Sprint Review and Retrospective for SNHU Travel Project

Dave Mobley

Southern New Hampshire University

December 14, 2024

CS-250: Software Development Lifecycle

1. Introduction

Throughout the development of the SNHU Travel application, the Scrum-Agile framework enabled our team to transition from a traditional waterfall methodology to a more iterative and adaptive process. This Sprint Review and Retrospective reflects on the contributions of Scrum roles, events, and artifacts to the project's success. Drawing on lessons from practical experiences and theoretical principles outlined by Schwaber and Sutherland (2020) and Rubin (2019), this document underscores how Agile methodologies supported flexibility, transparency, and continuous improvement.

2. Applying Scrum Roles

2.1. Product Owner

The Product Owner (PO) prioritized the product backlog and collaborated closely with stakeholders to ensure clarity and alignment. For example, during a stakeholder consultation about the "Explore Destinations" feature, the PO adjusted priorities to incorporate tailored recommendations. This action increased the feature's value by addressing key user needs, demonstrating how effective prioritization can align the team's focus with stakeholder objectives.

2.2. Scrum Master

The Scrum Master facilitated events, addressed blockers, and fostered team cohesion. For instance, when developers faced issues with integrating a third-party API for the "View

Itinerary" feature, the Scrum Master coordinated with external vendors and encouraged experimenting with local caching solutions. These efforts ensured steady progress and maintained high team morale.

2.3. Development Team Members

Development team members transformed user stories into deliverable increments while collaborating with testers to ensure alignment with the *Definition of Done*. Their close communication and iterative refinement contributed to a high-quality product that met stakeholder expectations.

3. Completing User Stories Using Agile Practices

The iterative approach to breaking down epics into smaller user stories allowed for incremental delivery. For instance, the "Search Flights" epic was divided into stories such as "Search by Departure Date" and "Filter by Price Range." Frequent feedback during Sprint Reviews ensured that each story aligned with stakeholder priorities. This iterative delivery model enhanced adaptability and kept the project user-focused.

4. Addressing Challenges and Changes

Scrum's flexibility was pivotal in responding to unforeseen changes. For example:

- Midway through development, stakeholders prioritized eco-friendly features. Stories like "Show Sustainable Accommodations" were seamlessly incorporated into the backlog and addressed in subsequent sprints.

- When a third-party API updated its authentication protocols, the team swiftly adapted by prioritizing integration fixes during Sprint Planning, minimizing disruption to the timeline.

5. Communication Practices

Communication played a central role in the project's success. Key practices included:

- **Daily Scrums:** These meetings facilitated transparency, enabling team members to raise blockers and align on tasks efficiently.
- **Collaborative Tools:** Tools such as Slack for real-time communication and Google Docs for shared documentation enhanced collaboration and ensured that all team members had access to updated resources.
- **Stakeholder Feedback:** Sprint Reviews provided stakeholders with opportunities to evaluate work and offer actionable feedback, fostering continuous improvement.

6. Organizational Tools and Scrum Events

6.1. Tools

- **JIRA:** This tool enabled task management, backlog prioritization, and progress tracking through customizable dashboards.
- **Confluence:** The team used Confluence to store documentation, meeting notes, and retrospectives, creating a centralized knowledge base.

6.2. Scrum Events

- **Sprint Planning:** Focused on establishing clear sprint goals and prioritizing high-value tasks.
- **Daily Scrums:** Ensured ongoing alignment and visibility into progress.
- **Sprint Reviews:** Allowed for stakeholder evaluation and refinement of deliverables.
- **Sprint Retrospectives:** Facilitated open dialogue about challenges and opportunities for process improvement.

7. Lessons from Agile Principles

Reflecting on Agile principles, three key insights emerged:

- **Transparency:** Burn-down charts and task boards provided real-time visibility, ensuring informed decision-making.
- **Stakeholder Engagement:** Regular feedback loops ensured that deliverables remained aligned with user expectations.
- **Adaptability:** Iterative delivery allowed the team to accommodate changes efficiently, enhancing overall project resilience.

8. Evaluating the Scrum-Agile Approach

8.1. Strengths

- Frequent feedback loops ensured deliverables met user needs.
- Transparency through visible boards and daily updates promoted accountability.
- Retrospectives supported continuous improvement.

8.2. Challenges

- Team members new to Scrum required time to adjust to iterative processes.
- Overcommunication during ceremonies risked inefficiency if not well-structured.

9. Conclusion

The Scrum-Agile framework proved instrumental in enabling iterative delivery, fostering clear communication, and responding effectively to changes. While challenges such as a learning curve for new members were present, the benefits of adaptability, transparency, and stakeholder engagement outweighed these drawbacks. These insights provide a foundation for future projects as the organization continues to refine its Agile practices.

References

- Beck, K., & Fowler, M. (2019). *User Stories Applied: For Agile Software Development.* Addison-Wesley.

- Cohn, M. (2020). *Succeeding with Agile: Software Development Using Scrum.* Addison-Wesley.

- Rubin, K. S. (2019). *Essential Scrum: A Practical Guide to the Most Popular Agile Process.* Addison-Wesley.

- Schwaber, K., & Sutherland, J. (2020). *The Scrum Guide.* Retrieved from `https://scrumguides.org/`

- Atlassian. (n.d.). *Agile Coach.* Retrieved December 14, 2024, from `https://www.atlassian.com/agile`

- Atlassian. (2023). *JIRA Software: The Project Management Tool for Agile Teams.* Retrieved from `https://www.atlassian.com/software/jira`