**Sprint Review and Retrospective**

Ezra Huffman

Department of Computer Science, Southern New Hampshire University

CS 250: Software Development Lifecycle

Prof. Clinton Bush

December 11, 2022

**Sprint Review and Retrospective**

This sprint the team used an agile methodology for the software development lifecycle. According to agile principles the team assumed roles within this framework, adopted common agile practices and tools, and did our best to embrace the agile principles. The specific roles that the team assumed during this spring were scrum master, product owner, tester, and developer. These roles came with expected responsibilities within the Agile methodology and were essential to the team’s ability to succeed. The scrum master planned and executed scrum events by setting up a shared calendar and sending reminders through a chat app or talking to them in person. The scrum master also helped maintain events such as daily scrums, backlog refinement, and sprint review. Prioritizing and overseeing the development team’s tasks was the product owner’s job. The product owner would define user stories, manage the backlog, establish a plan, and generally ensure that the product was delivering to stake holders. Testers within and Agile team are responsible for developing test cases and quality assurance throughout the development lifecycle. Unlike other development methodologies the testers are still part of the team, even if they have differing roles and expertise. The tester contributed to the success of the team this sprint by constructing test cases for the user stories and giving feedback to the rest of the team based upon the results of those tests. Lastly, but certainly not least, is the developer. During this sprint the developers were responsible for updating the Slideshow based upon new user stories created by the Product Owner. An additional way in which the developers contributed to the team’s success was by communicating with the team and offering transparency into the project’s goals and progress.

The Agile methodology helped the team succeed in changing directions this sprint by allowing for more iteration upon the product. A specific instance of this is the decision to pivot to a wellness/detox audience. The slide show was updated to cater more the wellness/detox community and was done so without trouble because that type of iteration is expected and welcome in an agile environment. The product owner updated the user stories and priority of tasks, while the development team implemented the changes and testers ensured the quality of those changes. Without the agile methodology this type of iteration could have been much harder to implement as it would not be planned for and there would not be as clear of a strategy for handling situations of this type.

One of the main points and advantages of the agile methodology compared to other approaches is the focus on communication. This focus on communication prevents silos and encourages inclusion and diversity of teams and ideas. During this sprint there were a couple of instances of communication amongst the team that are easily referenced. For simplicity I will list one here. The following is an email from the developer (Me, Ezra.) to the product owner (Bob).

Hi Bob,

I have just submitted my changes for the current user story I am working on Update Slideshow to Detox/Wellness (link if applicable). Can you give me more context into what drove the decision to switch focus on the client’s end? I would like to have a sense for the customers broader goals so I can better work towards those goals.

Additionally, while working on my last task I noticed that the blue background for the text makes the descriptions difficult to read. This is probably something we should consider changing moving forward.

Thanks,

Ezra

The above shows how team members are encouraged to communicate in the team and give their opinions and ideas for the product. This touches on another aspect of the agile methodology, ownership. Ownership just means that team members are given more freedom to make decisions regarding product development and assume more responsibility regarding the products success or failure.

To better implement agile methodology the team implemented some practices and tools. The most useful and prevalent being the daily Scrum meetings and task board. The daily scrum meetings are short (about fifteen minute) meetings where the team goes over their progress since the previous meeting, what they are working on, and to reach out if they are blocked. These are a central part of scrum-agile practices and encourage a lot of the goals of agile methodology. Having daily short meetings with the rest of the team encourage collaboration and a diversity of ideas. Task boards also help facilitate scrum-agile practices. The task board is an easy way to organize user stories, sprints, and a backlog. The task board serves many purposes, but one of the major advantages of using an online task board such as Trello, Jira, or similar services is that it makes the development process much more transparent. This allows for the team to realize what tasks are problems more quickly (“Fail early, fail fast”) and for stake holders to easily track progress of the project in its current state.

The Scrum-agile approach was effective for this project. With the ability to quickly adapt to changes and iterate on ideas, the agile approach lent itself well to this project. During the project the requirements changed at least once to focus on wellness/detox and this was quickly addressed by the development team by creating a new user story and test cases for the change. Without the focus on communication, iterative cycles, knowledge sharing, and transparency that agile provides, this type of change would have been much more difficult. There were not many cons in this project, so it seems that the Scrum-agile approach is a good choice for this project. If there were more of a focus on strict deadlines and budgets, then a completely Scrum-agile approach might not be appropriate and something closer to a waterfall approach might be better.