The scrum-agile methodology for development of code or many other products can be an impressive way of handling the developmental process while allowing for constant changes or reviews throughout the process. Each role in the Scrum-agile team plays a vital role in making sure that the customer gets exactly what they wanted with as little confusion as possible.

The Scrum Master is the support coach for the team. They tend to have the most knowledge on the agile methodology in the team dynamic. They facilitate the different scrum events that take place throughout the sprint iterations, such as the daily stand-in where the team goes over work from the previous day and any potential issues or questions moving into the current day. The team also has a weekly or bi-weekly meeting with all team members and possibly the customer to go over progress with the project. In the beginning of a project the Scrum Master helps the team to create a team charter that establishes the rules and practices that the team will adhere to throughout the development process. In our module two assignment I created a basic team charter for the SNHU Travel project team. This charter along with the support of the Scrum Master as needed helps to keep the team working efficiently.

The Product Owner works as the team’s driver for value and efficiency. They create and maintain the product backlog which contains a sorted listing of all work the team needs to accomplish. This backlog consists of user stories which detail the individual tasks that need to be accomplished or implemented for the product to be completed. The user stories that are within the product backlog are also created by the product owner. In our module three assignment I was put in charge of creating three user stories from a product backlog I created while watching a meeting. I needed to make sure to lay out every detail about the issue and what needed to be done for it. Testers and developers can create user stories that they can then bring up in one of the daily scrums for the team to go over and potentially add to the backlog. The Product Owner interacts the most directly with the customer and other governing bodies that would influence the product as it moves through the development cycle. The Product Owner must always stay on top of the product backlog to make sure it is as clear as possible for the rest of the team to read and understand so they can determine what work they will be doing during the sprint. The Product Owner and the product backlog are both very helpful for maintaining team efficiency.

The Tester is primarily in charge of testing and creating tests that ensure the product being developed meets the requirements set by the product owner and customer. The tester works closely with the product owner and goes over each user story to work out the exact requirements and any details that need to be accounted for when creating tests. With this information the tester creates test cases like in our assignment for module four. I created test cases detailing the individual requirements of each user story that needed to be met for the user story to be complete. I then went back and revised the test cases to handle the new presentation style presented to us. These test cases help to guide the developers and let them know when they’ve completed the work needed for the user story.

Finally, the Developers are of course in charge of developing the actual product. During the beginning of every sprint the developers and testers go over everything in the product backlog and select the work they believe they can complete in the allotted time. The Developers themselves play the most critical role in developing, updating, or fixing items in the product backlog to move the project forward. In our assignment for module five I was tasked with modifying a slideshow showing some top destinations. It’s important to note that I was only tasked with changing the information being presented, nothing about the slideshow was to be changed. This precision patching and editing is key for the user stories to be completed inside the sprint time period efficiently. Once these small changes are made, they can be presented in the next meeting to be reviewed for more potential changes or confirmation that it’s good.

During the development of the different user stories in our SNHU travel project I found the agile methodology to be perfect for this style of program development. The way the agile methodology is structured the program is constantly going through testing and review to determine what needs to be updated or changed. The user stories are typically given enough detail to cover what the user or the product owner expects from the completion of that user story. The original presentation of top destinations given to us to modify was a scroll list consisting of five arbitrary destinations. After the modification and review it was decided that it needed to be a slide show presentation compared to a scrolling list so that it focused on the individual locations being presented rather than the ability to view all of them at once. Since we got this information during one of our “scrum meetings” we were able to make the changes before what would be the next product review point. The inclusion of testing while the user story is being developed helps to ensure the program is working properly before marking the user story as complete and moving forward with the next one.

When it came time to communicate the most during our SNHU travel project it took the team a couple days to really get it kicked off and moving. Once we were able to establish what role everyone was going to fill, we each went over what key points from the agile process we wanted to implement into our business. Once our product owner took note of everyone's desired implementations, he summarized everything into a list so we could all vote one what we found to be the most important. I included in my vote, “Using short term sprints would give us smaller time tables that we could go back over in a retrospective. Daily meetings for the team members can keep problems minimal by providing support as soon as possible.”, to try and bring the tester and developer into the vote more directly rather than focusing on my part of the team only. Not long afterwards the other team members came in with their votes and we were able to determine the top choices.

The main principles that made our team successful during the entirety of this project is the concept of short sprints with small task testing and developing. We can look at our class as 8 sprints where we get feedback and review at the end of each modular sprint. During each sprint we were given only a couple of relatively small tasks that we could do in that week time frame. The discussions we did each week could be equated to a weekly scrum meeting where we would all go over a new or relevant concept that would further our work in that area. I’m unsure entirely how effective some of these were in getting everyone to thoroughly communicate with each other, but I personally found it nice to hear what everyone else thought on some of these ideas.

The Scrum-Agile approach was an overall effective way to go about the SNHU Travel project. It allowed everyone working on it to have small tasks that they could focus on for the week and get feedback at the end of the week to see how they did and what still needed to be fixed. The discussions allowed the students to talk to each other and ask questions, and the journals at the end of each week allowed for a continuous personal progress report. On the downside however we have the need to communicate thoroughly with your team. We had the tools necessary but many people weren’t very inclined to use them to their full extent. The precision editing could also be a hard part for some people. During one of our assignments the program sent to us for editing had a color scheme that made the text almost unreadable. The point of the assignment however was to only change the information on it and not the color scheme or anything else. This proved frustrating when it came time to change the program, since we have to consider if the customer truly wanted it that way. Overall agile still seemed like the way to go when working on this project. The waterfall process wouldn’t have allowed for these quick reviews and changes.