Dayna Eidle

Professor Rivas

Software Development I

January 26, 2017

Lab 1 - Essay

There are multiple software development processes out there, Agile Development being one of them. The main goal to Agile Development is being able to respond to change quickly and efficiently in an environment that may seem unfamiliar to the developers. It seems as though Agile Development is all about speed. It focuses on adaptive planning, which seems to say that the planning process is more of an ongoing thing as the code is being written and the project is being created. It also focuses on early delivery, showing that one of the main goals is to finish the project as quickly as possible for the client. Since there isn't as much structure when it comes to Agile Development, testing of the project occurs in the same iteration as programming. It also entails a good amount of collaborative group work.

Going along with the speed focus, Agile Development has some downfalls regarding that. In some cases, there tends to be a lack of product design because some of the designing is figured out as the programmers continue to work. Assigning tasks can also pose as an issue because with some companies, the same person gets assigned to the same task over and over again, which leaves no room for knowledge enhancement and training in other areas.

The process described in the textbook has some similarities to the Agile Development process, but is somewhat different. The process described in the book is defined as an engineering process. It is very structured compared to Agile Development. It contains seven key steps: Requirements Specification, System Analysis, System Design, Implementation, Testing, Deployment, and Maintenance. The one downside to this process could be that it might take longer to produce the final product because the process is so precise, however it leaves less room for error with the level of time taken at each step, including the possibility of repeating a step if need be.

If it were me as the developer, I would choose the process described in chapter 2.16. I am all about structure. I like when there is a set way to do things. Also, I find it easier to start off with a clear view of the plan and then if it needs to be changed later on, that's fine too. It seems like there would be less errors along the way with this process and if there is an error, it would be much easier to track and fix it. However, both processes have upsides and downsides and work better for different programmers. There is not one right, exact way to do it-- it can change from program to program.

Works Cited

"What Is Agile Software Development?" *Agile 101: Back to the Basics*. VersionOne, 2017. Web. 25 Jan. 2017.

Liang, Y. Daniel. *Introduction to Java Programming*. 10th ed. New Jersey: Pearson Education, 2015. Print.