Brian Engel

Sprint Review and Retrospective

In the SNHU Travel project, there were 3 main roles in our Scrum-agile team. There was the product owner, a Scrum master, and the development team. The product owner was the initial contact point between the stakeholders and the team. You can see this whenever Christy, the product owner, sets up a meeting with Amanda, the client and stakeholder, about what they are trying to achieve with the travel agency. She also brought Ron, the Scrum master, into the meeting since he has a better ability to communicate with the development team and has a better understanding of their capabilities. You could also see this whenever Christy set up a focus group with the SNHU Travel customers, also stakeholders, to gather information on what types of tools and offerings they would like to see. This is to set up the product backlog and prioritize it.

The Scrum master is the next role in the Scrum-agile team. They are responsible for sprint planning, the daily scrum, backlog refinement, the sprint review, and the sprint retrospective. As we saw already Ron, the Scrum master, can help at the meeting with the product owner and the client to provide expertise. He is also in charge of helping to transform the product backlog into user stories that can be accomplished quickly and ordering them by importance, keeping the daily scrum moving and on topic, and organizing the sprint review and retrospective.

The development team consists of developers and testers. For the SNHU Travel project, the tester, Brian worked closely with Christy, the product owner, to get clarification on some of the user stories so he was able to develop test cases for them. The developers also work closely with the product owner, the Scrum master, and the testers for any clarification they need on the project.

One of the changes to the SNHU Travel project was the transition from a list of destinations to a slideshow. This was overcome fairly easily with a Scrum-agile approach. The product owner just had to change the product backlog slightly, the tester had to revise the test cases to reflect the change, and the developer just had to refactor some code.

Towards the end of the project, there was a significant change to the focus of SNHU Travel towards wanting to focus on detox and wellness packages for their customers. In the Scrum-agile methodology it was not nearly as difficult to overcome as it would have been if using the waterfall approach. At first it sounded like it would be a major change, but as they talked it out, they realized that the tester would need to update the test cases, the product owner and Scrum master would deprioritize some of the other user stories, and the developers would figure out how much they could finish in the time remaining, since the deadline would remain the same. The Scrum-agile approach made a huge difference in how the team operated. If they were using the waterfall approach, they would have to start over at square one.

Communication is a very important element for a Scrum-agile team. Face-to-face communication is definitely the preferred communication method in Scrum. We see this with the initial conversation between the client and the product owner and scrum master. It is shown again with the focus group between the SNHU Travel users and the product owner. This also happens during the daily scrum meeting, when everyone gives feedback on what they worked on yesterday, what they are working on today, and any obstacles they are facing. While face-to-face communication is preferred, there are times when it is not an option. There are a number of emails sent by the team when they could not talk directly to each other. When I was sending emails about specific issues, I copied other team members on them instead of just sending it to the intended recipient, ensuring that everyone was included in the conversation so they would have the information I was requesting, or if they had the information I was requesting they could let me know without having to contact them directly.

There are many different organizational tools that the team can use during the Scrum-agile process. One is the information radiator. It is a large visual display put in a central location that shows the progress and status of all the user stories that the team is working on during the current sprint. There are also a bunch of different software applications you can use such as VersionOne, Rally, Jira/Greenhopper, Rational Team Concert, and Microsoft TFS that will help organize scrum projects. For sprint planning and developing the product backlog, Jira features include prioritization, progress tracking, road mapping, and task management. Almost all of these applications have these features in some way, and most of them also feature a digital information radiator so that if everyone isn’t in the same location, they can still see the progress of the team. You can also use these tools in the sprint review and retrospective to look back at the things that were accomplished and not accomplished during the sprint to help convey value to the stakeholders and team members.

I feel that overall, the Scrum-agile approach to the software development lifecycle is a good way to go for the SNHU Travel project. The project has unknowns in it and is likely to need adjustments throughout the course of creating the application, as we saw with the change to going from a list style interface to a slideshow and changing the content to focus on detox and wellness destinations. If we had used the waterfall method, we would have had to scrap the project each time there was a change to the requirements and go back to the planning phase. There are a few things that would make the Scrum-agile method not work nearly as efficiently as it should. A very large team is hard to incorporate Scrum with, which is one of the reasons Amazon has the two-pizza rule. It also needs all of the team members to be experienced with Scrum-agile and to be committed to working within the bounds of it. Another disadvantage is that there is no clear end date for the project, or a precise project cost. Although this might put a client off initially, it has distinct advantages as well, because you get regular deliveries that you can implement immediately, and not have to wait until the entire project is done, and the main advantage of being able to change things mid-project, and not have to scrap an entire project if conditions change.