CS 250 Final Project

Michael Jones II

The Software Development Lifecycle(SDLC) can be easily described as the steps that are involved in software project development from beginning to end. Usually, the Software Development Lifecycle has many steps and while they are contended between interpretation, I like to break them down into these seven steps. Step 1 is Requirements Gathering: This is where you are interviewing with the customer requesting the product. In this step you are trying to figure out the specific needs, wants, and additional features from the customers. This is the main information gathering phase of the project. Step 2 is Requirements Analysis: This is the step where the Scrum team begins to look at what the customer has requested. While discussions are on-going in this process the team decides if the project can be functional, what approach they need to take, and they sometimes begin breaking down how they will build the product. Step 3 is Design: This is the step where the team begins intricately planning how the product will work from start to end. They will also begin to prioritize steps and deadlines for the project. Step 4 is Development: This is the step where the actual development begins. The project is now being built, code is being written and deliverables start to take shape. Step 5 is Testing: This is the step where the testing team will be writing tests, debugging, and figuring out any potential issues going forward in the product. Usually, step 4 and 5 are on-going simultaneously and the development teams and testing teams will need to work in cohesion with one another to achieve a great product. Step 6 is Deployment: This is the step where the customer receives their completed product. End users will begin using the product in this step and they will begin gathering information on issues they find, changes they might like to make, etc. The final step and arguably the longest step is Maintenance. Maintenance and support are constantly on-going during this step. The end users and the development team must work together to fix any issues that come to light by long term usage of the product. If requested, this is where the development team could implement new features, patch issues/bugs, provide support and training on the use of the product. The maintenance step is the longest step because it remains for the lifetime of the product. Eventually, the end users/customers might wish to upgrade their current product with an updated UI, functionality, and other things. This potentially could remain within this phase, or depending on the size of the changes, completely start the process over again.

The Waterfall method was the first methodical process to be introduced. Normally, it remains very simple and easy to understand. Typically, within the Waterfall method, each step must be completed sequentially, and the next step cannot begin until the previous step is completed. Some advantages of the Waterfall method include phases will be completed one at a time, its very easy and simple to understand, and its much easier to manage and control. However, the Waterfall method does have its setbacks. Some disadvantages of the Waterfall method include difficulty of estimating time to completion and cost. Also, if requirements change you could potentially have to rebuild the product. The project is only seen by the customers at the end. This is a big disadvantage because if the customer wants it to work a specific way or changes their mind since the beginning, you will have to completely start the project over from scratch.

Agile methodology is an idea set that constantly promotes continuous iteration of development, testing and innovation throughout the life of the development process. The vast majority of this course focused on the pros and cons of Agile method of development. A few advantages of using Agile is that is suited greatly for a very large scale and complex project. Also, the customers will constantly be shown updates and normally will be given working prototypes throughout the development process. The biggest disadvantages for Agile is that the cost of the project is not a fixed budget and it can be difficult to manage at times. If the customers requesting a product don’t like certain deliverables, it can slow down the development process. While this doesn’t completely de-rail the project like in the Waterfall method, it will cost more time and money to complete the project. Also, anything large scale always has its challenges. It can be difficult to maintain a large team daily through the completion of the product.

As discussed previously, the biggest issue that can inhibit any team environment is communication. Ineffective communication will delay a project faster than anything else. The following excerpt is an email example from a previous assignment. I believe that this email does a great job of conveying to fellow team members what I would need to be successful for the betterment of the project. The email remains professional throughout and allows for responses and changes/suggestions from the initial interaction with the other parties.

(EXAMPLE) To Product Owner and Tester,

As you are aware, our team has moved to an agile methodology approach for this upcoming project. For the betterment of the team and the product, I will need the following things from you:

Product Owner:

* Give full requirements for the project
* Specify important dates and deadlines at the beginning of the project
* Print out of what outputs are needed in the program
* Thoroughly check and grant feedback on all deliverables to the customers

Tester:

* Please work closely and be as involved as you can in the development process
* I will need your assistance for testing before each requirement deadline date

Thank you,

Michael Jones II

(EXAMPLE)

Throughout the Scrum-agile process I believe the one thing that assisted the team in being successful was everyone completing their jobs. In a Scrum environment, every party has a specific role and tasks laid out. If one group or party does not complete their job, it will bring the entire group down. The Product Owner is responsible for being the middleman between the customer and the Scrum team. The PO will also monitor all work being completed by the team and provide suggestions to the team members during the process. The Scrum Master is responsible for the day-to-day management of the Scrum team. While the PO oversees the entire project, the Scrum Master’s sole focus is the Scrum team. The Scrum Master does not need to be a technical person, just a great manager of people, allowing for minimal micromanagement. The development team is responsible for the development of the deliverables. They are the primary programmers of the product. Most of the time, the development team works in cohesion with the Testing team. While the testing team’s main role is to test, they also have other responsibilities such as pointing out potential issues to developers, interpersonal conflict resolution between the two departments among other things.

Overall, I think that the Scrum-agile approach was correct for the SNHU Travel project. It allowed for constant user input, development oversight throughout the process and effective and efficient updates from the development team. In the specific scenario, the potential customers that were brought in was customers of competitors. The SNHU project team used this to develop necessary requirements for the project. If these users were requesting these features that were offered elsewhere, it would be great features to add into the SNHU Travel site.

References

(No outside sources were used during this report)