Guilherme Pereira

CS-250

SNHU Travel – Sprint Retrospective

During our time developing SNHU Travel, we performed a myriad of tasks that could only have been accomplished with a team. However, this task could only have been accomplished effectively using the agile model. The agile model differs from traditional models such as the waterfall model, as it anticipates changes and embraces versatility as opposed to a more linear cycle. Agile works to create a product that is more customer-focused as opposed to waterfall. During development, each team member had an important role to the success of SNHU Travel, that being the roles of Product Owner, Scrum Master, Developer, and Tester. Each role had a unique contribution towards the project which helped to bring the product to life.

Roles are important designations within the team to ensure each person adheres to certain tasks. In agile, there are four main roles that helped successfully create our product. The Product Owner is responsible for the product backlog management, which helps to have an organized set of user stories and prioritizes what functionalities should be done first. A user story is a practical example or scenario that explains what the task or functionality is about, what the customers intends to do with said functionality, and why the customer wants to do this. In this project, the Product Owner was able to communicate with the clients regarding what functionalities they would like to see, such as travel discounts, setting a price range, or being able to check if there are extra attractions nearby. Having collected this information, the Product Owner, along with the Scrum Master, were able to put together a list of user stories that were organized from high to low priority, based on customer needs. This product backlog is then passed onto the development team as to what should be completed.

The Scrum Master is responsible for overall management of the development team while not taking a permanent role as their leader. This role focuses on making the development team’s lives easier by providing accommodations or stepping in to assist them such as a user story explanation or a technical question. Scrum Masters are responsible for sprint planning, holding the daily scrum meeting, maintaining an organized and prioritized backlog, and conducting a sprint retrospective to conclude the strengths and weaknesses or each member for this project. For SNHU Travel, the Scrum Master was holding a daily scrum meeting that helped keep everyone focused and prepared for their day at work. At these meetings, three questions are always asked: “What did you accomplish yesterday?”, “What will you accomplish today?”, and “Are there any obstacles preventing you from accomplishing your work?”. Having these questions answered by each development member helps to Scrum Master to become more accommodating and provides more utility for the team. While nothing specifically was asked for this project, a Scrum Master may provide blinds for the team if the sun’s glare at them from a certain angle or retrieve a resource to help solve a certain problem a developer has come across. A list of strengths and weaknesses for each member is listed towards the end of this document.

A developer’s role is straightforward in terms of what needs to be done. However, there are more teamwork elements at play here compared to other models. In some traditional environments, every developer was given a set task depending on their experience or what they could handle. This sometimes led to developers having too much to do or too little, which also worked against a teamwork setting. While agile does not focus on sharing work necessarily, it helps support the teamwork role by having members come together everyday and explain what they have accomplished and what needs to be done still. Developers are able to utilize different tools to help them with their work, such as Gantt charts for precise works and deadlines, or Kanban which focuses on higher-priority tasks first. A developer should be able to share their knowledge with others if it helps them get through the task more easily. Online tools such as Microsoft Teams can also help developers set the pace within their team and exchange tools and information with one another. For SNHU Travel, developers were the creators of bringing the product to life. When changes arrived such as implementing a detox or wellness rating, some developers had concerns of whether the product had to be remade. This was not the case as they just needed to apply a new functionality, and with the expected change, the client was able to get what they needed. When developers experience change, a tester will come in to help see what errors may occur from the additional features in the product.

The final role in an agile work setting is that of the tester. While testers work to fix and note bugs caused by the product in development, they are also responsible with communicating with the product owner to understand what it is the clients want to do. A tester may need a detailed instruction as to a user story’s message, which helps them check things from logic errors to elements breaking down. Testers often communication with developers as well to notify of any issues that have occurred or might occur in the future depending on how something was written. Within SNHU Travel’s development, testers played a large role in testing for any issues that would be caused by adding the latest functionality of detox and wellness. While they have worked on testing other aspects such as price range and extra attractions, a tester can be very helpful when working on something newly implemented. This will occur many times in an agile environment.

The software development life cycle begins with planning, and moves onto analysis, design, implementation, testing, and finally delivering and maintenance. While the waterfall method would follow those steps linearly, agile embraces change and would have shorter versions of these steps in segments called sprints. One of the values of the agile manifesto is customer collaboration over contract negotiation, which aims to keep customer needs priority as opposed to what was written in a contract or document. People are subject to change, therefore making a product also prone to that. With further client communication, the product owner was able to assess how the user stories should be written and why the customer wants to do a specific thing.

During earlier segments of agile, we were tasked with creating sample emails between two people of different roles. Below is one example of three different people, A, B, and C communication with one another regarding a question of the newly implemented detox functionality. Recipient A is the developer, Recipient B is the product owner, and Recipient C is the tester. These are mock emails that shows a scenario in which a developer might need further information to complete their task. The purpose for communicating with the product owner is to obtain more detail as to what the functionality should look like in the product. The purpose for communicating with the tester is to notify of recent changes made to the system. Overall, having this communication and each member play a role is important for keeping a well-maintained development.

Hello Recipient B,

*I was made to understand today in our daily meeting that we have a new requirement to work on that should take priority over all the other items on our product backlog. In working with this new user story, specifically the one for adding a tool to help users view a detox / and wellness of a location, how would you like for us to go about that? While I do understand the idea of a customer wanting to know the cleanliness of a place, I fail to understand how they would gather that information. Do you want us to implement a system where users are able to provide a wellness rating of a location, or do you want us to list places that may serve to detox people such as spas, resorts, and the like?*

Thank you,

Recipient A

=====================================================================

Hello Recipient A,

*Thank you for reaching out to me for this clarification. From what I have gathered from our stakeholders, I believe that a list of places to help detox the customer will be best suited for this application. Being able to list out nearby spas, resorts, and other detox facilities is the functionality we are looking for our product.*

Good luck,

Recipient B

Hello Recipient C,

*I have recently made changes to the program that implements a new wellness system. This system will help display all nearby detoxing facilities for the main location listed in the travel app. However, I would ask that you test out whether all these locations are listed or if any are missing.*

Thank you,

Recipient A

=====================================================================

Hello Recipient A,

*I will investigate that for you and see if there are any businesses or locations that may go unnoticed for this function. I will keep an eye out for small businesses who may not have made a new for themselves and have yet to eb registered on the map. Also, I realized that the addresses for the sub-locations are not formatted properly, and this may also affect how the addresses for the detox facilities are displayed too. The text box appears to be overlapping the buttons, so see if you can fix that as well.*

Speak to you soon,

Recipient C

While no organizational tools were used for this project, one that does come to mind that would be great is Microsoft Team. Microsoft Team allows a wide variety of team sizes from 2 to 10,000, according to their product description. Team members can communicate in real time and send files to one another to edit or collaborate on. Microsoft Team provides unique tools to work with such as apps and workflows, market availability, and other commonly used features such as screen sharing and video conferencing. I believe in a scrum-agile environment where things are subject to shift a lot, this tool is perfect for teams of various sizes. Information radiators play a big role in obtaining information that everyone should have. An administrator may post and tag everyone in chat with that information on Microsoft Team, making it easily accessible for the entire team.

The Scrum-agile approach for the SNHU Travel project served greatly for what was needed during development. As seen through development, the product owner would connect with clients regularly to ask about what they want from the product. As time progressed, clients began to add additional features such as price range and detox ratings. While this may have extended the deadline for the product in a waterfall model, the product owner was able to adapt the change with the developers and testers, asking them to focus primarily on the new functionality. While I do believe agile shares more positive aspects than negative, the one notable factor that makes agile worse than waterfall is client power. Sometimes a client may strongarm the development of a product and make several changes near the end, which could severely impact the quality of the product in the end if everything needs to be done on time. This may also frustrate the development team, while they still need to make up for time lost on the change. While agile is about change, I believe too much in a short span of time is not good for development.

Finally, the retrospective to be listed. A sprint retrospective lists multiple feedback to each person on the team as to what they did well and what can be improved upon in the future. For this project, there were no team members to analyze or give feedback, but in a practical scenario, I believe each member would have feedback regarding their role in the team. A developer may receive more constructive criticism for how they built the application, while a tester may be asked to check different problems next time, such as login issues or password retrieval. I believe the sprint retrospective is important to help motivate members who are heading towards a new sprint, as it aims to encourage their growth as an individual and as a team.