Treyton Davis

CS-250

June 12, 2023

Sprint Retrospective and Review

During the week of being a Scrum Master, I learned a lot of things about how to handle the team, while also helping each person become a better version of themselves. I also learned how the main structure of an Agile team is built. The testers and developers are the normal employees, the Scrum Master is like an hourly supervisor, and the Product owners are like the salaried managers. As a Scrum Master, your main priority is to ensure everyone understands how the sprints are set up to ensure no one goes too far ahead but also making sure to help anyone that isn’t fully grasping the concepts. As I said in my journal from that week, I would like to see each person on my team learn from mistakes but also not forget where they came from. I would also like to have the team figure out each other’s strengths and weaknesses to make sure everyone is able to help each other when needed by the team itself.

During the week of being a Product Owner, I learned many things that I wouldn’t have learned before being in the real world on how time-consuming many things could become. Between meetings with the team and stakeholders, there would be just enough time to be able to manage the backlog to ensure the team is staying on track. The user stories were a little difficult because there wasn’t a lot to be able to go off while building them, on top of also trying to determine acceptance criteria after building the stories. The interviews did help though because they showed me how the meetings might be, but they also didn’t show if the meetings were only ten minutes versus two hours. It is helpful that the stakeholders or customers get to see what is being built to ensure that everything that is being done is of the highest satisfaction or the team risks losing the contract to finish the project.

During the week of being a tester, it helped to understand the acceptance criteria to ensure that what the product owner might be testing is tested while everything else is also tested to ensure everything works together before that piece of features gets integrated into the system and testing after that piece has been integrated to ensure nothing broke while integrating. Learning how to write an email to the product owner for clarification was also easy since I have had that experience before for a major project I had in a previous class. I was the main person between the team and our champion, relaying everything back and forth to ensure everyone stayed on the same page.

During the week of being a developer was the easiest week in my opinion because I am used to building software and ensuring that everything is working fine before releasing the feature to the testers for further acceptance testing and possible integration into the product. For the project I was mentioning, I was both Product Owner and Developer. The biggest thing between being a developer and being a product owner is that sometimes I knew about requirement changes or additions before the team did so I was able to add them to the backlog before they were able to read their emails. Agile is easier to handle due to being able to build working prototypes to show the product owner and the customers to ensure that everything is set up the way they would like it to be.

A Scrum-Agile approach to the SDLC helped each user story come because I was able to sit down and read the meetings multiple times before finally determining what was needed for the first set of features to ensure each user got something they would have liked to see, and even helped the acceptance criteria be built due to them saying what they would like to have seen in the first meeting during the second week. One of the stakeholders explained that they would like to customize their profile and see top destinations that they would like to see instead of being hit by a generic list. I built a user story around that to ensure that feature was added in at the earliest point.

The Scrum-Agile approach helped when the project changed direction since we are supposed to accept change from the beginning. Agile is meant to handle potential changes to certain items, even possibly changing everything if required by the stakeholders. One thing that helped with the change is being able to focus on one certain type of travel type instead of many to help SNHU Travel release as soon as possible.

I try to communicate as effectively as possible. Sometimes I just can’t find the words that are needed, and sometimes I don’t know the right time to speak up about something that might hinder the team or the project. One way that I was able to communicate effectively was during our group work in Module Six with our discussion posts. I did agree that a system that allows us to communicate in real-time could potentially help the team when a bug is found to ensure that the bug could be fixed as soon as a developer has time to be able to fix it. One thing that would help with a group chat would be making sure it stays related to work, while also ensuring everyone can have a place to ask for help in case they don’t understand something.

One Scrum-Agile principle that helped the team be successful was the meetings because they were kept short but still had a lot of information in them to help the team understand what is expected and needed from each member to ensure the project stays on time or even ahead but still allows each person to have their own time to determine what could be helpful or not. Eclipse was also helpful for building the prototypes due to being able to send them to the product owner so they could look at the pieces on their own time while still being able to show the customers what has been done on the project so far.

Some of the pros that Scrum-agile were effective is when the project did have its requirements changed without us knowing before the meeting but still ensuring everyone was able to help keep up with getting everything done for the sprint that week. I did not notice many cons; I believe the only con I noticed was my time-management was not on par with what should’ve been expected following Agile principles on getting everything finished when we needed to have it finished.

I believe that a Scrum-Agile approach was best for the SNHU Travel project since we had many changes happen during the project that could’ve been forced to be dealt with at the very end of the project. With Agile, we were able to accommodate the change as soon as we found out to ensure everything would be able to be done at the same time as expected.