

Team Reflection: Week 6

Team Ginger

Customer Value and Scope

- *The chosen scope of the application under development including the priority of features and for whom you are creating value*

In this sprint we have continued working with both code and visual elements. We have completed the animation for when collecting trash and also created icons for different kinds of categories of trash that you can pick up. We have also worked on implementing analytics and history over how much and what kind of trash the user has been picking up. The achievements framework from last sprint has also been implemented.

- *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)*

Nothing has changed in the success criteria. All of us are becoming more proficient with the tools, languages and API's we are using. We are also becoming more efficient during our meetings.

- *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*

Like last week some of our user stories will run across several sprints. This is even more clear this sprint. Our issue seems to be that we finish the user stories, or at least the coding part. But we never really seem to be able to test it in time. So we will have to review our size-estimations a bit and maybe change them. Or maybe change the way one handles a user story. Because we have had it happen a few times now that a user story was "done" but untested. So maybe we could say that the larger user stories are to be tested in the coming sprint instead.

- *Your acceptance tests, such as how they were performed, with whom, and which value they provided for you and the other stakeholders*

Once again, like last sprint, we performed all our acceptance tests together at the end of the sprint where we reviewed the code of each finished user story and its acceptance criteria.

After evaluating this week's KPIs we got the following result:

- Developed features per sprint/Planned features per sprint = 5/11
- Satisfaction level with project progress per team member 1-5:
 - Average: 3.57

- Median: 3
- Difference in time spent per sprint compared to target time: average: +1.46h per team member. This means on average every member spent more time than planned for this sprint. This resulted in 9 out of 12 planned features being completed, which is an improvement from last time when the average time spent was similar but the amount of completed features were less in number. Even though there were a lot of features being finished there were still 5 out of 11 user stories that were done. This is still more than last time and this would show that we have learned to better size our user stories since. This is also reflected in the team satisfaction level as it has risen since the last sprint.

Social Contract and Effort

- *Your social contract 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)*

Like previous weeks, we are very happy with our social contract. When we need help we ask for it and so on and so forth. Noone has felt the need to change it.

Perhaps we are lucky to have hit the mark on setting a good social contract at the very start!

- *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)*

Seems like we all kept the trend going this sprint because our KPI's tells us that we put in more time than expected. And we actually got a lot done this week!

Many new features were delivered and this is probably why our time spent was higher. But we also think that everyone is starting to push a bit harder because we are close to the deadline of the project and we awt as many things to be done and presentable as possible. Our goal is to have the app be in a presentable and coherent state.

Design decisions and product structure

- *How your design decisions (e.g., choice of APIs, architecture patterns, behaviour) support customer value*

This sprint we, as mentioned before, added many new features and of course this also means that a few new component libraries were added to our growing collection. One of the new libraries helps display stats in the form of bar graphs and circle graphs which is an immediate feedback value to the customers!

Besides the new libraries an architectural change was made to what we unofficially have decided upon. This says that we should have index files for

larger modules to make importing functions from them or connected modules way easier. This way you have all functions that for example, concerns persistence, be in the same place. This supports customer value in the form of us developers being able to use certain functions from different modules much easier which in turn allows us to produce new features faster.

- *Which technical documentation you use and why (e.g. use cases, interaction diagrams, class diagrams, domain models or component diagrams, text documents)*

We keep making minor changes in the way we use meeting minutes, but now they are tiny, like whether we should use tickmarks or not. It feels like we have settled upon a form that suits us all. We added some documents that explain how the achievements are going to work i.e the badge, the conditions to achieve it, title and such. This is very useful as it makes adding new achievements to the app easier since we can just keep on adding to the form and then add it to the app more or less directly. Besides this not much has changed.

- *How you use and update your documentation throughout the sprints*

As mentioned in the previous question, we usually settle on some template for documents that are going to be reproduced and then, when we see fit to change them, we do.

- *How you ensure code quality and enforce coding standards*

We use the previously mentioned prettier-code formatter and we also have a standard on how one should comment their code. It is easy to reproduce since it has been written in one file and we can take inspiration from that to reproduce it in each file.

Application of Scrum

- *The roles you have used within the team and their impact on your work*

We continue our work with no officially assigned roles as this has worked for us thus far. By now everyone in the team has implicitly taken on roles that fit them in the team. We also keep to the practice of having a code review during the last meeting of our sprint by having one member screen sharing.

- *The agile practices you have used and their impact on your work*

There has been no changes to the way we have daily scrums/stand-up meetings with one long review/planning meeting at the end of each sprint. With us exclusively switching to discord since last sprint, we have also expanded our infrastructure there with setting up multiple rooms where those working together on a user-story can work similar to peer-programming with shared screens.

The documentation required in the project is updated continuously and adapted to fit the project development.

- *The sprint review and how it relates to your scope and customer value (Did you have a PO, if yes, who? if no, how did you carry out the review? Did the review result in a re-prioritisation of user stories? How did the reviews relate to your DoD? Did the feedback change your way of working?)*

In this review we decided to continue as previously planned. We want to fix some bugs so that the app feels polished when it's finished next sprint. We have also decided to skip some unnecessary and hard testing in order to have time for more productive tasks.

- *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)*

Nothing has changed on this point.

- *Relation to literature and guest lectures (how do your reflections relate to what others have to say?)*

We have not had any lectures or read any literature. We would like to mention, however, that last week we remembered that in one lecture we saw that a scrum board contained a "On holiday" square. We used this when one of our team was away on holiday.