# CS 250 Software Development Lifecycle

Eric Vining

2/18/2022

**Sprint Review and Retrospective**

It’s that time of the development cycle to review the progress of the SNHU Travel Project and report back to the company. As my responsibility as scrum master entails, I will report back information regarding the contributions from the various roles on the team during our current project cycle.

To begin, I’d like to highlight out Product Owner and commend their work on being ‘the bridge between our communication gap’. The Product Owner kept in constant contact with the client and the users, making ample notes and developing the product backlog based off the requirements and feedback given to them. During our scrum meetings, the team held regular discussions with the Product Owner to ensure that the product backlog was in proper order, with changes made as needed, allowing the team to concentrate on the most important tasks at the time. Being the bridge between the communication gap, our Product Owner kept the team on track constantly, making sure that the SNHU Travel project met the needs of the client, even with major design shifts throughout the development process. Our team made suggestions to the Product Owner about some of the design choices and we were given proper feedback from the users to meet common ground on the user interface that the team was concerned about becoming too cluttered and inefficient. We can thank the Product Owner here, as they were very open to suggestions and instantly communicated our concerns to the users.

The development team met daily during the course of our development cycle and keeping the meetings on track was my goal. Largely, the focus on development was maintained and any inside or outside interferences, I chose to take head on and squash them to the best of my ability. Our team chemistry was some of the best I’ve seen, so there weren’t many problems faced that halted our development. The testers and developers kept in constant communication, often working in pairs. This paired-programming approach made sure that our sprints were more focused on ironing out a quality product, and a slightly slower approach, rather than risking sending something out that was done quickly but needed high maintenance and refining.

The biggest advantage of taking an agile approach to this project came when completing user stories. Without the agile approach, and focusing on a strict, outlined plan, the project may have hit some issues when the users requested and change in the design of the user interface. Being able to jump around and re-prioritize the product backlog was instrumental in navigating through the hiccups. If we had taken a stricter approach and designed everything with a specific design, it would have felt more like a rug was pulled from under us when the requested shift occurred. Agile allowed us to keep the door open to changes, with a loosely designed interface, and we were able to shift focus on the changes when they occurred. The changes were smooth and required less re-design because of this.

To continue with the discussion of these particular hiccups, the Product Owner reported back to us with interface changes midway through the design of the project. The users requested a more PowerPoint themed presentation for the top destinations list, different from the original sorted-list design. I, and the development team had emailed the Product Owner to express our concerns with this design and offered some recommendations to bring back to the users. This enabled us to flesh out a better design in the end and allowed for a communication avenue for meeting middle grounds on the user interface design. The open communication between our team was a major motivating factor, as the team began to feel that their creative ideas and input were being valued. The team felt more attached to the project, feeling as though they had some stake in the design.

This line of open communication and innovative approach was one of the contributing factors to the success of the team because the team as a whole grew closer because of the growing attachment to the project. I believe this innovative approach, where we left the door open for recommendations from the team, was a major motivating factor and helped the team grow closer not only amongst themselves, but to the Product Owner and even the users as well. Without this approach, the work on this project may have felt like an assembly line. No one wants to work like an assembly line robot.

**Assessment of the Agile Approach**

Pros –

* Open design
* Team input
* Shifting goals
* Team focused
* Task oriented
* Flow state

Cons –

* Laissez Faire
* Culture shock
* Pantsing

The open design approach really allowed the team to meet specific problems and changes head-on, instead of having to re-work things that were considered finished in the early product, due to changes in the requirements. This approach allowed for a more team focused approach, while the meetings also allowed the team to voice their concerns over some of the design choices and any of the backlog issues. An agile approach allowed for our goal shifts to feel less bumpy because our goals were never set in stone in the first place. Being open to an ever-changing design is a mindset that allows for our goals to be more loosely solidified and open to change. While being task oriented might scream waterfall design, the meaning here is that the team felt more comfortable focusing on tasks that had been broken down to a specific purpose instead of focusing on a major task as a whole. Lastly, I want to briefly express the importance of the ‘flow state’ and how our sprints got the team ‘feeling the flow’. Having short bursts of focused, small tasks, allowed the team to ‘get in the zone’ as you might call it, which made workflow much smoother. Focusing on major task, again, could be a detriment here, as the flow state is typically in short bursts. Spending a long time of a specific major task could’ve disrupted the workflow of our team to the point of mental fatigue.

Turning to the cons, I would like to point out that the agile approach is a major culture shift. Going from a task oriented, macro (or big picture) approach, that we’ve grown accustomed to, and changing the team’s mindset to be more micro-focused is a major challenge for any team. This culture shift process needs to be taken slow and you cannot force an entirely new system on people without facing backlash. The con here is that the major shift to agile could potentially cause some internal turmoil if approached haphazardly. Another con that has to be considered is the Laissez Faire approach to agile development, as it is very focused on getting content pushed out the door, choosing to make fixes as issues come along. The fear here is that this will lead to a too lenient approach to design, that we are already seeing in industries, where the idea is to release now and fix later. To offset this, our goal was to lean heavily on pair-programming and focus on open communication. This allowed our developers and testers to communicate any concerns with the coding and fix them immediately instead of waiting for a later date. The last con I want to talk about is very similar. ‘Pantsing’ is a term used by writers, which basically means ‘flying by the seat of your pants’. The term is used when writers write without a plan, outline, or goal, allowing the story to take them where it takes them. While this can lead to some amazing stories, it’s often the exception and not the rule. The fear of pantsing in terms of agile is that a less plan oriented approach might lead to a project falling off the rails because there aren’t enough guidelines. To counteract this from happening, our Product Owner needed to be on top of everything and keeping in constant contact with us and the users. While we never had a strict plan, we always kept up-to-date with what was expected of us.

**Conclusion**

Ultimately, moving forward with some of the agile principles is the best way forward. The team oriented approach and the focus on getting input from everyone to change the plan moving forward helps the team grow and become more attached to each project. However, I think there needs to be a system of balances in place to make sure that this approach doesn’t become to lax and lead to bad practices. If our company can maintain standards, and find a happy medium between using an agile approach and taking ideas from other design approaches, while maintaining a proper level of planning, we could have an efficient workflow moving forward.