Jacob Sullivan

CS 250 Final Paper

In every scrum team, proper execution by team members of their specific roles is pivotal to the success of any project they take on. This semester, we took part in a team that was hired to design a travel website for SNHU Travel that functioned in a manner desired by both the client and end user. We were approached by the client, SNHU Travel, with their original focus being on the need for a functional vacation booking service for their website dedicated to trendy, niche vacation packages. The scrum master and product owner met with the client and discussed what the client’s goals for their desired product is. After this meeting is where both the product owner’s and the scrum master’s work began. The product owner went on to begin creating and prioritizing the product backlog. Without a properly organized product backlog, a scrum team would essentially fall apart. While one of the principles of Agile is a “roll with the punches” approach focused less on planning and more on adaptability, the product backlog is essential to keep the team on track with the core needs of the project they’re working on. The scrum master began by selecting members for the team, consisting of a tester and a developer. The scrum master also created a schedule of scrum events such as weekly scrum meetings, daily stand ups, and sprint lengths that would be followed over the course of the project. The scrum master would also meet with the team and, together, they would all come up with an agile team charter. This is simply a list of rules and a code of conduct to ensure everyone is happy and able to work to the best of their ability. While the project unfolds, the scrum master’s role slowly transitions from that of a leader to that of a mentor. It seems to me through the various readings and this specific project that a scrum master is much more involved in the beginning. Once the team is on track, the scrum master seems to ease into the background and play a heavy support role, intervening only where necessary, and helping guide the project along. It is very important that a scrum master not overstep their bounds and micromanage, as that goes against the very principles of agile.

From this initial meeting, the developer and the tester can get to work as well. The developer’s role is fairly black and white. The goals for each sprint are decided as a team in a “what can we plausibly accomplish in this sprint” manner. The developer then begins to write code and build the product. Meanwhile, the tester analyzed data taken from a meeting with potential end users about what they would like to see out of the final product, such as lists catered to their travel histories and profiles, budget filters, and so on. Using these desires, the tester begins to build test cases to make sure these desires are prioritized and, when the time comes, pass these test cases to ensure functionality. While this is all going on, one of the most important jobs of all involved is communication. This keeps the entire team on the same page and allows them to see the strengths and weaknesses of their teammates, allowing them insight on where they could help and who they could go to for help with specific tasks. These roles together ensure that the product is completed on time and to the standards expected by the client and end users.

The agile process helped in the success of realizing user stories in a few ways. The product backlog, a key component of agile, must be kept organized and updated. In doing so, this allows the team at any moment to see the user stories in terms of importance and priority. With a waterfall approach, all the planning is done ahead of time and once the initial functionality is decided there is no going back to edit. With agile, as new requirements or desires came in and were added to the backlog the team was able to work on user stories and shift their attention and goals of each sprint to reflect the new and unexpected changes desired by the clients and end users. For example, there came a point where the client no longer wants a list of vacation packages, but a slide show like presentation. In a waterfall environment, this may have meant scrapping everything and starting over or even telling the client this isn’t possible due to the rigid plan set out in the beginning. With agile, teams can adapt to this new information. Rather than panic and try to start over, the team simply adapted these new desires to their current project by efficiently and concisely communicating with each other any changes that arose. The tester made a few simple alterations to their test cases, the developer made a few minor alterations to the code handling display, the scrum master made sure the product owner and team were all on the same page as well as updated sprint goals and meeting details, and the product owner edited the product backlog, editing and reprioritizing user stories to reflect the new information. Without agile, this transition into a different final product could have been disastrous. However, with the agile process in place, user stories were completed in a timely fashion even when minor alterations to them were added on during the project.

There came a point of the project where the client returned to the product owner and expressed the desire to change direction. Instead of trendy, niche vacation packages, they now wanted a focus on health and wellness vacation packages. If a waterfall approach was in place this could possibly have resulted in scrapping what exists and starting fresh. Everything had been designed and tested to fit a booking system for niche packages, not health and wellness. However, with an agile approach, there is a lot more wiggle room when major changes to the desired product are made. What it comes down to is, again, concise communication. I recall thinking that the product owner’s relaying of the new wellness focus was a bit vague. But with agile, there are weekly meetings and daily stand ups to help field all questions the team may have about the new requirements. By keeping the lines of communication open and consistent, as well as letting the teamwork these new problems in their own ways rather than working tied to a strict curriculum, the team was able to simply adjust their user stories, test cases, and working product to function around these new goals rather than go back and find an entirely new way to complete the project. Agile made sure there was no severe back tracking at this point, just a light retracing of steps and using those steps to continue towards a new, slightly different goal.

In terms of the prompt for this paper regarding communication with my specific team for the team project, I don’t think I can say I did a great job communicating, nor did anyone. This isn’t because of a lack of trying or uninterest, it was simply the time constraint for this part of the course. As I stated in a previous journal, this course is made up of people from all over the world with responsibilities in addition to school, be they work or family or community obligations. With agile, this hinderance is avoided by keeping that communication consistent using programs like Jira which give remote teams access to scrum boards, messaging platforms, and video conference software. Given just a week to communicate and come up with goals and procedures with little to no warning did not give myself or my fellow students ample time to properly complete this project in my opinion. However, I viewed this as a learning opportunity and imagined the project was a sudden communication passed down by a product owner that a scrum manager was not aware of. If this had been communicated effectively by the product owner to the scrum manager, our team could have had better results. But realistically, I found it very difficult to manage a group project given such little time and with everyone having their own schedules to go about around school. In a work environment, being that a career is a major focus of anyone, it would have gone much smoother in my opinion.

In terms of tools that contributed to the success of the team in the SNHU travel project, a largely beneficial tool was the scrum board. While it was never displayed, the scrum board seems to be the heart of any scrum team. Being able to view a board and see where everyone else is, updated in real time, is a impressive force behind the success of an agile team. Rather than traditional corporate environments where questions are directed towards management, then directed by management to those who have answers, and finally being recommunicated at a later point, agile teams have the ability to view a central “hive mind” of information that could answer most any mid-project questions they have. The fact that scrum boards and Kanban boards are updated in real time are also a key benefit as it lessens the chance that anyone is left out of the loop on important developments as well as allows team members to find things to do during their down time, whether that be helping someone struggling with a task they excel at or taking on a small task until a larger, more important one comes into play. The principles of scrum I found most effective were the focus on open communication and the lack of micromanaging. Open communication is key to the success of any business or project, and the fact that agile puts such a large focus on it through scrum meetings, scrum boards, and daily stand ups is one of the main reasons it’s so successful. The freedom to do that job how one sees fit, though, is one of my most admired principles of agile. I’ve worked in kitchens all of my life for many different chefs, and I have always found the most successful kitchens have been led by a chef with an agile mindset. In a kitchen with rigid protocols, creativity is stifled, and cooks regularly feel defeated, unenthusiastic, and undervalued. With the agile principle of empowerment, I’ve seen fry cooks turn into talented chefs because their ideas were nurtured and encouraged rather than silenced. Much of the reading this semester was on the practicality of agile in the workplace but I felt a large part of its success is due to the psychology behind it. Happier, more independent workers who are encouraged to think outside the box, in my experience, has always led to a more effective, efficient, and successful work environment.

In all honesty, I don’t see many cons in terms of the agile method. Any cons that come to mind are inherently dependent on the team practicing agile. If the team doesn’t actively want to work within an agile environment, this could negatively impact the entire project and those involved. From my perspective, while agile is a great way to manage a project and utilize a team, it could easily fail given a team of people who are consistently at odds with each other or who just aren’t enthusiastic about what they’re doing. We didn’t see it much in this project through the course of this class, but if the hearts of team members are not in the work, agile has a high probability of failure in my opinion. Another example of failure on the agile part was, again, not due to the use of agile but through the failure of the product owner to be more concise when relaying the change to wellness focused vacations. This communication came with very little detail and showed me that if communication is not consistent, concise, and open, agile could easily fail as the team falls apart. This is due to the laissez faire approach agile has with a project as it lacks a well-defined plan like the waterfall approach. Without a high level of communication, agile teams can quickly fall into confusion about their goals. However, if agile is practiced efficiently, the pros vastly outweigh these cons. Empowered and independent workers tend to be happier and more enthusiastic about their jobs, leading to better, higher quality results. The openness of communication and regular meetings ensure everyone is on task, in the loop, and feels as though they’re equal parts of a team rather than cogs in a machine. The ability to adapt to changing needs of clients and end users alike is facilitated by the very absence of that strict, rigid plan set forth in the beginning of waterfall projects and allows the team to go back and edit as needs develop rather than rule them out or start over. If used correctly and mixed well with some organizational principles from waterfall, depending on the project, the agile method can be extremely successful no matter what the task.