**Sprint Review and Retrospective**

**Scrum-Agile Roles**

Throughout this project I took on the various roles within an Agile development team as it transitions from Waterfall methodology to Agile. Starting at the top, the Product Owner is the representation of their stakeholders and voice of the customers. Communicating to the team based on what feedback is given from these sources is vital and in turn, helps the Product Owner maintain and manage the product backlog. The product backlog is key for ensuring project organization and efficiency, and product quality. Parroting information between the team and its consumers and stakeholders, being transparent with all sides makes for a successful Product Owner. In this role I created the user stories from a customer interview and reflected what was made into the product backlog, which was used to communicate the intentions of the product to the team.

The Scrum Master is responsible for ensuring the Scrum events are understood and enacted properly by seeing the team adhere to the proper Scrum practices and rules. (Cobb, 2015). As Scrum Master I facilitated the various Scrum events. For Sprint Planning, scheduling a dedicated meeting prior to each sprint aligns team’s goals and priorities for that section. Daily Scrums are short 15-minute meetings held to keep tabs on how each team member is progressing, keeping the team synergized and on top of any potential issues. During Backlog Refinement, conducting regular sessions for the team to review and update the backlog as needed ensures everyone is up to date with what is to come. Sprint Review is held at the end of each sprint for the team to present their completed work to the stakeholders for gathering feedback. Lastly, the Sprint Retrospective is a scheduled meeting for the team to discuss positives and negatives for the last sprint fostering continuous improvement as everyone moves forward to the next sprint.

The development team consists of both developers and testers. As a tester for the team, I was responsible for testing the product and building the test cases off the established user stories. Test cases are built to clarify pass and fail measures for the features intended for the product. The Tester needs to be in regular communication with the Product Owner and developers during testing to ensure pass and fail measures are relayed and any potential issues can be sorted out through each member’s communication channels. As a developer I needed to be cross-functional and self-organized. Developers in agile are much more active in their role, managing the estimation and planning of their tasks along with the progress reporting. In this role I worked on the code for the product while communicating with the Product Owner and testers for clarification on the vision for the product as it evolved.

**SDLC and User Stories**

The Scrum-Agile approach to the software development life cycle (SDLC) was instrumental in ensuring the completion of the user stories. The emphasis on organization and communication between all members of the team is fundamental with Agile and allowed for the user stories to be completed no matter what direction the project took. The format of the user stories themselves helped in clearly communicating each requirement. When tasked with completing the user stories myself I found how simple this format was to use and reflect the vision of the product laid out by the customers.

**Fluidity with the Scrum-Agile Approach**

The fluidity and adaptability that the Agile approach brings to development is great for when obstacles with development arise or when customer feedback causes a change in the product’s vision. If this project was operated under the Waterfall methodology, changes as seen with the SNHU Travel project would’ve caused severe delays in getting the completed product out to the consumer base. However, by utilizing Agile methodology, the changes to the initial requirements were able to be communicated immediately, user stores and test cases were updated to reflect the new requirements, and the project kept running. This fluidity is the biggest strength with Agile and it showed with this project.

**Communication**

The Agile methodology praised in the previous section is only made possible with effective and clear communication between all team members. The Product Owner needs to properly communicate with the customers, gaining and utilizing their feedback to be put into the user stories. The tester then builds test cases based off these user stories. Continuous collaboration between tester and owner is needed here to ensure the most up to date feedback from the users is communicated to the tester to allow for adapting the test cases. The developer needs proper communication with both the Product Owner and tester to ensure proper development of the project based on all available information they can get from these personnel. This chain of communication is a necessity for Agile to function as a development method as intended. An example of the communication between Owner, tester, and developer can be seen below, when as the developer I needed all updated information to reflect the changes made to the project’s requirements.

*Sample Email:*

*Good morning Christy and Brian,*

*I’m currently making the changes to the SNHU Travel project we discussed. I just have a couple questions to ensure I’m meeting the requirements properly.*

*Christy, can I get further elaboration on what the client is looking for? Specifically, are there any detox locations based on their preferences, or a specific design for the layout? Any information and feedback is appreciated!*

*Brian, based on the new requirements, could you provide updated test cases that fit these updates? This would help greatly!*

*Also, if you both can meet in person to discuss the information I’m requesting to ensure we’re all on the same page with what is needed from one another that would be amazing.*

*Thank you,*

*Steven*

Communication in manners such as the above clearly lays out the needs of the developer to effectively transition to the updated project requirements and offers a way for the other parties involved to give the items needed, as well as any additional feedback they may have for the developer in return.

**Scrum-Agile Principles and Organizational Tools**

As previously stated, Agile is built upon effective communication and coordination between the team. There are tools created for the sole purpose of enhancing an Agile team’s coordination and efficiency during development, Jira is one of these tools. As stated by Chard, E. in 2024, Jira “enables you to plan, track and report on the progress of your work. As you’d expect from a project management platform, you can also allocate tasks, identify inefficiencies and resolve challenges”. A tool like Jira is great for compartmentalizing all information needed for all parties during development. This makes it easier to track progress for projects and team members involved, further increasing cohesion and efficiency within each team.

**Evaluating the Agile Process**

The Agile approach has been praised throughout for its flexible nature, though this aspect does come with a downside not previously mentioned. The flipside to this flexibility is the lack of being able to provide a definitive project completion date. Agile development’s nature is great for constantly tweaking and adapting to the ever-changing views of the customer base, which expands the scope of the project. This makes any attempt for establishing rigid deadlines useless and only impedes on what Agile does best, in adaptation. This adaptation that promotes constant communication is what makes Agile the best approach for the SNHU Travel project.

**Citations**

Cobb C. G. (2015). *The Project Manager’s Guide to Mastering Agile: Principles and*  *Practices for an Adaptive Approach.* Wiley.

Chard, E. (2024). *The Top 3 Benefits of Jira: Why it’s a Must-have Tool for Project*  *Management.* Automation Consultants. [https://www.automation-consultants.com/the-top-](https://www.automation-consultants.com/the-top-3-benefits-of-jira/) 3-benefits-of-jira/