Sprint Review In Retrospect

Jacob Lawrence Walker

Southern New Hampshire University

CS-250-H2954: Software Development Lifecycle

Professor Tammy Morrison

December 8th of 2023

Sprint Review in Retrospect

This project allowed the simulation of several well-defined roles presented by the Agile framework such as the Scrum Master, Product Owner, Developer and Tester. Learning how each member of the team works together through sprints to finish bits of presentable product for stakeholders and clients is invaluable when comparing it to the traditional waterfall method. Coming from no project management experience it’s very clear that the agile framework is very scalable, implementable, efficient, productive and adaptive. Not only was each role experienced in this course, but also a shift in direction of the project was simulated to show how well Agile can handle change from top to bottom (or more accurately from left to right).

Each team member type contributed to the success of the project in multiple ways. Arguably open communication between all parties involved, sprint planning, proper team management (on multiple levels), and the willingness to adapt to change on the fly were some of the more important aspects that allowed this. However, each team member had a specific role they filled with responsibilities that enabled the continued success of this project.

From the top the Product Owner was possibly the most influential and a critical part of this process. An extremely important part of the success of a project is not only meeting the clients’ expectations, but also the end-users’ expectations. What the Product Owner provided was the ability for the client to receive a product of much higher quality than the expected result. Meeting with end-users enhanced the overall value of the client’s vision, seeing as the end-users would be the customers of the client. In the SNHU Travel project the Product Owner was able to establish that the end-users wanted search options related to pricing, types of destinations, hot deals, and tailored results. This was then converted into digestible user-stories presented to the team, which was further broken down by testers and developers into sprint related parts with fail/pass conditions and functionality. Had there been no Product Owner to act as the user representative then the likelihood that end-user expectations would not have been met would be much higher, leading to a dissatisfied client in the long run. Just as well the organization, prioritization, and transparency of goals and tasks are vital jobs the Product Owner performed that provided much needed insight and structure for the rest of the team to be successful.

Another important role in the Agile-Scrum system is the Scrum Master, which is comparable to a project manager without the traditional absolute authority one would have over a project. The Scrum Master essentially manages the team performance, relationships, and general environment. For example, the arranging of a short meeting in which each team member goes over what they have completed, will complete, and any issues or assistance that’s needed would be directed under the Scrum Master. Ensuring the team dynamics are stable, productive, on topic and positive are just a few of this members’ required tasks. During team meetings the Scrum Master would likely be the first to share this information (completed tasks, what to work on, issues) to set the stage for the other team members, as well as prevent tangents and collect information to be addressed at a future time. This was how this role was performed during the scrum meetings for the SNHU Travel Project. Simplified as much as possible, this member created a productive, efficient and positive environment for the other team members.

Developers are but just another team member critical to the success of this SNHU Travel Project. Developers work closely with Testers (defined next) to ensure the requirements (pass/fail) for user-stories are met. Not only this, but as the Developer it’s important to create plans for Sprints, making sure to work within the bounds of the timeframe to create a releasable and presentable portion of the project. As a developer it was important to focus on the Sprint at hand as well as be prepared to switch direction in the project when this occurred, as it did. Other important tasks of the Developer included the forecasting of how much work it would take to do each task, just as well be prepared to work with the inaccuracies of predicting how many hours it takes to complete work.

The final role which was studied was that of the Tester. If one was unfamiliar with what a Tester could be compared with, Quality Assurance would be it. The Tester takes user-stories that are provided by the Product Owner and creates requirements with pass or fail criteria for the Developers to meet. For example, one user-story in this project was an end-user’s desire to set pricing min/max for vacation packages. The tester took this user story and created a set of criteria to meet this expectation. This submission included the ability for an end user to select the price range on the web page and for the web page to display results within this range.

By utilizing user-stories a higher quality product can be produced. The Product Owner was able to relay what end-users wanted to experience from the client’s vision to the scrum team in a workable format. From there the Developers were able to develop Sprints or chunks of work to be completed and the testers design criteria for passing or failing these expectations. Had the user-stories not been included the client’s vision may have been met, but user dissatisfaction may have occurred in the long run. By communicating with the end-user’s real expectations could be laid out and that would provide the groundwork for truly successful software. Perhaps without the end-user feedback and suggestions some features would have been implemented, however much of it may have been overlooked. Not only this, but when the client decided to change the direction of the project towards a detox-based program features that end-users would want had become unclear. Meeting with end-users that would seek this genre of vacation is critical as many unfamiliar individuals may not have any idea as to what features users seeking detox packages would prefer to see in the final product.

Due to the focus on the sprint at hand the change in direction was not as detrimental as it would have been had the project been under a waterfall approach. The waterfall approach includes the maximum amount of planning ahead of the development, quality and release cycle. By accepting a system where planning is kept to the minimum required and keeping focus on these Sprints, the change in direction caused minimal damage to the project. If it was decided to scrap the current sprint, all was not lost because this was only a chunk. Decisions could be made to conserve parts of the project that had already been completed, while others be put on the backburner. Meetings with new end-users could provide similar expectations of the finalized product (vacation types, price ranges etc). It’s true that the Product Owner would need to reorganize, reprioritize, and create new objectives, but the Agile-Scrum system is designed to withstand this. Again, had this been ran under the Waterfall method everything would have to stop and the whole project would need to be planned out from start to finish, literally delaying anything to be presentable or releasable for a substantial amount of time.

Effective communication is most important in the framework. Typically in the Waterfall method there is a disconnect between many members of the team as each individual works on their own set of objectives within the allotted time. Stakeholders and the clients relay what they desire to see in the finished product and have little say or communication with the team until the final product is released. In the Agile-Scrum framework communication is absolutely critical between nearly everyone. During major shifts in objectives team members need to be on the same page so as to not complete unnecessary work or end up working on tasks that have moved to the bottom of the backlog. It’s especially important for developers and testers to be on the same page to prevent misunderstandings in what is being tested or developed. Just as well the team may have important questions the Product Owner can only answer before work can continue such as if the current Sprint is to be finished to scrapped, what is top priority, and what is salvageable. An example of quick communication via email that occurred between the developer, Product Owner and tester when the client expressed the desire change direction towards a detox/rehab based vacation search is provided below (from the stance of the developer). This ensures that everyone can make progress towards the new goals, and unnecessary work or confusion is avoided.

Dear Christy and Brian,

Seeing as we are shifting direction of the SNHU Travel Project towards a detoxification/rehabilitation based vacation search engine we need to go over any new requirements and user stories. My inquires include the following:

* What will the reprioritized list of objectives look like?
* What new features does our client expect?
* Does the client still want our previously collected feature list?
* Updated pass/fail requirements and new user stories

It would be most appreciated if we could set up a time and place to discuss these changes as I am sure there are other questions from the testers that need to be answered as well.

Regards,

Jacob Lawrence Walker

Another example of effective communication from the stance of the Tester to the Product owner following the customer focus meeting is provided below. This quick email encourages a mutual understanding of what exactly is to be expected, preventing unnecessary work from consuming valuable time and ensuring what needs to be done is done.

Dear Christy,

Your user stories are greatly appreciated and will be put to good use. As I’ve been working on the test cases I’ve come across the need for some clarifications

User Story #2: Search Preferences

* What specific attributes should be searchable?
* Should this be available on every page or a specific page?

User Story #2: Types of Vacation

* What are the main types of vacations?
* Should this be a search bar on every page?
* Should this be inherent to every search?

Thanks ahead of time for the clarifications!

Regards,

Jacob Lawrence Walker

The Agile-Scrum framework does encourage face-to-face communication for maximum understanding, however quick communication is essential to maintain the ability to have unscheduled meetings in order to proceed in many cases. Staying on the same page is again, critical to success.

A few tools from the Agile framework that are critical to the success of a project include test casing, daily scrum meetings, end-user meetings, sprints, consistent collaboration, and the team structure. Test cases are developed by the Testers and measured against the developer’s submission to ensure that expectations of the client and end-users are met, which promotes the true long term success of the project. Daily scrum meetings ensure everyone is on the same page, which ensures a higher level of productivity and efficiency. Meetings between end-users and the Product Owner allow for features that may have been overlooked by the client to be established, again, allowing for the true long-term success of a project. Sprints enable short runs to meet goals that are presentable to the client as well as boost morale among the developers. It’s important that tangible success is measurable by everyone. Consistent collaboration outside the standard scrum meetings yet again keeps everyone on the same page, increasing productivity and allowing for quick adaptation to changes that can take place at any given moment. The team structure of Product Owner, Scrum Master, Developer and Tester is very straightforward and even adaptable in different scenarios. In fact, team members can even switch positions as the project calls for or as new projects arrive.

The apparent effectiveness of the scrum-agile approach for the SNHU Travel project can be seen in the completed portions of work, ease of shifting directions when the client called for it, and in how all members of the team were consistently on the same page. Though the simulation was spread over 8 weeks, this was representative of a much shorter timeframe. From personal experience this does seem to be the most effective approach to this project, at least when in comparison to the Waterfall method provided by texts. If a waterfall method had been employed, several problems may have occurred. For example, all the work that had been completed may have been scrapped (aside from perhaps the developers attempting to salvage bits of code to reuse), the entire project would have had to start over (planning, standards for the finalized project, and in general redefining the whole project). The pros to using this agile approach was that despite the project making what appeared to be a significant shift, it didn’t necessarily negatively affect the development process. Everyone being in regular communication with one another paired with the inevitable instance of change caused minimal damage to the project and employee confidence. Due to the lack of personal experience with the waterfall method, however, it can only be assumed from the readings in this class that the agile-scrum framework was the best choice in the production of this software. This whole experience with this class has been extremely beneficial in analyzing projects outside the scope of software development as well, it has been very enlightening and should be employed at least in conjunction with the Waterfall method due to its highly adaptable framework.