**Sprint Review and Retrospective**

Michael Jordan

Southern New Hampshire University

CS-250: Software Development Lifecycle

Prof. Erin Tirrell

April 18, 2021

1. **Contribution of Roles**

The four roles in question are the Product Owner, Scrum Master, Developer and Tester. The product owner worked with the client and their users to gather information that would be necessary to create a useful product, then used that information to create user stories detailing the features and prioritize them within the product backlog, allowing the team to have a clear set of feature expectations. The scrum master worked with the team to create an agile charter that outlined the project goals and team guidelines. This charter ensured that every member of the team knew what to expect from other team members and what they were working towards. The tester worked with the product owner to clarify user stories and write detailed test cases that ensured the implemented features filled their purpose. The developer used the user stories and test cases to develop features and connect them into a working program that satisfied the client and their users.

1. **Scrum-agile & User Stories**

The Scrum-agile approach to the software development lifecycle (SDLC) aims to continuously design, develop, test, and review a program through short iterations, allowing the team to quickly adapt to any necessary changes. These iterations are called “sprints”, where a fixed number of user stories are chosen by the team to be completed within a certain time-box, generally 2-4 weeks, and the requirements for the chosen stories will not change during the sprint. Knowing that the feature requirements will not change and that there is a fixed amount of work during the sprint helped to focus on completing the accepted user stories, specifically the one related to viewing the top five destinations list.

1. **Handling Interruptions**

Because the Scrum-agile approach is adaptive and uses short iterations, it is well suited to handling changes in the direction of a project. Requirements are well defined only as far out as necessary, so that little work is wasted if a change is necessary. Additionally, teams only commit to completing a relatively small number of features at once, so in-progress work is less likely to be scrapped. This approach allowed the team to implement the focus on detox/wellness travel during week five and still deliver a completed project by the initial deadline.

1. **Communication**

The following is a sample from an email send to the product owner by the tester.

“After reviewing your user stories and starting to create test cases for each feature that determine whether the product passes or fails, I have noticed a few places where more detail would be useful to define accurate test cases. I would appreciate it if you could answer a few questions regarding the stories.”

There are several questions following this sample, but these introductory statements make it clear how the answers will help the tester improve the product. They also note that the stories already had enough detail to start on the test cases, but more detail would be even better. This places value on both the work already done by the product owner, and the work done in the future.

1. **Organizational Tools**

The product backlog tool was particularly helpful during the sprint planning event, as it simplified the process of deciding what to work on during the sprint. The user stories were also helpful during sprint planning and the sprint itself, as they provided an overview of the feature, the reason to include the feature, and what the feature needs to do to be complete. One Scrum-agile principle that helped was building in quality (Cobb, 2015) through test cases, ensuring that problems were found as soon as possible. Another principle that led to success was embracing helpful variability (Cobb, 2015) by accepting that the customer needs changed during week five and changing development priorities to reflect those needs.

1. **Assessment of Effectiveness**

There are a multitude of possible approaches and determining whether a Scrum-agile approach was the best approach for this project would be incredibly difficult. However, it was more effective than a waterfall approach, mostly due to a few major factors. First, although the client had a goal in mind, they did not have a clear outline of how to reach that goal. Because of this, drafting the requirements necessary for a waterfall approach would have likely taken a significant amount of time. Second, the deadline for delivering a working product was only five weeks, which would allow very little time for implementation and testing after the requirements were drafted. Third, the need to shift focus to a certain type of travel packages was discovered only two weeks away from the deadline. At this point in the waterfall approach, the requirements would have been completed and it is likely that the implementation phase would be almost over. Considering only the first two factors, satisfying the client with a waterfall approach would be difficult. Considering all three, doing so would be nearly impossible.

**References**

Cobb, C. G. (2015). *The project manager’s guide to mastering agile: Principles and practices for an adaptive approach.* Wiley.