

# Reflection week 39 HALO

Smith states that reflection is "assessment of what is in relation to what might or should be and includes feedback designed to reduce the gap" (R. Smith, *Formative Evaluation and the Scholarship of Teaching and Learning*, New Directions for Teaching and Learning, vol. 88, 2001, pp. 51-62) which can be boiled down to describing ...

---

... the current situation or "what is" (A),

## Progress

Ok - Slow but steady, a few bugs to address.

## Scrum effectiveness

- People should respond when others ask for help. You should not think you're done for the week only because you've completed your assigned tasks. You still need to respond to and help others.
- 

... what you want the situation to be or "what might or should be" (B), and

- -:-
- 

... a plan for getting from where you are to where you want to be or "feedback designed to reduce the gap" (A -> B).

- -:-
  - People should write more descriptive morning updates to let the right person be able to respond to described problems.
- 

## Design decisions and product structure

- how your design decisions (e.g., choice of APIs, architecture patterns, behaviour) support customer value
  - Collaborative design decisions with PO:s in sprint review and intermediary meetings ensure customer (PO) value.
- which technical documentation you use and why (e.g. use cases, interaction diagrams, class diagrams, domain models or component diagrams, text documents)

- None. Comments in code.
- how you use and update your documentation throughout the sprints
  - We don't
- how you ensure code quality and enforce coding standards
  - Testing the code, make sure it works as we want
  - Should make sure CSS files are separated per component.
  - Remove code that is not used

## Application of Scrum

- the roles you have used within the team and their impact on your work
  - The roles should have a more active role to ensure their respective domain's quality.
- the agile practices you have used and their impact on your work
  - Sprint reviews, team reflections have ensured we stay on track and or change direction for the better without unnecessary delay. However, with frequent meetings comes a significant time overhead which could impact the final product. Meetings are still definitely worth it though.
- 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?)
  - We did good. It was good. We had POs for each user story which provided relevant feedback for how the teams could improve customer value.
- 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 use git for version control. We have found video tutorials to be very helpful in learning the tools we are using for the project. We wouldn't say we have expertise in anything yet, though we have certainly developed some proficiency in the tools we are using as a natural consequence of using them.
- relation to literature and guest lectures (how do your reflections relate to what others have to say?)
  - Malte (SM) has been very irritating in the fact that he brings up people's names to avoid them falling asleep on the other end of the mic. He learned this from the Guest-lecture with the scrum master dude.