**Sprint Review and Retrospective**

At the heart of a Scrum-agile team is communication. Every team member must understand their roles and how they work together within the Scrum framework. In addition, working together on a project requires communication to safeguard the team's effectiveness and guarantee a functional product at the end of every Sprint.

A servant/leader is someone who works with the team to understand issues and work to resolve them. That could be within the group as a mediator of personal conflict, or it could be dealing with problems that are external to the team that may inhibit their effectiveness. It is the responsibility of the Scrum Master to be this servant/leader. In addition, the Scrum Master ensures the team is working under the guiding principles of the Scrum-Agile methodology. Keeping the team on task and helping to remove any obstacles that may prevent achieving the team's goals.

The tools or events the Scrum Master uses to aid the team with those goals are Daily Scrum Meetings, Sprint Reviews, and Sprint Retrospectives. These events help the team look at the project on a micro or daily focus and on a macro or whole picture understanding. The Scrum Master, in a Daily Scrum Meeting, will have each member address their work for the day and identify their needs to accomplish their task. Before the end of the Sprint, the Scrum Master will assemble the team for a Sprint Review to discuss what was accomplished and what (if any) tasks were not. The Scrum Master's role is to determine if there are necessary adjustments needed for Backlog Refinement or the next Sprint.

At the end of the Sprint is a Sprint Retrospective. The Scrum Master and team will explore the entirety of the Sprint and discuss what worked for individuals or the team and what didn't. The Scrum Master will note each of these from each team member and work to resolve those items that kept individuals or the team from being effective. Making changes and adjusting for the next iteration of the Sprint improves the team's effectiveness and helps meet the users' expectations.

The Product Owner is the link between the development team and the stakeholders. The Product Owner must also make sure that they understand the Scrum methodology as well as understanding of the developer's needs. This helps the Product Owner direct the information and feedback from the users and stakeholders into actionable User Stories that Developers and Testers can work from. The Product Owner will organize all the User stories into the Product Backlog.

Talking and engaging with those who will eventually use the software is a difficult skill and a valuable tool when understanding the user's and stakeholder's likes, dislikes, needs, and wants. Having the conversation to drill down specifics helps the development team when it comes time to start working on deliverables. Often, a user or a stakeholder may not have a fully realized idea of the function or feature they are requesting. Having an independent person to draw out the specifics can significantly improve how that function or feature is designed and produced.

Testers on the Scrum Team have a vital function that promotes confidence in the Product Owner and employs a measure of quality control for the Developers. The Testers work with the User Stories on the Product Backlog to define workable parameters for the Developers to work towards. Rather than setting to discover issues within the application before delivery, the Testers work with the Developers to address the problems as they are found at the time of their development. This can be in the form of test cases that the application should perform or a review of functionality. Testers ensure that the development of each User Story is to the Product Owner's satisfaction but also demand a high level of proficiency and skill from the Developers.

Developers must be proficient in creating applications to meet the expectations of the Testers, the Product Owner, and the client. The ability to respond to changes and adjust is imperative to the whole team's work. Developers must remain flexible and be effective communicators in responding to changes and identifying issues or solutions for the entire project. As they are working to implement the User Stories into a functional application, they are the first line in discovering (or creating) things that are working or not.

The Scrum-agile approach to the Software Development Lifecycle (SDLC) focuses on the dissection of the user's and stakeholder's requests into actionable tasks. The Team then reviews the Product Backlog and assigns each User Story to the Developers for the Sprint Backlog. Next, the Testers outline each of the User Stories with functionality expectations into parameters that Developers work toward completing. Finally, with the help of the Testers, the Developers create a functional application once the scheduled Sprint ends. Upon completion of the Sprint, the Scrum Team reviews the completed work and discusses obstacles and needs that will increase effectiveness for the next Sprint. In addition, the Team discusses the general process and information used to help set more precise definitions for future information gathering or processes.

If the Team discovered that there were aspects of the User's Stories that needed to be more transparent or if the information was required to define specific functions within the application, the Team would work with the Product Owner to ensure that information was received from the users or stakeholders. Clearly defined User Stories help the Product Owner communicate the development team's needs with the stakeholders. Additionally, accurate User Stories help the Testers and Developers outline the definition of "done" for each story. This aids the final product to better meet the stakeholders' expectations.

There was an instance during this evolution where the Product Owner met with the client. They expressed a pivot in the focus of the SNHU Travel application from general vacation locations to detox/wellness vacations. This is an example of how remaining flexible during the development process helps the team adapt to market conditions. The ability of the team to make this pivot ensures that the client would remain competitive in the market they hope to serve. To accomplish this pivot, the team would review the Product Backlog, deprioritize some of the user stories, and add or revise user stories to meet the customers' expectations. With effective communication, the team could adapt to this change without compromising the agreed-upon schedule or deadlines.

Effective communication is required to engage with the Team and get everyone involved. For example, in the initial client meeting with the Product Owner, the team worked on the original user stories to develop test cases that the developers could work towards for each user story. Shortly after, another meeting with the Product owner was held to discuss a slight change in focus. After each, it was imperative to ensure the Developers and Testers worked with accurate objectives. Each time many questions were asked, some background on each of the questions was given, either in the form of an opinion on the subject or in the way of a possible solution for the team to discuss further. That discussion enables each team member to have a say and offer solutions or identify later issues that need to be considered.

The following are some examples of emails that the Tester and Developer created:

**Example 1**

**To**: Christy (Product Owner)

**Cc**: Ron (Scrum Master); Nicole (Developer)

**Subject**: Latest User Stories from Last Focus Group

**Attachments**: ***CS 250 Module Four Test Case\_Jonathan Kleven.xlsx***

Good evening, Christy,

You will find the attached user stories and the proposed test cases for the items identified in the last focus group. Additionally, per your previous email requesting clarification, I have added 3 revisions to some User Story Test Cases. However, before we set this in stone for our upcoming Sprint, I would like to get your feedback and possibly Amanda's (Client) input, if any, to ensure we are all aligned.

**Questions:**

* Is a mobile app within the scope of this initial Sprint?
* How many criteria should we refine results for each user profile?
  + - Should the user be able to un-filter profile refined results if they ever wish to explore destinations outside their comfort zone?
* Should refining results by price include a price range, low end to high end, and allow destination results to be within the user's budget?

**Background:**

I agree that that mobile app may be a task for another Sprint, though the ability of this function would reach a wider audience. For example, suppose users only filter destinations based on a profile survey. In that case, the destinations may omit results that may not initially be to their liking but have activities and adventures that are more tailored to their taste, therefore having even a glimpse of destinations that were omitted or having the feature turned off would allow some users to explore areas they may not have seen initially as within their preference. Finally, allowing the user to set the low and high end of their budget would allow more confidence in the user of the SNHU Travel system to see the best destinations to their means.

Thank you,

Jonathan Kleven (Tester)

**Example 2**

**To**: Christy (Product Owner); Brian (Tester)

**Cc**: Ron (Scrum Master); Nicole (Developer)

**Subject**: Wellness/Detox Booking Pivot

Good evening Christy and Brian,

To ensure we make the application's correct adjustments, we need to understand a few details before moving forward.

With the wellness/detox vacations, are we looking for the most popular destinations nationwide, or are we looking to include some elements that make up a wellness/detox vacation, i.e., secluded areas, nature and wilderness, regimented programs for disconnecting from a digital lifestyle, and finally, therapy led courses for general wellness or anything specific?

As with most travel destinations, there can be no absolutely correct answer, but if the search results are too narrowly focused, that may turn off some users. Tagging destinations that have specific elements of wellness/detox may benefit the application and allow destinations within our existing resources to be included. Moreover, adding specific wellness/detox locations and tagging them with the elements that make them geared toward wellness/detox will provide a more extensive repository of search results to give the users more options to select the best fit for them.

Thank you,

-Jonathan Kleven (Developer)

The organization is key to ensuring nothing is left behind and that everyone has a clear sense of the full scope of the project. Communication is critical to the organization. While email can be an effective method of communication, it can leave much to be desired when it comes to organization. In the information age, the time of the spreadsheet to organize projects is dead. Today, many tools at the Scrum Teams' disposal offer organization, communication, and company and client favorite – metrics.

Information radiators are methods to aid in communication and organization. Information radiators like Jira or VersionOne provide the Scrum team a virtual location to view the project's entire scope and the team member's individual tasks. They offer a visual representation of the work that needs to be done and the work that has been completed. Additionally, communication among the team concerning User Stories can happen within the application. Requests for consultation and questions regarding User Stories integration with another can occur under the umbrella of one application.

Having a centralized location to discuss the project is essential when working with your team in the same room or building. The same is true when working in person or working remotely. Information radiators offer a Scrum Team a central organization tool that helps get the team involved, on the same page, and working collaboratively to complete the project effectively.

The SNHU Travel project was a fine example to show the merits of the Scrum-agile software development approach. One assumes that much of the information passed over the 6 weeks would have happened much quicker. It highlighted the necessity of communication among the team and stakeholders. It strives to curate an influential and well-working group that understands each other's strengths and weaknesses and those of the project. This assists a group in meeting the client's needs, boosting a company's bottom line. The rapid development of software is a competitive market. Teams and companies that can deliver are the currency of reputation.

Communication was a huge factor that was presented in this SNHU project example. The actual coding of an application is a skill that requires education and experience. The ability to communicate effectively may be lacking in that coding education and experience. A communication breakdown at any stage can break a project, at the very least, prevent a timely delivery and cause a slew of rework. That aspect of the Scrum-agile approach was not lost in this project. A con that can only be rectified by experience is communication's many forms. Working with difficult people and working with those who think they communicate appropriately but not within the needs of the Scrum Team are examples of experience-driven lessons that help the team work through those barriers.

While little of this project focused on the actual coding, it presented a methodology I had never experienced or heard of. I have worked with some incredible groups in the past, and the sad fact is that most times, as a group refines its cohesiveness, members of that group get reassigned, change jobs, or leave, leaving the rest of the group to learn how to be effective almost all over again. Much like the sales pitch of the waterfall method, it looks good on paper. Perhaps the lesson from that opinion is that keeping an open mind, communicating, and staying flexible is more core to teamwork fostered within the Scrum-agile method rather than producing applications that meet a client's needs.

The Scrum-agile method was the correct approach for the SNHU Travel development project. It started with an idea for an application that, for the client, did not exist and wrapped the production of the application in guidelines on how to conduct that creation. Not discussed over the last few weeks was backend programming. One assumes the database containing all the destinations to be presented on the SNHU Travel site must be parsed and integrated with the front end. Much of that work, the User Stories, testing, and quality control would require a lot of collaboration and teamwork to be presented to the client on an appealing front end. It would need guidelines for that team to operate under to stay focused and to emphasize communication to ensure the team was always working forward.

Works Cited

Charles G. Cobb. (2015). *The Project Manager's Guide to Mastering Agile : Principles and Practices for an Adaptive Approach*. Wiley.

CS 250 Software Development Lifecycle *Module 5: Product Owner and Scrum-agile Team* <http://snhu-media.snhu.edu/files/course_repository/undergraduate/cs/cs250/storyline/mod5/story_html5.html>

Overeem, Barry (July 2016) *Characteristics of a Great Scrum Team.* Scrum.org Whitepapers <https://scrumorg-website-prod.s3.amazonaws.com/drupal/2016-08/Characteristics%20of%20a%20Great%20Scrum%20Team.pdf>

Schwaber K., & Sutherland J. (2020) *The Scrum Guide* [*https://Scrumguides.org/docs/Scrumguide/v2020/2020-Scrum-Guide-US.pdf#zoom=100*](https://scrumguides.org/docs/scrumguide/v2020/2020-Scrum-Guide-US.pdf#zoom=100)