

# Team Reflection: Week 5

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*

This sprint we have continued with some animation for when collecting trash and also continued with making a list of achievements. The achievement framework is still in progress and the UI now exists to register a couple of trash types and how to collect them through a popup window. What we would like is to continue on our work and finish our user stories and finish our tests. We realised that we do not have a lot of time left so what we need to do now is to prioritize what kind of features we want in our application and remove those that are least important. We chose to not prioritize accounts and login/registration since it would not provide value until a couple of weeks later.

- *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*

Some of our user stories this week are combined with some continuance from last week's sprint. But the newly created ones are in the same standard pattern we have done previously. The difference this week is that we realised that we have adapted the size of each user story depending on the team member that it has been assigned to. We have not focused on the task breakdown because most of our user stories were quite clear using the acceptance criteria. The effort from each team member has been divided as equally as possible.

- *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.

- *The three KPIs you use for monitoring your progress and how you use them to improve your process*

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

- Developed features per sprint/Planned features per sprint = 3/7
- Satisfaction level with project progress per team member 1 - 5:
  - Average: 2.83
  - Median: 2.75
- Difference in time spent per sprint compared to target time: average: 0 hours per team member. This means we spent as a team approximately the planned amount of hours working. Although this sprint one of our members was off on a business trip so this result is calculated for only the six members of the team that have been actively working this sprint. The time result, including our non-contributing member, would instead be an average of: -2.82 hours.

From this sprint's KPI we can see, with the exception of one member, that we have spent approximately the planned amount of time, which is an increase from the former sprints, yet the satisfaction with the project has gone down as well as the amount of features completely developed.

Judging from the last sprint, as we managed to develop almost all the planned features yet still spent less hours than planned, one of the reasons for this is that we made a few of this week's user stories too large and thus not being able to complete them all on time, reducing the satisfaction level of the group.

## **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)*

We have, just like previous weeks, had no reason to change our social contract.

We have a very open and supportive environment so it is still working well for us.

If we do discover a reason to change it we will have a meeting about it before making any changes.

- *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)*

This sprint one of them members was away on holiday, but we had already planned for it so it was no issue. Now, for the time spent on the course: we achieved the expected amount of time! Everyone spent more time on the project, but an interesting aspect to this is our Planned Features Per Sprint-KPI (PFPS), as discussed in the point "*The three KPIs you use for monitoring*". Even though we

spent enough time this did not immediately correspond to a larger amount of completed features. One reason for this is that we did not expect to finish at least one of our user stories (#36 Achievement Framework) in one sprint, so that drags the result of it down. Another reason was that one of the team members' internet went down and it hindered efficient progress. A third one was that a previous user story had to be reworked.

It was good to hit the mark this week though, and we will keep putting the effort in like this, because it feels like we have gotten into the swing of it now.

## **Design decisions and product structure**

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

We previously focused on getting user authentication and remotely stored user data. This has been scrapped in favor of features that provide more direct customer value. We will not have the time to implement the features dependent on authentication.

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

We are still using our meeting minutes, standup meetings, guides and reflections in order to update the team and look after the workflow. This is a model that is working very well for our team, as we have tried many things previously and abandoned the ones that were hard to maintain these are the ones that have remained. These are also the ones that are most likely to change when we feel the need to. If we try something new in a meeting minutes or standup meetings one sprint and then go on and use it the next sprint again then that is a very clear indication of it being a good change, it is a very organic process. This is how we will continue in the coming sprints as well, trying out new things and either bringing them with us or discarding them.

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

We have not updated any of our documentation as there has been no need for it. If the need arises then we will have a meeting where we discuss it and then change them.

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

We automated code formatting on save across our IDEs so that it would be easier to follow the code formatting standard.

## Application of Scrum

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

We still have no official roles and we do not feel it necessary to have any. During our weekly review one team member is screen sharing so that the rest can watch. This overcomes one of the issues of remote working because it allows everyone to see what is going on and to literally be on the same page. This is, as mentioned in the last team reflection, also how we do our code-review. The screen sharing team member also handles the pull requests with the assistance of the other members.

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

We still do what we did last week, “daily” scrums/standups, sprint review where demos of features are shown and how the sprint has gone is discussed more thoroughly so that all members know in what state the application is. This is also when we discuss issues and/or rework user stories that might have to change as a result of the current sprint.

The Discord-server is coming along nicely and we have decided to abandon Zoom as a meeting-platform entirely in favour of Discord. The reason for this is that it becomes pretty messy when we use all these different softwares so we decided to limit ourselves to one. This is working well and we will continue with it.

- *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?)*

This review we scrapped user authentication and remote user data storage as it does not create much direct customer value. We do not have the time for creating the features that depend on user authentication and remote user data storage. We also had a bit of a special situation as we needed to merge code from a branch which did not meet our DoD as testing was not finished. We decided that the user story would be finished in the next sprint but that we would merge the code anyway.

- *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)*

We have continued to work as previously. We share information regarding best practices using Discord or our Facebook Messenger group. We have attempted to distribute the workload using different tools so that everyone can learn something about everything we are doing.

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

We have not read any literature nor had any guest lectures.