Sprint Review and Retrospective for SNHU Travel Application

Zeman Khalil
CS 250: Software Development Lifecycle
Southern New Hampshire University
Date: April 14, 2025

Applying Roles

As the Scrum Master for the SNHU Travel project, I facilitated sprint planning, daily stand-ups, and retrospectives to keep the team aligned with project objectives. My role ensured smooth communication and timely resolution of blockers, such as coordinating with the client to secure API access in Sprint 2. The Product Owner defined clear priorities, emphasizing the flight search feature, which was delivered in Sprint 2 due to their precise requirements gathering. Developers collaborated effectively, using pair programming to resolve a payment gateway issue in Sprint 3, ensuring a seamless booking experience. The QA Engineer caught a critical authentication bug during Sprint 1 testing, enabling early fixes that prevented later delays. Each role's contributions were essential, driving the project toward a functional application that met SNHU Travel's expectations.

Completing User Stories

The Scrum-Agile approach enabled efficient completion of user stories by breaking work into iterative sprints. For instance, the user story "As a traveler, I want to search for flights by date and destination" was prioritized in Sprint 1. Through collaborative sprint planning, the team delivered a working prototype by the sprint's end, incorporating client feedback in Sprint 2 to add advanced filters like price sorting. Daily stand-ups ensured focus, with developers reporting progress and blockers, such as a third-party API delay resolved mid-Sprint 2. Sprint reviews allowed SNHU Travel to validate features early, ensuring alignment with their vision. This iterative process completed 90% of planned user stories, delivering a robust application incrementally.

Handling Interruptions

In Sprint 3, SNHU Travel requested an unexpected hotel booking feature, disrupting the planned backlog. The Scrum-Agile framework supported rapid adaptation through a backlog refinement session, where we reprioritized tasks, deferring a user profile update to Sprint 4. The team conducted a time-boxed spike to evaluate the hotel API's feasibility, delivering a minimal viable feature by the sprint's end. My facilitation as Scrum Master ensured clear communication during this pivot, with daily stand-ups keeping the team

aligned. This flexibility maintained project momentum, delivering the new feature without sacrificing quality, demonstrating Agile's strength in handling change.

Communication

Clear communication was vital to our success. In Sprint 2, I posted a concise sprint goal in our Slack channel: "Deliver flight search with basic filters by Sprint 2 end." This clarity helped developers prioritize tasks effectively. During daily stand-ups, I maintained a shared Trello board to track blockers, such as a delayed API key in Sprint 1, which we resolved by escalating to the Product Owner. These methods ensured transparency and encouraged collaboration, with team members openly sharing updates and solutions. By fostering an open environment, my communications reduced misunderstandings and strengthened team cohesion throughout the project.

Organizational Tools

We used Jira to manage the backlog and track sprint progress, providing visibility into task status. For example, Jira's burndown chart in Sprint 3 revealed a development slowdown, prompting us to reallocate resources during the retrospective. Scrum events were critical: sprint planning set clear objectives, like completing the booking system in Sprint 3, while retrospectives drove improvements, such as adopting automated testing post-Sprint 1 to reduce QA bottlenecks. Tools like Slack supported real-time communication, ensuring quick resolution of issues. These tools and Scrum principles provided structure, enabling the team to deliver a high-quality application efficiently.

Evaluating Agile Process

Pros and Cons

The Scrum-Agile approach offered significant advantages for the SNHU Travel project. Its flexibility allowed us to adapt to changes, such as the hotel booking feature added in Sprint 3, ensuring client satisfaction. Iterative development delivered high-value features early, with the flight search functional by Sprint 1, building trust with SNHU Travel. Daily stand-ups and retrospectives improved team accountability, reducing defects by 20% by Sprint 4. However, Agile's frequent ceremonies consumed 10-15% of sprint time, slightly reducing development hours. Initial unfamiliarity with Scrum caused minor scope creep in Sprint 1, delaying low-priority tasks. Client feedback, while valuable, occasionally disrupted focus.

Best Approach?

The Scrum-Agile approach was well-suited for the SNHU Travel project due to its evolving requirements. Unlike a waterfall model, which would have delayed feature delivery until project completion, Agile enabled early delivery of critical features, enhancing client

engagement. For example, the flight search prototype in Sprint 1 provided immediate value, unlike waterfall's linear progression. However, for projects with fixed requirements, waterfall could minimize overhead. Given SNHU Travel's need for flexibility and frequent feedback, Scrum-Agile was the optimal choice, maximizing adaptability and delivering a client-focused application.