**Sprint Review and Retrospective**

At the end of the sprint, it is the Scrum Masters responsibility to host the sprint review where the team is ablet to gather actionable feedback on what they have completed via showcasing their work. The Scrum Master is also responsible for holding the sprint retrospective so that the team can come together and identify what did and did not go well during the sprint, identify potential improvements, examine how the sprint went regarding processes tools, relationships, and people, and to create a plan for implementing improvements to the way the scrum team accomplishes its work (*What Is a Sprint Retrospective? Agenda & Questions | Adobe Workfront*, n.d., "What is the purpose of a sprint retrospective?" section). During this course, I assumed the role of each member of the Scrum team, therefore this paper will serve as the review and retrospective for the team and serve to analyze how each of the various roles contributed to the success of the project and how effective a Scrum-agile approach was for the project.

The first role we inhabited was that of the Scrum Master who is the facilitator of scrum. As the facilitator the scrum master acts as a coach for the rest of the team and is committed to the scrum foundation and values while remaining flexible and open to opportunities for the team to approach their workflow. The main way we inhabited the role of the Scrum Master was specifically by creating an Agile team charter which helped contribute to the success of the project by setting behavior and communications practices for the team which helped them be more effective in their work. We also examined the role of a scrum master during a daily stand-up for a different project in which they helped their team by being an example for the team by being the first to answer the daily standup questions and helping the meeting go smoothly by clearing up any questions the team members may have as well as designating topics to the sidebar for discussion after the meeting concluded.

The second role we inhabited for the semester was the role of the Product Owner. As the product owner we are responsible with managing the product backlog and employing several techniques when engaging our users and stakeholders to develop requirements for the project. The user stories we created were integral to the success of project as they are the backbone of the software that we are tasked with delivering to the client. Without the user stories, the developers won’t know what to specifically program, making development almost impossible to proceed with. The Testers also will be left with an extremely vague definition of the user’s requirements making test cases very underdeveloped leading to a lower quality of product.

The third role we fulfilled was that of the Tester on the scrum team. The tester helps to ensure that the definition of done is met through the creation of test cases for each of the user stories. We were tasked with creating our own test cases for the user stories to help ensure the projects success. It was at this point in the semester that the project was interrupted and changed directions, however the Scrum-Agile approach allowed for easily adding changes to the project due to it’s focus being on small deliverables. It was quite easy to update the test cases for the user stories after changes were introduced thanks to this and the ability to communicate effectively with the rest of the team via email and daily stand-up meetings.

An example of this communication is the following email we were tasked with creating when creating a journal detailing our work as a tester on the project:

Subject: User Story Clarifications  
Dear Bill,

I am developing test cases for the different features for the SNHU Travel Project using the user stories that you have created. To properly determine the pass/fail conditions I should set for the test cases, I’m in need of some more specific details so that I can specify these conditions clearly. My questions for each case are as follows:

**User Story #1 + User Story #2**

* How will the filter appear to the user? Will the user need to apply filters via a pop-up panel, or will they appear listed next to the listings, or perhaps some other method?
* Is the user able to specify a price range for the price filter?
* Will the user be able to apply multiple filters at the same time?
* Will the user be able to change the sorting order of the filtered list?

**User Story #3**

* Will the user be able to see their previous purchases when logged into their account? If so, in what order shall the purchases appear and what information should each purchase contain on the list of orders?
* How will the system analyze the customers previous orders to make a recommendation? This story feels more like an epic that will need to be broken down with more details.

Thanks,

Vincent

This was effective and encouraged collaboration between team members as the message was short and to the point cutting out any unnecessary elements that would make it harder for the team members to get back to me with the information I requested. Team members are more likely to be collaborative and communicate effectively when it’s easy for them to get the information you need or complete things that need to be done.

The last role we were tasked with stepping into was that of the Developer. Developers are possibly the most important aspect of a project’s success as without them, there is no code and therefore no deliverables. Developers are the ones who bring the user stories to life and as the developer we were tasked with not only creating the initial deliverable but also refactoring our code when the changes to the project were introduced. Developers in a scrum-agile environment aren’t only tasked with producing the code but ensuring that it’s of quality instead of passing off that responsibility to a quality assurance team.

I believe that a Scrum-agile approach was extremely effective for the project, the various organizational tools such as the agile charter, product backlog, and test cases were very effective on top of the daily meetings. The approach also offered many benefits which I felt outweighed the potential cons of an agile approach, despite the potential cons of not having a strict schedule and changing requirements possibly causing problems in other areas of the organization, it left us able to easily handle changes to the project and the ability to break down the project into smaller deliverables so that we could meet the client’s deadline. Due to these reasons, I think that the Scrum-Agile approach was the best approach for the development project.

**References**

*What is a Sprint Retrospective? Agenda & Questions | Adobe Workfront*. (n.d.). Workfront.Com. Retrieved August 22, 2021, from https://www.workfront.com/project-management/methodologies/scrum/sprint-retrospective