### **Retrospective**

In my Scrum-Agile project, each week I took on a different Agile role other than Scrum Master that helped make the project successful and learn about the different roles a little more. I took on the role of the Product Owner. My job was to understand what the users needed and communicate those needs to the development team. I made sure we prioritized the right features so that the final product would meet the users' expectations. For instance, I would talk to users and gather their feedback to ensure the software we were building would be useful for them. Another important role was the developer. Developers were responsible for writing the code and creating the software. They needed clear and detailed instructions, which I provided by writing user stories. These stories explained what features were needed and why they were important. Developers would then use these stories to build the software. Developers needed clarification on how quickly search results should load. I provided that information, which helped create a better product. The tester’s role was to make sure the software worked as expected. This role used the user stories to create test cases and ensure everything was functioning correctly. They would also communicate with me if they had any questions or needed more details. Using a Scrum-Agile approach helped us complete user stories effectively. User stories are short descriptions of a feature or requirement from the user's perspective. They helped the development team understand what needed to be built. By breaking down the work into smaller tasks, we could focus on one feature at a time and ensure it was completed before moving on to the next. For example, one of our user stories was about allowing users to search for vacation destinations with the best deals. The development team used this story to create the search feature, and the testers used it to create test cases to make sure the feature worked properly. Because the story was clear and specific, it was easier for the team to complete it.

Sometimes, the direction of a project can change. The Scrum-Agile approach helped us handle these situations smoothly. Because we worked in short sprints, we could quickly adjust our plans and focus on new priorities. During our project, the client requested a change that required us to adjust our timeline. We were able to re-prioritize our tasks and still meet our deadline by focusing on the most important features. The flexibility of the Scrum-Agile approach allowed us to adapt to these changes without losing time. Communication is key in a Scrum-Agile team. We used different types of methods to stay connected and ensure everyone was on the same page. I sent emails to the developers and testers to provide them with the information they needed. I attended daily stand-up meetings where we discussed our progress and any challenges we were facing. One effective communication tool we used was the "information radiator," a big board where we could see the project's status in real time. This helped everyone know what needed to be done and what tasks were already completed. It made our communication more visual and easy to understand.

We used several organizational tools and Scrum-Agile principles that helped our team succeed. One important tool was JIRA, an online system where we could keep track of tasks and see who was responsible for each one. This helped us stay organized and ensured that everyone knew what they needed to work on. Scrum events, like sprint planning and sprint reviews, were also important in keeping our team on track. During these events, we would plan for the sprint, review our progress, and make any necessary adjustments. These meetings helped stay focused and made sure we were always moving in the right direction.

The Scrum-Agile approach had both pros and cons during our project. The main advantage throughout the entire project was flexibility. We could easily adapt to changes and focus on the most important tasks. However, one challenge we faced was that some user stories lacked detail, which made it harder for the developers to understand what was needed. Overall, the Scrum-Agile approach was the best choice for our project. It allowed us to stay organized, communicate effectively, and deliver a product that met the users' needs. If we had used a waterfall approach, which is more rigid and less flexible, it would have been harder to adapt to changes and complete the project on time since an extended deadline was not given.