Michael Bauer

Sprint Retrospective

The product owner did many things that were useful and started with engaging with users. When engaging with the users the product owner asked the right questions in order to figure out what it is the users needed. The product owner then created the user stories based on the user’s needs. These user stories helped define what the development team needed to work on. After developing the user stories the product owner prioritized them and added them to the scrum board. By putting the user stories on a scrum board it made the information visually available and accessible for everyone on the team. This accessibility to information is what leads to information transparency. All this allowed the team to work on the most important user stories first, be organized, and be well informed.

As the scrum master I executed the daily scrum or daily standup to make sure everyone was aware of what happened yesterday and what was happening that current day. The daily scrum made sure everyone was communicating and was aware of what was being completed as well as what needed to be. I created an agenda and made sure to keep the meeting focused and effective. During the sprint planning we would, as a team, come up with what we will work on for this upcoming sprint. Before the beginning of the sprint and throughout the sprint I conducted backlog refinement, along with the product owner and team members. This allowed us to make sure that the backlog reflected the current priorities of the users and that there did not need to be any items removed. This was needed because as the sprint goes on more information is gathered from users that may change what is needed or what is no longer needed. These events are important because it allowed for refinement throughout the process, this created more efficient work.

The developers worked together to accomplish the priorities of the sprint backlog. They communicated openly during the daily scrum which allowed for information sharing and everyone to be on the same page. If there were any issues, they were brought up so we could handle them quickly. This made sure issues did not compile and create havoc later. When there were changes made to the product backlog priorities, the developers were able to quickly switch focus to these new user stories. The developers are the ones that worked on the user stories to provide the deliverables to the users.

The product testers made sure that the user stories that were completed held up and that we provided quality to the users. If there was an issue with a user story the product testers insured we caught it early and this gave us time to fix any bugs in the user stories. If a product tester had an issue or question they were able to communicate these. The product testers made sure that any bugs did not accumulate and they made sure we did not wait until the end to find them. Knowing the bugs before the end kept us better informed with time to fix the bugs. Had we not known until the end we could not have properly planned enough time to fix these bugs.

The scrum-agile approach allows for a hands-on approach with users. The users can relay their needs and those needs are turned into user stories. After discussions with the users the product owner prioritized the user stories. These were then added to the product backlog and then to the sprint backlog. The developers worked on the highest priority first and worked down the list. This made sure that the users were getting the most important features they needed first. When the users needed to change their priorities they communicated this and new user stories were made. These were added to the product backlog and priorities were changed based on the new user stories. The developers then changed focus to the new priorities to make sure we delivered the user stories that the users needed.

Industries change and those changes can shape what needs a user may have. This is where scrum-agile is successful. Scrum-agile allowed users to have a hands-on approach to the project. This allowed the users to communicate any changes in their needs. These changes were moved into the product backlog quickly. This made always focusing on the user’s current needs easy.

Here is a sample of communication between myself and the team. “Before touching on the product backlog, I want to talk about user stories. These user stories will come out of discussions with the stakeholders/users. I will, based on these discussions, create user stories for what they need. If these are too large we will break them into smaller parts. These will be assigned story points, allowing us to predict how much we can get done in a sprint. These user stories will then populate the product backlog. The product backlog will be prioritized by me and updated after each sprint. I will make the user stories as clear as I can so there are no misunderstandings or room for judgement. The product backlog will be visible to everyone so that there is transparency with the information and everyone is on the same page.” Another here is to a developer, “Absolutely, we will have a sprint planning meeting to go over the upcoming sprint. Here we will look at the user stories and the current priorities. We will assign story points to the user stories and then add them to the sprint. The tool that Ali mentioned sounds like a really good tool to keep everyone on the same page. This will allow for information transparency and everyone to stay on the same page.” The first example effectively explained what was going to happen in each step going forward. Even if it was a step that did not involve all the members, this allowed for transparency with what the plan was. Everyone would be on the same page and have access to the same information. The second communication explained the sprint planning we were going to do and how we all would be involved. This showed that we would be working together as a team to complete the project.

There is not one principle or tool that made us successful. The scrum-agile approach is a combination of things and those together makes a team successful. It starts with the user’s being hands-on with the project. The user’s having their needs heard and then these needs being broken into user stories. Using a board to organize and prioritize what user stories are needed and keeping track of ones being worked on or that are completed. Having a daily scrum to allow any issues to be brought up and make everyone aware of what each member is working on. Having all this information accessible to everyone so there is information transparency among the team. Meeting together for sprint planning so we can decide as a team what we can accomplish in the next sprint. These are what helped the team be successful.

I have mentioned some pros of the scrum-agile approach above. Being able to change direction to stay working on the current needs of the users without losing much time. Allowed users to have interaction in the project, allowing them to change their needs if things change. Allowed testing of the user stories as they were developed, making efficient use of time and making sure we had quality deliverables for the users. Transparency of information made sure the team was on the same page. Some of the cons included a lack of experience needed with scrum-agile to benefit fully from it. Not being able to accurately predict sprint times because of the lack of experience the team had together. I believe overall the scrum-agile was the best approach since there was a change in the user’s priorities, scrum-agile allowed for that change to happen without starting over. With other approaches that change would not be voiced until the end where the project would have to be redone losing all the time we put in.