Module 7: Final Project

Tyeika Willis

Southern New Hampshire University

CS-250: Software Development Lifecycle

Professor Farley

June 18, 2025

ChadaTech has always used a very traditional way of making software called the waterfall method. This approach is kind of like a recipe where you plan everything from start to finish and follow the steps exactly. But in today’s fast-moving world, clients often change their minds or new ideas come up along the way. That’s where Agile comes in. Agile is more flexible and allows teams to adjust as they go, and one popular type of Agile is called Scrum (Highsmith, 2019). For this project, our team was chosen to test out the Scrum method while building a travel app for SNHU Travel, a client that wants to grow their business using smart technology. As part of the test, I took on different roles in the Scrum team and now, as Scrum Master, I’ve written this review and reflection to explain what we did, what worked well, and what we learned. This will help ChadaTech decide if all teams should switch to Scrum in the future.

Scrum teams have three main roles Product Owner, Developer, and Tester. I had the chance to try all of these roles throughout the project, and it helped me see how every role plays a big part in getting the job done. The 4th and most important role on a Scrum team is the Scrum Master. While they don’t tell people what to do or manage the project like a traditional boss, they do play a huge part in making sure everything runs smoothly. I had the chance to serve as the Scrum Master during this project, and it helped me see just how much this role matters.

As Scrum Master, I made sure our team followed the Scrum process and stayed focused during each sprint. I led daily stand up meetings, helped the team plan out each sprint, and reminded everyone of our goals. I also kept track of any issues that slowed us down, like confusion over a feature or problems with tools, and worked to clear those roadblocks so the team could keep moving forward. Another big part of my job was making sure everyone felt heard. If someone had an idea or concern, I encouraged them to bring it up during our sprint retrospective. That helped us grow as a team and constantly improve. It made people feel like their voice mattered, which led to better teamwork and better results. The Scrum Master role helped hold everything together. I wasn’t the boss, but I was the support system. I kept the team focused, helped remove obstacles, and made sure we stuck to the process that kept us organized and productive. Without someone playing that role, things could have easily fallen apart or gotten off track.

As the Product Owner, I focused on understanding what the client wanted and turning those ideas into a list of tasks called the product backlog. For example, SNHU Travel wanted customers to see the most popular travel spots on the homepage. So I wrote that as a user story and made sure it was a top priority in the backlog. I also had to answer questions from the team and make sure we stayed focused on features the client actually cared about.

As a Developer, I worked on building the features listed in the backlog. I helped code the section that showed the top five destinations, including images, short descriptions, and clickable links to travel packages. We broke the work into smaller tasks so different team members could focus on different parts, like the layout, the data, and the clickable link. I also helped debug problems when images weren’t loading correctly. It turned out some of them were saved with .jpeg instead of .jpg extensions, so the file paths needed to be updated in the code.

As the Tester, I had to double-check that everything worked properly. I wrote test cases based on each user story. For the destination list feature, I checked that each image showed up correctly, that clicking the link took users to the right package, and that nothing broke on different screen sizes. I also found a bug where two destinations had the same link, so I flagged it and the team fixed it before release. By doing all these roles, I learned that every part of the team is important. It also helped us respect each other more since we had all experienced what it was like to do someone else’s job.

User stories are short, simple ways of explaining a feature from the point of view of the user. For example, “As a traveler, I want to see the top destinations so I can pick a fun place to go.” This is much easier to understand than a technical request. Using Scrum helped us stay organized and complete these stories faster. Before each sprint, we had a Sprint Planning meeting to decide what stories we could finish during the week. During the sprint, we had Daily Stand Ups where each team member shared what they worked on, what they were doing next, and what problems they ran into. A real example was when we were working on adding images to each travel destination. One teammate couldn’t get the images to show up. They brought it up during the stand up, and someone else realized it was a file naming issue. We fixed it quickly and moved on. This teamwork and fast communication helped us complete the story on time.

Halfway through the project, the client changed their mind. They wanted to focus more on selling travel packages instead of just showing destinations. In the old waterfall way, that kind of change would have been a disaster because the whole plan would have had to start over. But with Scrum, we were able to adapt quickly. During our Sprint Review, we showed what we had done so far and talked about the new request. We added new user stories to the backlog for the package deals, sorted them by priority, and planned how to work on them in the next sprint. We didn’t lose time, and the team stayed focused because we had a clear plan. This experience showed me why Agile is so useful. Changes are expected, not feared (Highsmith, 2019). Instead of seeing change as a problem, we used it to improve the app and make the client happier.

Communication was a huge part of why this project worked. We talked every day during stand ups, and we also used tools like Slack to message each other and Jira to track tasks (VersionOne, 2020). Here’s an example of a message I sent to the team, “Hi team, for the new travel package section, should the results be sorted by popularity or price? I need to know so I can set up the sorting function the right way. Thanks” That simple message helped the team give me the answer I needed fast. I didn’t have to wait for a meeting. It also helped us avoid building something the client didn’t want. We also worked well together during retrospectives, where we talked about what went well and what didn’t. One time, we noticed that testing was being rushed at the end of each sprint. In the next sprint, we made testing a bigger part of the plan, which made a big difference.

We used some great tools to stay organized. Trello helped us manage tasks by using columns like “To Do,” “In Progress,” and “Done.” We could move cards as we worked, so everyone could see what was happening. GitHub helped us share code safely. Everyone could work on their part without messing up someone else’s code. Scrum events like Sprint Planning, Sprint Reviews, and Retrospectives gave us structure (Scrum.org, 2023). These meetings helped us make decisions, solve problems, and plan ahead. They kept the project moving forward and made sure nothing got forgotten. One change we made after a retrospective was giving more time to testing. Before that, we were rushing to test everything at the end, which caused stress and missed bugs. After we adjusted our schedule, the testing was more complete and the software worked better.

Scrum Agile worked really well for our project and helped us stay on track even when things changed. One of the biggest advantages was the flexibility it gave us. We were able to adjust our plans when needed without falling behind. The teamwork also improved because we communicated more, supported each other, and worked through problems together. We were able to identify and fix issues faster than we would have using a more traditional method. On top of that, the client stayed involved throughout the process and gave feedback early, which meant the final product matched their expectations more closely.

Of course, Scrum wasn’t perfect. At first, it was new to some team members, and there was a learning curve to understanding how it worked. It also required a lot of meetings, like daily stand ups, sprint planning, and reviews, which sometimes felt repetitive or time consuming. Another challenge was the risk of scope creep if we weren’t careful, we could take on too much work in a single sprint. Even with these challenges, I still believe Scrum was a great fit for this project. It helped us deliver a high quality product while staying connected as a team. I think other teams at ChadaTech, especially those working on fast-moving or creative projects, would benefit from using Scrum too.

Working on the SNHU Travel app using Scrum Agile was a great experience. It showed me how important it is to be flexible, to work as a team, and to listen to feedback along the way. Everyone on the team had an important job, and the Scrum structure helped us stay organized and focused. I think Agile would be a great fit for many other ChadaTech teams, especially in today’s fast moving tech world. It leads to better communication, better products, and happier clients.

References

Highsmith, J. A. (2019). *Agile project management: Creating innovative products* (3rd ed.). Addison-Wesley Professional

Scrum.org. (2023). *The Scrum Guide: The Definitive Guide to Scrum*. Retrieved from <https://scrumguides.org/>

VersionOne. (2020). *14th Annual State of Agile Report*. Retrieved from <https://stateofagile.com/>