**Final Project: Retrospective of Agile Development**

**CS-250 – Southern New Hampshire University**

**by:**

**Brandon Rickman**

**Table of Contents**

|  |  |
| --- | --- |
| **Product Owner** | 3-5 |
| **SCRUM Master** | 6-8 |
| **Tester** | 9-10 |
| **Developer** | 11-13 |
| **Team Collaboration Tools** | 14-16 |
| **Appendix A: SCRUM Team Charter** | 17 |
| **Appendix B: SCRUM Daily Stand-Up** | 18 |
| **Appendix C: Backlog User Story Example** | 19 |
| **Appendix D: Simple List View Java Program** | 20 |
| **Appendix E: Test Case Example** | 21 |
| **Appendix F: Developer to Tester Questions Email** | 22 |
| **Appendix G: Developer Code Example 1** | 23 |
| **Appendix H: Developer Code Example 2** | 24 |
| **Appendix I: Revised Program Slide Show** | 25-26 |
| **References** | 27 |

The Agile Software Development Process is versatile, scalable, responsive, quick, and allows for great team collaboration over its elder, the Waterfall Process. As we end of our Software Development Lifecycle course, we look back on each role we took in the Agile Process. Although, Agile is not hard in concept, it carries a great deal of implementation throughout a team. As each member becomes more comfortable with the Agile Process, the team will inevitably grow stronger, faster, and well-rounded in its skills.

**Product Owner**

The Product Owner can be viewed as the product manager. This person is responsible to their company for interacting with the client/customer and producing a product that is on-time, in budget, and works exactly as intended. The Product Owner is the head of the Team and the face of the company.

A Product Owner must meet with the customer to sort out the specifics of the product and what the customer would like to be seen from the product. This meeting with the client will help the Product Owner create and sort the user stories in the backlog. The Product Owner also must conduct any necessary focus groups to ensure their team has all the information to complete any necessary features of the product.

During these focus group meetings, the Product Owner will need to engage the users of the customer and interact with them to get their honest feedback of questions posed by the Team. Encourage the users to be honest about what like and don’t like. It is always easy for us to tell others what we like, however, people tend to shy away from causing disruption when it comes to saying what we don’t like. Make the user feel empowered to assist the Team in creating something the user would be proud to use. Sometimes the Product Owner may have to guide the answers from the users. In the video presented we were informed by all three users about their preference of vacation, however, each had already agreed to having a filter for type of travel, would be handy. This made for redundant information. To avoid this repeat the main concern or task in an ensuring way that is of a top priority for the Team.

These users' stories, created by the Product Owner with the help of the users, will shape the product’s direction, and vision. Each user story will add a feature or combine other features to the product. The Product Owner needs to know how to set these stories for both the customer to understand and the Team to understand. The phrase, “As a [user] I want to [some task or feature] so that I can [what the feature does].” helps to narrow down these stories to a thesis. This product statement makes the user story have purpose, a goal, and title. The Team then takes one of these stories and creates the feature as necessary. With the purpose given to the story, the Team then takes on that purpose, ultimately ending in the product having a new feature.

The focus meeting is essential for understanding the product concept and presenting new ideas, that maybe the customer had not even thought of. This would directly affect the customers bottom dollar, because in business without happy customers you do not have money. and without money you do not have a business. The Product Owner needs to be able to see where they can step in and assist the customer in acquiring new or loyal customer base. This can be accomplished through the customer focus group setting. As mentioned earlier, I did feel this video had repeated information and only allowed for two main features: a personalized list and filtering options. I added a user profile with preferences and previous travelled locations for enhanced user experience and list creation. I also added, software created travel tracking, to generate reports and statistics of user travel selections, preference, etc. This will help the customer in the future to gain better deals within the travel community at destination locations. This could also help acquire future users that were not available at the moment due to the limitations of package selections.

The position of the Product Owner is challenging, as the customer’s expectations of the product, and the Teams ability to complete the expectations of the customer clash. The Product Owner needs to have a strong sense of self and be able to understand all levels of the product production. The Product Owner also needs to be a people person, charismatic, and knows when to say something can or cannot be done. This position is a leadership role and should not be taken lightly. Ultimately the profit margin will fall onto the Product Owner’s shoulders if the Team cannot produce, and the Company loses the customer.

**SCRUM Master**

This week we learned about all of the roles of the SCRUM team. From the client to the product owner, to the scrum master, developer, tester, etc. After reviewing the best traits of each of these roles, we were instructed to take up the role of the Scrum Master.

The Scrum Master is a peer-to-peer leader role. This person is a person of knowledge and leadership. However, they cannot be afraid to set in and help pick up the legwork if something in the project needs it. The Scrum Master has to be a mentor, a manager, a multitasker, a mediator, a scheduler, and many, many more tasks. This person needs to be well organized and understands all aspects of the project. The are a leader that has no authority. They must know when to congratulate and when to push each team member. They help to ensure obstacles are out of the way for the Team Members to complete their respective task, and ensure the project is completed as planned.

As I sat down to take on this role, I had no clue what I was stepping off into. After all the Agile Scrum Team concept is new to me, as I have not worked directly in software production. I worked to complete the Team Charter, that laid out what the projects objective is, the vision of the team, the goal of the client, what team member would be taking on what role. What the communication conditions should be. Basically, the Scrum Master laid out how this specific team on this specific project was going to work together to complete the mission statement. This doesn’t seem too hard, until you are the one that must do it.

To conduct a sprint planning session, I think I would take the approach of a round-robin style table top. Starting with what we know, where we want to go, how can we get there, are there any know pitfalls along the way, can they be overcome, how long do we need to complete the first sprint. These questions would spark other ideas and questions from all areas of the team. Which would lead the team into the backlog planning and ultimately the backlog grooming (ordering of the stories).

To conduct the backlog grooming the must ask themselves what is needed and what is wanted. By keep these two questions in mind you can narrow down the stories to what is needed to make the program do exactly what the Client needs it to do, versus what the developers creative minds what to make the program do. Ordering the backlog will take time, but in necessary to keep the team running smoothing and help the sprints produce better quality results.

During the sprint retrospective, we would discuss what worked and what did not work during the most recent sprint. Did we not have enough time to complete everything? Did we have too much time? Do we need more team members? Do we need to go back to the client and ask more questions or present additional options? By critiquing the sprint this way, we learn what did and did not work and how to proceed further as a more cohesive team.

As the Scrum Master, I completed a Daily Stand-Up Agenda, which simply asked for each team member to state: What they did yesterday, what they are doing today, and if there were any obstacles in their way. It was then broken down across columns for everyday of the week. This made it easy for the team to go around the table and answer these three questions quickly, write them down, and they would be posted for all team members to see. If there are any obstacles between the team members then the Daily Standup could help alleviate these obstacles and help to produce a more productive team and project. If not, the Scrum Master would set to work fixing the issues so the team members could stay focused on the project. The Daily Standup can remain quick (usually 15 minutes) and still have meaningful insight into the pulse of the team and the project.

The ceremonies of the Agile Scrum Team seem to be set up to ensure a quality product is produced and present in a timely manner, while pulling a team together as one to show case the team and not the individual. The Scrum Master is the hardest position I see in the Scrum Team. With the need to be a master of all things, a mentor/leader, a manager, etc. it places a lot of responsibility on one person. However, the right one person, who is organized, well versed in the project concepts and tools, and able to teach others, would flourish in this position. The Scrum Master’s tasks and skills could easily be transferred into a managerial position in someone’s career path, with the hope that the mentorship, team concepts follow.

**Tester**

A tester in the Agile Development lifecycle is required to ensure programming functionality is correct, methods and process run as intended, and the client’s expectations match with the programs output. An Agile Tester has many functions and roles during the entire project but helps to hone the team’s skills during each sprint by trying to break the developers code. A good tester coupled with a good development team will lead to quick idea corrections or adaptations, an overall better performing team, and a team that is completely versatile.

The best user story elements for a Tester to have is the true picture of what the result should be. A Tester doesn’t need to be concerned as much with the final look of the project, however, this does need to be checked. The Tester needs to be concerned about the final functionality of the project or user story. A good question for the user to ask is, “What should this story do for the user?” Not only does the Tester need to be skilled in programming, but also in data manipulation, writing out test cases in English terms for anyone to read. The Tester needs to be ready to question the Product Owner on specific details from the client’s point of view.

During this exercise I found that the user stories presented were missing how the story was to function. I created them as a webpage and I believe this is what the result is, but I could not be sure due to the lack of detail in the email or in having a previous worked user story. If we could’ve gone back to the user stories we created and expanded them into the tests it would have given us more information to work with and created this entire story line. With only the email to go off of and without a product backlog developed for this story line it created some confusion as to what was expected. Ultimately, I was able to understand the purpose for this lesson and develop tests to conceptualize this lesson.

As a tester I would gather more information through meetings or follow up emails with the Product Owner and the SCRUM Master. These two resources would help to ensure I had the client’s vision and purpose in mind. I would also follow up with the Developer(s) to ensure that the result I was getting were the same as they were expecting when they wrote the code. Sometimes when writing code, you tend to get lost in the repetition and forget that each story has is own purpose or vision. A product backlog of the entire project would be a great resource to have to clear up any of the confusion, or the ability to use the product backlog generated in module 3. I know that uniformity is the best answer to writing a lesson plan and teaching several different students, but the direction seemed a little askew of what was required as the user story. By going on the “Tester’s Email” only there was not enough information to complete a test without developing some of the story’s functionality ad lib.

It was refreshing to complete this task though with the template. The template made it easy to understand all the parts necessary to breakdown any user story and complete and conduct a test. I did like that the test were first written out in English form, and not just code tests. Although there are times that coded Unit Tests will need to be conducted on projects, with the written English form it helps to create a set of rules for the Unit Test and creates a way that others can quickly understand what the test does and if the user story did what it was supposed to do.

**Developer**

There are several names for the developer, i.e. software developer, computer programmer, software engineer, etc. In the Agile Development Playbook, the developer is defined as:

* Taking responsibility for estimating, planning, and managing all of his/her own tasks and reporting on progress.  This role is essentially what a project manager might do on a very small scale.
* Collaborating closely with all the other members of the team to take shared responsibility for the overall efforts that the team has committed to.  This role is also similar to what a project manager might do but rather than being done by a single person with the title of “Project Manager”, the responsibility is distributed among all members of the team.
* Taking responsibility for the quality of the software the developer produces.  Instead of turning over some code to a separate and independent group for testing, the team, as a whole, takes responsibility for the quality of the work they produce.  A developer may or may not do the testing himself/herself, but the key point is that the quality of the code is not someone else’s responsibility.
* Interacting with users as necessary to clarify requirements.  Developers will typically not be given a well-defined set of requirements.  More often, the developer will get some general user stories that are intended to be a “placeholder for conversation” and the developer will be expected to interact with the Product Owner and users as necessary to better define what is needed.  This is essentially equivalent to a Business Analyst role on a very small scale (ChuckC3, 2016).

During this module we were tasked to make a last-minute major change in the direction of the company, SNHU Travel. The company decided to jump into a new growing trend and offer detox/wellness vacation packages. As the developer, who had already spent several hours writing the program to function as required for the previous ideas and conversations, new questions arose. Do we scrap everything and start fresh? What is the client specifically wanting with this new idea? Fortunately, these questions were quickly answer by the product manager. The client liked the program so far and only wanted to change the vacation packages. Therefore, the developer had to make minimal changes to incorporate the new company vision.

On this lesson, after determining that the only thing that truly needed to be changed was the vacation packages themselves, no further inquires would be needed to the product owner or tester. However, if functional parts of the program needed to change, I would request the Tester incorporate a new list of tests for the new functions. This would help the developer think of ways to implement the functions or add additional features to the function that may not have been thought of.

To ensure that I had the appropriate information to move forward, I would have the product owner supply me with a new or revised user story with the new functions in it. This would help when the developer or tester go to start their parts of the project, ensuring all the new aspects were conveyed. Through the Agile process there are opportunities to check and recheck areas even inside the sprint. With multiple eyes looking from different perspectives throughout the sprint and the project, each functionality is sure to have potential ideas thrown into the mix ultimately to achieve the goal of a finished well-shaped program. However, these different perspectives allow for new ideas along the way the way that one person may not have thought of. This also allows for failure to happen and not hinder production. A team that shows can fail fast but recover to a working product is a team that is on the leading edge.

As a developer the functioning program relies on your code and your knowledge to make the program work as intended. However, the Agile method takes a lot of the burden off the developer themselves to come up with all the ideas, tests, and stories to allow for a functioning product to be presented to the client in a faster time frame, while still allowing for expansion and thought-provoking ideas to be generated, whether they pass or fail.

**Team Collaboration**

The Agile Product Management Framework sets up the structure of the Software Development Lifecycle (SDLC). It also defines roles for each member in the team and their associated responsibilities. Agile also gives you suggestions for what abilities you will need in specific tools during your SDLC with Agile. It then becomes your choice to pick tools that work well with your team and the needs of your client.

When working with Agile several pre-formatted lists are needed to be created and possibly used multiple times. This sets up the perfect scenario for a template to be used. VersionOne is an all-encompassing software for Agile. It provides sections and templates for Agile documents with include daily stand-up, burndown charts, backlog, etc. If I had access to this program it would definitely be used to manage each Agile project. In our class though, it was shown that even on a small or non-existing budget team, simple Office Excel spreadsheets could be utilized for various required documents in Agile. We will go over three of the main areas, Full Agile Process Lifecycle, Team and Customer Collaboration, and Agile Portfolio and Program Management.

Full Agile Process Lifecycle

The entire lifecycle of a project in Agile spans from Planning to Code Integration to Testing to Release. We need a tool that can encompass all of these areas with ease of use (themes) and with collaboration of multiple team members (no matter their physical location) in my mind.

Team and Customer Collaboration

This area needs to responsive. Although emailing is a great option is there anything better? Yes, a program like Slack, a instant chat and file/code sharing platform, can expedite the question and answer process. When a developer has a question or needs clarification, they can request an answer by simply posting in forum type setting or direct messaging the tester themselves. Slack can handle all sorts of data and code snippets, as well as any team size. Each Slack channel can be added for each project, allowing for independency with the same program. Other programs to use could be Amazon Web Service Code Commit or Code Pipeline, especially if the costumer is already prepared to take the product into the cloud. Integration in AWS for the Agile Team will help with release times, response rate, code updates, bugs/fixes, etc.

Agile Portfolio and Program Management

Another area that will benefit from instant connectivity to the team when starting new sprints, setting up meetings, modifying the backlog and user stories. The ability to access functions like bubble boards, burndown charts, and a central location to store all of this will be key to the quick implementation of any Agile Team’s tool.

These tools are the three main areas I see needed to be integrated the best. If these three areas are properly focused and connected, then all other tools added to the Agile Team’s arsenal will only enhance the team’s overall success and set this Agile Team above any others. If available a developer could be tasked to ensure all of this is up and running, with any API calls and code functionality, will ensure that if any problems do arise, they can quickly be dispatched. The set needs to also be in the same location where most of the work is done and held onto. This helps with latency of any file transfer, and the ability to see any unforeseen problems.

**Appendix A: Agile Team Charter**

# CS 250 Agile Project Charter Template

**SNHU Travel Trendy Vacation Packages Add-on**

|  |  |
| --- | --- |
|  |  |
| Business Case/Vision:*(Value to attain)* | Booking system add-on for trendy vacation packages. |
| Mission Statement:*(Result to accomplish)* | Create a web application for booking additional trendy vacation packages to be added into the Client’s current web application. |
| Project Team:*(Team members and roles)* | Amanda: Client Representative  Christy: Product Owner  Ron: Scrum Master  Brain: Tester  Nicky: Developer |
| Success Criteria: | Start date: January 20, 2019  Expected completion date: February 12, 2019  Final deliverable: February 24, 2019  Key project objectives: Develop a timely application for presentation to client. |
| Key Project Risks: | Design integration with Client’s current web app; Database integration with vacation packages. |
| Rules of Behavior:*(Values and principles)* | 1. All team members will follow the “Golden Rule” and treat each other with respect. 2. Constructive criticism is the best way to correct issues and improve each team member. All members will not become offended by this feedback. 3. Don’t just bring problems up, suggest solutions to fix the issue. 4. There is an “open door” policy, with all communication wanted and received. 5. We recognize and celebrate all team member accomplishments together. 6. What happens in our team stays in our team unless additional resources need to be brought in. 7. All personal cell phones and communication devices will remain off during any and all meetings. 8. We take responsibility for our actions and will accept any necessary punishment. 9. The speaker has the floor and all sidebar conversations will be ceased. 10. We will collaborate when possible and work on our individual tasks as necessary to remain on task, never be afraid to ask for help. |
| Communication Guidelines:*(Scrum ceremonies and rules)* | 1. All meetings will be held at 9 am. 2. All members are required to be at all meetings in person and on time, unless otherwise approved. 3. Task list will be updated before 9 am. 4. Meeting Scribe will be shared on a rotating basis, and all minutes will be distributed within 24 hours. 5. Any cancelled or additional meetings will have notifications made by Product Owner as soon as possible. |

**Appendix B: SCRUM Daily Stand-Up**

**A screenshot of a cell phone

Description automatically generated**

**Appendix C: Backlog User Story Example 1**

|  |  |
| --- | --- |
| **User Story Number:** | 2 |
| **User Story Name:** | My SNHU Travel Profile |
| **User Story Size:** | Large |
|  |  |
| **User Story Value Statement:** | As an End User, I want to select preferences or places that I have travelled so that I can have a list of available destination packages that is tailored to me. |
|  |  |
| **Acceptance Criteria:** | • User Profile Database • Form for personnel information • Preference check list for:                 • Previous visited locations/experiences                 • Type                 • Price                 • Length of stay |

**Appendix D: Simple List View Java Program**

A screenshot of a social media post

Description automatically generated

**Appendix E: Test Case Example**

|  |  |  |
| --- | --- | --- |
| **ID:** | 2 |  |
| **Name:** | Search List |  |
| **Owner:** | SNHU Travel |  |
| **Type:** | webpage |  |
| **Method:** | browser |  |
| **Work Product:** | User Story #2 - Filtered Search List |  |
| **Priority:** | medium |  |
|  |  |  |
| **Pre-Conditions** | User at SNHU Travel Main page |  |
| **Test Steps** | **Inputs** | **Expected Results** |
|  | 1. Click link "Search" | Opens new browser window/tab with title "Search List" and Table appears with Column Headers "Destination", "Travel Type", "Price", etc. Search query at the top of the page for sorting list. |
|  | 2. Adjust slider for price range | displays user defined price for min and max. |
|  | 3. Click drop down list or enters location for destination | Text indicator prompts for input in text box and list destinations appear in selectable drop down list. As user types self populated destination match highlights. |
|  | 4. Click link of destination in list | Routes to detailed page of travel destination with full description and details. |
|  | 5. Click "Close" | Browser window closes and returns to SNHU Travel Main page. |

**Appendix F: Developer to Tester Questions Email**

**A screenshot of a social media post

Description automatically generated**

**Appendix G: Developer Code Example 1**

**A screenshot of a social media post

Description automatically generated**

**Appendix H: Developer Code Example 2**

**A screenshot of a social media post

Description automatically generated**

**Appendix I: Revised Program Slide Show**

|  |
| --- |
| **A large mountain in the background  Description automatically generated** |
| **A large glass window  Description automatically generated** |
| **A picture containing building, table  Description automatically generated** |
| **A screenshot of a tree  Description automatically generated** |
| **A screenshot of a computer  Description automatically generated** |

**References**

Chuckc3. (2016, May 24). Agile Project Management Training. Retrieved from https://managedagile.com/what-does-it-mean-to-be-an-agile-developer/