Each of the roles helped contribute to the success of the SNHU Travel project in their own way. The Product owner was the front man that communicated with the clients to get a comprehensive plan of what is to be expected from the team and what they hope to see with the SNHU Travel project. The product owner was also the one responsible for the product backlog and organizing the work in order of most importance to least. The scrum master was vital in delegating the work to be done to the team and creating an environment in which everyone could flourish. By setting up and running daily standup meetings the scrum master can get everyone on the same page and discuss progress and concerns that people may have. The development team are the backbone of the development process, they go in and create the foundation by following the guidelines set by the product owner and scrum master. They can use their creative integrity to create a collaborative and exciting user experience while maintaining the industry’s standards in operations.

The scrum-agile approach helped each of the user stories come to completion using test cases. As a developer/tester, our job was to read and document these user stories and create a test cases document which listed these user stories in order of importance determined by the team as well as the expected inputs and results. This allowed a comprehensive plan to complete these user stories in a way that would best represent the clients, the consumers, and the developers.

Sometimes during the development of a project new ideas will emerge or maybe things aren’t going as planned, so that means a change will be needed to keep development moving. In agile this is expected, the product owner will need to re-prioritize the product backlog to represent these changes and get the team on track with these adjusted priorities. The product owner will typically announce the adjusted features in this case a slide show was created to showcase the new requirements to the development team.

Communication is a vital part of the development process since it gets everyone onto the same page on what the expectations are and how the project is progressing. For example, as the developer, it is important to communicate with the product owner when changes need to be made so you are 100% sure you are meeting the new expectations. I wrote a letter to the product owner asking for clarification of the requirements in writing so I have something to refer to when needed, it also creates a dialog where we can discuss any impediments and work around for those impediments. As a tester I needed to communicate with the product owner to get a better understanding of the clients and end-user’s “stories” so I could get their specific metrics that would help me clearly define my test cases. Asking for clarification on these user stories shows that I care about the needs of the user and create a discussion in which the user and I can brainstorm off their ideas and create more collaborative ideas.

Organization tools can help a team transition to an agile SDLC. One of the tools the team used to help ease this transition was information radiators. Things like a whiteboard that is accessible to the whole team are vital in getting everyone connected with the happenings of the project. For some cases in person information radiators won’t work if all the team members aren’t present, which can be the case often with people doing remote work. To alleviate this, there are online options for these information radiators, things like Slack, or even a shared excel sheet can act as an information radiator to get the team on track. JIRA is a helpful tool that can help developers track their project and allows for bug tracking and is perfect for agile project management.

The agile approach in my opinion was the best approach for a project like the SNHU Travel Application. I say this because during the development, these were changed multiple times as the end users and clients changed their vision of what they wanted the app to be. This would have caused a major hiccup if we were using a waterfall development method, because we would already have the project preplanned and budgeted it would need a lot more refactoring and changes to get it working in a waterfall SDLC. This would lead to a lot of wasted time and money, especially if these changes were made later in development since things are usually done one at a time in waterfall, and testing is done at the end so any changes would have an unknown impact on the application until it reached the testing phase. In agile, when changes are made, they can be implemented immediately since testing is done concurrently and things are integrated in conjunction. One of the cons I can think of using the agile approach is that the deadline is less set-in stone and with the addition of each change it can potentially push the result further and further out which can accrue a higher cost of development.

REFERENCES

Charles G. Cobb. *The Project Manager’s Guide to Mastering Agile : Principles and Practices for an Adaptive Approach*. Wiley; 2015. Accessed August 13, 2023. <https://search-ebscohost-com.ezproxy.snhu.edu/login.aspx?direct=true&db=nlebk&AN=937009&site=ehost-live>