CS-250-T2785 Software Development Lifecycle

Southern new hampshire university

7-1 Final project: retrospective

Wills, Christopher

2022

Going through some of each of the roles present in in this course helped shed much light on how a successful Agile development project is supposed to look like. Starting off with the role of Scrum Master, I was able to contribute to the success of the project through means of facilitation and communication. I was able to assist the team in reducing pain points that popped in during development. One example of how this can work is if one of my team members is experiencing difficulty in performing work due to glare and brightness coming in from a window. It is my responsibility to find a solution there, such as installing blinds or a curtain, to help foster an optimal work environment. In being the Scrum Master, it fell on me to schedule out events for the team to come together to share their progress, issues, and anything else they would like to share. Hosting Scrums is one manner of completing this task, and by leading the team in how the Scrum ought to look like I can help reduce any apprehension to communicate freely.

The Project Owner was key to the project, as they are the direct line to the client the whole project is for. In collecting requirements for the project, as well as gathering user stories, they obtain the necessary information to steer the project. A team can’t really create a viable and concrete product if we don’t have someone working to translate a client’s vision into something actionable. The Product Owner also assists the team in clarifying dubious product items, ensuring that the team has a clear view of what the product ought to look like. The Product Owner also takes responsibility in updating the client on the progress made, as well as consistently checking if any requirements have changed. An example of this encountered in this course is when SNHU Travel desired focus to be given to detox/wellness vacations. The Product Owner promptly took this information and shared it with the team so that the developers were able to shift focus to this new Product Backlog item.

Speaking of developers, they are one of the workhorses of this development methodology. Without them, and testers, no product could have any hope of coming to light. Developers are tasked with working through the Product Backlog item list, as well as taking user stories and translating them into actionable items to be worked on. An example of this from this course was the user stories that were gathered by the Product Owner being sent to developers to sort out based on priority as well as creating task list to determine pass or fail on items being implemented. Developers also have a close relationship with testers. They often will work concurrently on issues to ensure that fixes are being implemented in a timely manner as their work progresses.

Testers are the other workhorse of the Scrum team. As mentioned before, they work closely with developers to share issues they have found so that bugs can be fixed much more quickly and more often than what the waterfall method would entail. Testers are also responsible for going through the pass/fail lists to ensure everything is properly working. Testers often try to take the view of the user, as they are essentially the last line of eyes to see the product item before it is pushed out as completed. Creating test cases in this course was one example of how testers use both a pass or fail mentality and a user viewpoint to ensure the product is coming out successfully.

In using a Scrum-agile approach, we were able to create actionable items to help create and implement solutions that came from issues raised in our user stories. In this approach, one major aspect that helped was the freedom of the team to issue priority to some items over others. While this may lead to some smaller requests being left out, it helps ensure that the major requests coming from users are realized. This was seen in our user stories collection and test case creation. We were able to gather pertinent information to determine what the most necessary features were so that they could be focused on. Another aspect of agile that can assist in seeing all requests being answered is the flexibility on work requirements. An agile team is much more likely to share information with each other that alleviates issues when they might arise. Instead of specialization occurring, many members can touch the same product item. This allows for more eyes and more minds to view a particular problem, which gives us more opportunities of a solution presenting itself through one of the members.

An agile approach needs to be open and understanding of change when it comes. An example that I used previously of the clients desiring focus on detox/wellness vacation had the team encounter change concerning the project. The ability to shift focus easily and the freedom to change priorities creates an environment that embraces change. In this instance, the developer had already created a list of vacation spots, so they were able to recycle some of the work already done to easily implement the change to the newly desired list. When coming into agile development, embracing change is a key skill to have.

Open communication is imperative when using an agile approach. With generally smaller teams compared to traditional development, collaboration is immensely important. If communication comes down in a small team like this, it can spell disaster for the product in terms of time requirement and completion. One example I thought was indicative of how agile communication should look like was our discussion post topic in which we would choose a role and communicate with each other from said role. By being open and honest during communication, we were able to share information more easily with one another that ultimately helps us grow as a team. When we grow as a team, we also see that the product improves because of it.

Once again, coming from the discussion post I mentioned above, some agile tools that exist out there help tremendously, both in item execution and communication. Having a physical board to place Backlog items being worked on can create a sense of ownership to the item being worked on but having a virtual board in addition to this enables for quicker information updates being more accessible. One tool that was mentioned in our discussion group was Asana, which would give us a space to meet up online to share progress on our items. These tools can prove beneficial to the client as well, as they are much more accessible to view the overall progress and problems that arise from the project.

When looking at the SNHU Travel Project, I think using a Scrum-agile approach was the best choice to develop the project. With it being on the small side of project size, using an agile approach is much more feasible than using a more traditional approach. If it were a large size project, agile may not have been the best choice, as it really shines in small to medium size workloads. In using an agile approach here, we were able to communicate more effectively and often with the client in terms of requirement gathering and updates. There weren’t any incredibly specific requirements that required advanced specialized help, so having a small team containing a general skillset proved to be effective. We were also able to implement changes much more quickly compared to traditional development when the request came in. It was almost immediate compared to the months it could have taken in a waterfall approach. Testing occurred much more frequently here, allowing for issues to be addressed during the initial stages of development. In the waterfall approach, we would have not seen testing done until the end of the project’s development. Ultimately, agile was the right approach for SNHU Travel.