Dre' Scheetz

CS 250

Denise Washington

10/13/2021

Sprint Review and Retrospective

          With the evolving tech industry, reforms to management and working environments have become a necessity. One innovation that has gained ground in being adopted by the industry is the Scrum-agile approach. Part of the approach is iterative design through sprints, and a Sprint Review and Retrospective is an opportunity to reflect on the success of a Sprint and how to improve for the next iteration. This essay is an introspection of the Scrum-agile team's performance for the past eight weeks, including observations and thoughts on the Scrum-agile process.

Each team member was indispensable to the success of the SNHU Travel Project. The Scrum Master executed the Scrum Events effectively, allowing for team cohesion, self-organization, and ownership. This was shown during the daily scrums, addressing the progress made and plans for what would be done that day. The Product Owner efficiently ordered and created the SNHU Travel Product Backlog. Communication was open with stakeholders, and the product vision was succinct regardless of requirement changes. An example of this was the Product Owner emailing one of the stakeholders on if the header should be implemented for the Detox and wellness change.

Clear tests from the Tester assisted the team with functionality for the Developer to use as a framework for the code's design and development. In addition, the Tester ensured the quality of the code before deployment through rigorous testing and code being flagged for non-functionality. Feedback from the Developer and Tester also proved vital so that the Scrum Master could remove impediments, such as the software complications of the IDE being used by the Developer.

           A Scrum-agile approach facilitated the product iterations well. The Product Owner first met with stakeholders to decide and order the Product Backlog, then addressed user story changes made by the stakeholders. The Product Backlog consisted of the user stories meeting what the stakeholders wanted from the product. The two that were deployed were the User Stories for a top-five destination list and the change to detox and wellness slideshow. These user stories were described in more detail through the acceptance criteria the Tester used for creating the Definitions of Done, allowing the Developer to implement the changes to the product iterations. For instance, "As an end-user, I want to click a link to view the top five destinations list, so that I can see the most popular locations" this would be expanded upon in the acceptance criteria through a scrollable list, ordered adequately. The Tester then further clarified this in the test case.

The Scrum-Agile approach proved to be vital to the process once requirements shifted. During the SNHU Travel site development, a pivot to vision was made from the most popular to wellness and detox-oriented destinations. Each team member supported this change by addressing their functions within the team, allowing for a smooth transition. With the Waterfall method, mid-stream changes could have been problematic, and the strictness of adapting to new requirements may have disrupted time, scope, and budget.

The development team demonstrated the values of Scrum. According to Schwaber and Sutherland (2020), these values are "commitment, focus, openness, respect, and courage." The Tester demonstrates these values with the following, "My primary concern is that the use of a slideshow as opposed to a list will be an inadequate user interface experience. As usability can deter potential customers, this format may be inappropriate for browsers."

The Developer's email shows effective collaboration by requesting information from the Product Developer and Tester, sharing why the information was being asked and their plans for openness and transparency. Per the Developer's email, "If so, can you email me the following user story to be implemented or specific story points? I will be updating the Kanban before our next Daily Scrum." Another section directed towards the Tester, "Can you email me the test case and acceptance criteria for the next user story once updated? I want to use the acceptance criteria as a framework for developing the functions of the following user story."

Some organizational tools can assist these interactions between team members allowing for better team cohesion and communication. Tools like Microsoft Teams or Slack can help team members better communicate or be used for Daily Scrum through video conferencing, predominantly for remote work. Potentially this would be useful for stakeholders, the Product Owner, and Scrum Master communications as well. Tools such as JIRA are practical during the software development life cycle by allowing real-time remote access to the Product Backlog, estimates, commits, product issues, and more. This tool can house each sprint iteration, including Sprint Planning and Scrum Artifacts, if properly utilized.

           When choosing how to proceed with a project, many factors must be taken into consideration. The scope is one of these primary factors. For a smaller project, multiple iterations may not be necessary to complete the project. A case for this would be the rapid development of tools for internal use. Optimization and code refactoring may be unnecessary, and a single developer may only do the project during the production of another project. Another example of this could be for indie developers or freelancers. The former most likely understands the majority of what will be implemented and has a clear goal for the product. A freelancer may only be hired to refactor a single system in a more extensive system with clear and concise requirements.

           The pros of agile are in its flexibility to adapt to different requirements, team collaboration, simplification of iterative design, reduced upfront planning, having a functional product during development to allow for testing to be completed proactively, and more transparency/openness between developers and stakeholders. For cons, many are part of the difficulty of correct implementation. As discussed by Cobb, Scrum-Agile is not a quick fix or band-aid to apply during a rough development. Scrum-agile cannot be implemented overnight, and during these times, a bottom-up approach using a hybrid "Manage Agile Development process" should be considered (Cobb, 2015, p. 290-297). Other potential issues that may arise could be from refactoring not being done due to time constraints of overly optimistic estimates, a backlog not correctly ordered, and too many requirement changes attempted to be implemented at the request of stakeholders. Trying to use agile in a fixed-price contract should be done with caution and adequately assessing the risks involved.

While agile may not be the most suitable choice for every project, the Scrum-agile approach was optimal for the SNHU Travel project. In the competitive marketplace for general business ventures such as travel agencies, being designed for fluctuation can help maintain the relevancy of the product and service with current trends. Pivot design through iterative planning and development can produce higher quality products during the Software Development Life Cycle with consideration to the evolving marketplace.

Reference

Cobb, C. G. (2015). *The Project Manager’s Guide to Mastering Agile: Principles and Practices for an*

*Adaptive Approach* (1st ed.). Wiley.

Schwaber, K. & Sutherland, J. (2020, November). *The Scrum Guide: the definitive rules of the game.* Scrum.org and ScrumInc.