Ken Rodriguez

Professor Morrison

Software Development Lifecycle

June 14, 2021

Final Project

Looking back on this release, it has become very apparent to me how each role in the Scrum Team is necessary for fast and cost-effective development. The Product Owner provided a way to communicate with stakeholders and the end user of the SNHU Travel product to develop and manage user stories utilizing focus groups. The Scrum Master organized Scrum Events such as the daily standup meeting, getting the development team to a place where they can organize these Scrum Events on their own, in turn increasing the amount of autonomy the development team has in the organization. In addition to working towards autonomy, the Development Team also worked on user stories based on the feedback from stakeholders and users to develop features. The Testers then provided feedback to the Development Team in response to test cases to address the features that were being added and whether the features the Development Team had submitted were working as intended or not.

After the focus group meeting, the entire Scrum Team developed user stories. The Product Owner would then take a Scrum-agile approach to order these stories based on their importance to the application; in the case of the SNHU Travel project, I had managed the product backlog to prioritize features that I believed would be most useful for a commerce website, such as a top five list of the best-selling vacation packages, a history of the user’s purchases, or a “sort-by-price” feature. Assigning degrees of importance to these user stories provides a way to determine how many user stories can be completed within a sprint and allows the Development Team to budget their time appropriately to ensure that a complete and functional product is shipped by the end of a sprint. For example, in the SNHU Travel project, the “top five” user story was a top priority in the product backlog and considered a medium-sized task. The “purchase history” user story was next on the list but considered a large task. While many smaller user stories followed the “purchase history” user story in the product backlog, the size of the Development Team might require that only the “top five” and “purchase history” user stories are accounted for during the current sprint while the remaining smaller user stories are saved for a later sprint to ensure the team stays on schedule. Managing a product backlog provides an opportunity to ensure that the Development Team and Testers aren’t overwhelmed by user stories while also ensuring those stories are completed in a timely and logical manner.

An important aspect of the agile approach to development is a focus on working in short increments to ensure that the organization can release and build upon a functional product within a short span of time to stay competitive within its respective market; this philosophy should ideally extend to its code. In the SNHU Travel project, this was demonstrated when the Product Owner requested that the product shift from the vague concept of “vacations” to a more specific focus on vacations that could provide holistic or opportunities detoxification. While the product had made a great deal in progress during development, the existing code was developed with the agile philosophy in mind and all that needed to be changed to adjust to this new vision were a few pictures, titles, and descriptions. If more code had needed to be changed, the Scrum principle of short sprint lengths would be conducive to remedying most problems that may arise; instead of having to refactor months or years of work, the Development Team should (ideally) only need to account for a few weeks.

Communication is an important aspect for the Scrum Team’s success. Every aspect of development from the Scrum-agile approach encourages communication, from the Product Owner’s meetings with stakeholders and focus groups, the Scrum Master’s various organizational tasks, and the entire team’s participation in the creation of user stories. The Scrum process provides ample opportunity to request context on what is going to be developed and provide feedback on what has already been developed. For example, I wrote the following to my peer Jose Ortiz recently:

“Hello, Jose! I hope you’re having a great day!

The importance of communication in the agile framework cannot be understated; the deadlines are tight and it's important to make sure that everybody's working towards the same goals. I believe this aligns well with adjusting user stories, as you've suggested. I would suggest that we implement sprint review meetings to get a better idea of how user stories can be improved or modified to better suit the product's vision. As product owner, this would give me time to gauge our progress and communicate with the stakeholders what features are working as intended (or better than intended) and give the rest of the team a specific and scheduled time to figure out what isn't working and how to better implement or totally remove those features moving forward. As the idiom goes, hindsight is 20/20, so why not capitalize on that clarity and prepare for the future?

Great work this week, I hope you have a great day!”

In this context, I was given an opportunity to acknowledge the context of the original message, the importance of communication, and an opportunity to suggest what Scrum practices we could consider moving forward in development, in this case, sprint review meetings. By leaving the suggestion of incorporating sprint review meetings open ended, I believe this creates opportunities for further discussion on the topic; perhaps Jose or another one of my peers has an alternative to sprint review meetings that I am ignorant of and may be more effective for our team. In my opinion, maintaining a positive tone encourages and environment that is also more conducive for constructive discussion.

Working on the SNHU Travel project, I feel that user stories have shown themselves to be one of the most important principles of Scrum development; they provide an opportunity for the entire team to suggest features, a suggested order for those stories to be completed, and a way to elaborate on the specific facets of each feature. From my perspective, without user stories, there is no product; the SNHU Travel team likely would have struggled without them. From a more general aspect, the Scrum principle of communication also seems instrumental in the Scrum Team’s success working on the SNHU Travel project. Meetings with the stakeholders and focus groups provided a vision of the product, meetings involving the Scrum Team in its entirety throughout development offer context and opportunities to adjust to better meet the vision of the product and reprioritize or adjust less valuable user stories. A good example of this communication in action would be the application of test cases. During the SNHU Travel project, I had written and submitted a few test cases to the Testers; upon review, the Testers communicated to me that the test cases I had written were incredibly vague and could be revised to better account for possible bugs. Without that feedback, it’s possible that the product I submitted could have bugs that nobody accounted for.

In closing, the Scrum-agile approach for the SNHU Travel project provided a fast and effective way to create and release a complete product with opportunities to add or adjust features after release. This development process allowed the Scrum Team to release a product to the market in a fraction of the time that the same product would have released under the waterfall development model. The Scrum process also allowed that product to adjust with the market, allowing for numerous opportunities for the stakeholders and focus groups to provide input on features they would consider important for a product. On the other hand, I could see this approach having consequences if the project’s scope expanded too much. As a project grows and more interconnected systems become necessary, additional planning might be useful for ensuring that this connectivity is coherent and easy to maintain. In the scale the SNHU Travel project took, however, I would consider the Scrum-agile approach the best approach to take. The features required for this project were simple enough and separate enough that splitting development into sprints comes across as a logical approach in my opinion; for similar projects, I am eager to try the Scrum-agile approach again.