the actual process of defining a study. Taken the UI into consideration, being solely responsible for creating a UI while also learning a new framework was also really, though, but I did get out from the process.

4 Impediments with regards to the team cooperation

Some impediments I encountered was the lag of communication and team planning at the initial end of the project. We did not delegate the work probably and that resulted in some group members having a greater workload than others. The lag communication resulted in some work being incorrectly made or some required work resources not being shared correctly so others can continue their activity. By example, when some UML was made one may forget to share these to the whole group so they can proceed with their section/activity.

Another thing regarding communication is the fact that we did not agree on common terminologies. Therefore, one may use one terminology while another uses a dierent terminology. Consequently, the report ended up being inconsistent and was therefore required to be xed. This could all be avoided if we had a clearly communication on what was to be done and how we achieved

Final edition
Something I really is missed was to meet up with my group in the weekdays. This would allow us to have a deeper discussion on issues instead of going around with them alone. I'm not implying that the others didn't work. They really did and I appreciate it, but since we only meet two times a week while also being confused on the application domain was not really optimal regarding creating a

proper program.

5 Retrospectively change something with regards to my approach to the cooperation within the team

I would have been clearer on my communication to avoid misunderstandings. I would also change the initial approach of the project in that sense of better planning and group coordination. It may have yielded greater results if we all had researched a bit more on the application domain, but also, how the requirement analysis document and system design document was to be written, so we could have avoided aimlessly assumptions on how they were to be made.

Final edition
Reflecting on the written above I did improve my communication. I was able to explain the application domain for the whole group which did help to steer development in the right direction. Nonetheless, I would in the future utilize TA's and other experts if I sense the slightest form of misconception because going through all of this again is not worth anyone's time and energy.

6 Cooperation improvement throughout the project

The group did, fortunately, mature throughout the progression of the project. The group had created xed work schedules and additionally utilized a Trello board. This has yielded great results in terms of planning and communication. The xed work schedules enable us to plan and propose what task has to be prioritized and accomplished while a Trello board allows the group to evenly distribute the workload among the members. If the group did this from the beginning, countless work hours and resources could have been allocated on some other sections of the project. Nevertheless, the group had apprehended this issue and has in that sense improved.

Final edition
Improvements did occur as mentioned above. Also, we ended up communicating more but that was, unfortunately, near deadline.

6.1 Individual Reflection Jacob

7 Individual Reflection Jacob

The biggest challenges I have faced in this project has without a doubt been the UML diagrams and understanding the the terminology used by the user. It was especially hard to understand the users needs and what was required of our solution. Not properly understanding the user domain made it especially hard to create scenarios and use cases which I struggled with. This could of course have been avoided by asking the client more questions, but regrettably. // Furthermore. In the code implementation we experienced once again our knowledge of of the application domain was lacking which lead to last minute changes to the program, redesigned tests and people were moved from developing on the project to patch-working, which hurt our ability to deliver a satisfying program.

7.1 Teamwork

I think the teamwork in this group has progressed smooth and without too many hiccups. There has been incidences in which the group had to deal with team members disappointing each other, but each and every time it has been resolved swiftly and without trouble. I think we have avoided a great deal of internal conflict by defining office hours in which team members could be contacted in. This has greatly helped group members with different sleeping patterns to separate work from spare time.

7.2 Retrospective

I think we could have saved ourself from a lot of hurt by implementing and experimenting with methodologies like Scrum earlier in in the process. I noticed we became much more organised after experimenting with a dedicated task board to visually represent our current progress. From my own experience, I can highly recommend this course to teach these methodologies earlier in the course.