Jordan Poston

CS-250-10680

04/17/2025

7-1 Final Project

Throughout this sprint (semester) we helped transition ChadaTech from a more traditional, linear project management methodology to the Agile methodology. This required us to implement specific roles for our team members and an increased emphasis on communication and collaboration. During a Sprint Planning meeting, our Scrum team (development team, Scrum master, testers, product owner and stakeholders) determined the overall sprint goal, discussed the initial plans for our product backlog, organized these items into tasks, and assigned these tasks to our team members. We also discussed our “Definition of Done,” set standards for our daily Scrum meetings and less-frequent, larger meetings, and collaborated about the overall vision for the project.

Our Product Owner collected feedback from focus groups to create User Stories, which we were then able to translate into direct tasks that the development team could work on to fix or improve our users’ experience with our product. Our testers also used these User Stories to build specific tests to ensure that these requirements (as identified by our own customer feedback) have been met before the product is deployed. For example, the idea for many of the filters in the search engine for our travel site came directly from these User Stories. Another user wanted to see custom advertisements for travel deals based on their personal taste in travel and destination. This kind of direct feedback is invaluable to the Scrum team.

Once we have organized the new tasks from the User Stories, our development team is responsible for writing the new code and adhering to our clear, clean standards for software engineering practices. Scrum teams are small and autonomous because that helps instill a shared language and communication style for the team, which limits confusion or lack of clarity on tasks, and ensures that each member of the team can plug in wherever they are needed. This means that any team member can jump into any task behind any other team member and have a clear understanding of how to continue said task.

Agile methodology’s focus on communication and collaboration make it a uniquely flexible framework for project management. This is extremely useful when encountering any kind of necessary changes in scope or other interruptions. For instance, at a certain point in the project, the client wanted to change the content of a particular slideshow from popular travel destinations to the top 5 health and wellness trip destinations. Since Agile is so flexible and iterative, it really only required one meeting for the Scrum team to change the scope for this overall task, and tweak other sub-tasks to begin implementing these changes and get right back to work on a deliverable product. The Product Owner and Scrum Master were instrumental in explaining how the customer’s feedback and changes would affect the vision for the product. After the Product Owner explained the new criteria and vision, the Scrum Master formulated a plan with the dev team on how to pivot and execute these new goals.

Good, clear and continuous communication is the keystone to success for any team project, using any management approach. A unified, whole team is more efficient than just the sum of its disparate parts. Sprint planning, daily scrums, sprint reviews and retrospectives are just some of the major meetings emphasized in Agile to keep the entire team on the same page regarding the vision and implementation of a product. Some of these communication practices are built into the fabric of Agile methodology. For example, our testers had to collaborate with the Product Owner on the User Stories to create the specific tests for our customer’s new requirements based on the focus group data. Contrary to popular belief, Agile practices also cut down on redundant communication. Agile is structured in a way that promotes communication amongst the team while also insulating team members from less relevant meetings and conversations that would take developers away from their tasks in the project backlog. The Scrum Master usually facilitates the organization of these communication protocols.

There is now an array of tools at our disposal that can boost communication and improve the utilization of Agile methodology overall. Applications like JIRA can act as a digital, communal space for members of the Scrum team to check for task assignments, notes from other team members, or overall job progress. Applications like this are especially useful when working on large projects with multiple teams. On these projects, more substantial tasks are split up amongst these multiple teams, who then break them into smaller sub-tasks to be performed by dev team members. In cases like these, it would be impossible to track progress via conventional means like a giant whiteboard or a single spreadsheet.

While other project management methodologies like waterfall could be useful for much larger, statically defined projects, the Agile methodology seems like a more natural fit for most software projects. Its endless iteration, abundant flexibility, and emphasis on constant communication keep Agile lean and effective. Customers also tend to love the feedback loops and constant delivery of usable products. Regarding the SNHU Travel site project, the Agile methodology was clearly a good fit. Had we chosen a more rigid framework like waterfall, our team never could have pivoted so easily when encountering customer feedback and needed changes during the development process. In the time it would take to get a usable product by applying the waterfall methodology (which would only meet the original product requirements), our Scrum team was able to engage with this customer feedback, implement it, and deliver a fully functional product with our customer’s requested changes.