

Team reflection week 6

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 week we have accomplished a usable app. The backend works as intended and the visuals are according to Hasts preferences. These have been developed together with Hast as our primary objectives for this sprint. We have prioritized very well this week and focused only on the points needed from discussion with Hast. Next week we would like to repeat this process with Hast and create more defined and valuable user stories. This will be achieved by creating a more detailed description of each user story although we find it to be very difficult to plan each user story in detail beforehand. We find that being agile during the sprints as a useful skill.

- **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)**

The user stories in this sprint were all done in time, but when we merged one thing stopped working. This is something we will need to handle. Since Dart and Flutter are still quite new to all of us, we programmed in teams to favor the process. This also favored the learning outcomes and teamwork. The effort around the team feels equal, everyone was part of a user story and each user story was equally difficult. We would like to keep programming in teams and keep user stories at the same level.

- **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**

Our user stories could be more detailed for the following weeks. Our acceptance criterias were followed well this week. We were better at writing specific tasks on the user stories this week, which made the process easier. But this is something we still can improve. We are a bit lazy with commenting the code. This is something we need to think more about next sprint, especially when the code is getting more and more complex.

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

We always have someone else in the group look at the code before the user story is defined as finished. This can be done more structured/formal, but for now it works.

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

Did we complete all our planned user stories in time (0-100%)? 100%

1. How satisfied are we with this week's work (0-10)? $62/70 = 89\%$
2. What has my stress level been this week (0-10)? $23/70 = 33\%$
3. Have I contributed and been listened to in this sprint (0-10)? $45/70 = 64\%$
4. How happy have I felt in this sprint (0-10)? $57/70 = 81\%$

Number 3 was a bit low this week. This is because some feel like they haven't contributed as much as they wanted. This can be because we had pretty big groups this week, and that makes it difficult for everyone to be very active and contribute a lot. Maybe we should have a little bit smaller groups next week, this week we had groups of 4 and 3 instead of 1-3.

Otherwise good numbers this week. Keep it up team, YAY!

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)**

No changes made since last week. No conflicts have risen.

- **The time you have spent on the course and how it relates to what you delivered.**

The time spent so far on the course feels sustainable. One big reason for this is thoroughly planned meetings, which makes the time feel valuable and productive. We have introduced timekeepers and "bullshit detectors" for meetings so that we stay on track. We don't track our hours because we thought that the value wasn't greater than the cost.

We feel that we have made a lot this week, the backend works and the UI looks great. We look forward to showing this to Hast on monday.

Design decisions and product structure

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

This week we have implemented providers architecture patterns for state management. We needed something to manage our states in the application so we read the Flutter documentation and saw that this is the appropriate way to do it. These helped us remove call back functions, helped further improve the backend and reduced coupling in our project structure!

In the current state doesn't add any customer value, but this will enable us to more easily add customer value and maintain the code in the future

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

How you use and update your documentation throughout the sprints

Currently we have comments in the code and we use figma to create visual examples of the application.

We need to update the documentation somewhat from now onwards as we have started implementing the backend properly. As the program grows we need to continue commenting the code and reminding each other to do it more.

We also need to use consistent filenames, variable names, and class names. We will discuss this further in the next sprint. We removed the requirement for credits in each file.

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

Currently we enforce comments on public classes and functions, as the app isn't particularly complex we have yet to implement any unit testing. We wish to have tests for the application to be sustainable in the future, so as development continues we will try to test it.

We have added a column on our scrum-board that is called test-run, where we place the user stories that are "finished" but haven't been run on another computer and looked through by someone else.

Application of Scrum

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

We have four roles in our group

Scrum Master - Responsible for keeping the meetings running smooth. The main “talker” when we have meetings with external parties as well for internal meetings.

Product Owner - Develop the user stories together with the group and make sure that the work continues. After the changes made this week the role feels relevant again. The product owner demonstrates the code on Mondays. To have someone responsible for demonstrating the code makes sure that someone takes on responsibility for the demonstration and that value is added each week.

Secretary - Keeps protocol of the meetings.

BSSARE & time keeper - Bullshitstopper makes sure we stay on track and on time. This has helped us greatly in keeping meetings on track. So that everyone can have a role we have added a second BSS role, this works well to make sure everyone is busy.

We want to reach a point where all roles are clear and contribute clearly.

We can reach this by continuously evaluating what was good and what was redundant in the roles during our sprints.

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

We have used the agile practices “sprints”, “user stories”, among others to work in a way that increments the total work in a moderate way and enables us to show something at the end of each sprint.

We want to reach a point where we have a steady work flow and get a lot of things done.

We can reach this point by writing crisp user-stories that have clear criterias and purpose.

The user stories have been increasingly more specific and value generating.

- **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?)**

Yes! We rotate PO every week. As the PO in this project is also a developer most of the feedback comes from the sprint review with the stakeholders. We do however reprioritise the user stories according to their wants and needs.

- **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)**

About half of our project group are not as used to working with the preferred tools for the project, so we split the sprint up in groups of those used to the tools and those that are not so used to them.

We want everyone in the group to feel comfortable with the tools.

We can reach this point by supporting each other with great sources on how things work, or through verbal collaboration.

We will try to split the work within the coding groups better so that all members can contribute equally.

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

We agree with literature that states that crisp user stories really helps the project move along in an apparent way. This brings a lot of joy to the group members.

We use a lot of the Flutter documentation to implement features in our project and to learn more.