

# Three Steps to Programming Excellence

Ulf Asklund, Lars Bendix, Görel Hedin, Boris Magnusson  
{ulf|bendix|gorel|boris}@cs.lth.se  
*Department of Computer Science*  
*Lund Institute of Technology,*  
*Sweden*

## Abstract:

For many years our department had a fairly traditional way of teaching program development to its students. The core curriculum was a course in programming and a course/project in software engineering. However, problems with the second course led to a redesign of our set-up.

The main problem with the software engineering course/project was that the students during the project part experienced that they were able to produce requirements, designs etc. without following the processes and methods taught – and with less effort. We were not able to scale down the process overhead to fit the scaled-down problems suited for student projects. So it was decided to introduce a new course/project such that the students now follow three steps of programming education: programming as an individual, in a team and as part of an organisation.

The second step of programming in a team is completely new and we have had very good experience using Extreme Programming XP as a basis for that step. At our department each term is divided into two study periods of seven weeks of teaching immediately followed by one week of exam. The XP course is placed in the second period of the autumn term and the XP project in the first period of the spring term. A corresponding XP coaching course for older students runs in parallel, having a course period and a practice period.

In the XP course part we introduce the students to XP as one example of software development methods. Besides introducing the general philosophy and practices, we go in more details with the practices of Test-Driven Development, Pair Programming, Simple Design, Refactoring and Planning Game. Furthermore, there is a lecture on Software Configuration Management for team needs, focusing on the practices of Continuous Integration, Collective Code Ownership, Frequent Releases, Refactoring and Planning Game. We also run a two-hour Extreme Hour with the students, in which they, using pen and paper and drawing a system, go through a couple of iterations to get the feel of how an XP project runs. To facilitate the use of tools and to anchor selected practices we run three computer labs using Eclipse, CVS and JUnit.

After this preparation students are ready to start the project part. About 110 students are divided up into groups of 8-12 people and each group is assigned two coaches from the coaching course. The projects go through six iterations (weeks) consisting of an 8-hour programming/production day and a 2-hour Planning Game. Each project does a release (including manual) to the customer every other week and the final release (including documentation) is also done to another group. At the final evaluation meeting groups evaluate each other's product in a round-robin fashion.

With respect to some other universities that have tried out XP in education we provide more extensive support to our students and follow them tightly. During the three years the course/project has existed we have found the following additional “educational” practices important for our success where others fail: Team In One Room (we have scheduled 8 hours of programming where the team meets and work together in the same room), On-site Coach (we have two coaches on-site with the team at all time, which means that they can be proactive and catch problems before they grow big), Reflection (with the help of the coaches we force the students to explicitly reflect on how their project is going after each iteration). Being a customer and producing stories is a big task, however, we use 4 customers for 12 teams and have had no problem in reusing the same product and customers for three years in a row. Another way we try to keep down the costs is to use older students (following a coaching course/practice running in parallel) as coaches.