Jonathan Boeglin

CS-250 Final Project

Sprint Review and Retrospective

For this Sprint Review and Retrospective, several parts of the Scrum-agile approach will be addressed. This includes how each role in the team contributed towards the project, how it helped complete user stories, how it handled sudden changes, team communication, organizational tools and principles used, and the overall effectiveness.

**Role Contribution**

To start, the Scrum team is made up of different specialists that help move the project forward with their own skills. In this case, the team included a Product Owner, Scrum Master, Tester, and Developer. The Product Owner contributed to the success of the project by receiving the initial request, interviewing the customer base for ideas on the project, and passing on any relevant information to the rest of the team. For the initial meeting with the client, the Product Owner received the project idea from the client, which allowed them to start creating the Sprint backlog. During the meeting with the userbase, the Product Owner gathered ideas from users representing those that would benefit from the project and implemented the most logical ones into the Product Backlog. For communication, the Product Owner made sure the team knew about any major changes in requests from the client in a timely manner.

Next the Scrum Master was also involved in receiving the initial request, assembling the rest of the agile team, create the charter, and schedule Scrum events. For the initial request, the Scrum Master heard what was expected, and determined what the size and make-up of the agile team should be. Then once the team was formed the Scrum Master lead them in creating the agile charter, which was a document stating what was expected of each person on the team, and the conditions for success of the project. Finally, the Scrum Master scheduled Scrum events such as the Sprint Planning and Daily Scrum. These meeting allowed effective communication between the team members and facilitated completion of the project.

The Tester on the team oversaw the development of accurate testing parameters for the project’s code to make sure that development was headed in the correct direction. This process included developing tests based on user stories, sending and receiving communication on any questions on what was expected, and by developing flexible tests that can be altered when plans change. For the development of and sudden changes to the testing code, the Tester had to read through the user stories and determine what was expected of the project. With the addition of any questions answered by the Product Owner, the Tester created an accurate set of rules that the project had to follow to be considered complete.

Finally, the Developer oversaw completing the project and making sure it passed the tests set up by the Tester. This included coding the project, intake and implementation of any sudden changes requested by the client and implementing the framework set-up by the Product Owner. Without coding the project, it wouldn’t exist, and not implementing requested changes would mean the project would never reach the agreed definition of “done.” Finally, by implementing any requests and framework set-up by the Product Owner, the Developer had a head start in the correct direction for completing the project.

**User Stories and Project Completion**

The next main area to cover in the agile process is the help the agile system contributed towards user story and overall project completion. For user stories, the Scrum-agile approach to the SDLC helped by giving structure to the cycle that allowed the team to work on the stories based on priority and difficulty. This was shown in the User Stories worksheet where the stories were categorized with short explanations for each. Other stories were not then lost, but saved for another sprint, or added to the sprint if a team member felt both could be completed at once and had relevance to each other. For project completion, the Scrum-agile approach allowed sudden changes in in direction or other interruptions to not completely ruin the completed work. Instead, the changes were only implemented in a way that related to the project currently, and all other ideas were added to the Project Backlog to be completed in a different Sprint. An example of this is when the Product Owner was informed that the project should focus on detox/wellness travel, instead of general niche vacations. This also allowed the Sprint to retain its original completion date and kept the project from falling behind schedule.

**Communication, Tools, and Principles**

Scrum-agile lends itself towards effective communication and has several tools to utilize and principles to follow that allow for a successful Sprint. For communication, examples include the use of email to pass along information and ask any questions regarding the project. Examples include the emails sent by the Tester and Developer to the Product Owner and the email sent to the Tester from the Developer. In the Tester email questions regarding each user story were sent, such as if the top choices should be in a list or slideshow format. With the Developer, the Product Owner received questions on the ranking system and specification of activity level, and the Tester received an email request for any currently completed tests, and the same questions the Product Owner was asked in case they had a different opinion on anything. These emails encouraged collaboration through active discussion of the project between different members of the team and allowed the whole team to be in the same headspace when working on the project.

The tools besides emails and principles besides open communication include the User Story and Test Case documents and the Rules of Behavior outlined in the Agile Team Charter. The documents and emails were a stand-in for the Daily Scrum and information radiator that a normal team would have, where the stories and testing updates would be listed similarly in the radiator, and the emails would be face-to-face conversations during the Daily Scrum. These replacements helped the team communicate and update each other effectively in the wake of an actual meeting and radiator. The principles that were listed include respect for team members, constructive criticism, accepting responsibility, and collaboration. Collaboration is clearly covered by the communication the team had, and the communication itself was always respectful. Constructive criticism and feedback came in the form of sudden changes needed, and the team took it in stride. The Product Owner facilitated this by taking responsibility for the needed change and having a plan of implementation. Also, the Product Owner recognized the work already done by the team at that point and made sure they understood their work so far had not been for nothing. This led to success by reassuring the team they were still moving forward, although the direction of forward may have changed.

**Scrum-agile Effectiveness**

The Scrum-agile approach has pros and cons when concerning the SNHU Travel Project. The pros include flexibility in content, ease of planning, and availability of help and tools. The cons include rigidity of deadlines and lack of defined project parameters. Project parameters that are well defined lead to a finished product that is closer to what the client had in mind, but due to the constant change in an agile setup, rigid project parameters are hard to establish. In addition, the deadlines for each Sprint usually cannot be changed, as they are decided ahead of time, and can mean that a project that just needed one more day of development still fails due to it being stuck in a rigid schedule. The amount of tools available, help with using them, and ease of planning can all help counteract these issues, however. With the correct planning and use of tools most projects are defined well enough to move on to the next phase and finish when planned. Finally, the flexibility in the content being created allows constant development while interruptions may change the entire project, such as what happened with the niche sites becoming detox/wellness sites.

For this project, the Scrum-agile approach was the best approach. It allowed the team to work on different facets of the project at the same time, kept the team small and organized, allowed for open communication which facilitated better output, and had a set of principles that made each member feel fulfilled in their work. Most importantly, though, it allowed for change while still progressing, where other systems would still be stuck in development and re-development due to these changes.