**FINAL RETROSPECTIVE**

**CS-250**

During this entire course I’ve learned a lot about the rules of a Scrum-agile Team. The first or second week of the course I took on the role of a product owner, the third week scrum master. The fourth week I took of the role of a tester, and perhaps one of my personal favorites: fifth week the role of a developer. Each role essentially taught me a set of skills that made me realize a lot of what each role does. When I get into a work setting, I will be able to understand how everything works. Playing the role as scrum master I learned that I would be responsible for holding meetings such as daily spring meetings, etc. The product owner role taught me how to be organized, and how mainly communication is key through out working in a team. Working as a tester, allows for creating the basis for the develop. It is like the guideline for the developer to follow. The developer role taught me to respond to change well, as well as being responsive in changes.

A Scrum-agile approach to SDLC helped each of the user stories to completion through a various set of five examples in this course that consist of Sprint, Daily Planning, Sprint Review, Sprint Retrospective, and most importantly the work. Every day the team undergoes a lot of sprint meetings to get a general overview of what should be required, and what needed to be changed, etc. This important to a team because it giving an overview of how the project’s progress is doing. The Product Owner can therefore determine whether to increase productivity in order to meet the deadline.

In week four or five the product owner decided to make a few tweaks to the project. While doing so, she was quick to inform her team about these changes to the Tester, and the Tester therefore has to relay this information promptly to the developer. During this changing process the communication was demonstrated through email informing and assuring that the correct changes were made. Sometimes these changes can be a little daunting especially if it is a spur of a moment type of situation. But in these moments, developers cannot take the same amount of time they did for the initial plans. Often times developers only have a short time frame to make these changes.

During both week four and five, it was important to communicate any concerns to the source. I had to contact the Product Owner and Tester to relay the information to me about the changes so I could add what was needed for the development. It’s always important to contact the correct person for this information. Asking someone that isn’t as well informed about the information wouldn’t be the best, and would create a delay for finishing the project on time. For example, contacting the Scrum Master wouldn’t be the best idea, because they would probably be on the same knowledge base as the developer. When writing emails, it’s also important to keep straight to the point and not beat around the bush. Time is of the essence, and it should be wasted on things that aren’t really important. I had to essentially prioritized my questions, and then make the changes as quickly as possible.

When working on the SNHU Travel project, the team implemented a lot of principles from Scrum- agile. Concerns and progress were concluded with daily sprint meetings. Testers utilized creating product backlogs, and relayed information about user inquiry. The product owner handled the what needed to be done.

Having a Scrum-agile approach for the SNHU Travel project has a lot of pros and a few cons. Listing the pros first—having a scrum-agile approach allows for the team to be more flexible about the changes. In week four or five, when the Product owner decided to make changes, using the waterfall approach in the first place would have created problems because it is so ridge and doesn’t allow for much change in plans in situations like these. Another advantage, is that working in a Scrum-Agile approach allows for team members to plan out any potential risks that may result in delaying the project or the success rate of the project. A few cons to working in a Scrum-agile approach would rely on the level of the commitment from the team. Another con would be that teams can get side-tracked fairly easily.

In this situation, working in a Scrum-agile approach would be the best approach for the SNHU Travel development project. Given the previous situation, where the Product owner had to make changes to the project so that it would achieve its best potential this would be a valid example of why it would be best for the team to use a Scrum-agile approach. Using a waterfall methodology wouldn’t allow for a change such as this. Also <Give more reasons>