TEAM 20, SPRINT 2

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

Customer value & scope

This week we have prioritized user stories that will make future work on the app easier for us. This includes work such as dynamically creating objects from a JSON-file and designing better looking lists so that when they are filled later in the development process it will look as it should. This work would simplify future sprints as much of the framework would be complete and it would just be for us to start using it. As we will further discuss later in the Team Reflection we had some trouble finishing some user stories this sprint. We accept a user story as done when the formulated acceptance criteria is met, and this was not the case for two of our user stories. The user stories we did complete however does provide value for both us and our external stakeholder. We have simplified the future work for us as developers by being able to dynamically create objects and also completed more of the essential design for our stakeholders.

This sprint we had some problems regarding the user stories. The main problem was that the team did not think through the user stories properly before choosing them, which made it very difficult to fulfill all of them. Especially two of the user stories were dependent on another user story in this sprint, which we realised during the week. The other two did not depend on another user story which made it possible and much easier to complete them.

Looking forward, in order to be able to complete all selected user stories, we will think through the user stories more in detail and in a technical way before we determine the sprint. We need to make sure that each user story is isolated when working on them in the same sprint. We will by having a more thorough discussion in the sprint planning meeting try to better choose and discuss the technical aspects together so that every sprint pair will have a clear idea of what needs to be done.

KPI:s

Until sprint two, we have worked with the three KPIs. These three have been Effectiveness, Quality and Productivity. But this sprint we realised that our KPIs don't reflect on our application. We are building an application where the main focus is to digitalise a catalog. This means that most of the coding is design oriented and the visualisation is more in focus than actual functionality. The KPI Quality is about code quality and it is hard to write good tests on code that is written for the view. With this realisation we have decided to choose another KPI until sprint 3 to better be able to measure our product. The new KPI is **User Satisfaction** and is better suited for our application. User satisfaction is measured through surveys. By asking the stakeholder to rate their experience of the application after each sprint, by answering two questions, we can better understand what is good in the application and what can be enhanced in the next iteration. We have decided to use the following two questions for the stakeholder to answer:

1. Meeting user expectations on functionality
2. User interface convenience

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

We estimated for all user stories to take 17 hours in total since this sprint was a bit smaller than the other ones considering it was a national holiday on Friday. The actual time we spent on the user stories was 25 hours which gives us a percentage of 147 % and therefore our planning for this sprint is considered bad. In addition we didn't manage to solve two of the user stories. Most of the "extra hours" were spent on those stories and despite that these user stories weren't finished. Therefore we choose to calculate a separate percentage for the finished user stories which we estimated would take 8 hours in total but actually took 10 hours. The percentage therefore became 143 % which made us realize that we have to reconsider our procentual limits so that the KPI isn't as depending on as few hours as it was this week.

For this sprint the percentage limits were set as follows:

- ○ < 60 %: Bad
- ○ 60 % - 80 %: OK
- ○ 80 % - 120 %: Good
- ○ 120% - 140%: OK
- ○ > 140%: Bad

For the upcoming sprints we will user these limits:

- ○ < 50 %: Bad
- ○ 50 % - 75 %: OK
- ○ 75 % - 125 %: Good
- ○ 125% - 150%: OK
- ○ > 150%: Bad

If these limits were set for this sprint it would mean that our planning for the finished user stories worked out okay. Our planning in total was considering the limits okay but should probably be considered bad since we didn't finish half of the user stories.

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

This week we had estimated the user stories to totally cover 18 points. As mentioned earlier only two of the user stories were finished which in this case led to that only 9 points were done. This week we got 50 % done which means that considering the limits presented below this week's result was bad. For the upcoming sprints we will have to take technical aspects into further account to make sure we create fair user stories.

- < 60 %: Bad
- 60 % - 80 %: OK
- 80 % - 120 %: Good
- 120% - 140%: OK
- > 140%: Bad

Social contract and effort

As last week, we worked together over Zoom in different pairs, and used the screen sharing function to be able to code at the same time. We found a function in Zoom where the screen sharer could remote the computer so the other person could click around and write code in the Android Studio program. This took us one step closer to work in the same environment. It also made it easier to collaborate and to ensure that both persons felt involved in every step. We will continue using this function.

Like we described in the KPI's, we had some problems with estimating the time to each user story. We spend more time than we thought we would do, but with only managing to deliver two out of four user stories. Next time, we hope to have more knowledge about creating better user stories and estimate time for them.

Design decisions and product structure

As we create this app one of our key values is customer satisfaction, which has led to design decisions being heavily based on what the customer wants, expects and requires. In a first meeting with the customer, our mockups were reviewed by the customer, giving us important feedback on design decisions as well as functionality.

In every User Story the design element is core as it drives and directs other parts of functionality. For example, in this sprint one of the User Stories which were to be implemented was "As a user I want to be able to see an advice-page for the students on the homepage to be able to better be prepared when attending the fair". Whilst creating the page, we faced a dilemma. This page was to contain information from the customer's website, which had a lot of text. We needed a way to fit all this text into the app's page, whilst maintaining usability for the customer. We found a solution by grouping the text into segments and creating a sub-menu for the text. By grouping the text with this menu, it is possible to create a page free from too much text as well as too much scrolling. This design-decision directly supports customer satisfaction and value.

As in the previous sprints and as decided in our initial social contract we use Github as the version control software/ platform. We strive to commit as often as possible to make it possible for us to retrieve previously working code in case of unexpected failures. We've experienced with android studio that in many cases the app just crashes when we run it in the emulator without any warnings from the compiler. So in our experience when it comes to working with android studio it is especially important to have frequent commits.

As mentioned we've been very detailed and particular with our documentation during the sprint. One huge advantage of this is that when someone experiences difficulties with a task another member can easily jump over to the branch in question and help them out.

Because of the fact that most of the app is purely design-based the need for testing the code decreases. Thus it is harder to control and ensure code quality. As of this sprint, we have yet to implement a lot of methods with functionality. As we move forward, we will make sure to write tests for these to ensure code quality. This, with the fact that we aim to make the app easily extendable as well as maintainable for the foreseeable future will further improve the quality of the code.

Application of Scrum

The application of Scrum has not changed much since last week. We thought that we have found a way that works for us. We rotated the scrum master, and had two other team members taking the role this week.

During this week we clarified what kind of responsibility the scrum master will have. We decided the main responsibility is to lead the meeting and ensure that our work continues to go in the right direction. We will still create the user stories together and do the estimation as a team. We want the team members to be dedicated to the development of the app and therefore we want to make decisions together. This decreases the responsibility of the Scrum master. This has been working great for us the last few weeks.