CS 250

04.16.2025

Julliane Pamfilo

[Julliane.pamfilo@snhu.edu](mailto:Julliane.pamfilo@snhu.edu)

Final Project

**Sprint Review and Retrospective: SNHU Travel Project**

Applying Roles:

During the SNHU Travel project, each role within the Scrum-Agile team contributed heavily to deliverables and the project moving forward. Organized daily standups, sprint planning, sprint review and retrospective as the Scrum Master. I helped team members with blockers, and timelines. When the development team encountered late-stage test case alignment issues, I organized a mid-sprint backlog grooming session to realign priorities and strengthen acceptance criteria.

SNHU Travel Client:

The Product Owner defined the product vision and client-side initiatives. The product backlog was maintained and user stories prioritized according to customer value. My first major contribution was to redefine a user story in the middle of a sprint to account for new business requirements, specifically new filter criteria on travel. A development team that comprised both front-end and back-end developers was able to process user stories to satisfy working software increments. Through all this, they pair programmed on core functionality and worked with the QA role to fulfill acceptance criteria.

Completing User Stories:

Time squeeze of the user stories was significantly supported by the scrum-agile framework. Each sprint was organized around a sprint backlog pulled from the product backlog. For example, a user story related to a user-friendly booking interface was divided into subtasks and delivered promptly in Sprint 1. Daily standups kept lines of communication open and led to the identification of an early bug with form validation which was resolved as a team by day three of the sprint.

Additionally, agile principles of working software and customer collaboration enabled us to demo partially completed features to stakeholders. We used their initial feedback to make improvements to a user story for the itinerary suggestions, which was completed by Sprint 2. It is through this iterative and adaptive testing cycle approach that the final product met client expectations with no need for redesign later in the cycle.

Handling Interruptions:

At the halfway point of development, the project experienced a major shift to where the client wanted support for multiple languages. In a Waterfall model, this would have caused scope creep. Yet, adaptive planning in Scrum-Agile allowed us to convene an emergency sprint planning meeting. We opened a new epic for language integration and applied a lower priority on parts.

In Sprint 3, features incrementally matured language toggles and what key UI elements to have dynamically translated. The Product Owner closely collaborated with the stakeholders to make sure the new user stories were correctly prioritized. We managed to have the multilingual support delivered in the last iteration without compromising goals in other iterations just by tweaking our backlog and exploiting the cross-functional skills that the team had.

Communication:

It was critical for the success of the project to have effective communication. As Scrum Master, I used Slack, Trello, and the like to keep the information visible. I posted a mid-sprint status update in our group slack channel for example:

“Heads up: Developer team found a browser compatibility issue with the itinerary builder. Resolution expected by EOD tomorrow. Tweak the testing schedule also.”

This notification notified the QA team to redirect their attention in the meantime to avoid sitting idle. Team members also provided updates on Trello cards about their progress towards various tasks and blockers. This enabled a collaborative environment and ensured that asynchronous updates were easily traceable.

Scrum Artifacts and Scrum Events:

A few Scrum events and tools helped to keep the project organized and moving. During Sprint Planning, team capacity was aligned with story point estimations. We used Daily Scrum to maintain visibility and resolve blockers early. Sprint Reviews gave stakeholders an opportunity to provide feedback early and often.

(Things like Jira and Trello were very helpful.) Jira assisted in assigning and keeping track of user stories (and their status) over the sprints. There were Trello boards to visualize the tasks and groom backlog. When combined with the usage of Scrum ceremonies, these tools ensured that we continued to have our Agile cadence, even when it was disrupted from the outside.

Evaluating Agile Process:

It mentioned how well the SNHU Travel project did with the Scrum-Agile approach.

Pros:

* Enabled continuous engagement and feedback from the client
* Scope adjustments where possible and not timeline busting
* Encouraging team collaboration and sharing knowledge
* Provided potentially usable software at the end of every sprint

Cons:

* Need for frequent coordination and meeting time
* Added overhead managing changing priorities and epics

After few initial hitches, Scrum-Agile was the best thing at that time. With mid-project scope changes, they would have had problems with the Waterfall model, delaying release or creating expensive rework. Agile brings an unprecedented degree of responsiveness, transparency, and iterative delivery—invaluable assets to client-facing projects such as SNHU Travel.

Conclusion

Acting as Scrum Master for SNHU Travel demonstrated the best use of Agile in complex adaptive systems, specifically user-centric software development. The Scrum-Agile framework allowed them to deliver work quickly while enabling them to be adaptable in their sprint planning and building long lasting collaborative relationships with stakeholders.