**The Scrum Master kept the team flowing smoothly. He helped organize the daily scrums that kept everyone in alignment and allowed any setbacks to be brought forward. He helped in creating the team charter. The charter outlined the goal, the team members and their roles, and general rulings agreed upon by the entire team. Not only did it give the goal clarity it helped keep the team engaged and invested by evolving as the team made changes and completed tasks.**

**The product owner’s most important task was bringing value to the project. To do this the product owner kept in contact with end-users and stakeholders. The project Owner derived user stories from the client's requests and prioritized them, creating a clearly defined product. Developers and the Product owner utilized users’ stories to fulfill the needs of the client. Throughout the process, the user stories were cleared from or added to the backlog. If needed the Product Owner gathered more information and revised the user stories. The Product Owner maintained the backlog throughout the process by deciding what tasks will solve the problem needing to be solved.**

**The Developers worked on the "How?" to solve a problem, where other team members, like the Product Owner work on the "What?" and "Why?".**

**The developers turned the user stories into the real-life features that make up the product. As new information came in they were flexible and communicated clients' needs to complete the project successfully. They delivered working programs that could be tested and reviewed and brought value through continuous revision. Working with the Testers, the Developers frequently created working code that could receive feedback from the clients.**

**The testers helped determine the passing and failing test cases for the user stories and products. They were able to establish what was acceptable and marked as complete. When we were unsure of a client's needs, the Testers reached out and incorporated new information to create a more refined solution. working in tandem with the Developers, the Testers provided fast and reliable results while also relieving the burden of Quality Assurance from the Development team.**

**Using a Scrum Agile approach user stories were used to communicate user needs. Creating user stories gives persona to the problem being solved. Instead of being just an application, it is a solution to a users' specific need and builds value by being characterized by its function. Requesting additional information provided clarity and customer satisfaction. By breaking up Epics into smaller stories, the immensity of a task could be completed in parts or slimmed down. We were able to tackle each task to completion and have little wasted effort due to the prioritization of the backlog. Forward progression lent to a snowball effect as each sprint was completed.**

**An agile approach allowed us to resolve setbacks promptly or catch them preemptively. Frequent communication allowed any issues to be addressed promptly. Between the daily scrum and cooperative development, unexpected changes were handled and dealt with within the sprint timeline. Agile methodology thrives even when unexpected changes arise due to small tasks and quick development cycles. After creating an initial slide show, we had to update the travel slide show to display the updated preferences given by the product manager. Using preexisting code, we were able to update test cases, then the development team adjusted the .jar file to fit the new features of the product.**

**Communication is a priority in agile methodology. Throughout the sprint, we had to request information to have a more accurate interpretation of the user stories. In place of direct communication, we reached out electronically. We addressed a specific person, the required information, and gave a sense of urgency while remaining professional.**

***Hello Mary,***

***I’m reaching out today because I’m needing some assistance with the user interface design for the SNHU travel app. If we could get details of your request about not seeing places you weren’t interested in. We would like to know if you would like to select the areas you are interested in or to choose areas you are not interested in, also would you like this to be a setting in your profile or whenever you are generally searching within the app. Please respond at your earliest convenience, if needed we can also meet in person or virtually.***

***Sincerely,***

***Thor (Ruler of Asgard)***

***#1 Tester***

***Southern New Hampshire University***

**In this E-mail Thor, one of our testers needed more information about a specific feature a user wanted. Although direct communication is preferred, there are several electronic means of communication to keep the SDLC running even during a pandemic. We had success using Jira an electronic project management tool and an E-Whiteboard. Tasks, setbacks, user stories were all available to the team whenever needed. Having these electronic tools allowed us to stay connected during bad weather or a pandemic.**

**Agile principles make the software development life cycle flexible and driven by what matters most. I have spoken about the collaborative benefits of scrum earlier which is the strongest benefit of agile in my opinion, other beneficial principles include time boxing and iterative development. timeboxing and iterative development are components of agile that focus on breaking the project into manageable pieces. Time boxes set specific time aside for each task and encourage immediate action and results. Quality assurance testing, user testimonials, Scrum, or sprint meetings can all be scheduled within time boxes so the team has a clear idea of how much time they will have to complete their other tasks. Iterative development works in tandem with time boxes by separating large projects into pieces to be developed, reviewed, and assessed before moving on to another task. Iterative development allows for changes to occur without derailing an entire project. Both principles build value by focusing on what will provide the most benefit and decreases wasted effort by limiting wasted time and development.**

**The SNHU Travel development project could have been completed through a waterfall approach, but it would have taken more effort to adjust to the changes that were brought up midway through. The project also would not have had the added user-chosen features like the travel preferences. In addition to lacking user-chosen features, other developed features would have to be thrown out wasting hours of development time.**

**whereas with the Agile approach, the changes were able to be configured within a normal sprint. Numerous features were added midway as a result of communicating with the users and team members. As a result of utilizing agile methodology, only a small amount of development and testing were wasted as a result of feedback or changes.**