While we were going through the SNHU travel project I had the ability to do multiple roles inside of the Scrum Team. One of the things that this has taught me when looking at agile development is that it works best when you understand how each role works together. With this learning throughout the course, I would recommend the company goes and train each employee on the agile team and I will go through into how each part had impacted our project starting with the Scrum Master.

One of the things that I want to start of with was the Scrum Master. The Scrum master helped make sure that the team was able to collaborate. The Scrum Master is responsible with planning the sprints, backlog grooming, daily standups and the sprint retrospective meetings which were all important parts in the development. These ceremonies and meetings allow for the team to build a sense of communication with each member being able to be involved.

When we did our discussions, we had a scrum master who main role was leading the group. They were helping format any daily standup meetings we would have. When we watched the videos for those meetings as well in the other parts of the project and I acted as one I noticed how important this role is for making sure communication is throughout the team. Communication seemed to be one of the more vital parts of this and the daily stand ups were essential in facilitating this. Without the meetings and everyone understanding our progress as a group I don’t see how the agile method would have succeed. The meetings allowed for our members to communicate where they were and with what issues we may have been encountering. Beyond just meetings the scrum master was important for making sure that the teams had clear goals for each member as well as making sure there was a way to hold people accountable for there work, but also praise and congratulate members and the team for the progress we are making on something.

The next roe I wanted to look at was the product owner. The product owners focus was for helping work with the development team and providing leadership and support. They would be responsible for proving the route we were taking and helping provide roles to members of the team. With this role it showed that the product owner needs to understand the team that they are working with. Assigning roles to the team without understanding the strengths of the members could lead to an inefficient team with problems because of the lack of experience in that role. However, with agile development the product owner also needs to mitigate the creation of Subject matter experts. We want to make sure that the knowledge and experience one person is gaining is spread throughout the team. The product owner is going to also be a liaison between the client and the development team as well. Understanding what the client wants as well as how best to distribute that between the teams and developers will be a major responsibility for them which is why they need to have that understanding of the developers’ strengths and weaknesses as well as help continue to allow them to grow. This job like all of them in the agile team are going to have to make sure the vision for the project is aligned with the feedback and balancing it properly. We don’t want to over engineer a solution from feedback or under engineer it. We must make sure on each level we are making sure to have an alignment of our goals.

On the topic of feedback when working on the project we would be given user stories that would help us delegate and prioritize tasks for our development teams. For the product owner they would have to be able to understand that feedback and decide what is best for us to prioritize based on previous meetings and our goals. When I was given the user stories, I would notice at times that there wasn’t enough information and would wish that there was more so we can properly create our plan. The product manager would need to make sure that they are getting actionable feedback to help further the project. As well as feedback we will need to create a backlog from the user stories as the project manager. When doing the SNHU project I noticed that by having the visual representation of this backlog and seeing how the prioritization would plan out it helped me better communicate these goals to our team. These would be some of the functions that the project manager would have to do.

Next, I would like to look at the tester. When I was the tester for our project, I noticed that as a tester it isn’t just testing it. We are also providing feedback and need to make sure we have an open line of communication to help provide that feedback to improve the project. In more standard waterfall developments, the tester would just look for bugs and report it. If the team were to move to an agile style of development, they need to understand that for all the roles we can’t look at them as 1 to 1 translation of what was done previously, and the tester was one of the best ways to show that. Besides me providing the feedback to the product owner or the rest of the development team the tester will also sometimes fix the failures if they understand it. Some failures I noticed in the snhu project could be fixed quite easily and I could just communicate what was changed. The tester not only would have to do those roles but also, we would help create test cases. We could receive user stories from the product owner and notice that we may need more details that they didn’t. We would contact them for that detail to help create proper test cases that align with our project.

The last role I want to go over is the developer. The developer is another role that changes a lot from the standard in a waterfall development team. When we are working in an agile development space the developer will also interact with the user test cases from the tester and help provide feedback to the product owners. The product owners would also continue to communicate with them to make sure that whatever they are creating meets the team goals. This helps to show why communication is important with the developer role.

When we were working on our SNHU travel project as the developer we received feedback from user stories that made the project transition into a different focus. With this it showed me how there should be a focus on modularity in our programs as developers. We may receive feedback that has us target a different use case or different target demographic and applying modularity in our design process for code will help us iterate faster and provide new samples back and forth for us to get more and more feedback on. Other than modularity when working on that project it showed me how important communication with the product owner and others were for the developer. When I received the feedback from them on changing how travel sites were delivered, I saw that I had a lot of questions that needed to be answered for me to create something that solved that user case. I was able to create a sample of it, but also asked for that information and had that communication so I can continue to improve the project.

While looking at the software development life cycle us having this agile approach from scrum helped a lot with moving user stories to completion. For example, when we looked at the test cases, we were able to break down the importance of each case and then we could further assign it out to other teams based on that priority, It would allow us to continue to get the important features completed first with quality of life (qol) features done as the project progressed. Another example of where this agile nature allowed us to help with moving on and completing user stories is when we received the feedback on the change in focus to relaxing locations. Because of the ability of us to have working code already at this stage of the project we can go in and make the changes without having to start over or wait until the stage was complete. If we were to combine this with a focus on modularity in our programs, we would be able to implement feedback more effectively from user stories into our program in a faster time frame.

When we were going through changes for the focus of the snhu travel project us having the agile development helped a lot. As I went over earlier, we were able to already have working code so we knew what was working. With that understanding we knew that if any problems came forth with the changes it would be in a new aspect which would help speed up debugging a lot. Besides just having a working project that we can easily iterate on with that feedback the agile method allowed us to take that feedback and start working on it immediately. In a waterfall environment we would have had to wait until we completed our current parts before proceeding to change our target. Because of this the waterfall method would have had a lot of limitations of us building out a project just to get it working and then changing a lot of that work instead of being able to pivot almost immediately based on feedback.

TO: Project Owner and Tester

Subject: Project change

Dear Project owner and Tester,

I received the feedback in wanting to change our target delivery of vacation spots to wellness and detox locations as discussed earlier. The user stories on this feedback will help me understand what type of details they are wanting to be shown with them. Also, we will need some new test cases to make sure the product meets the requirements. I have completed a sample based on the initial information given but will need some specific things addressed to further improve this.

1. Will we be applying previous user experience and preferences when delivering the top 5 destinations?
2. Is there a definitive type of destination that fits the detox and wellness list that is needed to fit this or is it more general?
3. Will we only be including those type of destinations or just prioritizing them in the list
4. Will we include an filter to remove those for people looking for something else, but just steering them to results that include it first to not detract from the larger base that wants these showing up first.

Thank you for your time in this!

This is an example of our communication with our team for the snhu travel project. I would say that this was effective because it allowed me to gather more information about what changes were being made from the new feedback in the user stories. It also had a copy of the current changes that I made based on my best understanding so that the product owner and the rest of the team can comment on it and help provide further direction. I also listed questions that I felt would help me understand the direction and those questions with my submitted file showing where I was would help them provide any guidance ensuring we tackle what the users would be expecting from our application.

When looking at the organizational tools that we had for this some of the tools that really helped me be successful in this was the user stories and weighing them. Understanding the user stories and being able to create the test cases. Understanding the aspects of the test cases like priority and any pre requisites allowed me to best prioritize what I should be completing. Being able to have that and also speak with the rest of the team to ensure that they felt the weighting for the user stories were correct would allow us to deliver key features earlier in the project based on that document and then continue to iterate in the future if possible, with other features from the user stories that were lower on the list in terms of importance.

When looking at the SNHU travel project there were some pros and cons of us using the Scrum-agile approach. For the pros it allowed us to have a product that was working, and we could continue to modify based on user stories and any changes we had in targeting. The other pros was the ability to understand what the user wanted from feedback and weigh the importance as well as the size of that feedback so we could work into implementing that into future versions of the software. For the cons though what I have noticed with the agile project is the reliance on other team members. It showed that if the team didn’t have good leadership and a great source of communication across the teams the project would fail. However, with that major caveat I would still say that the agile method for this was the best one. It allowed me to help understand feedback and implement it fast as well as get more feedback to better change what we were doing. Getting that feedback in a waterfall cycle wouldn’t be in the progress of the project but at the end when we had a deliverable. Being able to have working code throughout it to get that feedback was one of the major benefits of the agile approach.