**SNHU Travel: Sprint Review and Retrospective**

Arica N. Bryant

Southern New Hampshire University

CS-250 Software Development Lifecycle

Professor Tammy Morrison

**Sprint Review and Retrospective**

All the various roles of the Scrum-agile team contributed to the overall success of the SNHU Travel project in different ways. As Scrum Master, I worked with the development team and arranged scrum meetings. There were Daily Scrums that lasted 15 minutes and allowed the team to update each other on the current progress of the project and any impediments they were experiencing. It was my job to ensure these meetings went smoothly and that everyone attending knew exactly when and where they were happening as well as how they were expected to behave. A Team Charter was created that portrayed the Business’s vision, mission statement, team members and their roles, as well as rules for behavior. It outlined the criteria for success and key project risks.

The Product Owner was responsible for making sure the value of work done was prioritized correctly within the Scrum Team. They were always available whenever there were changes needed and they were ready to answer any questions about the development progress. They facilitated interviews with the users and stakeholders of SNHU Travel to find out what the final project specifically needs to be successful. A product that does not meet the goals of its users is an unsuccessful product. By speaking with the client, they were able to learn that SNHU Travel wanted to offer trendy niche vacation packages. They created User Stories based on the feedback and organized the product backlog accordingly.

The Quality Assurance tester was responsible for creating the test cases needed for the development team to follow. They worked closely with the Product Owner and took the user stories and arranged test cases for them. The tester had a few issues with the user stories at the beginning. For example, the “Set Price Limit” user story needed to include more details on how the listing would look. However, these issues were resolved when the tester reached out to the Product Owner through email for clarification. In the future, we will work on making sure the user stories are more detailed and keep the line of communication between the tester and the product owner more open.

The developer was in charge of creating functional software and implementing user stories based on the created test cases. For the SNHU Travel project, they worked closely with the Product Owner as changes occurred. When the developer needed further information on the current requirements, they also reached out to the product owner. The code they created was functional, correctly commented, and the revisions made were only related to new requirements. Each role was individually important, but they all still required working together and communicating.

Creating a Product Backlog of user stories labeled in priority from large to small helped to determine which user stories were the most important to work on. This was an effective timesaving move as well. Time was spent on the most valuable features first and the less valuable tasks were prioritized further back in the product backlog. Test cases were created based on the submitted user stories. Next, the developers worked to create code and functionality that passed these test cases.

During development, we learned that the client wanted to change the requirements. Based on new data, SNHU Travel found that detox/wellness vacations would be very profitable in the near future. Using a Scrum-agile approach, the new requirements could be modified without jeopardizing the entire project. The tester updated their test cases to account for the new requirements and the Product Owner deprioritized stories in the Product Backlog so that development could focus on the new tasks.

Communication was vital throughout the project. At times, there were not enough details to complete a task and it was necessary to reach out for further clarification. When the Tester needed more details to create the test cases, they sent an email to the Product Owner containing questions such as: “Will the description for the locations be placed below or to the side of the destination picture?” and “Will the price limit button need to be listed alongside other search settings? If so, in what order would be preferred?”. The Product Owner responded promptly and sent a slideshow that depicted how the project should look. Following this example, the tester was able to complete the test cases.

Organizational tools that were useful for team efficiency were information radiators like the Kanban board and Jira. The Kanban board was used to keep track of in-progress tasks. It was utilized during daily scrum events and members could update the board while the meeting occurred. In a future project, the online project management tool Jira would be useful to include in the project. It would allow me to track the progress of the current sprint and see the assigned members from anywhere. Team members could openly edit and update other members on their progress, and it even allows for collaboration on certain tasks. When combined with daily scrum meetings, it would ensure there was open communication and would work well in a larger team.

Some pros of the scrum-agile approach for the SNHU Travel project are that it encouraged teamwork and functional products were created throughout the development process. With a functional product being created before the end of development, this allowed for easier testing and changes to be made. Some cons are that there were a lot of risks involved in the process and it highly depends on customer interaction. When a customer is unsure of what they want in a product it can potentially push back production time.

I believe that a Scrum-agile approach was the best approach for the SNHU Travel project. It allowed for better adaptability and put the needs of the customer first. When changes were needed, the team was able to quickly revise the project without slowing down development. It allowed team members to work collaboratively because no role on the development team was ranked higher than the other. Everyone shared common skill sets and was able to help their fellow team members. The constant communication with the client/users helped give the team a sense of progression and accomplishment knowing that we were working on requirements that are valuable to the user.