**Sprint Review and Retrospective**

**Applying Roles**

It provides a framework within which the system is structured. Based on the data grouped in cells, during the SNHU Travel project, our group practiced Scrum-Agile, and each of the members assumed a role - for example, there was the Scrum Master, the Product Owner, and one or several Developers. These roles were vital in ensuring that the project was a success. There was also the role of the Scrum Master, who conducted daily meetings at which each team member presented their progress and outlined any issues faced. The Scrum Master was also responsible for eliminating all potential blockers and sending the team in the right direction. For instance, when there was a hold-up due to the absence of comments on critical features, it was the responsibility of the Scrum Master to contact the relevant Product Owner to obtain the information necessary for the project's progress.

In my case, being the Product Owner meant that I was obliged to outline the list of tasks (the backlog) in the correct order and ensure that the final result we achieved was the one the client could benefit from. Setting priorities was an ongoing task that was particularly important since it inherently had to center around the client. One of the main tasks was building a payment system that made it possible to travel the world only with the mobile application. If this had been a focus right from the start, the people in the team would have worked on developing something already known to be important to the client.

The developers took users’ stories and implemented ideas into the mobile application. They operated on sprints, and each time after completing a sprint, they would report on achievements. For instance, rather than focusing on the final version of the destination filter, the developers presented a working copy of the filter and then developed it with the Product Owner’s input on subsequent iterations. This kind of work allowed the project to move forward according to the plans and at a reasonable pace.

**Finalizing User Stories**

With the Scrum-Agile approach, we were able to handle user stories while maintaining momentum. One of the principles of Agile is to present parts of the finished project and receive feedback throughout the process. For our project, the client’s user stories were prioritized according to the client’s requirements and completed within short sprints.

An excellent example of outcomes from such interviewing is a user story requesting the possibility of filtering travel destinations by price, weather, and how many nights you are willing to stay in that one country or city. Due to its complexity, we decided to split this into smaller parts, which can be completed one at a time. The developers designed a basic structure of the filtering system in the first sprint. The second sprint included saving user preferences for future searches. By the third sprint, we had a filter tool that was functioning and ready for the targeted users to test.

Short sprints allowed us to work with the client and ensure that we were building something that they needed. In addition, they could change the feature at an early stage. All in all, this made it easier to expect that the feature would be adequately functional at the end, that is after the application was completed.

**Managing Distractions**

One of the factors I appreciate about the Scrum-Agile method is that it gives you room to administer any interruptions or changes of focus direction. As most software projects have new requirements during implementation, this was also the case for the project. Agile made coping with those changes without derailing the whole project easier.

Everything changed when the client “asked us” to change the app UI and optimize it for mobile. In the first place, we were more focused on creating the desktop version, but the client understood that more and more of his customers used gadgets. According to a traditional waterfall method, this kind of change would have resulted in much time being lost because it is predetermined that all the phases would be carried out in the specified sequence. However, we updated our current plan with Agile and pushed UI enhancements to the next sprint.

In Scram, due to the frequently occurring sprint reviews and retrospectives, we were able to bring about this modification without wasting time. After each sprint was completed, we focused on our strengths and weaknesses, thus enhancing planning for upcoming sprints. This way of working enabled us to remain focused when the project objectives changed.

**Communication**

Good communication was essential to ensuring the Z statistics synergy project was completed without disruption. In Agile, as we had daily updates, at least one of the teams provided what they had been doing, which issues arose, and linked with the rest of the team what was going on at the pain. Other tools were also available, for example, the use of Jira, which 'helped in allocating jobs and dispensing information as well'.

For example, when the client demanded an update for the mobile UI, I created a task in Jira and assigned it in collaboration with the Scrum Master. Thus, what every person was supposed to do and what changes had been made were visible to everyone. Jira organized us more effectively and ensured all team members were clear about their roles and the project status.

During the sprint review, we showed the work performed to the Product Owner and the client. This enabled us to resolve issues pending at the analysis stage and incorporate advice before proceeding to the next sprint. These developments helped ensure that all parties were informed of the latest developments and what was to be done next.

**Project Management Tools**

The projects we conducted, together with the core activities of Scrum, were part of this project's winning formula. One tool that we all loved a lot was Jira since it allowed us to manage and monitor tasks, scope different team members, and observe the project's overall status. We integrated a Kanban board to depict our workflow, thus allowing one to know what needed to be done quickly.

We adhered to the underlying tenets of Scrum and even observed the regular sprint planning and retrospective meetings to avoid congestion. During the sprint planning meetings, we reviewed the items in the backlog and selected the most desirable ones to do next. The daily stand-ups were aimed at allowing everyone present to explain what course of activities they were currently undertaking and any other problems. Each sprint was followed by a retrospective to discuss the most recent sprint: what worked and what could be improved in the following one.

Using these tools and observing the events enhanced the team's coordination and enabled us to *produce quality work within the set deadlines.*

**Evaluating Agile Process**

Overall, the Scrum-Agile method was very effective for the SNHU Travel project, although it had both positive and negative aspects. The people behind the project were probably the most flexible regarding the framework's applicability; they appeared to be very comfortable adjusting to new conditions. Early and continuous project delivery to the client for feedback was critical, as it established the certainty that whatever we were developing was something the client truly yearned for. It was simple to accommodate client requests for changes.

On the negative side, using Agile also had its downs. For one, it would sometimes take a lot of time to receive a response or have a change made. For instance, when we received input on the filtering feature, it prompted a rework that required some time to complete. In a Waterfall approach, such changes are usually made towards the end, so the entire exercise becomes easy to manage.

Considering everything, Agile was the best approach for this project. The travel agency's requirements changed greatly during development, and Agile addressed them more readily. A Waterfall approach was too rigid for the project, in which the requirements change constantly, and it would have been difficult to make amendments after the development phase commenced.