

Brianna Peoples

### Sprint Review and Retrospective

ChadaTech experimented with Agile as an alternative to the traditional Waterfall method, using the Scrum framework to guide the process. In order to evaluate its effectiveness, we introduced it during the development of an application for SNHU Travel, a travel agency aiming to strengthen its digital presence. The project followed Agile principles, breaking work into sprints and adapting to changes as needed. This retrospective reflects on the roles taken on throughout the project, the completion of user stories, how obstacles were managed, and the overall impact of Agile on development. The insights gained will help determine whether adopting Agile across all ChadaTech teams would be a beneficial shift.

Each role in the Agile team had a specific purpose in keeping the project on track. The Product Owner set the vision for the project, prioritized features, and ensured that development focused on the most important aspects first. The Scrum Master facilitated meetings, helped remove obstacles, and made sure Agile principles were followed. When delays occurred while integrating a third-party API for flight bookings, clear communication was maintained, and collaboration with developers and testers allowed for a quick resolution. The development team was responsible for building, testing, and deploying features, which made it possible to get the main functionality, including searching for travel packages, filtering results, and booking flights, completed in the early sprints. Testers played an essential role in catching early issues and providing feedback before each sprint was completed. Because each role worked toward a common goal, the application met the needs of SNHU Travel efficiently.

Agile made it easier to complete user stories without being overwhelmed by the entire project at once. Instead of developing the full application in a single phase, the work was broken down into smaller, manageable features. The first sprint focused on implementing a basic search function for travel packages. In the following sprint, filters for destination, price range, and travel dates were added. When SNHU Travel later requested a customer reviews feature, it was possible to add it to the next sprint without disrupting the overall timeline. This approach made sure that a functioning product was available at all times instead of waiting until the very end to have something usable.

One of the biggest advantages of Agile was how well it handled unexpected challenges. In traditional Waterfall development, changes or interruptions could cause major delays, but Agile made it easier to adjust without affecting the rest of the project. During development, an external API provider changed its integration requirements. This could have caused significant delays, but since the work was done in sprints, priorities were adjusted, and work continued on other tasks while waiting for updated documentation. Another unexpected challenge was a situation where a developer had to take leave unexpectedly. Rather than letting this slow things down, daily stand-ups and backlog tracking made it possible to reassign tasks and keep progress moving forward. There was also a point where SNHU Travel decided that mobile responsiveness was more important than certain desktop features that had been planned. Because Agile is designed to adapt to change, the focus was shifted in the next sprint, keeping everything on schedule.

Good communication was key to ensuring everything stayed on track. Agile promotes frequent and open communication, which was done through daily stand-up meetings to check on progress, discuss challenges, and plan the next steps. These meetings were short and focused,

helping to stay aligned without wasting time. Slack and Jira were used for ongoing communication and task management. Slack made it easy to have quick discussions and solve problems in real-time, while Jira kept track of user stories, assigned tasks, and documented progress throughout the project. At one point, a critical bug was found in the payment processing feature. A team member reported the issue in Slack, and within minutes, both developers and testers worked together to identify the problem and deploy a fix before the sprint ended. Sprint retrospectives also played an important role in improving communication. After each sprint, discussions were held about what went well, what could have been done better, and what improvements could be made moving forward. One major takeaway from these discussions was that breaking down tasks more clearly in Jira improved tracking and made workflows smoother.

Several tools helped keep the project organized and running efficiently. Jira was the main tool for tracking user stories, managing sprint progress, and handling backlog items. Confluence was used to document important project decisions and meeting notes. Version control was another critical component, and GitHub was used for managing source code. A structured branching strategy was followed to avoid conflicts and ensure smooth collaboration. Regular code reviews were conducted to maintain consistency and catch potential issues early. Sprint planning, daily stand-ups, and retrospectives all contributed to keeping the project on track and allowing for continuous improvement.

Agile had several advantages in this project. It provided the flexibility to adjust to changes without disrupting progress. The incremental approach ensured that a working product was always available instead of waiting until the end of development to deliver something functional. Continuous feedback from sprint reviews allowed for refinements to be made based on real user needs. Constant communication made it easier to stay aligned and quickly resolve

problems as they arose. However, there were also some challenges. Since this was a new way of working, there was a learning curve that slowed things down at first. The frequent meetings, while helpful, also reduced the amount of time available for actual coding. Because Agile allows for flexibility, it also made it easier for new feature requests to come in during development, which required careful backlog management to avoid distractions and delays.

Despite these challenges, Agile proved to be a great fit for this project. The ability to quickly adapt and keep progress moving forward without major setbacks was a huge advantage. The sprint-based approach meant that SNHU Travel always had a functional version of their app, and improvements could be made along the way rather than having to fix everything at the end. If ChadaTech decides to implement Agile across all development teams, it could lead to faster project completion, higher client satisfaction, and a more collaborative work environment. There would be an adjustment period, but in the long run, the benefits would outweigh the initial challenges.

This was a class project, and there was no actual team involved. Instead, different Agile roles were taken on throughout the course, and tasks were completed related to those roles. The insights in this retrospective come from coursework rather than a real-world development experience.