Jadan Morrow  
Southern New Hampshire University  
CS 250: Software Development Lifecycle  
Goran Trajkovski  
August 17, 2025

The SNHU Travel project was developed as a pilot initiative under the Scrum-Agile framework for ChadaTech, a company exploring the possibility of moving away from its long-standing waterfall development model. The goal of this pilot was not only to deliver a functional application but also to evaluate how Agile principles, specifically the Scrum framework, could improve responsiveness, collaboration, and overall product quality. Throughout the project, I rotated through the roles of Product Owner, Developer, Tester, and Scrum Master. Experiencing each role in turn provided me with a comprehensive understanding of the software development life cycle from multiple vantage points. This reflection examines how those roles contributed to the success of the project, how Agile practices supported user story completion, how the team adapted to interruptions, the communication strategies that promoted collaboration, the organizational tools that kept the project on track, and an overall assessment of Scrum-Agile in comparison to the waterfall model.

Working as the Product Owner required me to focus on maximizing product value by ensuring that each user story reflected real customer needs. I engaged directly with stakeholders, using their input to create backlog items that addressed practical challenges. For example, after observing travelers struggling to interpret cancellation and refund policies, I wrote a user story with detailed acceptance criteria requiring that refund timelines for refundable tickets be displayed in a calendar view. This not only addressed a usability concern but also provided clarity for the development and testing teams. My approach was shaped by The Scrum Guide’s emphasis on the Product Owner’s accountability for value delivery (Schwaber & Sutherland, 2020), as well as Lucassen et al.’s research showing that well-structured user stories improve productivity and quality (Lucassen, Dalpiaz, Horkoff, & Joosten, 2021). The ability to translate raw stakeholder insights into precise acceptance criteria proved to be one of the most valuable contributions of this role.

Shifting into the Tester role revealed the practical benefits of those acceptance criteria. They offered clear, measurable expectations that made it easier to determine whether a feature met its intended purpose. However, I also discovered where stories lacked the detail necessary for complete and accurate testing. Some did not specify interface formats, such as whether a feature should display in a list or a slideshow, or failed to note platform constraints and navigation flows. In these cases, I initiated direct communication with the Product Owner to obtain missing details. This often involved requesting wireframes, user journey maps, and technical requirements so that test cases could accurately reflect the intended user experience. By clarifying these points early, I reduced the risk of invalid test results and prevented potential misalignments between development and quality assurance. This reinforced Lucassen et al.’s (2021) argument that effective story writing with clear acceptance criteria is essential to both testing and overall software quality.

Serving as the Developer reinforced the importance of adaptability in Agile. During the course of the project, there were moments when priorities shifted mid-sprint. Instead of continuing blindly, I sought immediate clarification from the Product Owner about which stories had changed, which acceptance criteria were updated, and how the revised priorities affected ongoing development work. I also coordinated closely with the Tester to ensure that any new or updated test cases were aligned with the latest requirements. This responsiveness minimized wasted effort and kept the sprint deliverables aligned with stakeholder needs. Agile’s emphasis on adaptive planning and incremental progress (Schwaber & Sutherland, 2020) made these adjustments seamless, allowing development to progress without significant disruption.

When I took on the role of Scrum Master, my focus shifted to maintaining team alignment and transparency. I facilitated daily stand-ups, sprint reviews, and retrospectives, ensuring each team member had the opportunity to share updates, raise concerns, and contribute to planning. I promoted the use of information radiators and digital task boards, which displayed the status of backlog items, dependencies, and blockers in a way that was visible to all. These tools reduced ambiguity and made it easier to identify issues before they became major obstacles. My efforts as Scrum Master underscored the importance of structure and consistency in fostering a collaborative environment, which is a central principle of Scrum team accountability (Schwaber & Sutherland, 2020), even when team members were working asynchronously.

One of the strongest demonstrations of Agile’s value came in the way it supported the completion of user stories. The refund timeline story serves as a clear example. As Product Owner, I defined its acceptance criteria and ensured it was prioritized appropriately in the backlog. As Tester, I created targeted test cases to verify the feature’s functionality. As Developer, I implemented the feature with careful attention to those criteria. As Scrum Master, I kept the team aligned and ensured that feedback loops were short. Because Agile allows for iterative refinement, the story evolved over multiple sprints as stakeholder feedback was incorporated, preventing costly misalignment and ensuring the final implementation matched the intended vision (Schwaber & Sutherland, 2020).

The project also highlighted Agile’s ability to handle interruptions effectively. At one point, evolving business requirements changed the scope of certain features mid-sprint. Under a waterfall approach, such changes would have forced the project back to earlier phases, delaying delivery and increasing costs. Under Scrum, however, these changes were absorbed without halting progress. I immediately clarified which stories were impacted, confirmed new acceptance criteria, and adjusted the development approach accordingly. This ability to pivot without derailing the entire project is one of the most practical advantages of Agile (Schwaber & Sutherland, 2020).

Communication proved to be the backbone of the team’s success. Structured Scrum events created regular opportunities to realign priorities and share updates. Outside of these events, asynchronous tools such as discussion boards kept the project moving forward. For example, the Product Owner posted a prioritized list of stories to the board, eliminating confusion about development order. Targeted emails further supported collaboration. In my Developer role, I sent a message to the Product Owner and Tester requesting clarification on backlog changes, updated acceptance criteria, and QA insights. As Tester, I sent another requesting wireframes, user flows, and technical constraints to ensure test cases were accurate. These exchanges were direct, specific, and actionable, which reduced rework and kept momentum high. Effective communication like this is recognized as a critical factor in Agile team success (Schwaber & Sutherland, 2020).

Organizational tools like Jira played a crucial role in keeping the project on track. An up-to-date Jira board provided a shared, real-time view of the backlog, ongoing tasks, and dependencies. This visibility supported effective sprint planning and reviews, allowing the team to respond quickly to changes. Information radiators supplemented these tools by making the state of the project instantly visible, fostering transparency and accountability. When combined with Scrum events, these tools ensured that work remained aligned with both short-term sprint goals and the broader product vision (Schwaber & Sutherland, 2020).

Reflecting on the Scrum-Agile process for this project reveals a number of strengths. Flexibility allowed the team to adapt to evolving requirements while maintaining delivery pace. Frequent communication kept feedback loops short, ensuring stakeholder needs were addressed promptly. The collaboration between roles ensured that user stories were defined, implemented, and tested with precision. However, there were challenges as well. Some stories lacked sufficient detail at the outset, slowing development and requiring clarification during the sprint. These gaps reinforced the need for strong story-writing practices, including clear acceptance criteria and complete functional descriptions (Lucassen et al., 2021).

Comparing this experience to a waterfall approach underscores Agile’s advantages for the SNHU Travel project. In Waterfall, the changes to cancellation and refund policies would have required revisiting completed requirements and design phases, delaying subsequent development and testing. Agile’s iterative approach allowed the team to adjust mid-sprint, incorporate changes immediately, and deliver value continuously. This responsiveness not only kept the project aligned with user needs but also reduced the risk of costly rework, reflecting one of Scrum’s most significant advantages over traditional models (Schwaber & Sutherland, 2020).

In conclusion, the SNHU Travel project demonstrated how Scrum-Agile can deliver value through adaptability, strong communication, and collaborative effort. Each role I assumed contributed in unique ways, from translating stakeholder needs into actionable stories, to ensuring quality through precise testing, to keeping development aligned under changing priorities, to fostering transparency and cohesion across the team. Organizational tools and disciplined communication practices provided the structure needed to support these efforts. Although the project encountered challenges when stories lacked detail, the overall results confirm that Agile was the right choice for this work. For ChadaTech, this pilot serves as strong evidence that adopting Scrum-Agile more broadly could improve responsiveness, enhance collaboration, and better align products with client expectations.

## *References*

*Lucassen, G., Dalpiaz, F., Horkoff, J., & Joosten, S. (2021). Engineering and analyzing agile user stories: State of practice. Journal of Systems and Software, 173, 110811.*

*Schwaber, K., & Sutherland, J. (2020). The Scrum Guide (November 2020 ed.). Scrum.org.*