Ian Cluett

CS250

Professor Judy Mason

12/11/2024

Utilizing the various roles found in a Scrum-Agile Team was the sole reason for the success of the project, without them things would be much more disorganized. A large benefit of a Scrum-Agile Team is that no one individual has sole responsibility for the team, individuality is utilized more for support rather than expertise. A Scrum-Agile Team’s hierarchy is more akin to a spiderweb than the classic pyramid or tier system, with each role having its own purpose. Starting off with the Scrum Master, their job is to facilitate the flow of agile practices such as daily standups. In one of daily standups at SNHU Travel, the Scrum Master recognized that an issue a team member was having regarding the Product Owners’ absence was leading to a tangent. They respectfully acknowledged the issue but requested it be kept for a separate discussion so that the meeting would end quickly as the standups are not supposed to go on for long. Regarding the Project Owner, their role is crucial for outward communication to the stakeholders as well as inward communication to the team. In the SNHU Travel project, the Project Owner talked with the users of the website to determine how to best introduce new features, then relayed those ideas to the rest of the team who would otherwise be unaware of said ideas. Testers in Scrum-Agile work slightly differently than testers in a regular waterfall approach. Here, they have less of a focus on quality assurance at the end of development, but have more involvement with the developers, pushing a test-driven style of development that introduces quality assurance sooner. During development of the SNHU Travel project, the tester worked with the Product Owner to clarify some user stories so that they could create meaningful test cases as well as showed the database developer in module two how to program databases with testing in mind. Developers are the last role that complete the Scrum-Agile Team, by being able to react quickly and adapt to the new requests given by the Product Owner of SNHU Travel they were able to ablet ship out a successful product.

User stories are one of the fundamental features of the Scrum-Agile approach, they are essentially the “tasks” that get worked on during each cycle. Scrum-Agile utilizes many different tools that allow teams to tackle these user stories in an efficient and successful manner, however the largest tool put into play is in how user stories and deadlines are defined. Agile pushes the idea of shipping a usable product, not a perfect product, because focusing on perfection leads to issues like delays and frustration. Testers work with the Product Owner to find a compromise in the user stories, they reach an agreement on what the specific user story should complete – usually just a few points of functionality – and then the developers program the functionality only to those specifications and no further. If the team decides later that they want to improve the feature, they create a new user story to complete during another cycle. User stories can shift throughout development as well, for example in the SNHU Travel project, there was a user story that outlined a new feature for the website that consisted of creating a list of vacations. The Product Owner talked with shareholders during a meeting and discovered they wanted a few features to be included in the list. Updating the existing user stories was the next step for the team, who discussed the new information during a meeting. After the new user stores were created the tester updated their requirements and the developers took over to complete these new stories.

A Scrum-Agile approach has a few methods for handling interruptions or change of directions due to its open nature and encouragement of collaboration. Utilizing short cycles called sprints, usually about two weeks, allows teams to reconvene and tackle any new ideas or issues that arise during development. Helping tackle this issue too was the principle of transparency, nothing is hidden from specific members of the team so issues that arise are seen by everyone, allowing everyone to have input on how to move forward. In the SNHU Travel project, there was a major shift in development when it was discovered that the Product Owner decided that a slideshow was a better direction to take rather than a list that a user scrolls through. Initially, the team members were surprised and voiced concerns about deadlines and completion. Assuring everyone that the shift wasn’t as major as initially thought, and that deadlines would stay the same, the product owner reprioritized some of the current user stories to keep up with the shift.

One effective method of communication was emailing, this is a straightforward way to get questions answered while having the added bonus of a paper trail to back up anything that was said. There were a few times people needed to get answers, such as the tester needing information from the Product Owner about new test cases they were working on. While I worked as a developer, I had to request information from both the Product Owner and the tester because I did not receive enough information during the daily meeting. I wrote an email and included both of them to encourage transparency and collaboration, and it read as follows:

“Hello Product Owner and Tester,

I was just informed of a change in the new feature we had just been working on during the latest sprint. I wanted to ask a few questions I was unable to get answers to during our daily sprint.

*Product owner:*

You mentioned they wanted the feature to include a new section, did you want us to prioritize this or continue with building the rest of it before working on this? Should this new section become a new user story? It is starting to feel like an epic with all the changes we’ve seen requested after the survey, and the team is starting to worry about meeting the original deadline.

*Tester:*

Once the Product Owner answers the above questions, would you like us to continue sending the files we have been working on or do you want to develop new test cases depending on what the Product Owner says.”

These were simple questions that, while I could have asked in person, I opted to ask over email so that I could have a way to make sure I had the correct information. One drawback to using emails like this is having to wait for a response, sometimes people do not check their email often and one might have to wait longer than is necessary to receive a response. Asking people questions face to face, whether during meetings or by seeking them out during the day, is one of the quickest ways to get a response to questions you might have.

SNHU Travel utilized something similar to a Kan-ban board, where we tracked stories that were not started, stories that were in progress, and everything we had completed using sticky notes on a white board. This board was updated every morning during the daily standup, where each team member would go around and update the team on their progress of a specific task. If a member completed their task, they would move the stickie note with their name and task on it to the completed section, then would choose another task in the higher priority section to start working on. They would write their name on the new task and move it into the “in progress” section of the whiteboard. Communication was also a driving factor for the success of our team, specifically in the communication of our Product Owner. They made sure to update the team as soon as possible whenever any sort of new change occurred, such as when they realized there needed to be a shift in the layout of the top destinations list in the travel project. Individual communication from the testers and developers to the product owner led them to being successful in their own tasks as well, making sure they were working on the correct things.

I believe that a shift to Scrum-Agile was the best approach for the SNHU Travel development project, it allowed them to stay on top of shifting user ideas, to complete tasks on time, and increase collaboration between team members. During the waterfall stage of the team, there were islands of expertise that significantly slowed down development due to specific team members being the only ones able to complete certain tasks. These successes follow the pros I identified for the team, the pros being increased communication, increased collaboration, and better deadline completion. There are a few cons to point out as well, such as the fact that the new approach introduced some major changes last minute to the team, increased the level of communication required for the team which some found difficult, and it introduced a level of uncertainty to the project that some team members did not like. Overall, shifting the team to a Scrum-Agile approach was the right move for the team, allowing them to be more successful than if they continued in their original waterfall approach.