In this project, each member of the team had a significant role to play in the project. The Scrum Master, Product Owner, Developer, and Tester all made fantastic and unique contributions. Because of the contributions from each role in this Scrum Team, the project was a success.

The Product Owner took the time to meet with a client and understand their needs. By asking the client clarifying questions, the product owner was able to uncover the features that would need to be developed. From there, they created a backlog and were able to prioritize items on the backlog for development. The product owner also met with a focus group to gather feedback on their needs to understand the user base for a client and development of the initial booking system. When the user requirements changed to focus on wellness vacations, the product owner did an excellent job in readjusting the product backlog to deliver the product on time.

The Scrum Master organized the team and ensured that they worked in an agile manner. The Scrum Master led the creation of the agile team charter at the beginning of the project. This charter outlined each member's role and clearly specified the vision and mission of the team. The Scrum Master also scheduled and lead events such as Sprint planning, daily scrum, sprint review, and sprint retrospective. Each of these events was important to maintain open and transparent conversation amongst the members of the team as well as communicate status updates on tasks. When there was a blocker, the Scrum Master was able to step in and help ensure that each member had a clear path to complete their work.

The developer for the team was responsible for the execution of developing the product. The developer took items from the product backlog and, using the requirements specified for each ticket item, understood the needs to build the product. The developer was able to translate user requirements into technical details to develop features such as the booking system, the gallery, images, and information for each location. When the user requirements changed from having a gallery view to a carousel view, the developer was able to rewrite that part of the application to reflect the updated specifications.

The tester played a particularly important role in this project beyond just testing. The tester was responsible for the creation of initial test cases. These test cases were taken from the acceptance criteria that the product owner provided to ensure that each of these requirements were well tested to ensure that the product not only functioned correctly, but that the user’s needs were met. They also use tools such as wire frames to understand the visual needs of the product. The tester also had to revise test cases as specifications changed for the project. This was important because we wanted to ensure that all the test cases were in parity with the current needs of the user.

The Scrum-agile approach to the software development lifecycle helped each of these stories come to completion. The product owner created each of the user stories from conversations that they had with the client and focus groups as well as market research that they had conducted. These requirements were clearly outlined so that the developer was able to take these requirements and translate them into technical details to create the product. The tester also took these user stories and crafted test cases that reflected the acceptance criteria from each of the user stories. Because we were developing in an adult manner, the tester was able to test each of these stories as a developer was creating them. The product owner was also able to update these user stories in which the team then could shift and adjust to deliver each leg of the project. This is contrary to a waterfall approach where the entire project would have been a single deliverable instead of creating vertical slices and iterating on the project as the team learned more along the way.

As the scrum master, I needed to utilize different forms of communication. Some of which were traditional forms of communication such as email when I needed to email as a tester regarding updated specs. I also communicated initially with the team using the team charter which outlined our vision and goals for the project. Had it been a real agile setting with a full team, we would have utilized daily stand ups for in person communication as well as sprint reviews in sprint retrospective. These would've allowed the team to communicate together, give each other feedback, share thoughts, as well as give status updates regarding the sprint. An important tool that was also used were user stories which outlined the needs of the user for features of the product. These are essential for development as they gave the developer the ability to understand the user’s needs to translate into technical requirements for feature development.

The Scrum-agile approach was critical throughout the entire project to ensure that the team delivered a high-quality product on time. There were several instances along the way that the initial requirements for the project had changed. One example of this was when the layout for the gallery changed to a carousel view instead of the traditional gallery view. After the product owner gathered market data, they also learned that the most popular vacation market was for wellness trips. Because of this the requirements for the project changed and parts of the application had to be rewritten to support health and wellness vacations. All these changes were valuable adjustments to the project because they were the exact needs of the client, and they were updated to reflect the current market which helps support the success of the product for our client so they could have an edge on their competition. Had we taken a waterfall approach, the entire project would have been developed from the initial specifications and these features would not have been able to have been created.

In summary, the Scum-Agile approach was the best approach for the SNHU travel development project. This approach allowed for the team to iterate as requirements became more defined or as they changed so that they could shift their workflow to accommodate the users' needs while still meeting the deadline. This approach also allowed each role to focus on their individual tasks and work in an unblocked manner while still creating a collaborative environment. In a smaller project like this a waterfall approach may still have worked but there would have needed to be additional planning done at the forefront of the project. That may have saved Time from having the team needing to adjust those requirements changed within the scrum approach. However, the waterfall approach would not have been able to shift to meet the customers' changed needs during development.