**Michael Medvedev**

Dear Team,

Congratulations on your hard work and effort for the past few weeks. One of the last steps we are taking as we near the end of Chada Techs first agile effort is to take a step back and look at what we have achieved. The goal of this retrospective is not only to understand if agile has the efficacy to deliver a high quality product, but if shifting over to an agile framework at the enterprise level will benefit the company culture as a whole.

To start off, there were a variety of different roles filled and hats worn throughout this process and demonstrating how each role impacted the SNHU travel project is crucial in reviewing our work here. The roles in our team were the product owner, scrum master, developer, and tester. The product owner was the frontline of the team who collaborated the most with our client at SNHU Travel and absorbed every detail of what was expected from us for this product. He then socialized this vision with the team by creating user stories and made sure to keep a detailed log of these stories and which had the highest and lowest priority, which was always subject to change. Specifically, during the product owners meeting with our SNHU Travel client, his role was essentially to create user stories based on the vision and requirements voiced by the client and deliver those stories to his development team who would functionally break it down and begin the development process. Next, the Scrum Master had a very important role in making sure the development team had the resources allocated to being iterative development, provide clarification on user stories and backlog refinement, as well as reduce bottlenecks and impediments for the development team. Specifically, during the SNHU Travel project our team had daily standup meetings for 15 minutes that helped each member of the team voice achievements, questions, or concerns. These 15 minutes were crucial in tracking progress, motivating and inspiring progress, as well as providing help and reducing bottlenecks. The Developer had the lowest macro overall but carried out the execution of each development iteration in developing the actual project. In the SNHU Travel project, an information radiator was used to display current tasks and progress. The developer would work on assigned tasks everyday and used user stories developed by the product owner as a benchmark for what was expected and used the scrum master for clarification. The success criteria for what was expected was outlined by the tester. The tester was more than a QA monkey in this agile framework. Since we decided to explore test-driven development, our tester actually outlined the “correct output” of our user stories that our developers implemented. In essence, testing was done first, and the code was developed just enough to pass these tests which ensured efficiency and modular/maintainable coding.

With these roles clearly established and agile principles being implemented on a daily basis, there was a clear increase in the quality and efficiency of tasks being completed. The scrum-agile approach to the SDLC significantly helped each of the user stories come to completion by not doing too much preplanning up front, rolling wave planning, iterative development, test-driven development, daily scrums, face-to-face collaboration, and information radiators. The idea in the SNHU project was to wait until the last responsible moment to make decisions and decrease the planning phase while embracing change and adapting to new requirements. Specifically, requirements were gathered from meetings between constant collaboration between the product owner and SNHU Travel clients and passed down to the development team through user stories. Consequently, constant daily communication was held between the scrum team and scrum master to ensure the clarification of user stories/backlogs therefore aligning the vision of the development with the end user/stakeholder.

Embracing change is one of the key benefits to an agile approach since reducing upfront cost requires lots of flexibility at the development level. During our SNHU development phase, there was an email sent out from the product owner describing new changes that our client at SNHU wanted from our product. In a traditional project, this would have been catastrophic as resources would have not been allocated for a change, it would not have been documented, and lack of collaboration surely would make it difficult to adapt to change. Fortunately, in our new scrum-agile approach, we adapted by re-aligning our vision with the new product change through grooming the backlog, re-creating test cases which therefore redefines our success criteria, and committing to those changes in our development. Because we have an iterative approach to project development, it was very easy to adapt to a change with the constant testing whereas a traditional project has a testing phase after the development phase.

When discussing the topic of an agile team, strong collaboration is certainly a major point for successful development. This is because one of the principles of the agile manifesto is face-to-face interactions, which is the best mode of collaboration. The reason for needing so much collaboration is the fast-paced environment with constantly changing requirements and need for adaptation. In a traditional project plan requirements are all pre defined and developed sequentially without room for change. This is unrealistic in today's world with constantly changing needs, so with the modern agile project planning, teams can meet the volatile needs of end users/stakeholders all while maintaining a consistent deadline. Specifically, we had an abrupt change given to us by our client at SNHU and we persevered and used our agile methods to quickly embrace this change and at the team level completely shift our priorities by using collaboration and iterative development, here is an example of such important collaboration. “Dear Product Owner and tester, in order to align our product with the new vision you socialized in our recent meeting, I will need the product backlog to be groomed so we can prioritize new tasks to retain our current deadline, as well as new test cases for the changes. Since we’re doing test-driven development, lets start off with sending me some new test cases so I can start developing with the expected results in mind which will help our code be more optimal and our level of effort more efficient.”

Just as important as it is to have constant collaboration, so is transparency. Traditional Project planning did not constantly show progress and allowed managers to cover their mistakes and be dishonest about progress to evade criticism. In our scrum-agile framework, we embrace failure because failing is part of the iterative process and does not cost as much. Moreover, current tasks in progress and backlogs are visible to anyone on our information radiator. This was any physical or digital tool that allowed members of the team to update the radiator with complete tasks, tasks in progress, and future tasks. Not only was this more transparent and honest, it also gave the team small wins and kept morale and motivation high by constantly seeing progress. Also it is easier to monitor bottlenecks and develop story points based on previous completed tasks.

Ultimately, the effectiveness of Chada techs first scrum-agile approach was immediate and showed even more potential. The main benefits of this approach were faster products, less risk, higher quality, less bureaucracy, and honesty. Less upfront planning allowed development to start faster, iterative development allowed for less risk in failure and embraced change. Higher quality was evident in test-driven development and pre-defining “success” criterias. And lastly, less bureaucracy from the removal of a hierarchy, more collaboration, and transparent progress. However there were two cons we noticed, firstly less upfront planning meant more initial ambiguity, especially for a first time agile team, effort required for tasks were hard to measure. And Finally There was a lack of macro knowledge amongst the development team, since they did not deal directly with the product backlog, nor the clients, it was difficult to have the products vision fully aligned with the development team just through user stories, they had a smaller visible scope than the product owner and scrum master. Overall, the pros definitely outweigh the cons, and the more familiar a team becomes with agile, the easier it becomes. Therefore the scrum-agile approach was certainly efficient for this project and should be used again.