**Sprint Review and Retrospective: Leading Agile Development at ChadaTech**

Abel Amedu Babatunde, Abel

Southern new Hampshire university

06/19/2025

**Sprint Review and Retrospective: Leading Agile Development at ChadaTech**

Being the Scrum Master and leading the pilot Agile project in ChadaTech, I have managed to transform our development team that used a traditional waterfall model to an Agile methodological framework with a Scrum-type approach. The objective was to design a traveling app on behalf of SNHU Travel—a project that offered an ideal environment that would show the value of Agile in improving the quality of the product, teamwork, and responsiveness to change. This retrospective is a brief account of how our team has performed on the project, an assessment of the Scrum-Agile framework, and advice on how ChadaTech may wish to turn its company-wide.

**Applying Roles**

This high level of success of the SNHU Travel project depended on how every Scrum role was distinctly characterized and fully performed. The product owner kept and ordered the product backlog. This position influenced how the development team focused on the most valuable functionalities that were meant to handle real-time reservations and secure payment methods depending on the needs of clients. The development team was cross-functional, meaning that they worked together to achieve increments at the end of every sprint. They used pair programming and a continuous integration process that enhanced the code quality and shared knowledge within the team (Beck et al., 2001).

As Scrum Master, I enabled daily stand-up meetings, sprint planning, reviews, and sprint retrospectives. My task was to eliminate obstacles and keep the team on the track of Scrum practices. As an example, an issue with the access to the API of the booking system created a delay in the performance of the entire team, and I was able to coordinate the support of IT as well as external vendors to address the issue as quickly as possible, reducing the downtime of the development. It was this leadership that allowed us to adhere to our sprint schedules and ensure that team morale did not go down (Schwaber & Sutherland, 2020).

**Have finished user stories**

In our Agile development, user stories played an important role. Every story was written in the user perspective (e.g., I mean, as a traveler, I want to get a confirmation of my trip through my mail so that I can trace my itinerary). These stories helped us to concentrate on providing user-centered features. In the sprint planning, the team chose product backlog stories and said how many story points they would require to implement them.

The Scrum enabled us to make anything in short and manageable sprints, and each time the sprint delivered a potentially shippable product increment. As an example, during Sprint 1, we successfully finished the log-in and sign-up module. We managed to apply the itinerary customization by Sprint 2, and Sprint 3 featured the integration with secure payment. Such iterations gave room to solicit feedback, and with this, it was possible that our solution met the changing needs of SNHU Travel (Highsmith, 2009).

The Scrum process furthered backlog grooming and prioritization, which enabled us to remain focused on user stories that are of most value. Shared responsibility and close interaction with stakeholders warranted the fact that our work could always be validated and deployed.

**Handling Interruptions**

Flexibility to change is one of the greatest advantages of Scrum-Agile. In the third sprint, SNHU Travel asked to add a new payment gateway to their system since they had changed the vendor. In the waterfall model of development, such a diversion at an advanced stage would have ruined the whole scheme and raised the project costs. But with Scrum, we just conducted an emergency refinement on the backlog, we reprioritized the work, and we incorporated the new gateway into the following sprint, so we were not going to miss our schedule.

This ability to respond to change reflected the fundamental concept of Agile, whereby the changing requirements were embraced even at a late stage of the development in order to deliver competitive edge and customer delight (Manifesto for Agile Software Development, 2001). Agile helped us to switch gears quickly and not lose the project despite its benefits and the confidence of the clients.

**Communication**

Our team succeeded mainly because of communication. I conducted daily stand-up meetings as the Scrum Master and understood the updates, the challenges, and the plans of the respective members. Such brief occasions were transparent and accountable and allowed early identification of blockers.

In one of the sprints, such as Sprint 2, one of the developers mentioned in the stand-up that unit tests on the booking feature were failing randomly. This timely communication allowed us to set up a timely pair debugging, and within the same day, we determined a data race condition and fixed one. We were also able to access such tools as Slack that helped us collaborate in real-time and Jira, where we could trace user stories, bugs, etc.

I sent sprint summaries to the team at the completion of every sprint. This is one of the excerpts from a Sprint 3 summary:

“Hi Team,

Bravo on completing the feature of sharing itinerary. During Sprint 4, let us concentrate on the testing of payment integration. Time logs and blockers should be updated in Jira tickets. The time we should have our retro is Friday at 10 AM.

Best,

[Abel] Scrum Master”

This regularity of communication guaranteed coherency, minimization of misunderstanding, and cohesiveness of the team.

**Scrum Events and Organizational Tools**

To maintain the project on the right track, we utilized some tools of organization and Scrum events. Jira has given us the ability to create a product backlog, assign the tasks, and track the progress via burn-down charts. We took Miro to collaborate on sprint planning and retrospective.

The formal Scrum meetings, namely, Sprint Planning, Daily Scrum, Sprint Review, and Sprint Retrospective, were very helpful. We, for example, persistently got better by using the method of Start, Stop, Continue in our retrospectives. Another effect of this situation was the creation of such a decision as automating regression testing, the beginning of which was Sprint 3. This allowed us to accelerate the delivery and slacken the bugs.

Schwaber and Sutherland (2020) propose that not only do these Scrum events facilitate transparency, but they also empower teams in order to inspect and adjust, which we discovered was the case during our development process.

**Assessment of Agile Process**

The Scrum-Agile methodology was very effective for the SNHU Travel project.

**Pros:**

* Flexibility and Adaptability: New client requests were taken well.
* Customer Engagement: There was a higher frequency of feedback, which meant a closer adherence to client expectations.
* Team Autonomy: Self-managed developers and enhanced ownership.
* Early Detection of Problems: There are fewer defects found as continuous testing and integration happen.

**Cons:**

* Initial resistance: It was unknown to some members of the team.
* Scope Creep Risk: The risk here is that the changes in the backlog might be frequent, thus frustrating the project.

Agile helped us to launch on time a powerful and user-friendly travel app despite all minor inconveniences. It can be concluded, based on the experience, that Scrum-Agile would be strongly recommended to be used in future development projects at ChadaTech, particularly the one where requirements are to change and stakeholders are to be actively involved.

**Conclusion**

Among the results of this pilot project, it was shown that implementing Scrum-Agile enhanced teamwork and collaboration as well as the quality of the products and client satisfaction markedly. With excellent communication, clearly established roles, and receptive planning, we shipped high-value features efficiently and progressively. Agile scaling within the company can be very beneficial to ChadaTech, as long as it is done with due care and assistance in the process.

**References**

Beck, K., Beedle, M., van Bennekum, A., Cockburn, A., Cunningham, W., Fowler, M., ... & Thomas, D. (2001). *Manifesto for Agile Software Development*. <http://agilemanifesto.org/>

Highsmith, J. (2009). *Agile Project Management: Creating Innovative Products* (2nd ed.). Addison-Wesley.

Schwaber, K., & Sutherland, J. (2020). *The Scrum Guide*. Scrum.org. <https://scrumguides.org>