Sprint Review and Retrospective

Jonathan Olson

Southern New Hampshire University

CS 250

October 17, 2021

During each iteration, the Product Owner assisted in identifying product backlog items, proposing Sprint targets, and sorting our backlog items by priority. Working with the Product Owner, I assisted in the facilitation of Sprint Planning sessions, ensuring that relevant backlog items were added to the Sprint backlog for each iteration and that Sprint targets were agreed upon.

The Product Owner and Development Team (including testers) updated each other on weekly progress and impediments during the Daily Scrum standup meetings, which were supported with Gantt chart graphics and Burndown charts. As a Scrum Master, I assisted in the removal of specific roadblocks and provided ideas for the development team to overcome such obstacles. A major makeover of the UI was one such occurrence; it required adaptation on the part of our entire team, but the underlying functionality remained mostly unchanged, and we were able to complete the Sprint on time.

Our entire team met once a Sprint for Backlog Refinement meetings, where the Product Owner delivered the backlog items that needed to be revised and solicited feedback from developers and other members. The Product Owner and Development team were in charge of developing the product, and as a Scrum Master, I assisted in providing feedback and removing roadblocks during the refinement phase. We convened a Sprint Review at the end of each Sprint to assess progress on that increment and, if necessary, adapt the backlog with the Product Owner. We also worked with the stakeholders to assist them optimize the value at each increment.

The formulation of our User Stories for SNHU Travel was aided by open communication between the Product Owner, consumers, and developers in the form of an interview; our Product Owner formulated our User Stories from the interview notes and categorized them into the product backlog. These User stories indicate the functionality and value that users anticipate from our software, so we included them in our Sprint Gantt chart and Burndown chart to precisely forecast their completion and "burn them down" over the project's lifespan.

The Product Owner could easily communicate the stories to developers and prioritize them based on User Value by providing a clear user value statement for each because the user stories were independent, useful, size-estimable, and tested. The team aided the Product Owner in writing user stories for "Custom Profile," "Customized Destinations," and "Choose Destination Package Type," to name a few. The Product Owner incorporated these User Stories into the Backlog for each incremental refinement, allowing the development team to prioritize items and estimate development scope, and providing us with a general schedule.

The Product Owner at SNHU Travel has entirely changed the direction of the Flight Packages interface in the middle of the project. If any essential functionality needs to be changed, the Developer conveyed that the Velocity chart and Gant chart should be updated to provide an estimate of whether these modifications could be completed within the existing timetable. Because this was mostly a theme update, the essential code functionality remained unchanged, and our team was able to complete the graphics overhaul ahead of schedule.

The testing team was asked to write appropriate JUnit tests, ensuring that tests for GUI functionality were written; Pair Programming between the lead Developer and lead Tester was arranged to ensure that all bugs were identified and fixed during that Sprint iteration, and the Definition of Done was met. This Scrum-agile strategy resulted in faster bug assignment and resolution.

Clear communication between the lead developer and product owner was critical during the previous example of the graphical user interface overhaul to avoid making unnecessary changes or providing a poor customer user experience, while also communicating realistic standards in the scope of possible functionalities and timeline.

The importance of frequent, clear communication was emphasized at SNHU Travel. In AGILE, a Sprint Planning meeting is held to establish our product's deliverables, followed by Daily Standup meetings to identify barriers and provide status updates. Product Demo sessions allowed us to demonstrate what our Pair Programmers had created, and Retrospectives allowed us to share lessons learned and improvements made. Stakeholders also communicated with the product owner and users, providing our team with a clear understanding of the business value and User stories we should aim for in our Sprint targets. For instance, during SNHU Travel, the UI was changed to a slideshow-based interface. The Product Owner communicated UI modifications and reasons, while the development team communicated time constraints and technical specifications. Because the core code remained mostly unchanged, the Sprint was completed on time. We aided with this by maintaining clear open communication about the current Sprint goals, Gantt chart estimation, and Burndown chart estimation for the week, after which we emailed the updated expectations to the Developer, Tester, and Product Owner so that we were all on the same page.

It's best to employ a combination of Scrum events like Daily Standup and Pair Driven Programming, as well as information radiators like Gantt charts and Burndown charts, to maximize a Scrum team's efficiency, working potential, and communication. Our team switched from Waterfall to AGILE technique during a Scrum event. In that case, I implemented Jira development workflow and cultivated a culture of communication by responding quickly, updating Project Requirements, incorporating Burndown charts into our workflow, keeping parties on track through automated text and e-mail alerts, and meeting the deadline with our Shareholders so that we could share the progress of our current Sprint and backlog items.

With my effective guidance, we communicated the critical goals, user stories, fresh information, and critical knowledge via Daily Standup (as the Scrum Master). This fostered an atmosphere of openness among the participants.

I'd want to focus on a single Scrum event: our initial de-investment from the Waterfall technique and introduction to the AGILE methodology. We implemented Jira in our first scrum, which offers a range of information radiators and tools that aided the entire team throughout the project, from stakeholders' User stories to developers' code bases, Project Management for Product Owners, and charts to view project length until completion. Jira also offers developers continuous integration and deployment, product roadmaps with integrated Gantt charts and Burndown charts, and shipping software workflow options. Our Product Owner was about to manage her backlog and track our initial release's progress. All these factors combined have made communication and distance development easier. Gantt charts offered us a notion of where we were in the project, giving us a timeline that was accessible by any member; burndown charts helped track the team's velocity and production, improving productivity; and burndown charts helped track the team's velocity and output, improving productivity.

The most difficult aspect of introducing AGILE into our workflow and switching from the more straightforward Waterfall technique was the implementation. To adopt the AGILE techniques company, time, money, and effort were necessary to learn, and an investment to comprehend the features of AGILE methodology was required on all parts throughout the culture transformation. Furthermore, AGILE introduced an element of unpredictability that was not present in the Waterfall methodology. It necessitated honing our project scope and adaptability prediction skills, as well as complicating the budget and marketing/shareholder strategies. We were concerned about AGILE's scalability in terms of development team size because, with over 40 engineers on staff, it was difficult to maintain coherent 5-man teams. On the good side, it ushered in rapid development, feature release flexibility, a willingness to embrace uncertainty and adapt to changes, and sharpened responsiveness to change, with a focus on user stories and changing market conditions for shareholders. In the case of the SNHU Travel Project, AGILE proved to be the optimal strategy because the user requirements changed over time, and our team was able to leverage rapid AGILE development to generate a functioning prototype to demonstrate to shareholders and consumers, resulting in a customer-centric solution.