7-1 Final Project

Sprint Review and Retrospective

Brandon Shands

Southern New Hampshire University

CS-250: Software Development Lifecycle

Professor Trevor Hodde

October 12, 2022

**Applying Roles**

Software developers often find themselves working on large projects with many moving parts. As such, it is most common for teams to use an adaptive style of project management that requires a high level of collaboration between members. This course has focused on the Agile methodology for management, or more specifically a certain framework called Scrum. In order to understand the philosophy behind it I have been tasked with taking on the role of various members of a Scrum team.

In order to understand the methodology, it can be helpful to focus on the Scrum Master first. Often seen as a support role, it plays a vital role in upkeeping the style’s principles in order to ensure the team’s success. When serving as Chada Tech’s Scrum Master I facilitated meetings and events, helped coach others on the Scrum philosophy and removed obstacles that could have impeded the rest of the team. My overall goal was to help the team collaborate effectively. For example, I would schedule the daily scrum meetings, kick them off with a brief prompt, and then sit back and observe how everyone else approached the project. I would only step in if I felt the team needed some guidance or other coaching. In fact, the less I need to directly intervene the better. It means that the team is already operating effectively and I’m succeeding in my task.

Scrum’s focus on collaboration means that all roles play a crucial part, and as such there is generally less of a focus on a traditional “big boss” that micromanages their subordinates. The closest equivalent in Scrum would be the product owner. Their main goals are to serve as a liaison between customers/stakeholders and the team, as well as maximize the value of the product through backlog grooming. As Chada Tech’s product owner I played a crucial role in meeting with customers to determine the requirements for the product. I needed to ensure that I had the best possible grasp on the details because I would be regurgitating it to the rest of the team by creating user stories. These are simple explanations of specific software features written from the perspective of an end user. This is an important step in breaking down the complex project into more manageable chunks.

Developers serve as the backbone of the Scrum methodology. They are the ones that will be creating the actual product code. However, Scrum’s focus on collaboration allows them to have a direct impact on the way the team will approach the project. The product owner will groom the backlog, but developers will help determine how to approach the necessary user stories for each individual sprint. Developers will also act as testers and create test cases detail the requirements for specific features. This can even help reduce the number of bugs that might appear throughout the rest of the life cycle, making subsequent testing easier. Scrum is a highly adaptive methodology and as such the development team must adapt daily in order to reach the sprint’s goal. As a developer for Chada Tech I attended the Sprint Planning meeting to lay the groundwork for the upcoming sprint. Once it was in motion we had daily scrum meetings where we reevaluated our approach and communicated our daily tasks and potential issues. Finally, I used industry best practices to create code that met the success criteria detailed in the test cases. At one point SNHU Travel changed the project’s focus, and I had to adapt my already written code to account for it.

**Completing User Stories**

Scrum played a major part in how the team handled the SNHU Travel project. First, the product owner met with the customer to determine the requirements and broke them down into user stories. We then had a sprint planning meeting with the whole team where the product owner explained the goal of the upcoming sprint. The team worked together to create the sprint backlog, a plan for success where we determined which user stories to focus on first in order to receive the most value. User stories were assigned to team members, who would attempt to implement the feature and pass the test cases determined by the tester. We had fifteen minute daily scrum meetings in which we discussed our individual progress and any issues. We were also able to adapt the sprint backlog to account for new information, such as SNHU Travel’s decision to change the focus to health-related vacations. At the end of the sprint we held a sprint review to receive feedback on our implementations.

**Handling Interruptions**

One of the primary benefits of an agile approach is its adaptability. Midway through the project SNHU Travel changed the project’s focus to health-related travel and decided it would be more beneficial to use a slideshow format. The product owner needed to reorganize the product backlog in order to account for these changes. Some user stories needed to be adapted and test cases updated. Developers were then required to change their code in order to meet these new requirements. Luckily our short sprint durations and daily scrum meetings meant that we were always willing to adapt to potential changes. Had we used a waterfall style of project management it would have most likely been a much larger issue due to the amount of concrete planning involved. Chada Tech’s approach expected a level of uncertainty and definitely helped lessen the impact.

**Communication**

Scrum’s focus on collaboration and adaptability requires good communication to function properly. Team members should be as detailed as possible when discussing the project. Poor communication can lead to needing to change already written code or redundancy, wasting both time and money. Below is an example email demonstrating good practices:  
  
To: Christy (Product Manager)

CC: Brian (Tester)

From: Brandon (Developer)

Dear Christy/Brian,

Now that the customer would like to focus on detox/wellness packages I had a few additional questions. Given the importance of changing our focus, would it be best to try and implement the wellness aspects and moving other features like personalized user suggestions further down the product backlog? I would also like to ask Brian to clarify the exact ways in which we will implement the new focus on health. Will we simply highlight the health “type” of packages on the homepage or will we just include a higher percentage of health packages in user search results? Please let me know when you have finalized the test cases so that I can begin working on this section.

Thank you,

Brandon

**Organizational Tools**

Luckily there are several tools that can help the team operate efficiently. Things such as email, video chat software, and cloud-based storage can help ensure that team members are able to interact and share data when not in the same location. Skype was especially useful for when someone was unable to attend the daily scrum in person. There are also tools with a more direct focus on project management. Jira is a project-tracking software that was extremely beneficial to the team. It provided the use of Agile and Kanban boards and the ability to track real-time performance reports. Team members could view the product backlog and see roadmaps for the future. We were even able to create and manage user stories and test cases. Jira allowed the development team to stay better connected to the project whether they were in the workplace or working remotely.

**Evaluating Agile Process**

An agile approach was hugely beneficial to the completion of the SNHU Travel Project. Short sprint periods and daily meetings allowed us to quickly adapt to any potential issues. We were able to pivot without any major complications. This would have been a major wrench in the gears if we had more concrete plans. An agile approach also allowed us to break the project into digestible chunks and focus on completing the most essential functions first, resulting in less bugs, fewer revisions, and an overall superior final product.

A standard waterfall approach would have likely kept the teams more compartmentalized. This has the potential to result in major wastes of time and money. That said, there are certain downsides to scrum as well. Some will find the sheer number of meetings required an annoyance. The collaborative focus only works if all team members are willing to put in the required effort. It also works best for small teams, becoming increasingly difficult to manage for large teams. Perhaps the biggest con would be scope creep. Waterfall’s strict end dates can be stressful, but the same can be said for not having any definitive project end date. However, I believe Chada Tech made the correct decision when embracing agile. Our small team was able to adapt quickly to changes, communicate well, and keep costs low. Each company and project is different so I can’t say that an agile approach will always be the correct solution, but it works well for Chada Tech and I believe we should stick with it as we move forward.