Chada Tech

SNHU Travel Project: Sprint Review and Retrospective

Derek Cun

CS-250

**Review and Retrospective: Applying Roles**

For the duration of this class, we assumed the different roles found in a Scrum Team. The setting of the class was based on a company transitioning from a waterfall methodology to an agile one which involved the use of a Scrum Team. We were tasked as a scrum team to develop an application for SNHU Travel. The roles included were a scrum master, product owner, developers, and testers. This retrospective will cover an analysis of the roles, their effectiveness and how each part contributed to the overall process. In this section, each individual role will be analyzed.

**Scrum Master**

As the scrum master, our job was to help manage the backlog and maintain the current status of it, and keeping the team updated on our current progress. As the scrum master, we helped facilitate communication amongst the entire scrum team and coordinate our main focus or any disagreements. We also planned daily scrums, backlog refinements, sprint reviews, and sprint retrospectives. These activities help keep the team updated on task status, and how we resolved any issues to further increase efficiency. During one of the daily standups, a conflict occurred where a side issue was brought up about another team member always arriving late. The conversation side tracked to another topic the team felt was important, but was not meant to be discussed during the daily standup. As the scrum master, we placed that conversation subject into a side parking lot to be discussed further, but not at the time currently. Facilitating the team conversation helps the team remain objective, efficient, and focus on the highest priority tasks at hand.

**Product Master**

As the product master, we helped defined the project requirements desired by our customer. Additionally, we created user stories from those requirements to add into the product backlog and prioritize the most important ones. This job is very important, as it defines the main goals the scrum team will aim to solve for during each sprint. As the product master, communicate what are the most important desires from our customer to the developers and the entire scrum team. By creating user stories, the development team can understand the information concisely, letting them focus on their main job as developers. We also help resolve any uncertainty in the user stories or project requirements that were defined. For example, during this class the developers were uncertain about a specific requirement desired by the customers. As the product master, we communicated with the customer to gather further details and information so the developers can create exactly what is desired. This is extremely important to ensure we meet the customer’s requirements and work effectively, benefiting the agile methodology greatly.

**Developer/Tester**

As the developer, we are tasked with creating the product that meets the customer’s requirements. Our job was to create a product that solves each user story previously defined. The tester creates test cases for the features created by the developer. This helps quality control the features built by the developer and test that it meets the end user requirements. The development team are the direct producers of the end product, so they are essential to the agile process.

**Review and Retrospective: Completing User Stories**

Completing User Stories is an integral part of the agile approach in the software development life cycle. Often times, building a product becomes very complex and can get lost in details without any planned methodology. Creating user stories helps the team identify all of the desired requirements and prioritize them by importance concisely. The user stories are collected from our customer or end-users, from there, the product owner will prioritize them based on importance from a conversation with the customers. A user story is written concisely following a sentence format as follows: “as a… I want to… so that… I can…”. The goal of framing requirements in this structure is to concisely present the information in its simplest way possible. By doing this, we can break down the requirements separately in the simplest way possible. This helps the scrum team manage their workload more effectively in all ways.

**Review and Retrospective: Handling Interruptions**

During the project, the customer desired to switch the product focus to detox/wellness travel. The agile methodology and framework the team followed, allowed us to easily switch focus for this new prioritization. The team was able to take the already developed code and revise it to focus on the new priority.

**Review and Retrospective: Communication**

As a tester, I reached out to the product owner and developers on the team to validate the Test case scenarios.

“To: Christy (Product Owner), Development Team

To develop accurate and effective test cases for the upcoming features for this new system, we need to get more information for the following user stories. Development team, please verify the test cases for each user story make sense with the feature of the system. Product owner please clarify any requested additional details.

User Story One:

- Clarification needed on how the “Top 10” trips should be listed (same page or slideshow format).

- Preferences on listing order (counting down from 10 to 1 or starting with #1).

- Specific types of vacation packages to be included.

User Story Two:

- Desired column headers for trip details and sorting options.

- Allowing user-specified price ranges.

- Length and detail level for trip descriptions.”

This communication is effective be cause it clearly identifies what is needed from each specific team member. Additionally, it validates the test case scenarios before beginning testing, to avoid wasted efforts or testing irrelevant requirements or features.

**Review and Retrospective: Organization Tools**

One organization tool used in the sprints was Jira. Jira helps organize the tasks planned for a sprint and update the status on each of them. One principal of agile is being frequently up to date on the status of the project amongst the team, which Jira helps you do tremendously. Additionally, Jira helps track tasks’ status during a sprint and acts as an easy way to stay up to date for the status on any specific task. During sprint reviews and retrospectives, we can also easily reference each task and any events that transpired for them with Jira as well, making it a very effective tool to use during the agile process.

**Review and Retrospective: Evaluating Agile Process**

For the SNHU Travel project, the agile process performed fairly well. Designing the new website for the travel agency was a complex task with several requirements, using the agile process helped the team manage the workload simplistically and efficiently. The benefits from using agile were improved efficiency, effective communication, and flexibility with prioritization. One double edged sword using agile is early and frequent involvement with the customer during project development. Involving our customer during the product ensures we deliver a quality and desired product, however, it also subjects the team to changing requirements or priorities. During this project the team faced one change from the customers, however this could be much more in other projects. I think the agile methodology served well during this project, but given the duration of the project and single sprint we used to review it is hard to determine if it was the best. I think if we had more sprints to reiterate off of, we could surely determine if it was the best method. With that being said, I think it functioned solidly in this project execution.