# Project Three – Sprint Review & Retrospective

As part of the organizational shift from a traditional waterfall development model to a new Agile and Scrum framework at Chada Tech, our team was chosen to test this new way of working, before moving the entire organization onto this new model. This would require each of the members on my team to take on various agile "roles", to fully experience the new methodology, as we take on the task of creating a web application for the “SNHU Travel” travel agency. As a result of using the Agile/Scrum framework, taking advantage of the various elements that make up the methodology, and the hard work from the rest of the team, ultimately, we were able to create a solution for the SNHU Travel team, in a quick and consistent way, even in the face of changing priorities!

There are a few roles that members of a team can assume as part of the entire Agile team: Product Owner, Scrum Master, Testers and Developers. First, for the Product Owner, this is an essential role as part of an Agile team, and one that is generally leading the "direction" of the team or project. This person would be responsible for reaching out directly to the stakeholders of a particular project, and to keep some sort of relationship with them, so that we can keep a constant form of communication between the project team and the stakeholders of the project. For the SNHU Travel project, it was a single project that was easy enough to keep on top of. However, in a real-world situation, a team member would likely be working on multiple things, so it's very important to have a strong Product Owner that can not only reach out to these stakeholders and allow the testers/developers/etc. focus on their work, but also to have someone that can advocate for the rest of our team, for this project and future ones. The Product Owner will also be responsible for creating and maintaining the Product Backlog, or list of items or features that still need to be implemented or finished to consider the project complete. For the SNHU Travel project, the Product Owner helped lead to our success by meeting and working directly with the SNHU Travel team on a regular basis, and taking feedback from them as soon as possible, as well as creating a product backlog for the developers and testers to have a vision of what to work on next.

The Scrum Master is somewhat in between the Product Owner, and the testers and developers. They are considered a "servant-leader", which is not necessarily a leadership position, but rather one that helps to facilitate the other roles within the Agile team, to keep things running as smoothly as possible. One part will be supporting the Product Owner, helping them to make strong decisions about the future of the project, since they are so much closer to the work than the Product Owner would be, as well as helping to maintain the Product Backlog to make sure the appropriate things are being worked on, and in the most appropriate order. On the flip side, they are also there to help support the testers and developers, as well. Since the testers and developers will be mostly focused on implementing features and fixing bugs, the Scrum Master helps to keep other elements of the job on task, such as specifically which items to work on next, and to help them in any other way that they might be blocked, so that any stoppages of work could be avoided.

The tester is another vitally important role to a Scrum team. While developers will be trying to implement these features into whatever system or application is being built, it is up to the testers to come up with a suite of tests that these features can be tested against. Developers tend to just get things working and may overlook certain aspects that a tester would think of first-and-foremost. By having this role available to the team, we can have several people considering how things could go wrong, as well as specifically how things should be working. Once these tests are created, then the testers can either work directly with the developers themselves to ensure things are working properly, or the developers can utilize these test cases and make sure that their code is still upholding whatever the test case may be. Without this role, an application would still be made, but the level of confidence that the application will be working as intended and without issue would be much lower.

Finally, the last role is the Developer role, and this is the member(s) of the team that would be creating the application and working the various items on the Product Backlog. Development is already tough, so having to manage working with stakeholders, setting up various meetings and trying to resolve any blockers that might be impeding you from finishing any development work would make it even more difficult. Thanks to the format of the Agile team, the Scrum Master helps facilitate most of these tasks so that the developers can focus on the difficult task of just creating the application! Apart from doing the actual implementation, the other ask of the development team is to be as transparent as possible when it comes to the work being done. Being able to express any of these issues to the Scrum Master, or to the team as part of a Daily Scrum meeting, will lead to success as we'll be able to tackle these issues as soon as they arise.

The SDLC, and the Scrum/Agile approach, was hugely beneficial to the success of the SNHU Travel project, and the completion of the user stories for the project. First, simply having the entire backlog which makes up the entirety of the project in a single system that all members of the team could access and edit at any time was a huge first step. It allows the development team to have no confusion about what should be worked on next, it gives the entire team a quick glance at how the project overall is going, it allows various members of the team to edit the different stories, if any changes were to occur, and it allows the Product Owner to have confidence is speaking about the progress of the project. Lastly, by only selecting certain stories to work for a particular block of time (or a sprint), the team was able to focus on those tasks specifically.

For the SNHU Travel project, about midway through development, the stakeholders requested that a fairly large change be made, as compared to the direction that was originally given. Since their market research indicated that "Wellness/Detox Vacation Destinations" were going to be hot in the upcoming vacation season, they wanted to have that advantage as part of the site we were creating for them. In a more traditional waterfall approach, this kind of change would have been nearly impossible for several reasons. For one, in the traditional way, there would've been very few meetings between our Product Owner and the rest of our team, were any of these changes to arise, so we would've never been made aware that they wanted to make this kind of change. Not only that, but if the development of the product had also followed a traditional approach, it would've been much more difficult to make that kind of a change. However, since we decided to take a small number of user stories which we would implement during the current sprint, it was easy for us as the development team to adjust just one of those stories, to make sure we showcased those "Wellness/Detox" vacations in that same sprint.

Communication is key in an Agile team, and we were able to utilize that open communication within our team when it came to some clarifying questions. Once the SNHU Travel team made their decision for different vacation destinations, while the change seemed straight-forward, it would've given the development team much more confidence to get some clarification from our Product Owner. At the same time, some of the test cases that were originally designed would need to be adjusted based on the team's request. As a result, I sent an email to both the Product Owner and Testers, to let them know which questions I had regarding the change, as well as just keeping the testers in the loop about what I was thinking. In that same email, I also suggested that the development team could work directly with the testers in this case, if need be, to make sure things continue smoothly for the rest of the sprint.

The way that project data is organized, as well as the various Scrum events that make up the Scrum/Agile methodology, certainly helped the team create a solid product for the SNHU Travel team. For one, having all our user stories in a single place known as the Product Backlog, where any members of the team could view and edit them, was a huge success, as just having visibility into what still needs to be done and being on the same page as all other members of the team can be extremely beneficial. The Scrum "Sprint" is another concept which should be highlighted. When working on a new application, it can be overwhelming to consider all the things that still need to be implemented to have a working solution, especially since those requirements can be an ever-growing list for a long time. By selecting a set number of user stories to implement in a 2-week window, it allows the development team to focus on just those items, giving each one as much attention as it needs, rather than trying to implement a ton of things, which may result in a worse product in the pursuit of trying to get more done. Lastly, the Daily Scrum meeting also would've led to our success on this project. Various blockers may stop us from being able to continue development, and in a traditional project structure, it might be a while before those issues are brought to who needs to hear it to start making changes. By utilizing a daily meeting where all members of the Agile team attend, and give a quick breakdown of what they've done, what they will do, and where they may be stuck, any issues that arise can be resolved as soon as possible, which will lead to more effective development time, and finally, a better product!

As for a comparison, I don't think there were many downsides to the Scrum/Agile approach. Even in a traditional project approach, meetings will still be had, and development and testing will still occur. The only difference is that in the traditional approach, it is assumed that all the development and testing can be planned out completely ahead of time, which increasingly is becoming less and less common. In the Scrum/Agile approach, all these things still occur, but with much more frequency, and with the expectation that changes will happen throughout the project, and we'll need to be able to act when they do. Specifically, for the SNHU Travel team, they were an existing company trying to expand their capabilities, and as a result, changes are constantly happening to them as well, and because of the Agile approach we used, we were able to adjust to those changes when they reached out to us for new vacation destinations to showcase in their site. In the traditional approach, this kind of change would've been nearly impossible, and for those reasons, I think that the Scrum/Agile approach was the best one to take for this project.