CS-250: Final Project

Christopher M. Biele

SNHU

CS-250: Software Development Lifecycle

Professor Mason

CS-250: Final Project

Throughout this term, I have learned a great deal about the Agile development process, adding insight into how Agile teams function and work together. The roles of each team member are crucial in their own regard. Every role is valuable because each role adds unique perspectives and skills. Starting from the top, I want to highlight roles that impacted the success of my SNHU Travel project, beginning with the Scrum Master.

The Scrum Master functions as the monitor of the team. They enact Agile practices, ensure team collaborations, and oversee guidance on transitioning to Agile development. For this project, the Scrum Master helped team members comprehend the value of the Agile development process, especially utilization of feedback in iterative development. As Scrum Master, I was key to our events, where the team could meet and ensure consistent communication happens. Stand-ups, sprint planning sessions, sprint review sessions, and sprint retrospectives created the foundation of communication and fostered an atmosphere of learning from mistakes and feedback. These meetings and sessions allowed the team to communicate updates, give real-time feedback, and adjust the distribution of workloads. All these factors provide the team with a workflow that focuses on cross-team communication and improvement through accountability.

In the SNHU Travel project, the next role I want to address in our success is the Product Owner. The Product Owner’s role is that of a liaison: they provide communication between the development team and the stakeholders. This cross-team collaboration allowed the development team to ensure our goals were aligned with the client’s. When we defined user stories in the project, the Product Owner was crucial. The clarity the Product Owner provided us when defining user stories allowed us to ensure consistency between the business-side's and technical-side's vision. Throughout the project, the Product Owner was tasked with ensuring the backlog was evaluated for refinement. For example, once features were applied and demonstrated, the Product Owner provided feedback on how we can refine the feature to meet the client’s needs. Receiving constant feedback allowed our team to pivot to newly prioritized tasks, ensuring time was spent strategically. In the project, our Product Owner helped us adjust the user profile customizations feature. When this feature was first presented, it seemed feasible, but as time went on, the Product Owner identified the lack of necessity in this feature when compared to others. This allowed our team to focus on higher-priority key features, like streamlined booking options. These changes gave us the insight needed to comply with the client’s goals for the SNHU Travel program.

The next role I want to discuss is the developers. As mentioned earlier, each role on an Agile team is critical to ensuring success on a project. Developers played a significant role in translating user stories into functional features and components, ensuring user and client needs align with technical requirements. In our sprint planning sessions, developers clarified how user stories can be sectioned into achievable tasks. In their clarification, the developers provided analysis of potential roadblocks, like dependency issues. The identification of these issues, in the initial stages, was crucial to ensuring alignment between the business side and technical side. This information was then discussed in daily stand-ups, where issues were addressed, adjustments were made, and goals were kept aligned. Developers also play a crucial role in testing phases of the development process, by testing coded features in increments. These increments allowed developers to analyze the code, make adjustment notes, and pass the notes to the testing team. When implementing the filter features, the development team was in constant communication with the testing team, verifying all requirements were met. In Agile development, a system called pair programming is enacted. In pair programming, two developers work on the same code, allowing quicker analysis of successes and failures. This practice allowed our newer team members to learn from our tenured members and fostered a collaborative atmosphere. It cannot be understated how crucial developers are to Agile development because they are a key to cross-team communication and collaboration.

Pulling from the developer role, testers are the next role I want to discuss. I touched on their contributions in the developer role summary, but I want to home in on a couple of things: accuracy and quality. Testers provided us with detailed test cases, utilizing the user stories, and applying the two in ensuring each met the acceptance criteria. In our sprint reviews and retrospectives, testers communicated their findings to other teams, ensuring continuous improvement and accountability. Testers gave us accurate feedback that we applied to each phase, while respecting any changes requested by the client or other teams.

Each role played a part in the success of the SNHU Travel project. Previously, I mentioned how the Agile approach helped us define user stories, adapt to changes and roadblocks, and how communication was effectively enacted. I want to touch on the organization tools we used and how they were utilized in the development process and events. By having daily scrums, sprint planning sessions, sprint reviews, and sprint retrospectives, I created a structured framework and plan for the development process. In sprint planning sessions, levels of priority for each task were assigned, ensuring alignment with the product’s backlog. During daily scrums, team members openly discussed their successes or issues encountered, playing a significant role in fostering the atmosphere of constant improvement and transparency. When sprint reviews were held, the team members could present their finished work to the client, allowing us to gather feedback and make any necessary adjustments to ensure alignment between the client and the development team.

The Scrum-Agile approach provided us with many pros and cons that I want to touch on, starting with some pros. For example, the defining characteristic and benefit of the Agile process is the adaptability allowed through the flexibility provided. As the project progressed, our team was met with shifting requirements that needed to be met as we developed the software. When the Product Owner told the team about the changing requirements of the filter features in our sprint. The new requirements were communicated across teams, allowing us the flexibility to be on the same page and implement any changes. In VersionOne’s S*tate of Agile Report (2022),* my observations are supported by their findings that Agile practices allow quick adjustments across teams. These practices allow us to create a great final product that meets old and new requirements, ensures customer satisfaction, and improves team communication and collaboration.

Next, i want to discuss the cons of the Scrum-Agile approach. When implementing the Scrum framework, there is an admittedly large learning curve. This curve does decrease as time goes on but going from waterfall to agile means adjusting to new practices, roles, and team dynamics. Research done by *Hoda et al (2011)* found that organizations transitioning to Agile would often face challenges in new team dynamics and team members accepting their given role. Another con that I would like to discuss is a technical term: scope creep. If management of the project were to falter, this would lead to scope creep. During the SNHU Travel project, we received feedback that had us implementing new features, increasing the scope of our work. With good management, adjustments, and feedback, we ensured this would not become scope creep.

Now that the project has concluded, I can say with confidence that the Scrum-Agile approach was perfect for the project. Although there was potential for scope creep and early difficulties in learning the approach, adjustments were made to ensure they did not occur. Team members received support from other team members, creating a pipeline of knowledge between new and old members. The iterative approach that Scrum utilizes allowed our teams to deliver a great final product that met changing needs and requirements. The roadblocks and cons we discussed served as lessons for the future, another feature of the approach – improvement through feedback and experience. These experiences give valuable insight regarding transitioning to the Scrum-Agile approach throughout ChadaTech.

References

* Hoda, R., Noble, J., & Marshall, S. (2011). **The impact of agile practices on the satisfaction of software developers**. *Information and Software Technology*, 53(9), 959-968.
* VersionOne. (2022). **State of Agile Report**

*For additional information on APA Style formatting, please consult the* [*APA Style Manual, 7th*](https://apastyle.apa.org/style-grammar-guidelines) *Edition.*