This week we've decided to write a short introduction since some important thoughts might not fit under a specific discussion point.

Last week we struggled with our startup meeting which resulted in a bad sprint. We did not have clear designated tasks and had problems picking up the pieces as the week went on. Therefore we did not get enough work done and changing this was the focus during sprint 3. By planning this sprint better we've improved and produced a much better result. We've also started having daily scrum meetings and this contributed to a much higher work rate.

Customer Value and Scope

- the chosen scope of the application under development including the priority of features and for whom you are creating value
  - Unchanged, the scope of the application is still the same.
- the success criteria for the team in terms of what you want to achieve within the project (this can include the application, but also your learning outcomes, your teamwork, or your effort)
  - We've managed to raise the bar when it comes to our work rate. This was an important mission which we all agreed upon at the end of sprint 2. Our teamwork also improved since we decided that the bigger stories would be executed in pairs of two. This resulted in a much better workload for us as individuals.
- your user stories in terms of using a standard pattern, acceptance criteria, task breakdown and effort estimation and how this influenced the way you worked and created value
  - This was something we improved quite heavily this week. We spent more time on reworking our user stories and making a better estimation. Two of our members decided that a spike was necessary to evaluate the time cost of a user story they were assigned. This turned out well and the corresponding tasks got completed. Another result was that everyone had something to do since we divided tasks which had an appropriate scope.
- your acceptance tests, such as how they were performed, with whom, and which value they provided for you and the other stakeholders
  - We've introduced pull requests which helped in making sure that the deliveries were functioning properly and also helped define when we move tasks to done in our scrum board.
  - Another addition is the new interaction protocol which serves as proof when it comes to fulfilling our acceptance criterias.

- the three KPIs you use for monitoring your progress and how you use them to improve your process
  - The group has integrated sonarqube to check the code quality. We are using this to identify potential vulnerabilities in the code that needs to be amended. We also decided on creating a work breakdown structure in order to keep track of the sprint deliveries.
  - We introduced a table to track success rate of every sprint to trace whether or not changes to the scrum process would improve our capability to deliver.

Deliveries

	Week 1	Week 2	Week 3	Week 4	Week 5	Week 6
User Stories in backlog(nr)	7	7	3	5	5	
Finished User Stories(nr)	0	4	3	6	4	
Success Rate(%)	0%	57%	100%	83%	80%	

Code Quality

Feelings-table:

	Daniel	Hampus	August	Jesper	Anthony	Henrik
Motivation	4	4	4	4	4	4
Productiveness	3	4	4	3	4	3
Stress	3	3	3	2	2	2
Team communication	4	3	4	3	4	4

Social Contract and Effort

- , i.e., the rules that define how you work together as a team, how it influenced your work, and how it evolved during the project (this means, of course, you should create one in the first week and continuously update it when the need arrives)
  - Because we had daily scrum meetings we weren't in need of formal meetings and therefore haven't had chairmen/secretaries.
  - Our teamwork improved since we communicated more often and also because some of us worked in pairs.
- the time you have spent on the course and how it relates to what you delivered (so keep track of your hours so you can describe the current situation)
  - We've all spent more time on the project and we believe this is a result of better planning and a proper division/delegation of tasks.

Design decisions and product structure

- how your design decisions (e.g., choice of APIs, architecture patterns, behaviour) support customer value
  - We were thorough when prioritizing our user stories and tried to focus on the ones that will give the customer the maximum value.
  - We've started to work on some minor specific improvements that supports customer experience such as the ability to swipe between tabs.
- which technical documentation you use and why (e.g. use cases, interaction diagrams, class diagrams, domain models or component diagrams, text documents)
  - Unchanged except for interaction protocol.
- how you use and update your documentation throughout the sprints
  - We've created a document which tracks success rate when it comes to delivering the user stories that we plan in our startup meetings.
- how you ensure code quality and enforce coding standards
  - Pull-requests and sonarqube have made an positive impact for the group and project in ensuring code functionality and identifying code quality.

Application of Scrum

- the roles you have used within the team and their impact on your work
  - Unchanged(?). We thought that we were in need of designated roles but our other changes resolved most of our previous concerns.

- the agile practices you have used and their impact on your work
  - This sprint our group started to have daily scrum meetings which resulted in an noticeable improvement of communication and awareness of the project process within the team. Through the teams weekly retrospective in team reflections we have identified problems in the process and discussed how we can overcome the problems in the next sprint.
  - Our Trello board came to much greater use this week. It was easy to keep track of each members work and how the team progressed.
  - We improved a lot when it came to reworking user stories so that they're independent. This week we did not have to rely on other people implementing features that affect your own assignments.
- the sprint review and how it relates to your scope and customer value (in the first weeks in terms of the outcome of the current week's exercise; in later weeks in terms of your meetings with the product owner)
  - We still have some issues with our deliveries. Features are not fully completed when pushed. This has been hard to work on because of our previous focus on improving work rate and structure. If we had more time a next step when it comes to group improvements would be tackling this issue. We think that using more kpis such as tracking user story lifespan could put even more pressure on the team to reduce the scope of user stories. This leaves more room to "perfect" deliveries and hopefully make rushed features more uncommon. For the last sprint we want to make sure that our deliveries have a good quality rather than a large scope.
- best practices for learning and using new tools and technologies (IDEs, version control, scrum boards etc.; do not only describe which tools you used but focus on how you developed the expertise to use them)
  - A lot of the features implemented this week and during the project have been uncharted territory for our group. Previously it has mostly been up to individuals to research and find solutions. This week we partly worked in pairs of two which improved the rate of learning. Our members helped each other, divided the research areas and held a dialog that made finding a fitting solution quicker.
- relation to literature and guest lectures (how do your reflections relate to what others have to say)?
  - Some members have spent time on research about agile practices. We've always had a notion about this although we all could benefit from getting a broader theoretical knowledge. This helps to reduce time spent on finding methods for solving our problems and could also help with identifying what exactly we are struggling with.