Sprint Review and Retrospective

SNHU Travel Project

Prepared by Nathan Lively

Applying Roles

Throughout the SNHU Travel project, I experienced firsthand how different Scrum roles contribute to project success. As a Development Team Member, I implemented core features of the travel booking system, including the user authentication module and flight search functionality. I maintained code quality through rigorous testing and documentation, demonstrating the development team's responsibility for technical excellence.

When serving as Scrum Master, I focused on removing impediments and facilitating progress. For instance, when we faced technical challenges with the payment gateway integration, I coordinated with stakeholders to quickly obtain necessary API documentation and access credentials. I also maintained the sprint board and ensured daily stand-ups remained focused and productive.

In the Product Owner role, I prioritized the product backlog based on business value and technical dependencies. For example, I prioritized the basic flight search functionality before the advanced filtering features, ensuring we delivered core value early. I wrote clear user stories with detailed acceptance criteria, which helped maintain alignment between business requirements and technical implementation.

Completing User Stories

The Scrum-Agile approach significantly enhanced user story completion through its iterative nature. For example, when developing the hotel booking feature, I broke down the user story into smaller, manageable tasks:

1. Design database schema for hotel information

2. Implement basic search functionality

3. Add room availability checking

4. Integrate payment processing

This incremental approach allowed for regular feedback and adjustments. During sprint reviews, stakeholder feedback helped refine the user interface, leading to improved user experience in subsequent iterations.

Handling Interruptions

The Scrum-Agile methodology proved particularly valuable when handling project changes. A significant example occurred when new security requirements were introduced mid-project. Instead of disrupting the entire development process, we:

1. Evaluated the impact on current sprint goals

2. Added new security requirements to the product backlog

3. Reprioritized upcoming sprints to accommodate the changes

4. Maintained progress on existing features while incorporating new requirements

This flexibility demonstrated how Scrum-Agile practices help manage unexpected changes while maintaining project momentum.

Communication

Effective communication was crucial throughout the project. During daily stand-ups, I provided clear, concise updates following the standard format:

- "Yesterday I completed the user authentication API endpoints"

- "Today I'm implementing input validation"

- "I need clarification on password requirements"

This structured approach ensured efficient information sharing and quick identification of blockers. Documentation in JIRA included detailed acceptance criteria and technical specifications, enabling clear understanding across roles. For example:

User Story: "As a traveler, I want to search for flights by date and destination"

Acceptance Criteria:

- Search form validates date formats

- Results display available flights

- Sort options include price and duration

Organizational Tools

The combination of tools and Scrum events significantly enhanced project organization and efficiency. JIRA proved invaluable for tracking sprint progress and managing the product backlog. Git facilitated version control and code review processes, while daily stand-ups in Microsoft Teams enabled effective remote collaboration.

Sprint events demonstrated the value of these tools:

- Sprint Planning: JIRA's story point tracking helped estimate effort accurately

- Daily Stand-ups: Teams' virtual meetings maintained team alignment

- Sprint Review: Screen sharing facilitated effective feature demonstrations

- Sprint Retrospective: Digital collaboration boards captured improvement ideas

Evaluating Agile Process

Pros of the Scrum-Agile approach in the SNHU Travel project:

- Rapid adaptation to requirement changes

- Regular feedback through sprint reviews

- Clear visibility of project progress

- Structured approach to managing priorities

Cons encountered:

- Initial learning curve with Agile tools

- Challenge of accurate story point estimation

- Documentation requirements sometimes slowed development

- Remote collaboration complexities

The Scrum-Agile approach was indeed the best choice for SNHU Travel because:

1. Requirements evolved throughout development

2. Regular stakeholder feedback was essential

3. The project needed flexibility to adapt to market changes

4. Incremental delivery allowed for early value realization

Conclusion

The Scrum-Agile methodology proved highly effective for the SNHU Travel project. Through experiencing different roles, managing user stories, handling changes, and maintaining clear communication, I gained practical insight into agile development practices. The combination of structured processes and flexible adaptation enabled successful project progression despite challenges and changes. This experience has provided valuable lessons in project management and agile methodologies that will benefit future software development efforts.