Justin Hardin

CS-250 Software Development Lifecycle

7-1 Final Project Retrospective

December 12, 2024

Learning about the Agile methodology has been an eye-opening experience for me. Before taking this course, all my software project experience (as limited as it is) was always done in a waterfall methodology. I had never heard the term “waterfall” to describe a development approach, but as I read about it, it occurred that it perfectly described how I approached my projects. I thought you had to have an air-tight plan in place that served as the unchanging “gospel” for the project. This could explain why I have started many projects but finished very few. When I hit a road-block of some kind, I would go back to my design document and start redoing this, that and the other. I spent all my time meticulously planning and very little time actually getting anything done.

While working on the SNHU Travel Project, it was useful to assume the different roles on a Scrum-Agile team. Each role is very important, and they’re treated as such within the team. The Product Owner gathered the requirements from the stakeholders and created a backlog of user stories. Each user story had acceptance criteria that the team could use to make sure that we’re actually giving the customer what they want. The Scrum Master guided our Scrum events like daily stand-ups, Sprint Planning, Sprint reviews, etc. which ensured constant communication and transparency. Including the stakeholders for our Sprint Reviews was a major help in development. They informed us that they wanted to change some requirements mid-development, and because we were following agile principles, we quickly identified the change they wanted, and were able to implement it without derailing the whole project. They wanted to change from a list structure to a slide-show. Fortunately, the underlying code remained largely the same, so we mainly just needed to update how the data was displayed. Due to our Agile approach, we efficiently implemented the change and delivered value to the customer very quickly.

I think where the team really shined was its effective and efficient communication. Communication has such a strong focus in an Agile environment. Not only the planned meetings like the daily standups, but the spontaneous communications efforts as well. One good example is when the tester was attempting to create test cases, and realized that some of the user stories needed clarification. Rather than waiting until the next scheduled meeting, the tester emailed the Product Owner with the inquiries. If the Product Owner is unable to answer the questions themselves, they can simply contact the relevant stakeholder or stakeholders, to get clarification straight from the source. Once the details were sorted out, the tester was able to create targeted test cases based on the acceptance criteria of the user stories.

The main organizational tools we used were the product backlog and the user stories in it. Our well-thought-out product backlog was the bread and butter of the project. During each Sprint planning meeting, the team determined which user stories to focus on for the upcoming sprint. Then, the developers implemented features based on a user story’s acceptance criteria. That same acceptance criteria was then used to create test cases for the user story. During the Sprint review meetings, we would try to gage how the Sprint went. Did we accomplish all of our goals? Did we deliver value to the customer? Did we meet the acceptance criteria of the Sprint’s user stories? Did we properly test what we implemented? Doing this in iterations and keeping the product backlog up to date, we were able to give the customer what they wanted and in a timely manner.

While Agile proved to be a useful methodology for this project, it did come with some challenges. When the project started, we didn’t have a well-defined set of requirements. We initially got some vague desires from the client, so we had to make changes about mid-way through. Challenges like these can lead to frustration within the team, but fortunately Agile is good for adaptability and we were able to make the necessary changes quickly and efficiently.

I think Scrum-Agile was the way to go for this project. Through the use of user stories, we stayed on track, adapted to changes, and delivered value to the client. I don’t think the waterfall approach would have worked well for this project simply because the client didn’t know exactly what they wanted at the start of the project. They knew what they wanted on a high level, but weren’t certain on the details. So we would have spent a lot of time up front waiting for the client to decide exactly what they wanted so we could get the requirements. Since we had a fairly short time window, this would have wasted valuable time. With Agile, we were able to get started immediately. And while we did have to address a major change once the client decided on the specifics of the display, we were able to handle this change gracefully. I think Agile was definitely the way to go.