CS - 250

Professor Mason

Billy Lewis

October 18, 2025

**7-1 Final Project – Sprint Review and Retrospective**

### This retrospective summarizes my experience as part of the Scrum-Agile team for the SNHU Travel project. Over the course of several sprints, I rotated through key Scrum roles—Product Owner, Developer, Tester, and Scrum Master—gaining firsthand insight into how collaboration and iteration improve software outcomes. The purpose of this review is to reflect on how Agile principles guided our development process, helped us overcome challenges, and fostered continuous learning within the team. By analyzing our practices, tools, and communication methods, I aim to highlight lessons that can strengthen future Agile implementations.

### **Applying Roles**

Throughout the SNHU Travel development, every Scrum role contributed to our success.

* **Product Owner (PO)**: Defined user stories and priorities. As PO, I clarified acceptance criteria and ensured the Slide Show feature aligned with business value—showcasing top destinations to engage customers.
* **Scrum Master**: Facilitated stand-ups, kept the team aligned, and removed blockers during the image-integration phase.
* **Developer**: Modified and tested the Slide Show class, debugging image-path issues and updating code per new requirements.
* **Tester**: Validated that navigation, images, and text matched acceptance criteria.

This cross-functional collaboration prevented silos—something Cobb (2015) cautions against when teams adopt Agile mechanically without shared ownership (*p.* 356).

### **Completing User Stories**

Agile’s incremental flow helped complete user stories faster. Rather than waiting for all requirements upfront, we iterated—adding and testing each feature in sprints. For instance, incorporating destination images and descriptions was handled in small, testable increments, allowing early feedback. Scrum ceremonies such as sprint reviews ensured stakeholders verified value at each step. This reflects Cobb’s note that iterative delivery “reduces risk by validating small portions of scope early and often” (*p.* 359).

### **Handling Interruptions**

Midway through development, missing image assets halted progress. Instead of stalling, we adapted—redefining sprint priorities and verifying resource paths. Agile’s flexibility allowed the project to continue without derailing milestones. Cobb (2015) describes this resilience as the essence of adaptive leadership: teams must “treat disruption as data, not disaster” (*p.* 362).

### **Communication**

Our communication mirrored real Scrum culture:

* Daily stand-ups clarified blockers and progress.
* Direct chat between Developer and PO resolved configuration issues quickly.
* Informal retrospectives captured lessons after each sprint.

These open feedback loops promoted trust and visibility. Dingsøyr et al. (2022) found that short, structured communication cycles drive continuous learning and high team performance (*p.* 9). Within SNHU Travel, these conversations prevented misalignment and kept morale high.

### **Organizational Tools**

If our team had implemented JIRA or Azure Boards, we could have visually tracked user stories, bugs, and acceptance tests in real time—serving as digital “information radiators.” Such tools reinforce Scrum’s transparency principle. Sprint planning clarified goals, daily stand-ups maintained momentum, and retrospectives encouraged adaptation. Cobb (2015) highlights that visual workflow tools “amplify team accountability by making commitments and progress public” (*p.* 364).

### **Evaluating the Agile Process**

Scrum-Agile clearly improved SNHU Travel’s workflow.  
 Pros: adaptability, early feedback, shared ownership, and visible progress.  
 Cons: initial confusion when switching roles and untracked assets.

Overall, Scrum was the right approach. A Waterfall model would have delayed validation until the end, increasing rework risk. Agile allowed incremental delivery and continuous customer alignment—echoing Cobb’s conclusion that “Agile succeeds when learning replaces prediction as the control mechanism” (*p.* 369).

### **Conclusion**

The SNHU Travel project proved that communication, transparency, and collaboration drive Agile success. Every role—from PO to Tester—contributed to continuous improvement, validating that Agile is not merely a framework but a mindset of learning, adaptation, and teamwork.

### **References**

Cobb, C. G. (2015). *The Project Manager’s Guide to Mastering Agile: Principles and Practices for an Adaptive Approach* (pp. 353–370). Wiley.

Dingsøyr, T., Moe, N. B., & Seim, E. A. (2022). Team learning in software development: How do Agile teams learn? *Empirical Software Engineering, 27*(3), 1–23. https://doi.org/10.1007/s10664-022-10145-2