Peter Bou Saada

CS-250 Software Development Lifecycle

Linda Gaston

10/15/2021

SDLC: Sprint Review and Retrospective

The agile software development lifecycle has different roles to maintain things in order, and keep steady progress, those roles are the product owner, scrum master, tester(s), and developer(s). Each member of the team plays a very vital role in the success of the agile process. The product owner is responsible for communicating the progress to stakeholders, communicating with users to get user stories, and creating the product backlog, this is the backbone of any project, communication is key, and finding out what needs to be worked no is essential. The developers are responsible for creating the software that can fulfill the user stories, and make it maintainable, and expandable. The testers are responsible for creating test cases based on the user stories and making sure that the user stories are being fulfilled by the developers correctly. The scrum master is responsible for holding the daily scrum meetings and making sure that everyone has what they need to succeed, both developers and testers are responsible to communicate this during the daily scrum meetings. I personally experienced every single one of those roles throughout this course, and all these roles relate to each other in some way or another. The product owner had to develop the product backlog for developers to work on. The tester had to get information from the product owner on what makes a test successful or unsuccessful. The developer had to get more specifications from the product owner, and get feedback from testers, and as the scrum master, I had to facilitate the communication between everyone, and make sure the developers and testers had everything they needed.

The agile approach to SDLC allows for a lot of agility (hence the name agile) in production, it allows teams to efficiently adapt to changes in requirements. This makes production more efficient when compared to a method similar to waterfall. From personal experience, things like the daily scrum meetings allow team members to communicate what they need, and what they are working on, ensuring that everyone is working on different things, or helping each other with the same thing. Agile also provides a backlog of user stories that the team can use to prioritize their work, and at any point, the importance of a user story is what tells the team members what they need to work on. During this course, we experienced all of those. The product owner made a product backlog, which allowed the developer and tester to use the product backlog and get their work done.

Halfway through the SNHU travel project, stakeholders decided to make changes to the travel destinations that we are suggesting, and because we are using agile, it was extremely easy to do. The product owner just needed to update the story point to match what we needed and give it a higher priority for the team members to work on, we could not change the deadline, but the team was not rushed, they were told to do as much as they can and not worry, this helps improve morale.

Communication was key for this project, as scrum master, it was my responsibility to make sure that all team members were communicating what they were working on, what they needed, and even direct side conversations that should not take place in the daily scrum meeting. The developer sent an email to the product owner and the tester, asking about more specifications and definitive pass or fail for test cases. The developer sent the following questions to the product owner when the changes were made to the product backlog:

“I have questions for the product owner on the new changes, I would need to know a few things:

* Are we considering the user’s personal preferences when recommending detox/relaxation vacations? If so, how?
* Is the user going to be able to turn off this feature? Or are they going to be forced to look at these kinds of vacations even if they do not like them ?”

This is a perfect display of communication, short and concise.

The team requested that a tool be used to keep track of the ongoing issues and user stories. After thorough consideration we have decided to use Jira, it is marketed as “The #1 software development tool used by agile teams” and it has done wonders for our team, it made organization much better, as it is instant, and we do not need to go check a white board to update/check the progress of a user story. It allows the division of the SDLC into sprints, and releases. During the daily scrums, we just had to look at Jira in order to find out what has already been done, and we were able to update what is being worked on through it within seconds.

A scrum-agile approach to the development process has provided us with the many pros that I featured already in this sprint review, but it also carries some cons, the change that was requested by the stakeholders, put strain on the developers and testers, as they had to adjust and remove parts of their earlier work to accommodate for the new changes. However the benefits that it has provided far outweigh the cons, our team was extremely efficient at getting user stories done, and as some team members are not yet used to agile, we only expect the performance to increase as they get used to it more and more. I conclude with the opinion that a scrum-agile approach was very beneficial for the SNHU travel project, we already have a ready product within the first sprint, if we had continued with waterfall, we would have needed longer time to get a working product, and any changes would take a lot longer.