**SNHU Travel Retrospective**

by Tray Bridgewater

Over the last two months, our team here at Chada Tech has produced a travel booking web application for our client, SNHU Travel. We had initially planned to create a general purpose vacation booking web application, but midway through the product our client informed us that they would like to change the product direction to something a little more focused – particularly, a booking application for wellness and detox traveling. Despite a wrench being thrown in our plan, our team was successfully able to use the Scrum framework and the broader principles of Agile software development to rapidly respond and re-function the code we already had instead of starting from scratch.

Our team had a strong division of labor. It included a Product Owner, a Developer, a Tester, and a Scrum Master. While everybody is expected to have a baseline understanding of how the rest of the team functions, each role is vested with certain obligations. The Product Owner serves as the intermediary between the development team and other stakeholders in the project, the Developer is the technical expert responsible for implementing the software, the Tester designs and manages test cases to ensure the product is functioning properly both technically and from a business perspective, and the Scrum Master manages the details and implementation of the various Scrum events. Each of these roles are critical to the success of an Agile project: if you remove any of them, we lose an extremely valuable part of the team and compromise the larger project.

I would like to briefly circle back and discuss the metaphorical wrench thrown in our plan – the sudden shift in direction. Were we using a Waterfall development strategy, where everything is planned out in advance, this would have necessitated very large-scale changes to the entire workflow. We were able to avoid this issue, however, due to our use of Agile as a software development framework. Our code itself was highly modular, and all we had to do was re-arrange elements of our sprint backlog to make room for our developer to go back and retool certain aspects of the website. By conceptualizing the feature backlog as a flexible, shifting thing we are able to respond when rapid changes are needed.

Another success throughout this project was in our team’s communication. Take, for example, this email from our developer to the product owner:

*“Product owner,*

*I’m reaching out to ask you to review my implementation of SNHU Travel’s new search feature. I’d like to make sure my work is up to scratch before we move on. Please visit the website and play around with the search engine. Let me know if you think there are any changes I could make to improve user satisfaction. Additionally, please let me know what objectives ought to take priority for the remainder of the sprint.*

*Thank you, Developer.”*

In this email, our Developer very clearly communicates a specific request that he needs fulfilled by the Product Owner in order to continue the development of the product. Reviewing completed work is a central role of the Product Owner, as they are intermediaries between the development team and other people, such as users and clients, who have a vested interest in the quality of our product.

The larger Scrum framework is a tremendous asset. It provides a distinct workflow that ensures constant communication between members of the team, provides a flexible mechanism by which we can change priorities on a whim, and keeps a relentless pace of development via the Sprint. The sprint constitutes a brief burst of activity with pre-determined goals drawn from the “Sprint backlog”, a list of features to add or changes to make that is maintained by the Product Owner. Each day of a sprint is punctuated by a brief Daily Scrum in which the team members communicate concerns and interface with each other about the development process.

These different parts come together to allow the development team to do a tremendous amount of work in a relatively short period of time. The most important part of the entire process, in my opinion, is the sprint itself. It forces features to be sub-divided into atomic parts, and in doing so forces the team to adopt a level of focus and detail-oriented development that is simply hard to achieve without it. The short length of the sprint also forces the team to develop the product quickly, which is how we churn out high-quality software in a short period of time.

The flexible nature of the Sprint Backlog comes in as a close second for the important feature, though. It is how we were able to completely reorient our project on the drop of a dime. The backlog is constantly changing and being maintained by the Product Owner, who is vested with the authority to make consumer and business-side decisions about the development of the project. When our client changed their requests, we were able to move “adjust program to center about wellness and detox vacations” to the front of the Sprint backlog without causing a serious rupture in the broader workflow. Were ever aspect of development planned out in advance, this would have been a much larger problem for us because it would fundamentally change our entire software development workflow.

Ultimately, I believe Agile and Scrum are highly effective ways of developing software. Waterfall development works well for smaller and less sophisticated projects because of the relative simplicity of its workflow, but when the nature of the project is complex or could conceivably be changed, a less planned-out and more agile approach to development is called for.