Response to Dr. Robert’s Presentation on Extreme Programming

Written by James Yoder

The project was a mess. It was an old program containing mostly legacy code written by programmers that had left long ago. Customers were getting frustrated because many of their requested features were not being implemented or were too buggy to use. Bugs that had been fixed were reappearing with every change and new feature. The project was far behind schedule and was costing the company dearly. Something needed to change drastically, and change soon.

That’s what Dr. Don Roberts found when ProfitStar called him in for consulting advice. First, Dr. Roberts looked at their team structure and communications. What he found was not encouraging. Team members were separated in cubicles and used instant messaging to communicate, which was very inefficient. Another issue was the rivalry with the testing group, who was located in another area of the building. The result was an inefficient system that did not communicate well. Dr. Roberts proposed changing the entire programming system to the extreme programming (XP) model, which stresses communication and testing as the solution to the problems he found. Although the changes were sweeping, the management and programming department decided to try XP.

The change was tough, but the results were dramatic. The programmers moved all their computers to a single table so programmers could communicate just by talking across the table. Test cases were laid out for each module of the program so that new changes did not recreate bugs. Planning and scheduling was improved by better time estimation and daily stand-up meetings. Refactoring helped clean up the old legacy code that was complicated and buggy. Testers were also moved in the same room with the programmers, which also improved communication and eased rivalries. In the end, the company was able to modify the program to transition away from the Paradox database and allow support for other databases, and since then, they have consistently hit target dates and delivered new features to their clients. For them, the XP model was a resounding success.

For me, this was another success story for changing the way we think about software development. Although many of the techniques used by extreme programming are not suitable for all developers, key ideas such as constant automated testing, better communication techniques, and real life time estimation can be implemented in
almost every software company. I think that extreme programming is another step to producing better quality software in a realistic timeframe.