**ChadaTech for SNHU Travel – Agile Methodology Adoption**

**Sprint Review and Retrospective**

Craig O’Loughlin

Computer Science, Southern New Hampshire University

CS250: Software Development Lifecycle

Professor Rangitsch

December 11, 2021

**Introduction to the project:**

Our development team at ChadaTech was tasked with pioneering the company’s transition to an agile methodology. The focus of this paper is on providing a retrospective of our first sprint, where we will cover what practices were beneficial to development along with lessons learned before planning and moving forward into our next sprint. We will take particular interest in cases where agile methods were applied successfully towards problem solving and project outcome.

Our team going into first sprint cycle consisted of a single **developer**, a single **tester**, a **product owner**, and a **Scrum master**. The goal for this sprint was to deliver the first working phase of software for our client SNHU Travel. The basic requirement to be fulfilled was an application display consisting of “top five travel destinations” that can be viewed by the end user. Java AWT was chosen for this first iteration as it is a safe, straightforward API for GUI development that is platform independent (Javatpoint, 2011-2021).

**How the different team roles contributed to success:**

Our product owner set the concrete requirements to be delivered by our development team. They were responsible for working with the customer and stakeholders and understanding what our product must do. Traditionally in a Scrum-agile environment these customer requirements take the form of ‘user-stories’ each defining a particular use case for the end user and a solid deliverable. This worked particularly well for our project as these user-stories formed small, workable pieces of the project that directly met customer requirements. It ensured we were always producing something of value to our customer.

The Scrum master of our project took the lead in adoption of agile practices among the entire team. The goal is to encourage adoption on an individual basis. A team charter was created by our Scrum master at the beginning of our project which set the tone of our agile changeover. By working within the guidelines of the team charter, we had a solid foundation in which to begin our agile methodology shift. Our Scrum master also worked to organize and set the example at our daily Scrum meetings, and this benefited us by enhancing communication among our team and improving our ability to track progress.

The software tester during this sprint cycle worked directly with our product owner to ensure that what we develop meets exactly the requirements from our user stories. By having concrete test cases, we were able to utilize ‘test-driven’ development in order to develop our software to meet the requirements. This worked well to simplify development and eliminate superfluous effort towards something not required by the customer.

Finally, the developer of our team kept in constant contact with all members during this cycle, as understanding the requirements from both the product owner and the tester was vital to a successful outcome. This open communication was necessary in order to be flexible enough to work in an agile environment and deliver a product that meets requirements. They were also able to provide direct feedback on the software itself to the rest of the team from the coding perspective.

**How our Scrum-agile team completed user stories – an example:**

The highest priority user story from our backlog, and the main story developed in this sprint, was: “*As an end user, I want to click a link to view the top five destinations list, so that I can see the most popular locations for travel to educate myself on the best places to go.*”. This requirement was first defined by our product owner, who, after a meeting with our company and with the customer, was able to distill requirements into an actionable user story. The user story was then given a priority, an estimated size (in this case, we used small / medium / large), and the acceptance criteria by which the user story would be considered completed. Our tester then was able to create specific test cases for this user story, in which for each an ‘input’ is defined as an interaction with the software, and ‘expected result’ defines the behavior the software must perform. These test cases are then used to guide the efforts of our team’s developer. The story would be considered complete when it passes all test cases.

**Handling changes in project requirements:**

During the project sprint, we were tasked with changing the output format to a slideshow instead of a one-page list, as well as focusing on a different type of destination for SNHU Travel: Health and Wellness.

One of the most important principles toward success here was open and quick communication. Our agile team is set up towards being in continuous contact with each other throughout the project sprint. We have our product owner available at any step in the project, who is constantly working to define and understand requirements and any changes therein. By having such a system, our team was able to quickly identify and understand the change.

A second principle towards success here involves our iterative development. Our tester, working along with the product owner, was able to revise test cases instead of starting from scratch. Our developer was able to rewrite only pertinent sections of code to meet the new cases. We attribute this reactive ability to the agile methodology of working on small requirements, which means we don’t get ‘too far’ into development to where changes are difficult to implement.

We identify an area for improvement here, however, towards being sure to set requirements during sprint planning, and striving to maintain those requirements at least through a cycle. Not doing so will lead to extra or redundant work. Thanks to our agile setup we have minimized the amount of work required to meet a mid-sprint change, but improvement opportunities still exist.

**Effective communication – more project examples:**

Communication channels opened at the very start of our project with the team charter, from which we were able to discuss expectations and roles with each other. Included in this was value/principle #2: “*Communication is crucial for success; we will make every effort towards open and transparent communication throughout*.”. We continued this open communication through our sprint planning meeting, then continuously with our daily scrum meetings.

Face-to-face communication is highly valued in any agile team, however we were also able to utilize email communication where needed. For example, the developer of our team was able to keep in contact regarding specifics of project requirements as in the following email excerpt:

*Hey [Product Owner], [Tester],*

*I would like to go over some requirements about the new slide show format of our project. Can I get a clear definition of ‘health and wellness’ with some examples, in order to make sure I am including the proper information?*

*Also, I noticed that the sample slideshow presented has blue background with black text, making it more difficult than necessary to read. I haven’t changed anything there yet but would like your opinion on it.*

We felt that using a combination of communication methods is effective in making sure the entire team understands our project status and what must be done next.

**How organizational tools and Scrum-agile principles contributed:**

Agile methods are well explored and documented among many development teams. Therefore, we were able to utilize a variety of templates for things such as our agile charter, user stories, and test case definitions. We were also able to explore collaboration tools during this project sprint and evaluate their usefulness to towards managing communication channels. We identified Jira software as a possible tool that can be used to great effect in future sprints for coordinating team efforts in one single, easy to access space.

All 12 of the principles included in the Agile Manifesto (Agile Alliance, 2021) are also applicable for our agile methodology shift. Among these we particular appreciate 7: *Working software is the primary measure of progress* and 4: *Business people and developers must work together daily throughout the project.* These principles in particular guided our efforts during this sprint and stand out as major advantages to this methodology.

**Conclusion – effectiveness of agile in this approach:**

Many attributes of an agile methodology contributed to the success of this project:

* Our team being able to work directly with the customer through our product owner contributes to the quality and value of our product in our customer’s eyes.
* Our team developing in short increments (sprints) allows us to adapt and assess our product dynamically.
* Our team is built around and primed for open communication with each other which contributes to project success.

However, some potential limitations of using an agile approach still exist. This includes:

* There is difficulty in company-wide adoption of what may be a much different approach to development than what we are previously used to in waterfall project planning.
* Agile is not one size fits all, and there are times when requirements are clearly defined and must remain clearly defined throughout.

In general, a hybrid approach can be used as needed for any given project. For this particular project, however, we feel that a Scrum-agile approach well-suits the requirements and enables our team to develop high quality software more efficiently and with a better outcome for the customer.

**References:**

Javatpoint (2011-2021) *Java AWT tutorial.* <https://www.javatpoint.com/java-awt>

Agile Alliance (accessed 2021 Dec) *12 principles behind the agile manifesto.* <https://www.agilealliance.org/agile101/12-principles-behind-the-agile-manifesto/>