Dakota Condurso

Professor Mason

April 15, 2024

CS-250

In order to successfully implement and utilize the benefits of a Scrum-Agile Methodology four roles were named. These roles were the product owner, scrum master, developers, and finally testers. Leading off with the role of product owner, the product owner did an excellent job in gathering the needs of our stakeholders. In addition to this, changes within the scope were carefully thought out and priorities as well as needs were altered as needed and in alignment with both developers and customers. One such example of this, was the shift to a focus on wellness/detox travel, in which the product owner acknowledged the constraints this change was facing and in turn deprioritized other stories in the backlog to accommodate the planned deadlines. Moving forward, as a scrum master I feel I did adequate job within this role. Elaborating upon the previous example, I played a large role in this reprioritization and was the first to question the product owners scope change in regards to timelines and priorities. In addition, I also regularly facilitated events critical to the functions of agile and with that assisted in creating a well working team and environment.

Changing paces to the more technical roles, our developers did a great job in maintaining a high level of work quality as well as exchanging information throughout the software lifecycle, this including both progress and setbacks during daily standups. Finally, our testers did a wonderful job in developing test cases and interpreting test cases in a way that helped to develop the product in a meaningful way. In addition, clarification and recommendations were asked for and stated as needed for development.

The Scrum-Agile approach helped to bring each of the user stories to completion through the allowance for adaptation as well as through the empowerment of both testers and developers to make decisions. This is especially evident in the design of test cases in which user stories were utilized by our testers to expand upon testing requirements and add features that may have been glossed over had a traditional waterfall methodology been utilized. Changing topics, and now going back to the example concerning project scope changes, Scrum-Agile provided a great methodology to manage inevitable interruptions and changes in direction. As the previous example suggested, when priorities for the project changed the team was able to quickly change course by physically alerting the prioritization of goals within the product backlog. This is an attribute that’s unique to agile, and had our methodology still been the traditional waterfall method, this change would have likely had lasting repercussions in the form of quality or timeliness.

One example of communication I feel we did exceptionally within the team was during our use of both story boards and stand-ups. For story boards in particular, our tester was eager to gain more info regarding requirements and wrote

“Hello,

While the initial user story has been exceptional in providing a basic set of criteria for test cases, I do feel the product could benefit from further elaboration into user needs and desires. This having been stated, would it be possible for an epic to be provided to further elaboration on the user story?” This in response led to the creation of an epic that resulted in even more communication regarding the needs and desires of our customers and helped to better tailor the product. As for standup’s, individuals setbacks and progress were routinely voiced and helped to create a more well-informed team throughout the software’s lifecycle.

For tools utilized during the Scrum-Agile process one of the most integral was the use of project management tools such as Jira. This tool allowed our team to help visualize our product backflow through various charts and through its use as an information radiator. This feature was especially useful during standups as a supplementary way of describing both setbacks and progress.

Having gone through these elements of Scrum-Agile in relation to this project, Agile was a greatly effective methodology through this software’s development with only some considerations to be made. Starting with the pro’s however, Agile allowed for an incredibly fluid and collaborative environment. This was evident in the use of story boards in development and how both testers and developers utilized this information to intuitively create features in accordance with customers wants and needs. Along with this inevitable scope changes were dealt with in an incredibly efficient way with minimal setback. As for cons the main problem with this methodology is maintaining sight of bigger picture with goals inherently being sectioned into small tasks. This drawback, however, is entirely manageable with proper techniques and communication. For these reasons, I feel Agile was absolutely the best path forward for this particular project and any like it in the future with many of the previous drawbacks accustomed to waterfall being all but dealt with.