DIT 257 Agile software project management
Isabella Fransson

Individual reflection: Week 7

What do I want to learn or understand better?

So, the final individual reflection.
I finished and updated the achievement-framework this sprint. Users are now able to gain achievements from reaching a certain level. We don't have an achievement like that but at one point we did, so we all agreed that this was a feature to be implemented anyways. I also wrote tests (have only done it once before, in a course almost two years ago!) and updated, added and edited the comments to my functions.

Turns out, during the code-review, that I had misunderstood how you were to describe async functions. I had put parameter descriptions like:
* @param {Promise <Number>} id
For all async functions. Emanuel explained that were I to look at the function, then I never treated id as a promise and thus it should be excluded from the description. This was really insightful for me!

I practised what I preached in the last sprint review. Whenever I hit a problem I couldn't solve in a reasonable time, I did something completely different for a while. It worked great and this is something I will bring with me, not only to other projects but to other courses as well.

My reluctance to ask for help or give up is actually one of my greatest insights during this course and I have a feeling that this is something I will have to work on for the rest of my life. Many of us are brought up to be very independent and many have an ingrained belief that asking for help is something akin to admitting defeat. Like it places a big red stamp on your forehead that says "I know less than .. ".
Maybe it's the ego speaking, I don't know. What I do know is that, by asking someone that knows more than you on a subject for help, you will learn what they know at a much faster pace than you would by trying to figure it out on your own.

I'm not saying it is bad to try and figure things out on your own, it is very beneficial when you do in the right circumstances. But if you lack that sort of boundary or limit where you can recognise something as "Too Difficult to solve in a Limited Amount of Time" you could stagger the progression of an entire project. Now, luckily for me, I acknowledged my issues before it became a project-wide concern.

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

Were I to look back on my contribution to the team learning new things I have been "lagom". At the beginning of the course, I wrote a document detailing how one should go about getting the development-environments to work properly as some of our team members were having issues. I also created the document "Useful links" but this was created just prior to me creating our Discord-server, to which all of our interactions and documentations moved. So the document itself is abandoned but the practice of posting interesting and useful links lives on in the server!
Besides that I answer questions in the chat when people have issues, posting links and telling the team about interesting things I have discovered.

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

The majority of my contributions have been centred around creating user stories and their corresponding tasks and acceptance-criteria. This has involved discussing back and forth with the majority of the team about how one should name user-stories properly (surprisingly difficult!) and what it should actually entail. It has been a hard balancing act trying to provide customer value when we all, as developers, strive to create a robust and modular foundation in which to build our app (very vertical slices).

It sort of feels counter-intuitive to hardcode values and actions one sprint which are to be changed in the next sprint just to provide customer-value. But I do understand why it is important because you will *always* have something new to present to the PO and stakeholders and such. This bit I feel like I struggled with the most. After I got used to having a more customer-focused approach it became easier.

The Scrum board has been very useful for me! I was always able to see what the others were up to and at what point in their development they were. It is a great tool and I will definitely bring it with me to future projects.

What is my contribution towards the team’s deliveries?

As I mentioned in the first question, I finished all that I set out to do the last sprint as well as implementing that a level increase can grant achievements and also I wrote tests.

I struggled a bit with writing the tests as it wasn’t as straight forward as I thought with async functions. It was one of the times where I had to take a break...
In the beginning, I tried to test it through the collectTrash-button, which ended up causing a lot of issues with persistent memory. Eventually, I realised that I should test my functions directly, i.e via:

```
await notifyAchievement([{
    amount: 10,
    id: "3af87f4f-a83b-4ff1-8f42-393f4ca15154",
    time: 1602486346740,
    type: "bag",
  }
]);
```

Closing Comments

I have learned SO MUCH during these weeks! I hadn’t written a single line in javascript before and now I have written over a thousand! I have learned about the versatility of react native and it’s component-libraries, how to store something persistently, got an understanding what JSON does and just how *annoying* JSON.lock is because it always seems to have one or thirty conflicts with master…

I have learned how a standard development process can look from start to finish, with scrum-board, user-stories, code-reviews and daily scrums to mention a few. It has given me a greater understanding of what might be expected when I graduate and start my first job as a software developer.