

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 chosen to implement most of the Weather API:s features for the user. This includes a symbol for the weather condition, and also the outside temperature. These two things have been our main priority this week while working with the Weather API.

We even finished the basic UI. Progress on the more refined UI will be postponed to give room for the algorithmic calculations such as creating schedules in the upcoming weeks.

The chosen scope was a bit vague before this week. At the end of the week a minimal viable product was chosen to make sure certain features get done. The minimal viable product is set to send notifications at a set time and recommend outdoors activities for clear skies and indoors activities for rain.

- 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)

One of our criterias is that we as developers learn how to build an android app. Regarding this goal we're on a good way to achieve.

- 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

This week we decided our user stories on the monday and then how difficult each task would be as well as what each person would work with. This was a good way to divide the work among us which eased our work and created value. We did not focus much on acceptance criteria this week which is something we need to do better next week.

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

We mostly just presented our work to the PO and the developers and then decided on whether or not to accept them. We also presented our work to the stakeholder who could then receive value to see results.

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

created this week for the first time. uncertain that the 3rd should be. KPIs will be used to reflect in the following weeks and assess our outcomes.

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)

There is a [survey](#) you can use for evaluating how the team is perceiving the process and if it is used by several teams it will also help you to assess if your team is following a general pattern or not.

We feel that so far, our social contract has been really successful. Everyone in the group agrees and attends every meeting, in person or online. The environment is welcoming in that way that everyone can express their opinion if they feel like it. Especially this week, there has been really good teamwork within the group specifically with making sure GitHub works with Android Studio for everyone. When we had a meeting and one person was sick and joining the meeting on the computer, we regularly asked that person for his or her opinion, since it can be a bit difficult to add input via the computer.

- 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)

4 team members had a lot of work to do in another course this week in order to meet deadlines. This led to this project being deprioritised. Obviously this is not a wanted situation, and should be avoided in the future. In order to avoid this issue, we should make an attempt to evenly distribute our time each week.

Design decisions and product structure

- how your design decisions (e.g., choice of APIs, architecture patterns, behavior) support customer value

We chose an API from public swedish weather service SMHI, since SMHI is a government agency we retrieve trustworthy data from a reliable source. We have Controllers and views so far, but will start with models soon as we start scheduling.

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

We will start with a class diagram the coming week. Since we had no idea about how android apps are built, we weren't able to make out how diagrams will build up.

- how you use and update your documentation throughout the sprints

We do not check our documentation that much currently other than to check what each person should do, however we try to make sure that we have documentation after each week.

- how you ensure code quality and enforce coding standards

Our biggest focus right now is to make sure that everything works so we do not really have a high code quality. Our plan is to go through our code more towards the end to make sure that the low quality code and small tests aren't there in the final product.

Application of Scrum

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

Since this week tRAINing was of lower priority for 4 of 5 team members the total workhours on this course were lower we had not too many cases in which we could act in our dedicated rolls. Still PO had feedback from stakeholder and scrummaster did some organizing such as booking rooms for meetings.

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

Since this was a very busy week there wasn't much work done compared to last week but dividing up the project as well as having multiple meetings and a meeting at the end of the week as well still helped us with what we did.

- 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?)

Last sprint review we learned that our structure of making tasks as sort of sub-user stories made it a bit hard to follow which user stories we're currently working on. We have therefore changed the method so that we create tasks on each user stories, a feature the tool we use trello, already has.

- 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)

So far the tools used are Android Studio, trello, and github. We have learned to use branches in github to enable each developer to work on each development and not make destructive commits onto the main repository. This knowledge was mostly gained from trial and error, due to some relatively helpful errors when attempting merges, combined with some googling. In the end most of our problems were solved by a .gitignore file.

Searching on the internet for solutions for a basic version of the problem, then taking that solution and tweaking it until it fits the program.

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

We felt a similarity to the guest lecture when he was talking about how it was hard to make estimations on the work. Also we found it hard to show something valuable to the PO when most of last sprint was backend work.

