**Final Project**

Evan Nagy

Southern New Hampshire University

CS-250 Software Development Lifecycle

02/22/25

The Scrum Master duty is to make sure that the scrum team stays on track. It is similar to a team lead. A team leader knows everything that pertains to debugging problems in his

department and keeps things running smoothly. The Scrum Master adheres to industrial

principles and practices. He acts as a conduit of the product owner. He collaborates with him

and makes sure that all requirements are met and retrieves any new ones. After meeting with

the Product Owner, he will have a scrum meeting with the rest of his team. He will address the

new requirements, gather questions and concerns and collaborate with his team members to see

if they have any new ideas.

The Product Owner`s responsibility is to not only collaborate with the Scrum Master. It is to

collaborate with clients, manage the product backlog, create user stories, develop and organize

the backlog if additional information appears. An example from my time in this class, during

our SNHU Travel Software assignment. We had to meet with our clients, gather their

requirements, create user stories and a backlog, and update our stories when someone asked for

more details. For instance, one of the requested items to be expanded upon was the SNHU app.

It was requested we break it down into smaller stories since the scope seemed too broad. Later,

the client switched the scope of our project to include destinations that catered towards detox

travels. After which, the product owner would have to re-evaluate the product backlog and

rearrange the order of the backlog in order of importance. Afterwards, the Product owner will

create or modify User Stories to help understand the reason behind these features. This is in

hope that it will provide better context to these features and encourage the developers to suggest

new methods of getting the same result.

After the Product Owner is done, it is up to the Scrum Master to plan sprints for the

development team to work on. The goal of these sprints is to create bite sized work that the

team can finish in 1-4 weeks. This is the main focus of the Agile Process. This is due to Agile`s

ability to take in feedback and create new sprints based on new information. Just as we saw in

our SNHU Travel Software. We can create new sprints to modify existing code based on our

new requirement of detox vacations. We do not have to redesign our entire software system due

to 1 new requirement.

The Scrum-Agile approach to the Software Development Life Cycle (SDLC) makes creating

stories simpler. We can break a large application into smaller more manageable sprints

that we can document the completion of. We can also set the groundwork of our application as

higher priority than ancillary features. Thus, they will get done by the first available man rather

than having to delegate it. As noted previously, Agile is flexible. If any new features come to

our attention, Then we can simply add them to the backlog and sort them based on priority. We

can thus adapt to expanding requirements rather than trying to plan them all out.

In a previous lesson, we had an interruption of our client wanting to add detox destinations to

our requirements. There was some worry about how that was going to affect the project.

However, with some explaining, It came down to modifying a few lines of code and moving on.

In other words, due to the Agile process, interruptions are handled easily and without much

worry.

Previously, We have had meetings with the Scrum team about how we wanted our

application to look ahead of time. We went around the team asking each to give their input until

someone that was late showed up. One of the members started to cause contention within the

discussion due to the late member. So, both were taken out of the discussion so that the meeting

could continue, and the leader could manage the controversy after the meeting. This incident

showed that not only provided an example of being able to communicate within a team

environment. It also showed conflict management ability by being able to move agitators out of

the room and address their concerns once the meeting was over.

One of the organizational tools at my disposal are scrum events. They allow me to make sure

everyone is on task and focused for the days work. Sprint Review are also great. They allow me

to show the progress being made by our team. This allows transparency for all those involved

and allows stakeholders to see the actual progress being made. Sprint Retrospectives allow a

Scrum team to understand how everyone and everything worked together and look for ways to

improve performance for next time, “assuming there is a need”.

The major pros of the Agile process is the ability to break down large tasks. From personal

experience, the tasks that I choose for myself often seem daunting. However, a simple exercise

of taking one issue or feature and deconstructing it into its smallest actionable step can cause a

problem to become a non-issue. However, A con of the Agile process is it doesn't lead to well

architected code. Because you didn't plan out your steps from beginning to end. You are left

with small sprints of unorganized code. There are no abstract classes and there is likely little

inheritance going on. Thus, you may have a lot of repeated code that could be recompiled and

slashed from the project.