Tutorial on Extreme Programming: A New Way of Approaching Programming, Team Practices and Processes in Software Development

Arturo Sánchez-Ruiz

CIS Department
University of North Florida
Jacksonville, Florida, USA
arturo@acm.org, www.unf.edu/~asanchez

Goals:

At the end of this tutorial, the attendee will be able to identify different practices associated with Extreme Programming (XP) that touch upon the way programming, team practices and processes are approached in software development. More specifically, the attendee will know about "test first" (using the JUnit framework), "pair programming", "release planning", and "iteration planning". Furthermore, the attendee will be exposed to timely and relevant literature in the area.

Contents:

XP principles, variables, and values. Elements of XP that influence programming, team practices, and processes. Lightweight versus heavyweight development processes. Programming in XP: an example of "test first" using JUnit. Pair programming in XP: an example of how it works in practice. Planning under XP: an example of release planning and iteration planning. Is there anything new under the sun? Current trends in XP.

Relevance:

Currently there is a big debate about heavyweight versus lightweight processes. The former being characterized by complex tasks in number and structure, and the latter being characterized by few and simple, iterative, and incremental tasks. An example of a heavyweight process is any that come from a CMM-like model (e.g. level three and up). Different actors must develop many different artifacts, and these artifacts must be kept in synch as the software evolves from vaporware to an actual implementation. XP is an example of a lightweight process that is currently being discussed, studied, and applied by people that consider themselves part of the avant garde of Software Engineering. It is therefore important to know what XP is all about to be able to determine its potential usefulness in both the academia and the software industry.

Presenter biography:

Dr. Arturo Sánchez holds a "Licenciado" diploma in Computer Science (*Magna Cum Laude*), from Universidad Central de Venezuela (UCV), and MS and PhD degrees in CS from Rensselaer Polytechnic Institute. He has taught Software Engineering-related courses from 1993 to 1998 at the CS Department of UCV (as a member of the Software Engineering and Systems Center – ISYS), from 1998 to 2000 at the CIS Department of University of Massachusetts-Dartmouth, and since 2000 at the CIS Department of University of North Florida. As a member of the graduate curriculum committee, he has been responsible for designing the newly created graduate Software Engineering Track, with a well-defined Object-Oriented component. His active research areas are Generic Programming, Interoperability, and Computer Science Education (see www.unf.edu/~asanchez for a sample of his recent work). He is a member of the ACM, and the Computer Society of IEEE.