DIT 257 Agile software project management
Isabella Fransson

Individual reflection: Week 6

What do I want to learn or understand better?

This sprint has sort of been like a continuation of the last one. As I was expected to finish the user story in two sprints I have kept working with it this sprint. The last sprint I left off at a complete stop because I had run into an issue I couldn't solve on my own. So when we started this sprint I asked Emanuel if he could help me with the async-problem and what turned out to be the underlying issue - that the page wasn't refreshed on update!
Luckily he, Bashar and Benjamin had run into this issue before so it was an easy fix. We also decided to implement the observer pattern, something which I had failed with previously as a result of messy code and a lack of knowledge.

What stood out the most the previous sprint was that I have a hard time asking for help. I'd rather spend hours trying to solve a problem on my own because I don't want to give up. This is ridiculous and ineffective when working towards a deadline. I set a time-limit for trying to solve issues on my own and if I hadn't figured it out by that time I would either do something else or ask for help. Sometimes all you need is to take a break because staring yourself blind on a problem usually won't solve anything.

This is something I would like to keep practising. To either ask for help or take a break, because I do feel like this is something I will have to put some effort in to maintain and re-learn.

How can I help someone else, or the entire team, to learn something new?

I actually taught the team something new on the sprint review! I had found the function foo.some(item => item.property === outsideProperty) and I found it really neat as we are working a lot with lists of objects and we often want to access or compare a certain kind of property of the object. I used it frequently in the achievement framework!

It works really well because the majority of the team is present at the sprint reviews so you know that everyone hears what you are explaining and they are able to ask questions. During the sprint review, we also have out code review and during this, we are able to learn a lot from how everyone else writes their codes. So if I find something interesting or confusing in someone else's code then I can ask. This serves two purposes, one is that I learn something new but also, by asking, the rest of the team hears the answer as well and can ask follow-up questions or simply get a refresher.

What is my contribution towards the team's use of Scrum?

The last sprint retrospective was very useful for how what and how we all delivered this sprint. I feel like we all got a bit more serious and focused more on just how many features we could churn out to the user that would be of user value. Besides that, I have been doing the regular, helping drive the meetings in the right direction and the things I have described in my previous reflections.

What is my contribution towards the team's deliveries?

I have now finished the achievement framework! It works well. When you collect one or many trash of one or many different kinds then the achievement framework is signalled and will the go on to fetch the stored data, separate them into categories by type and returns a list of all of the categories with the corresponding amount of collected items for each one. Using this list all achievements will be looked through to see if any of the conditions to grant an achievement is met or not. If they are met then the id of that particular achievement will be stored to the persistent memory. When a new achievement is granted a signal is sent to the achievement-screen to signal it that it is time to update. It will request an updated version of the acquired achievements and the achievement framework will match what is stored in memory to the actual achievements and pass that list to the achievement page!

What I would like to do for the next sprint is to make it possible to remove granted achievements, as we have now implemented the feature of being able to remove previously collected trash.