Writing Assignment 1

David Houngninou

CS 390

Dr. Hwang

15 September 2008
On Monday 15th 2008, Dr. Roberts was the guest speaker for our software engineering class. He is currently consulting for ProfitStars, a company owned by Jack Henry Associates. The company is specialized in services designed to enhance the operating performance of financial institutions, automated systems, fraud detection and much more. Dr. Roberts presented the stages of work accomplished to improve the infrastructure and working environment of the engineering team for a more effective output. First, I will provide a brief overview of the preexisting structure of the development team. Next, I will enumerate the main steps and modifications brought by the XP life-cycle model to improve the service quality to clients. Finally I will explain how these changes affected the productivity of the development team.

Several reasons motivated Dr. Roberts to suggest a change to the XP life-cycle model. First, the former setting inside the development team was not allowing a good communication flow. In terms of statistics, customers were reporting a high rate of technical bugs in programs. Gradually, a loss of confidence in the team started to grow. One of the main reasons invoked was a lack of guidance for the overall development team, and a poor testing process on projects. In the first place, he proposed to rearrange the physical working environment. I believe that it was a bright idea to reorganize the working space to enhance communication between collaborators. Since large projects are generally divided into small chunks, if one part of the projects is corrupted due to a lack of communication, the entire team becomes accountable for a late submission. By opening the physical space in the office, programmers were now able to discuss issues instantaneously and share knowledge. One member of the team can possibly solve an issue that has been a bottleneck to another colleague. Similarly, by encouraging stand up meeting every morning, different teams can share an overview of tasks and make plans for the rest of the
A strong interchange needs to exist between programmers and testers. By introducing the idea of breaking projects into small chunks and testing them independently, the testing team can reduce the chance of leaving bugs. With such method, the probability of leaving errors in programs is reduced. Each tester can focus better on debugging with a small piece of code to scrutinize at a time. As a result of this change, feedbacks from clients revealed less bugs in the final product. However, there is also one main inconvenient for breaking projects in small pieces. To make a robust overall project, the team needs to make sure that all the parts can fit nicely into a working code. Otherwise most of the work has to be redone. As an example, I can refer to the implementation of Longhorn by Microsoft in 2004, where engineers implemented the project more like Lego blocks and were finally unable to meet their submission deadline. Lastly, I believe that Dr. Roberts did not define his appropriate methods to face sudden changes while using the XP life-cycle model. Such changes can be variations of the client requirements in the middle of the project implementation.

The XP model provided benefits to ProfitStars’ development team and affected customers rating positively. I definitely agree that pair programming is one of the best methods to write code accurately and efficiently. However, in most of our programming courses as computer scientist or computer engineer, we barely have the opportunity to work in pairs or teams on projects. I believe that our educational curriculum does not encourage team programming enough. As a result, we end up being self-centered programmers very unfamiliar with the notion of knowledge sharing on large projects.