Sprint Review and Retrospective

Felix A. Carela

Southern New Hampshire University

Sprint Review and Retrospective

**Roles**

## As the Scrum Master, I facilitated scrum events, removed obstacles hindering the team’s progress, and ensured adherence to Scrum principles. The Scrum Master was also responsible for ensuring the team was focused and motivated, and ensured collaboration and communication. Specifically in this project, the Scrum Master was involved with communication between the Product Owner and the development team about implementations of user stories. I acted as a facilitator for the development team, by ensuring the user stories were going to be possible to complete during a sprint. The Product Owner was more involved with communication from the stakeholders to the team. She was successful by providing clear requirements, and making crucial decisions regarding the product.

The Development Team was responsible for designing and developing the software, while the Testing Team was responsible for testing a wide range of test cases. Although these two teams had different roles from the both the Product Owner and Scrum Master, they still were responsible for applying agile principles. For example, they were responsible for contributing in the daily standup meetings, Spring Planning, Sprint Reviews, and Sprint Retrospectives. They were also responsible for the entire project succeeding, which meant they had to do a lot more than just developing or testing within their area of expertise.

**User Stories**

The Scrum-agile approach played a significant role in completing user stories effectively. This approach works by focusing on delivering small, but frequent valuable increments. The user story for the SNHU Travel development project had a user story, which required there to be a top 5 destinations list. This user story was able to be completed in a 2-week sprint since it did not include any additional features. By being specific and as simple as possible, this user story allowed the developers and testers to be able to focus and therefore increasing productivity. The sprint backlog also helped plan and manage tasks, which resulted in the completion of user stories within the specified time frames.

**Project Interruption**

The Scrum framework allowed for a structured process to address the project interruption, and any other changing circumstances. When the client realized they wanted a top 5 destinations for detox/wellness travel, it initially created fear among the developers since they thought the entire user story changed. With the help of the Scrum Master other user stories were deprioritized, which allowed the developers and testers to focus on the new user story. The Scrum-agile approach allowed for an update of the product backlog which allowed for a smooth transition into the new user story.

**Effective Communication**

The Scrum-agile approach promotes effective communication by encouraging collaboration among team members. Effective communication is achieved through a variety of different approaches which can be used in different circumstances. Face-to-face communication is effective, but when not possible it is also effective to use video conference tools and even instant messaging platforms for circumstances where a lot of information does not need to be communicated. I encouraged effective communication not just by encouraging active participation, but by also creating a safe space where team members’ ideas and concerns are both respected and appreciated. This is important to implement since you cannot encourage participation if team members are afraid to share their thoughts in the first place. When team members feel appreciated and respected, they are more likely to not only want to participate but also to be more creative.

**Tools and Principles**

There are many organizational tools and Scrum-agile principles which helped the team be successful. The use of a burndown chart is effective for monitoring the progress of a Sprint and helped identify any deviations from the planned work. As the Scrum Master I maintained the Burndown Chart, I used it to identify any potential bottlenecks and adjust our efforts accordingly. We also used a Scrum board to visualize and track the progress of tasks, user stories, and impediments. It also helped enhance transparency and facilitate collaboration amongst the team.

There were also many Scrum events that helped my team be successful such as Sprint Plannings, Sprint Reviews, Daily Scrums, and the Retrospective. The Sprint Planning ensured collaboration which yielded well defined requirements for the Sprint. The Sprint Reviews improved the team by having team members share what could be done better, and therefore, improve the effectiveness of the team for the next Sprint. The Daily Scrums were shorter meetings that allows team members the opportunity to share conflicts and solutions to problems while they were still relevant. The Retrospective is a conglomerate of all the other meetings where problems and solutions are analyzed in more detail.

**Pros and Cons**

The Scrum-agile approach for the SNHU Travel project was effective since it provided collaboration, flexibility, and iterative development. The emphasis on frequent communication and cross-functional teams improved the collaboration in the team. The flexibility of the Scrum-agile approach helped the team maneuver through the changes, such as when the addition of the addition of the top 5 detox/wellness relaxation user story was added during a sprint. Iterative development is another effective Scrum-agile approach that provides continuous feedback to stakeholders, and breaks down the project so the team members are able to focus on smaller tasks. The Scrum-agile approach did also present some cons, such as the learning curve of implementing it. Shifting from a waterfall model to a Scrum-agile model requires not only knowledge of agile but also an open mind to try something different. Another significant con of implementing agile is the amount of documentation it requires. Corporate level projects almost always require extensive documentation, but the agile model documentation is different and focuses on aspects that may be unfamiliar to people who are used to the waterfall model.