TEAM 20, SPRINT 3

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

Customer value & scope

Up until now we haven't had a clear view of our Definition of Done. This week we formulated three criterias for our DoD which will work as a success criteria for the group. These are as follows:

- All java code that can be unit-tested should be (except setters/getters).
- All code has to be documented.
- All new code will be checked by another coding pair to make sure that the code is of good quality.

KPI:s

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

This sprint's estimated total time was 23h and the actual spent time this week added up to 23h which gives us a percentage of 100% and we conclude that this week's planned time was on point and we consider this as a "Good" rating in our Effectiveness KPI. This is not the same as concluding that every user story took the exact amount of time calculated. The user story we ended up rating as largest in terms of time consumed ended up taking a bit less (2 h less) and one of the other user stories ended up taking a bit longer.

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

This week we estimated the user stories to totally cover 19 points. We had to revise our estimated points due to some user stories being more difficult than anticipated. After a

reflection together in the team we measured the user stories to have a total of 19 points, where some of the user stories lost some points and some gained some points.

We had some problems with one part of a user story. That user story was graded with 5 points before, but after the sprint we think it is worth 6 points instead. As we did not completely finish this one, we only got 3 out of 6 points. So as a result we did not get 100 % of the estimated points this week because we only finished 16 out of 19 points. This led to 84 % coverage and is considered “Good” in terms of our Productivity KPI.

Last week, we estimated the user stories to cover 18 points but we only got 9 points from them. So we had nearly the same difficulty this week, but the result was better. This week was an improvement for the team.

User Satisfaction: A measurement of satisfaction from our stakeholder.
We have agreed on two questions we will ask our stakeholder every sprint on her view of what we decided in the beginning of the sprint. The stakeholder will in turn grade these questions from 1 to 10 where 10 is optimal and 1 is “in drastical need for change”.
The questions are:

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

This week we haven’t done anything that she can see functionality in, therefore only the second question was relevant for her to answer. She gave us 8,5/10 for user interface convenience. She considered the logos showing in the company list giving a great touch and that they help you recognize the companies real quick. She thought that the new design on the company detail page was good overall but that it could be complemented with some more colors and frames.

Social contract and effort

In the beginning of the project we wrote a social contract for the team. So far we think that the teamwork has been great and the social contract hasn't been used. In our social contract

we had set up a couple of goals regarding our personal development in agile practices as well as delivering a product that both we and the product owner can be proud of. So far we feel like we are on the right path to achieve our goals, we're on schedule when it comes to the app's evolution and we've learned a great deal of agile practices. Our user stories now compared to the first sprint have become much better which increases our probability of achieving each sprint's goals as well as working in a much more efficient way as a group.

Since the social contract was written in the beginning of the project before the sprints started we didn't have much knowledge about Scrum and agile practices. Therefore we think that we need to readjust the contract and add information about the Scrum masters. Every sprint has two Scrum masters that share the role for one sprint, and every new sprint we choose two new Scrum masters. In this way everyone in the team can try out the role as a Scrum master. We shall add this to the contract and also specify what the Scrum master is expected to do.

Design decisions and product structure

Last sprint we had two user stories that were not completed. These user stories touched the same area and none of the coding pairs could solve the problem. When we planned the user stories for this sprint, we had to discuss whether or not to rethink a new way to solve the issue or to continue down the same road. This issue regarded the structure of how the lists would be presented on the search page and on the my-page page. Since it was a central and important part of the visualisation we prioritized these user stories and decided to stick with them this sprint as well. We rewrote them into one user story. Due to the fact that we still thought it was the best solution to go with, we tried with the same user story this sprint and as it happened to turn out, it worked. The coding pair that worked on this user story found a solution. With this said, we therefore chose to stick with the same design decision and it resulted in a huge win. It will have a lot of customer value as it is something that later on will be used when navigating through the application.

This sprint we have also had a longer discussion regarding how to ensure code quality. With some help and tips from our supervisor, we have decided to start with pull requests from the next sprint. We will use pull requests and let others in the team read and double check the

code we have written, it will thereby be easier to detect errors and mistakes that could help us avoid problems in the future. Since we decided this in the middle of the sprint, this is something we will start using in the next sprint.

In order for others to easily understand code that they didn't write themselves we plan to write Javadoc for all of our logical functions. By writing Javadoc we as a group can quickly and easily get a grasp of what a particular method is doing. The same goes for our external stakeholders if they choose to receive what we have done. However, previous weeks we have not written proper Java documentation for all of our methods, so starting this week we tried to write Java documentation for every (logical) method we implemented. But we still want to become better at this, we went back to write Javadoc for some of our previous sprints but there are still some methods, classes and constructors that are missing. To ensure that in the future everything is properly documented we aim to change our merging strategy to always review everything before merging but also to keep reminding each other to write Javadoc. By doing this, others in the group, and potentially even the stakeholder in the future, will have a much better time understanding the function of every part of the code and make our workflow more efficient.

Application of scrum

The second sprint was challenging and we ended up not finishing two out of four user stories. This made us evaluate our way to create user stories and plan sprints. Therefore, we focused on creating a doable and good sprint for this past week. With help from our supervisor we formed a new way of thinking and tried to be more precise when we estimated the work each user story would require. This helped us create a great user story which we to most part managed to carry through.

For future sprints we got the advice to not create user stories which are adapted to a specific amount of workload and also totally independent. Instead, we will from now on create more and smaller user stories that aren't totally separated from each other and which in the end add up to the desired workload.