**Final Project Week 7**

**Wesley Blackwell**

Changing from the waterfall method was a great choice for the company. The roles of the team, Scrum Master, Product Owner, Tester, and Developer, created the framework for our agile strategy. Utilizing the structure and each roles particular duties in the process, we were able to be ready for changes and customize our software to the customers’ needs more specifically and efficiently. Communication was streamlined and fluid between roles and in meetings. The communication framework of the Scrum-agile methodology helped everyone stay up to date on the status of the project and changes. Organizational tools from the methodology helped us to maintain our focus, efficiency, and communicate. Overall, I would say that the Scrum-agile methodology was a complete success. This Sprint Review and Retrospective will provide explanation, analysis, and description of the various phases of our trial run.

The Product Owner played a key role in the team. This role was important because they engaged the customer directly and often. The initial information gathering performed by the Product Owner allowed us to get a grip on what the customer needed. Creation of the product backlog was simplified because we had such quality information directly from the customer. The user stories allowed our testers and developers to really focus on the customers’ expectations. When the changes came up in week 5, our Product Owner was able to quickly provide the team with effective and focused requirements to meet the changes.

As the Scrum Master, I facilitated the first team meeting so we could begin the Backlog Refinement, or Grooming, process. This is the most important step as Scrum Master in my opinion. In this phase of the agile method, you set the stage for the entire project, identify potential disruptions or distractions, and most importantly put the customers desires first. I engaged the team daily with our morning Scrum meetings and chose the Kan Ban Board organization and planning style for us to keep the tasks and goals neat and easy to follow. The Sprint Planning meeting I facilitated allowed us to assign tasks to each member of the testing and development group and determine the ultimate objective of the first sprint. We had a review and retrospective after every sprint. It was important because reflecting on the last sprint identified strengths and weaknesses of the team, unfinished work from the last sprint, and any other issues the team encountered. This knowledge helped us learn from mistakes and how to work together to be more productive, work together more effectively, and make a better product for our customer.

The Tester was an effective role. Quality products and customer service provided by them helped the team to meet the expectations of the customers. Our Tester had to email some questions out to the Product Owner about the user stories they provided. The stories needed to be updated and more informative to meet the customers’ changes in week five. We could have had the Tester meet with the customer for a face-to-face meeting but it ended up not being necessary.

Our Developers were very motivated by the new method as well. Their responsibility of project deliverables was broken down into smaller parts. This allowed us to utilize an incremental development approach. Building the project a little at a time allowed the developers and testers to target the bare minimum requirements for the customer. When the changes arrived, communication and agility allowed the developers to gather the new requirements from the Product Owner and work with the Testers to adjust the focus quickly. The Daily Scrum was crucial to handle the changes. One quick huddle with the team and everyone knew exactly what was happening and how to adjust to meet the customers needs.

The Scrum-agile approach allowed the team to work through the software development life cycle with relative ease. The user stories were categorized by difficulty and importance as a part of the planning and analysis phases of the SDLC. Design, implementation, and testing were made simpler by the user stories. Utilizing the data gathered by the Product Owner, the development and testing teams were able to focus on the bare minimum requirements of the application. With the targeted data from the user stories, the testing team was able to assist in building routine maintenance for the product.

When the changes came in the middle of development, we had tools to handle it. The Product Owner had met with the customer regarding changes in their desires and informed the team in the daily Scrum that day. The developers and testers responded quickly with questions about the details surrounding the changes. Once the team had come up with their list of information, they required to make the adjustments, the Product Owner met with the customer again and gathered the data and emailed it out to the team right away. Testers went through the tests they had been working with and removed the useless ones and altered others to fit the new requirements and data. New tests were also written to comply with the changes. Developers were able to quickly alter the code they had already written since the style requires only the bare minimum during development. Additions and subtractions were made, and the development and testing team were back up and running.

As Scrum Master it is important to encourage development in my teams’ skills. My personal belief, “Teamwork makes the dream work”, motivates me to engage my team and promote communication and teamwork. The Agile method works great and gets better with increasing teamwork. Facilitating the meetings is the major method I used to engage. I wanted my team to develop their own dynamic to be able to quickly and efficiently relay information to one another. This was crucial during the week of the changes. The Tester and Product Owner were a perfect example. They communicated via email and face-to-face meetings in order to refine the user stories. The Tester was then able to make efficient changes to the testing that accommodated the change.

As I previously mentioned, we used a Kan Ban board. Everyday in the Scrum we discussed the tasks of the sprint and updated the board based on completion and assignment. Another tool used in Agile is the Sprint Backlog. This log was a representation or the Product Backlog but in a broken down and organized format. Using this tool, we were able to organize our Sprints and utilize incremental development. The most important principle of Agile we utilized is “customer collaboration over contract negotiation.” We worked with the customer daily to ensure the product was exactly what they wanted. Gathering the user stories from the customers allowed us to refine our vacation search engine and tailor the results to the individuals. When we experienced the change phase, the transition was made easier by our frequent communication and access to the customer.

Some of the pros for the Agile method include customer satisfaction, better control, risk reduction, continuous improvement, and many more. The morale of the team was much higher than in the waterfall method. The only con I could see during the process was that the product was developed incrementally so we had “fragmented output.” This is not so much a con to me as it is the point of the way we did things. Another con I see is that we only developed with a minimum requirement mindset. Again, this is not really a con at the same time since the customer can always add to the project once the basic program is up and running. The Scrum-agile approach is great, and I feel was the best approach for this project. The most specific reason is because the customers of a travel agency need results that are specific to what they like. Surfers are looking for places to surf not camp. Snow boarders are looking for cabins and snow, not beaches and sun.