

Jacob S. Hary

Final Project

Sprint Review and Retrospective

CS 250 - Software Development Lifecycle

Southern New Hampshire University

10/18/2025

Applying Roles

Throughout the SNHU Travel project, every Scrum-Agile role played an important part in helping our team succeed. Taking on different roles during the project helped me really see how teamwork and communication form the base of Agile development.

When I was the Product Owner, I focused on creating and prioritizing user stories that matched what the client needed most. One of the first stories I wrote was about letting users search for both flights and hotels on one screen. I made sure it had clear acceptance criteria, so the team would know exactly what counted as done. By prioritizing this story early, we made sure we started with the most valuable feature first.

As a Developer, I realized how helpful daily stand-ups and sprint planning were for keeping the team on track. Each of us shared what we worked on, what we planned to do next, and any issues we ran into. That helped us spot blockers early. For example, during one sprint we had trouble getting the booking form to connect with the payment gateway. I brought it up in the stand-up, and another teammate suggested a workaround that ended up saving us time and avoiding delays.

When I was the Scrum Master, I led meetings, kept things organized, and made sure everyone followed Scrum principles. I tried to encourage communication and make sure everyone knew their role. It also taught me to stay flexible, especially when priorities shifted or unexpected problems came up.

Completing User Stories

The Scrum-Agile process really helped us complete user stories more effectively by working in small, manageable pieces. Instead of building the entire SNHU Travel app at once, we broke it into short sprints with specific goals.

One sprint focused only on building the search and filter feature for destinations. By narrowing our focus, we could really fine-tune that one function and then show it off at the sprint review. The feedback we got from stakeholders helped us make small improvements before moving on to the booking and payment system. This made it easier to catch bugs early and fix them before they became bigger problems.

Another great part of the process was how visible everything was. We used a digital Scrum board to track progress from “To Do” to “In Progress” and finally “Done.” Seeing tasks move across the board was motivating and made everyone more accountable.

Handling Interruptions

One of the best things about Agile was how flexible it was when the project direction changed. Halfway through development, SNHU Travel asked for a new feature that would show trip recommendations based on a user’s past searches. In a traditional Waterfall approach, a change like that would’ve meant major delays or even starting over.

But with Agile, we just added it as a new user story in the product backlog. During sprint planning, we re-prioritized tasks and decided which stories could wait until the next sprint. That way, we kept our schedule while still delivering what the client wanted.

That experience showed me how Agile doesn’t just manage change—it actually works best with it. The incremental nature of Scrum made it easy to fit in new ideas without ruining what we already finished.

Communication

Communication was one of the main reasons our project went so smoothly. We had daily stand-ups, sprint reviews, and retrospectives that kept everyone in sync. During stand-ups, we’d quickly share what we did, what we’d do next, and anything blocking progress.

For instance, I once mentioned being stuck on a bug in the search results page. A teammate said they’d dealt with something similar before and immediately shared a fix. That quick chat saved me hours of work and showed how much open communication helps in Agile.

We also used tools like Trello to organize user stories and instant messaging to check in fast when someone needed help. These tools made our teamwork smoother and kept things moving.

Organizational Tools

Our team used several Agile tools to stay efficient, like the product backlog, sprint backlog, and burndown charts. These made progress easy to visualize. Sprint planning helped us split big goals into smaller, more realistic tasks. The daily stand-ups encouraged steady communication, and sprint reviews gave stakeholders a chance to give feedback early.

The sprint retrospective was especially valuable because it gave us space to reflect on what worked and what didn't. One thing we learned early was that we underestimated how long integration testing would take. So in later sprints, we scheduled extra time for it. These events and tools helped keep our workflow smooth and flexible.

Evaluating Agile Process

Overall, I think the Scrum-Agile approach worked really well for the SNHU Travel project. The biggest advantage was flexibility. When the client's needs changed, we could adjust quickly without wasting time. Agile also made teamwork stronger because the short, repetitive sprints forced us to communicate and share progress often.

Pros of Agile:

- Easy to adapt to new client requests
- Frequent feedback improved the product
- Encouraged teamwork and accountability
- Delivered working software after every sprint

Cons of Agile:

- Needed consistent participation and communication
- Harder to predict final completion dates
- Could be tricky at first for anyone new to Scrum

Even with those small challenges, Agile was definitely the best choice for this project. SNHU Travel's needs were always changing, and Agile handled that perfectly. A Waterfall method would've been too rigid and slow to adapt.

In the end, Scrum-Agile helped us build a better, more user-friendly app while also improving how we worked together as a team. After seeing how effective it was, I think ChadaTech would benefit a lot from using Agile instead of Waterfall. Being able to quickly respond to feedback while staying on schedule is exactly what today's software projects need.

References

Scrum Alliance. (n.d.). *What is Scrum?* <https://www.scrumalliance.org>

Atlassian. (n.d.). *Agile project management*. <https://www.atlassian.com/agile>