Michael Rosati

Final Project – Sprint Review and Retrospective

During the SNHU Travel Project, each member of the Scrum team contributed in important ways to the success of the project. The Product Owner’s job was to collect and coordinate the users’ and other stakeholders’ requirements and needs for the project and determine the relative priority of each user story in the backlog so that the most vital elements—for example, the search functionality and its siblings, the application layout and appearance, and the product’s focus on detox and wellness trips—would be addressed and implemented first. The developers worked to estimate the complexity of each user story in the backlog so that they, and the rest of the team, would have a clear idea of what could be accomplished in each sprint. This helps the product owner and the team agree on sprints with specific goals that deliver measurable and visible improvements to the user at the conclusion of each increment. And, of course, the developers worked to implement each user story in code and verify the correctness of their work using test-driven development. The test group was responsible for cross-checking the developers’ work to ensure that their test-driven process did not leave any holes in the automated verification suite and worked with the product owner to devise and implement automated validation and acceptance tests, which is vital to ensuring the customers’ satisfaction with the product. The Scrum Master, meanwhile, helped to coordinate and facilitate the efforts of all the team members, organizing the daily scrum, sprint planning, sprint review, and sprint retrospective meetings and ensuring that no external factors would distract or hinder the rest of the team from achieving their goals.

The Agile approach helped the team’s user stories reach completion in several ways. First, thanks to the efforts of the product owner and development team, each user story was clearly defined and reasonably well estimated before work on the story began. This helped to avoid wasted work or midstream course corrections that would have resulted in lost time and effort. Second, the daily scrums served to keep the developers and testers abreast of what each team member was working on and what problems were arising so that a team member having difficulty could get help quickly without wasting undue time and effort spinning their wheels. Finally, the sprint review served as a demonstration platform and pencils-down deadline at the end of each sprint, providing a boundary to the work as well as giving the product owner and other stakeholders direct vision into the team’s progress over the course of the sprint so they could evaluate the work and declare it complete. Seeing the product in action on a weekly basis also gave the customers and stakeholders a chance to propose new user stories based on the work that had just been completed.

Our team experienced an interruption when the product owner determined, through customer engagement and feedback, that the SNHU Travel app should focus on providing detox- and wellness-oriented vacation trips to its users. Because the process was Agile and the product had been developed in an Agile way, this change in direction had little effect on the team. There was nothing in the code that had to be reworked to accommodate this change; all that had to change was the underlying data being displayed. In fact, the narrower focus for the app probably saved the team some future effort: the team would no longer have to develop features like filtering based on different kinds of trips.

When questions arose between team members, ad hoc meetings, daily scrums, and emails were all useful ways of getting those questions answered. For example, when the detox/wellness focus arose, the development team wrote an email to the product owner asking about the priority of this change compared to other features in the backlog. Additionally, the testers sought clarification on how to objectively determine, in a manner suitable for automated testing, whether a trip qualified as being focused on detox and wellness. By writing these questions and their answers down in emails, the team created an electronic paper trail that could be referenced in the future and copied verbatim into their user story descriptions so that these questions would not have to be repeated by other team members.

Since we were a physically distributed team, we did not have a Scrum room or a physical Scrum board. Instead, we used Jira as our enterprise tool for user story tracking and sprint management. Jira was invaluable to our efforts because it gave every team member immediate visibility of the entire sprint and backlog; everyone could see, from their home desktops, what everyone else was working on without having to wait for the next daily scrum. It allowed product owners, developers, and testers—people who are very comfortable with mouse and keyboard—to type their user stories and detailed descriptions into a reliable enterprise application and use familiar, computer-based gestures to manage the stories’ status rather than hand-scrawl them on notecards and stick them on a physical board that is prone to many kinds of failure. This helped most during sprint planning, where it is much easier to manage and track digital assets than physical ones. Jira allowed the scrum master to see the team’s progress and velocity, helping the product owner communicate progress and overall project estimates to the stakeholders while being on the lookout for any user stories that were proving more difficult to resolve than anticipated. And the digital scrum board was an easy reference for the product owner and stakeholders to see what was about to be demonstrated to them at the sprint review and what decisions had been made by the team in the implementation of each story.

The Agile process had many benefits for the SNHU Travel project team. It helped all the team members, not just a privileged few managers, collaborate at all levels of product design and development. It helped the team stay tightly focused on small, user-centered increments of development that could be implemented, tested, and delivered quickly, allowing for a fast feedback cycle between the team and the stakeholders. It helped ensure that the product and the team would be responsive to changing priorities and requirements, an inevitable truth in all software development. And it helped the team deliver a quality product on time and on budget that aligned with the customers’ needs at delivery time rather than as initially conceived months or years earlier.

While waterfall development and its associated artifacts such as Gantt charts may give some degree of comfort to project managers at the outset of a project, the stories those artifacts tell are rarely more than fiction. They offer only the illusion of predictability. Inevitably, a key requirement changes or an unforeseen delay presents itself, and the effects of such an obstacle ripple like so many shockwaves across the entire project plan, leaving chaos in their wake as managers and developers scramble to put things back together and get a product delivered on time. But without feedback mechanisms, the product they deliver at the end of the lifecycle, while it may resemble the initial vision for the product, is unlikely to meet the customer’s ever-evolving needs. In the end, Agile development with Scrum was the best approach for the SNHU travel development project.