SNHU TRAVEL PROJECT

SPRINT REVIEW AND RETROSPECTIVE

Vernon Dodo

Southern New Hampshire University

CD-250 Software Development Lifecycle

June 22, 2025

# Introduction

Welcome to the SNHU Travel Project sprint review and retrospective. In the context of Agile and Scrum, a sprint review is a good way of looking back to look forward because Agile is dynamic and always evolving. To review the process, we will look at its composite parts and how they worked together to move the project forward.

# Scrum Team Roles

We cannot look at how the process evolved without looking at how the individual roles contributed towards it. Agile, combined with Scrum, is such a cross-functional and multi-skilled process that each role was of pivotal importance to push the project forward.

The team consisted of the following roles:

## Product Owner

As the link between the client and the team, the product owner drove stakeholder engagement on both sides of the project. In our opening scenario of the SNHU Travel project, the product owner, Christy, conducted a focus group as a means of gathering requirements for the project. Once she had the requirements, she could create a product backlog, which was pivotal as a starting point for the team.

## Scrum Master

The Scrum Master played an important role in sprint planning, conducting daily standups, sprint reviews, and sprint retrospectives. Being closer to the team, and organizing and facilitating team activities, the scrum master was an important liaison between the team and the product owner. Being the de facto leader of the team, the scrum master guided the team activities to align with the client’s goals.

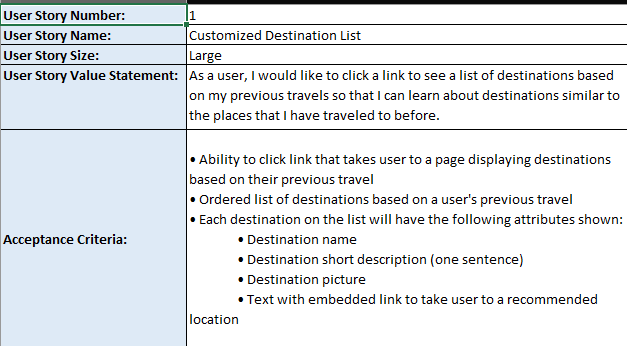
To ensure that the team achieved the objectives of the sprint, the Scrum Master had to:

* create an environment of openness and honesty where team members feel free to contribute
* encourage working relationships between all stakeholders in the project
* encourage ownership in the project
* clearly establish team responsibilities so each team member knows what is expected of them

## Tester

Unlike in the Waterfall approach, where testing took place at the end of the process, testing resources got involved at the start of the project. As soon as a product backlog had been established, and backlog items had been selected for a sprint, testing resources could start crafting user stories that satisfied the backlog items. The added advantage of these user stories, and how they were crafted, was that it assisted the developers in designing their code in such a way that it conformed to user expectations.

Use stories were typically crafted as follows:



## Developer

The developers were not just crucial in creating the code to create the product, but also in assisting the product owner and scrum master into selecting backlog items that could be worked on. At the start of the sprint, some prioritization was done with regards to which backlog items needed to be in the sprint. Together with the product owner and scrum master, the developers then had to decide which backlog items could reasonably be implemented, based on the amount of information available for an item. In the SNHU Travel Project case, it was decided to implement the top destinations list feature first, and later, when there was more information available, to then implement the top 10 and top 5 destinations list by refining the code for the top destinations list.

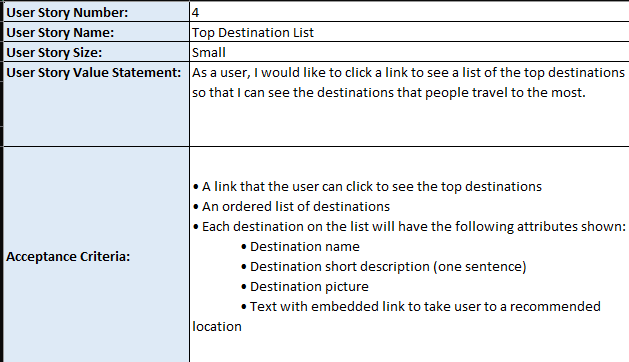
Once they have selected the backlog items, the developers then liaised with the testers, looking at user stories, to clarify what the expected outcome of the backlog item is. The way the user stories were crafted, let them know exactly what the expected inputs and outputs were. Doing this type of test-driven development, allowed the developers to create a product that was closely aligned with user expectations, since both developers and testers worked off the same user stories.

# Completing User Stories

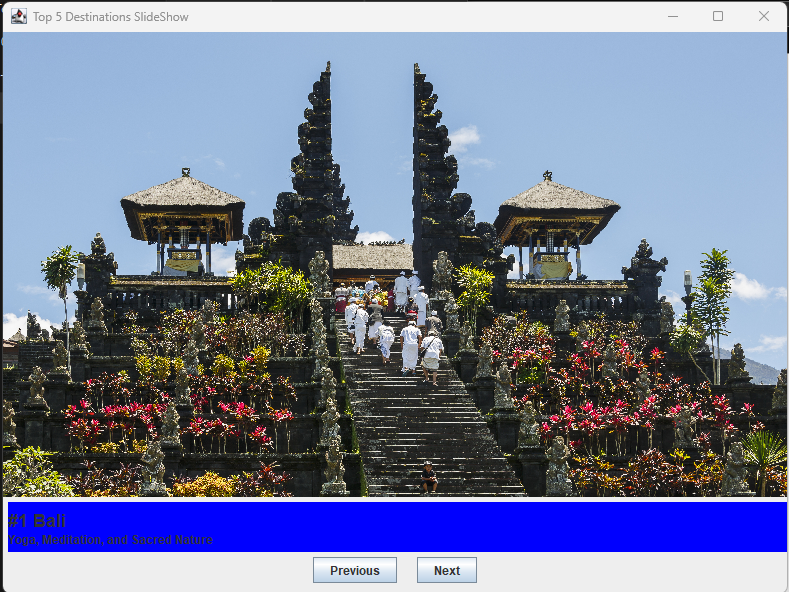
The push to complete user stories was a holistic approach that started with the Product Owner gathering requirements from the client, the tester taking those requirements and crafting user stories out of them, and the developer using those same stories to design the code. This approach ensured that all team activities were aligned to meet client expectations, with everyone working off the same blueprint and having a common understanding of what needed to be delivered.

A typical scenario in the SNHU Travel project went like this:

The tester crafted a user story for the Top Destinations List



The developer developed the Top Destinations List feature.

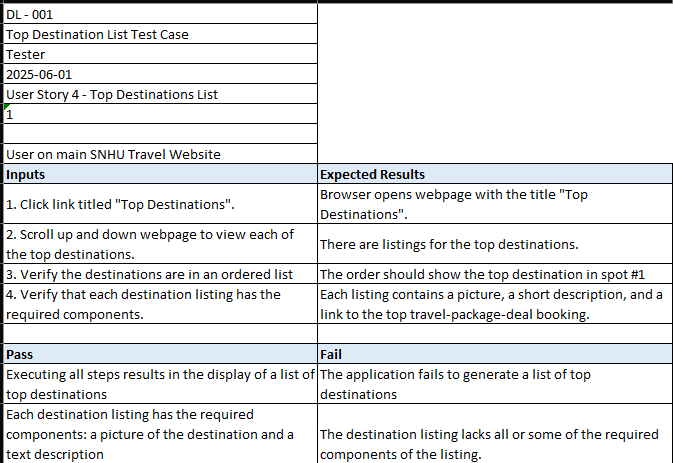


# Handling Interruptions

This project was a perfect example of how to handle interruptions or changes in the project. At one point, the client changed direction on how they wanted the Top Destinations list to be implemented. This change required a revision of the test cases as well as changes to the code because, instead of a list, the client wanted the results to be displayed as a slide show.

Given the change in requirements, the test cases were revised as follows:

Original Test Case



Revised Test Case

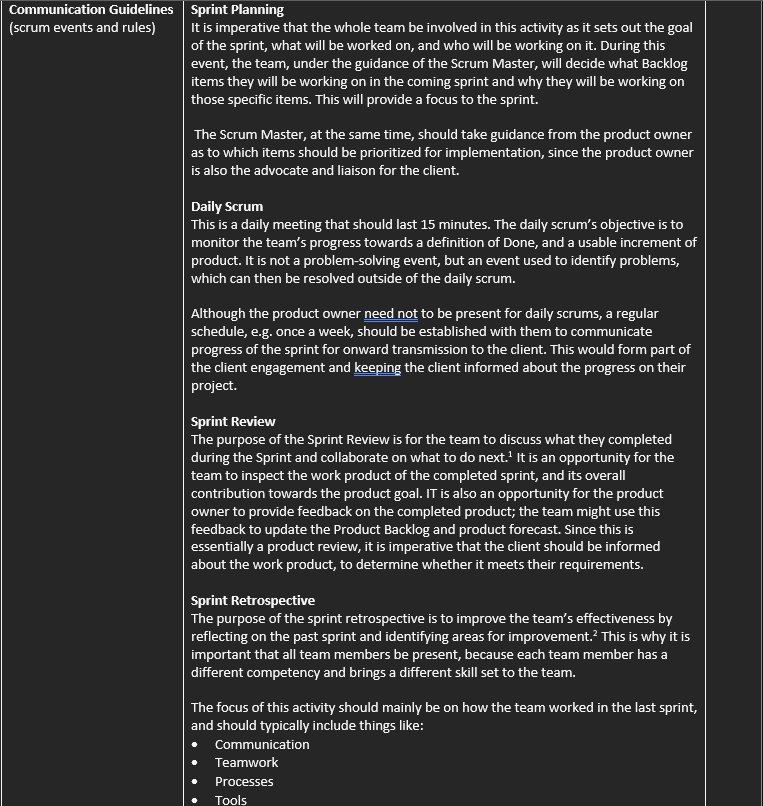
A screenshot of a computer

AI-generated content may be incorrect.

In summary, the changes were handled smoothly, with minimal interruptions to the sprint or project.

# Communication

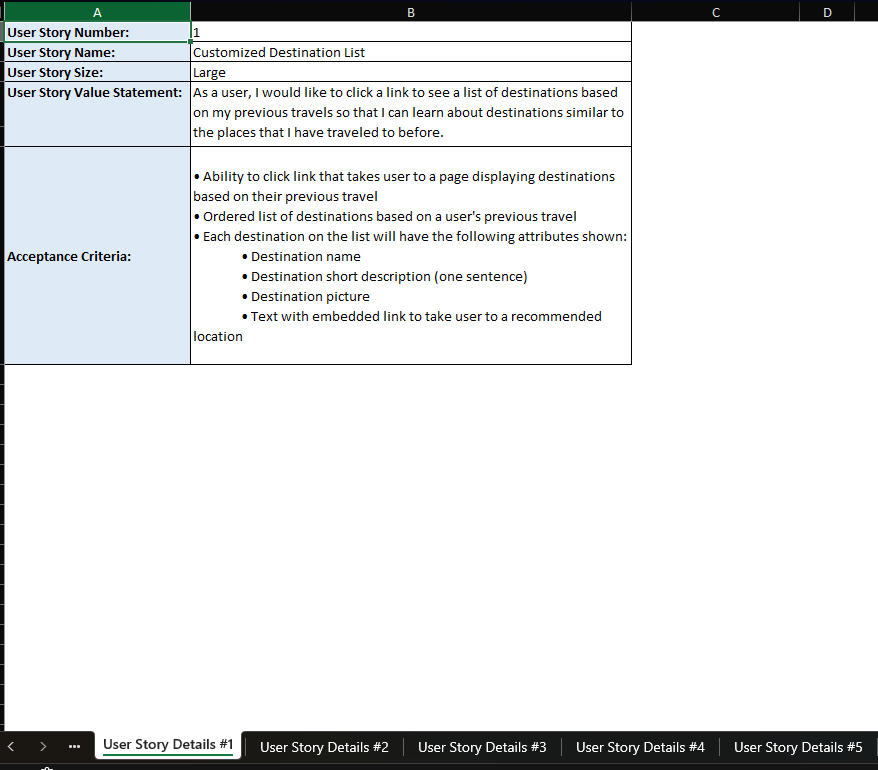
At the outset of this project, it was established that communication was going to be key to completing this project. For this reason, it was included as one of the items in the Agile Charter – see excerpt of Agile Charter.



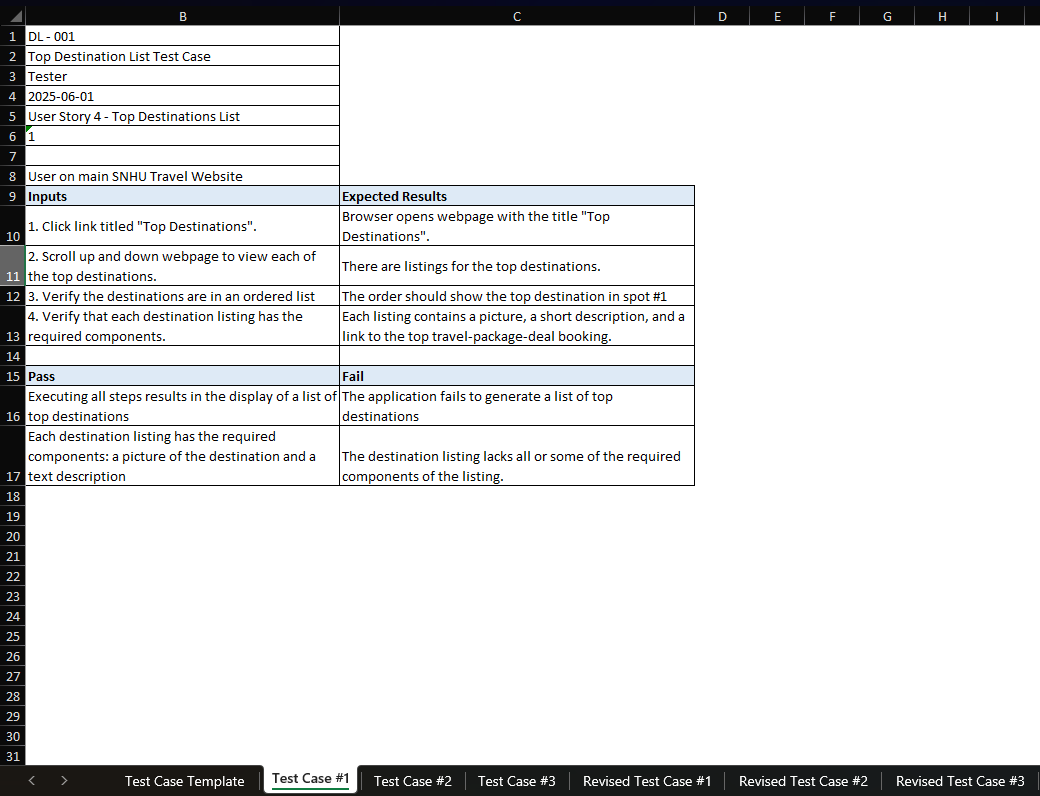
Establishing a communications framework was aimed at the project so that it could include all stakeholders. To give effect to this framework and what was set out in it, the Product Owner conducted a focus group with the client, to determine their requirements.



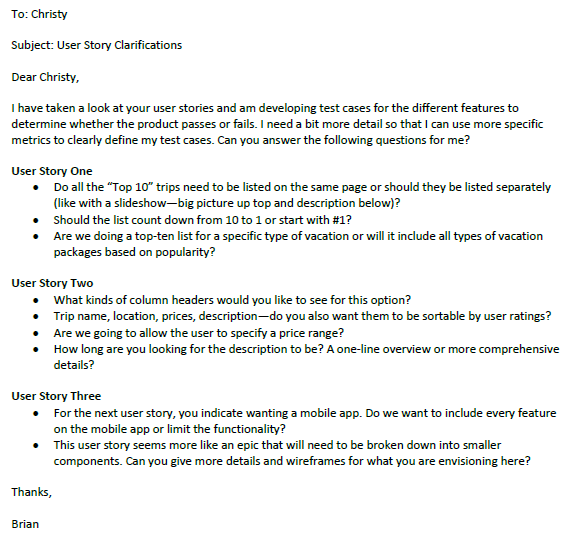
It has to be understood that communication is not just verbal or written communication, but any form of communication that would drive the project forward.

Once requirements had been established, and the a product backlog created, the testing resources, in conjunction with the product owner and development resources, could craft user stories for the various requirements.

These user stories provided a common framework that both test and development resources could use as a reference for their own tasks in the project. Following on from the user stories, testing resources could now use them to create test cases for the various user stories.



Part of a continuous feedback loop, and always keeping all stakeholders within that loop, is communicating with the Product Owner when clarification is needed for backlog items; without clarification, the team would not have had enough information to proceed with an implementation.



Even after clarification was received, the team could effortlessly pivot their efforts to accommodate the new information that they received e.g. the testing resources revised their test cases and the development resources changed their code accordingly. The end result of this alignment of activities, brought about through effective communication, was a quality product, aligned with the objectives of the client.

# Organizational Tools

The one organizational tool that the team used to great effect, was JIRA. It was so effective that it really was the only tool that they needed to manage the project.

With its Plan feature, the team could create a project plan that aligned the team, resources, and deliverables, with the company objective.

The Product Owner could use its backlog feature to create the product backlog, which then allowed the Scrum Master to use its Sprint feature to move items from the product backlog into a current sprint, making it excellent for sprint planning and execution.

Due to giving visibility to the project, and centralizing all project items and activity, JIRA helped to coordinate and increase efficiency in the team. Since information was centralized and accessible, it saved valuable time in retrieving it, giving team members more time to devote to backlog items.

# Evaluating the Agile Process

Was the Agile process the best way of implementing the SNHU Travel Project? We would have to weigh the benefits against the disadvantages to answer that question.

Starting with the advantages, I would say that Agile combined with Scrum, provided the project with a flexible and adaptable working environment. Because Agile is so lightweight, and did not need a whole lot of documentation upfront, the team could start to work on the project as early as possible, with just the information at hand. The communications framework, established in the Agile Charter, provided a means for the communication of further information. Being common to the project, the communications framework also provided all team members with a common frame of reference, which helped them to align their activities with the objectives of the client. An added advantage of having different skill sets in the same team, was that the project team functioned as one cohesive unit, as opposed to in a Waterfall situation, where development and testing would have been separate business units – this all but eliminated those knowledge silos that commonly forms under a Waterfall approach.

Thinking about some of the disadvantages of Agile, the lack of all the information upfront could definitely be viewed as a disadvantage. For any resources to start work on a project, they would need as much information as possible. Also, since Agile has more of a focus on process and on producing a workable product, it shifted the focus away from producing documentation, which is actually more important for the management tiers of the company for accountability and audit purposes.

In summary, Agile, as an implementation methodology, has more advantages than disadvantages over the traditional Waterfall approach. It was thus the correct approach to use in the SNHU Travel Project given that we did not have all the information upfront, and that the requirements changed during the duration of the project.