CS250 Sprint Review and Retrospective

Anthony L. Fillmore

Southern New Hampshire University

Table of Contents

[CS250 Sprint Review and Retrospective 3](#_Toc95493478)

[A. Demonstrate how the various roles on your Scrum-agile Team specifically contributed to the success of the SNHU Travel project. 3](#_Toc95493479)

[B. Describe how a Scrum-agile approach to the SDLC helped each of the user stories come to completion. 3](#_Toc95493480)

[C. Describe how a Scrum-agile approach supported project completion when the project was interrupted and changed direction. 3](#_Toc95493481)

[D. Demonstrate your ability to communicate effectively with your team by providing samples of your communication. 3](#_Toc95493482)

[E. Evaluate the organizational tools and Scrum-agile principles that helped your team be successful. 3](#_Toc95493483)

[F. Assess the effectiveness of the Scrum-agile approach for the SNHU Travel project. 3](#_Toc95493484)

[Describe the pros and cons that the Scrum-agile approach presented during the project. 3](#_Toc95493485)

[Determine whether or not a Scrum-agile approach was the best approach for the SNHU Travel development project. 3](#_Toc95493486)

[References 4](#_Toc95493487)

CS250 Sprint Review and Retrospective

# A. Demonstrate how the various roles on your Scrum-agile Team specifically contributed to the success of the SNHU Travel project.

Each Scrum role played a part in our SNHU Travel project that directly impacted its success. The Scrum Master coordinated all scrum events and helped set the guidelines for team ceremonies. They actively searched for and removed blockers from the scrum team, facilitating open communication between stakeholders that helped identify key patterns. They facilitated the exchange of important details when the business decided to shift the project's focus to Health and Wellness travel destinations. This ensured that the entire team was aligned and could swiftly adapt to the new requirements.

The Developers successfully implemented necessary changes by writing user stories, revising stories, and prioritizing them based on changing project goals. Their technical expertise and agility allowed them to complete the work and ensure the project was delivered in time for a critical business advantage.

The Product Owner helped set requirements that fed the stories being created by the developers and testers. They helped prioritize the backlog of stories ensuring the Developers focused on features that mattered to the business. They also fostered open communication by relaying changing requirements to the product team, ensuring a quick change of direction to meet business deadlines.

Testers focused on quality control, identifying bugs and surfacing issues that required code changes. They worked closely with developers creating a tight feedback loop that improved the quality of the product.

# B. Describe how a Scrum-agile approach to the SDLC helped each of the user stories come to completion.

The process began with our Product Owner collaborating closely with the Scrum Master and project team to gather and relay the requirements. We engaged end users through surveys to identify the features that would provide them with the most value. These insights were then translated into user stories, capturing the general theme of what our customers were looking for.

The Developers played a crucial role in turning these user stories into task driven outcomes. They translated business requirements into technical ones, defining tasks and establishing clear done criteria. They ensured that the features were delivered as expected, meeting the evolving needs of our customers as the project requirements were set, and then changed.

The prioritization of user stories by the Product Owner was another key aspect of our Agile approach. By prioritizing the backlog, the Product Owner provided a roadmap for the developers, guiding them towards delivering features that mattered the most to our customers. This alignment with customer priorities had a direct impact on our company's bottom line.

The iterative nature of Agile allowed for constant collaboration and communication among team members. We were able to address challenges and obstacles quickly, ensuring the smooth progress of each user story towards completion.

# C. Describe how a Scrum-agile approach supported project completion when the project was interrupted and changed direction.

I think the key to the successful direction change was the tight feedback loop between the Product Owner and the business. This interaction kicked off a series of events that the Agile framework facilitated. The business identified a target area of opportunity and alerted the Product Owner. The Product Owner worked with the Scrum Master to identify requirements and goals, and then worked with the developers to create stories to modify features. The Developers being advocates of technical excellence were able to pivot on previously written features and alter them to meet the new requirements. The testers ensure the code was bug free and the features were delivered on time to hit a crucial business window.

This streamlined collaboration highlighted the adaptability of the Scrum-agile approach and allowed the project to change direction seamlessly. The iterative nature of Scrum enabled quick feedback and course correction, ensuring that the project remained aligned with the business. The close collaboration between the Product Owner, Scrum Master, developers, and Testers fostered a shared understanding of the new requirements and goals. By leveraging the strengths of the Scrum-agile approach, the team successfully adapted to the interruption, enabling the project to stay on track and meet the evolving business needs.

# D. Demonstrate your ability to communicate effectively with your team by providing samples of your communication.

Here is an example of my communication to a Product Owner from Week 5:

Hello Product Owner,

First off, I’m extremely excited to be working with you on this product! I’m looking forward to learning from you and building something that delights our customers!

As we begin our project, I was hoping to work with you on getting some critical meetings scheduled where your participation will be needed, specifically our backlog-refinement meetings and our standups. Your voice will be an important one as we look to ensure our user-stories have enough required detail and they have clear goals. We will also be heavily relying on your vision to prioritize our story backlog, which should give our developers and testers a clear pipeline of work and streamline our planning process!

Our standups will be 15 minutes daily and our backlog refinement meetings are 30 minutes per week. Please suggest good times that you are available, and I will get email invites sent out asap!

I was also hoping to pick your brain on relevant documentation the team might need to get up to speed, project goals and what a minimum viable product might look like. Would you have time for a status this week?

Can’t wait to meet you!

Anthony Fillmore

---------------------

I take a positive tone in my email and start by asking for collaboration in the form of participation in critical meetings. I also acknowledge the importance of the Product Owner's input and expertise. This demonstrates the Agile principle of "Customer Collaboration over Contract Negotiation" and the Scrum value of "Customer Collaboration" by actively involving the Product Owner in decision-making and ensuring their voice is heard throughout the project.

I also highlight the Product Owner's role in prioritizing the story backlog, emphasizing their vision to guide the team. This aligns with the Agile principle of "Deliver Working Software Frequently" and the Scrum value of "Focus" by ensuring that the development team has a clear understanding of the priorities and goals set by the Product Owner.

I wrap up by showing I’d like to collaborate further by seeking the Product Owner's insights on relevant documentation, project goals, and the minimum viable product. I hope to schedule a status meeting, so they know I’m commitment to regular communication and feedback. This aligns with the Agile principle of "Customer Collaboration" and the Scrum value of "Openness" by fostering transparent and ongoing collaboration between the sender and the Product Owner. Overall, I feel like I did a good job in treating my Product Owner with respect and keeping a tone of collaboration whilst still asking for action.

# E. Evaluate the organizational tools and Scrum-agile principles that helped your team be successful.

One of my favorite Agile tools that we covered was the Agile Team Charter. A charter establishes clear expectations and helps our Scrum team operate effectively. It allows all members on the team a voice, fostering open communication, collaboration, and a sense of ownership. It provides a sense of project purpose to the team by outlining our mission, values, and goals, providing a solid foundation for agile teams to iterate on.

The product backlog is another critical tool as it provides a work pipeline for Developers and Testers. The backlog represents the work needed to achieve project goals and is a direct reflection of the business requirements gathered by the product owner and other stakeholders. By having the backlog refined and prioritized, we can ensure that the work pipeline is adding business value and creating an efficient pipeline to deliver important product features. It also provides visibility to stakeholders, giving a general understanding of what will be worked, and in what order. This creates a nice feedback loop that can help adjust priorities on the fly and ensure stories are meaningful.

User Stories describe the work engineers need to do to achieve project goals. They epitomize the Agile principle of “collaboration of contract negation”, focusing on the needs of the user rather than detailed specifications with timeline agreements. They also require open communication with stakeholders, creating a tight feedback loop that allows the project to change direction quickly. For the Travel Project, User Stories allowed us to deliver features that our customers were looking for. We didn’t need a trial-and-error phase because we knew what they wanted – this increased our efficiency allowing us to focus on delivering features we knew would add value. It also allowed us to change direction quickly when required, understanding that the market had changed and profitably depended on that change allowed us to reuse the existing code, making small changes to adjust rather than completely rework the code.

Test Cases ensure quality of the product, increasing reliability of the product and ensuring the business has as little down time as possible. They also make the team more efficient by preventing the Developers from must rework features down the line, creating sound products early on the process and allow for a quick feedback loop to fix issues. Test cases reinforce the agile principle of technical excellence and good design by ensuring a feature done criteria includes rigorous scrutiny and standards.

The three questions in the Daily Standup (What did I accomplish yesterday? What will I do today? Are there any obstacles?) foster an open communication model, providing visibility to stakeholders and introducing accountability as a daily affirmation. The goal isn’t so much as to hold people accountable (even though that is a by-product) as it is to identify areas of opportunity for the scrum team. Blockers are identified quickly, allowing the Scrum Master to assist in moving tasks forward. Opportunities for pair programming and swarm coding become apparent and collaboration becomes a focus. Shared goals are identified preventing duplicate work and ensuring goals are aligned.

# F Assess the effectiveness of the Scrum-agile approach for the SNHU Travel project.

## Describe the pros and cons that the Scrum-agile approach presented during the project.

**Pro:** Agile accommodates environments where requirements often change.

The travel market seems to change often, with destinations and trends shifting rapidly based on various factors like social media trends, customer preferences, and the economy. The iterative approach promoted by Agile allows for continuous refinement and adjustment of requirements throughout the project lifecycle. This flexibility enables the development team to respond swiftly to changing market demands, ensuring that the product remains relevant and aligned with the evolving needs of travelers. The Travel project certainly benefited from being an Agile project, proven when the developers delivered crucial features on short notice.

**Pro:** Agile delivers value early, and often.

Shortly after receiving our requirements, we were able to write user stories and produce a prototype Top 10 slideshow for our stakeholders, showing project value and gaining valuable feedback about project direction. This helped create a close delivery -> feedback loop that allowed us to iterate on our prototype, expanding features that mattered to the customer and delivered results for the business. It also allowed us to build modular features that were easily interchangeable, allowing us to pivot on new features quickly, ahead of the market.

**Pro:** Realistic Approach to Software Development.

Acknowledging that requirements can change based on the market or business objectives promotes teams iterating in small, manageable increments. This helps avoid lengthy and complex development cycles that run the risk of being thrown away. In our Travel project we were able to stay ahead of the competition and deliver value to our customers in a critical window of time because we changed directions quickly and reused code that didn’t lock us into any one certain feature.

**Con:** Agile depends heavily on customer interactions.

There is a risk for the Travel project to get bogged down with deadlines and customer requests due to the nature of Agile and heavy customer interactions. When we had our direction change and needed to write stories, I felt like I had a lot more specific questions about “done” criterion because I knew the deadline was close at hand. This gives developers the feeling that the have to get it “perfect”, which has a byproduct of spending more time planning and asking probing questions. The project runs the risk of losing efficiency and agility if these type of deadlines start to become the norm.

**Con:** Agile minimal documentation means knowledge can get in a silo’d

Having multiple developers contributing to a project is critical to its success. The more complex a system is, the more dependencies there could be on documentation to ensure new developers are onboarded quickly and efficiently. There is a risk with the SNHU project of creating a bottleneck when a new feature needs to be added that requires time and training to fix. As the SNHU project grows and the features start to compound, the risk of knowledge silos’ grows, and documentation may be unavoidable.

**Con:** Complex dependencies may cause issues.

The algorithm for the Top 10 list based on User preferences and weight had the potential to become a complex dependency that might require a team of its own to tackle. (Separate from the GUI implementation). The feature requires a constant feedback loop to perfect and should be altered routinely to ensure its keeping up with user demands and other critical factors.

## Determine whether or not a Scrum-agile approach was the best approach for the SNHU Travel development project.

I’m probably biased here but I think Agile was the correct choice for the SNHU Travel Project. The project wasn’t extremely complex and did not have a plethora of cross-team dependencies, which lends itself to Agile right off the bat. The requirements for the Travel project changed often due to the nature of the Travel business, requiring a team and project methodology that could shift quickly with the desires of the customer. Agile provided the flexibility and adaptability to make these adjustments, utilizing the tight feedback loop of the Product Owner and business customers combined with the Scrum master to facilitate the team’s direction change quickly, and in time for crucial deadlines. User stories were created, the backlog was adjusted and prioritized, and the development team reused already implemented features, making slight tweaks to satisfy the new requirements. Waterfall wouldn’t have provided the agility need to meet the deadline after our direction change and we would have missed a big business opportunity and lost market share.

References

Cobb, C. G. (2015). The Project Manager's Guide to Mastering Agile: Principles and Practices for an Adaptive Approach (p. 10). Hoboken, NJ: Wiley.