Sprint Review and Retrospective

CS 250

Justin Farquhar

12/10/2023

To begin, the scrum master allowed for a lot of the initial set up for the transition to the agile methodology, from getting the initial information from the client, to the sprint planning, and now to the sprint retrospective. As stated in the initial journal, for Sprint planning, at this time those are really as far as we could plan unless we were to fully wing it and hope the client is satisfied. For the daily scrums, we could go over what each team member had accomplished, and hopes to accomplish, while also allowing for time to provide any roadblocks that may have been met by any team members. This was going over the initial planning stages, unfortunately we did not get the most information right at the start, leading to a bit of a delay on actual work, but it has since worked out. Each of these events are very important in their own ways. Sprint planning allows for a layout of a timeline of the development cycle for the product. The daily scrums gives everyone a daily overview of where progress is at and allows for help if someone needs it. Backlog refinement is a natural process as more and more features are expressed, so there is a reference for what all is done and what still needs to be done. The job of the scrum master went very well, allowing the team to gain knowledge, a feeling of belonging, and a sense of pride for being able to work through this massive change.

Next we go to the role of the product owner, who oversaw communication with stakeholders, management of the backlog, and management/adjustments to user stories. The interviews/user meetings were the driving force in writing each user story as each user was able to provide detailed information on what exactly they wanted out of the product. If needed, there was the option for the product owner to ask for clarification or additional details to help the scrum team develop what is wanted out of the product. This process also avoids the issue of very open-ended expectations of the program that could be present in other development processes where a client could send an extremely vague expectation of a product they want developed, that may very well not serve their wants or even be feasible to create. Being able to actively listen wildly helped the role of the product owner, having to go through the initial interview to determine what the stakeholders/customers wanted out of the app can take a lot, especially if you can’t read more into what exactly they are looking for, as they are very unlikely to give a clear cut answer every single time. This also leads into knowing what all is possible, the product owner should have a good grasp knowing what exactly the team is able to do, and know where some bounds could be overstepped or where to lay back on others to prevent over promising. The product owners management of the product backlog lead to efficient completion of each user stories based on importance, with this the project was able to be completed at a steady pace with everyone knowing what was next on the chopping board.

Next we move onto the role of the tester, who served an extremely important role in making sure we actually were able to deliver a working product. Without their services the entire process would have taken substantially longer, if we were even able to catch all of the issues to be able to deliver a viable product. The tester did run into issues with clarity, as stated in their journal. A bit more clarity would have been very helpful. Thankfully, with agile we get more freedom to add in features as they come up or are asked of us, but more thorough documentation of what exactly was wanted would have made this initial testing significantly easier and more straightforward. Though I do partially place that blame on us, as we were not asking the right questions or trying to delve into the customers wants more. This information will have to come with either the customers asking for additional features or through us asking for more clarification on some of the features they had presented to us initially. Unfortunately, we are left at a sort of standstill where we must wait on that feedback, but there are still plenty of additional features to be added. While waiting for this clarification, work was still able to be done to help propel the project forward.

Finally we move onto the role of the developer, who was responsible for providing the code for the program to move forward. As the developer, there was a substantial change in direction requiring some change in code that was already set. But as stated in their journal, agile methodology is just about perfectly tailored for flexibility. As more tasks or requirements are brought upon developers we have the ability to flex around them, decide where they fit within an updated backlog, and either begin working on them immediately or at a time when it is more appropriate, all without setting back development too dramatically. If we were to be using a different approach, this otherwise small change would have set development back dramatically, potentially to the point of affecting the end delivery date. This alone shows the power of agile when faced with a project being interrupted and changed, without, the project would have likely been delayed massively, requiring a lot more restructuring.

Communications was initially a harder point, being fully remote and only having emails, discussion boards, and web cams lead to a few hiccups. But the team was more than able to overcome these. I personally think the entirety of the agile process helps facilitate communication within a Scrum Team. So much of the design plays into allowing for more time and ability for additional communication within a team. Having the smaller teams, in additional to the frequent scrum meetings allow for more consistent forms, as well as closer knit teams to allow for feedback loops between team members.

Then for the organizational tools and principles that helped the team be successful. The daily scrums were a very large help. This allowed for team members to provide pain points they were dealing with and either have a fellow team member acknowledge or provide help with the issue. Without this, we could have run into the issue of the team sitting on the issue for too long and just not accomplishing anything at all. The Sprints provided the absolute background for this entire process. Having the ability to migrate everything within the project into smaller two week intervals allowed for smaller initial projects that all worked towards the bigger project as a whole. The sprint planning gave us the start for the entire process, having the ability to plan out everything, especially going into this completely blind was a great alleviation of stress.

As a whole, the scrum-agile approach for this project was an incredible success. For pros, the project was able to move on even with the change in direction from the stakeholders, the team was able to learn off of each other better being in the smaller teams, and motivation was greatly improved working in the smaller sprints rather than having everything blend together with a large project. For cons, there was a lot of confusion, since we did not get the full overarching goals from the stakeholders/customers initially, we were effectively running wild trying to get what we could out of that initial interview and were waiting for additional guidance from them. This could have just been an issue on our end, either being inexperienced in the process or just a part of agile as with waterfall you would absolutely have to have everything all at once before you could even begin trying to do anything. Overall, I think the agile approach was the best approach we could have taken for the SNHU Travel development project and would not have done it any other way, the successes outweigh the negatives by large, and this was a great experience for the entire team.