

TEAM 20, SPRINT 5

Team members: Rebecka Jakobsson, Cecilia Michelsen, Jacob Messinger, Tina Samimian, Elin Nilsson, Linnéa Fransson, Norbert Laszlo & Oscar Helgesson.

Customer value & scope

We have prioritized to implement features that deliver the most value for our stakeholder in this last sprint but due to it being the very last sprint we've unfortunately not managed to implement all of the features that we wished to have from the start. Therefore in this last sprint we focused on the features that according to our stakeholder would bring the most value to the end user.

For this sprint we had a quite good estimation for how long we thought the user stories would take. What we did differently in this last sprint however compared to previous sprints, was that we intentionally low balled our estimations to be able to prioritize the more important features and had a couple user stories with features that were not as important. We called these the bonus features which we didn't assign to anybody but if any of us would finish with time to spare they would continue with these features, which some of us did. This allowed us to also complete all of our bonus features.

Overall we consider this project as a success even though we didn't implement all of the features that we had in mind. However we did manage to implement all of the features required for a MVP that we from the start would consider as a success.

KPI:s

Effectiveness: Estimated number of invested work hours VS Actual number of invested hours for each sprint.

This week we estimated the total time of the user stories to **15,5 h**. Since this sprint was a bit shorter due to the holiday we planned fewer hours than normal. The actual time we spent this week added up to **18,5 h** which gives the percentage of **119 %**. This rate is considered **Good** according to our KPI limits of effectiveness. From this good KPI we can see that the estimation in total was very good the last sprint, and that we have developed our ability to estimate correctly. However there were a few deviations because especially one of the user stories was overestimated, while two others were underestimated. In total these small deviations evened out which led to the good KPI.

Productivity: Total score of points by completed user stories in a certain sprint vs. estimated total score one sprint.

This week was a bit shorter than a regular week so we decided to make a smaller sprint. We planned the sprint to cover **17 points**, and we managed to solve all of them which gave us **17 points**. That gave us **100%** which is considered **Good** on our KPI limits of productivity. This was one of our best estimated sprints, which proves that we have learned how to estimate during this project. This week we chose to have more but smaller (with lower points) user stories than usually. That may have had an impact on our result. The points that we estimated for each user story were accurate for the difficulty of the user story.

User Satisfaction: A measurement of satisfaction from our stakeholder.

1. Does our work meet the user's expectations on functionality?
2. Is the user interface convenient?

This week our stakeholder gave us 9,25/10 for the functionality and 8,5/10 when it comes to the user interface convenience. She considered our work this week to bring a lot of value and that it tied the knot for most things discussed throughout the project. Foremost she thought that the company detail page was a lot clearer now and that it was great that we had added links to google maps and the Facebook page in the home fragment.

Design decisions and product structure

During the course of this project our goal has been to create as much value for the customer as possible. We have therefore focused on working in an agile manner and completing as many value-creating user stories as possible and therefore haven't paid as much attention to following the architectural design pattern MVVM, but we still haven't totally disregarded it. If we were to redo the course, paying greater focus or having a generally better understanding about the MVVM pattern would in the end create value for our customer. If following it were second nature to us it would not have slowed us down and created value. We do however believe that if we had focused on the pattern too much it would have slowed us down a bit and wouldn't have enabled us to create as much value to our stakeholders in other places as we have.

We also planned out to write tests for all code but we deprioritized that because of other things we deemed more important. Of course having Unit-tests for everything would be beneficial to code quality in the large scheme of things. However, we focused on spending that time creating more value to our customer instead. Doing Unit-testing was a part of our Definition of Done (DoD) but we barely used that part of it. In hindsight reformulating our DoD would have been beneficial to us as we didn't use the entirety of it and this is something that, if we did the project together again, we really would pay attention to. Looking through alternative DoDs other people have used in other projects could have helped us find one that was more useful and beneficial to us.

Application of Scrum

During the whole project we have had two new scrum masters (as far as possible) each week and we have all shared the responsibility of the product owner. It has worked out really well

for us. The whole team has been very engaged and therefore the scrum masters' biggest area of responsibility has been to lead the meetings. Regarding our sprint reviews we have been in contact with our stakeholder when doing the reviews every week. Thereby we have been updated on her opinions and priorities and she has been able to comment on our work continuously. By sharing the role of the product owner we have all been part of the weekly meetings with our stakeholder which has brought clarity to all of us.

We have developed our expertise in working agile and gotten better at writing user stories, acceptance criterias and tasks. We haven't really changed our way of doing it but learned from our mistakes and developed our mindset and understanding on how to do it. We have also gotten a lot better at estimating the time and points each user story takes.

During this project some of us have learned to use git, since the majority of the group had worked with git before it went pretty smooth to teach the beginners. As mentioned earlier we worked in teams where at least one of the members had experience of using git. For further projects we would need to develop our way of working by using pull requests from the beginning to ensure code quality. In addition we have all learned to use Android Studios, since all of us were beginners we have used google to learn more and also tried out a lot. Therefore some code has taken a lot more time than expected but at the same time we learned a lot.