

SOFTWARE DEVELOPMENT LIFE CYCLE



Goals: Learn the basics for the right software development process, in order to reduce its costs and avoid pitfalls

Target: Developers with basic skills and little work experience

Format: 40 hrs. (40% Theory - 60% Workshop)

Notes: The specific tools used in the course are previously checked out with the client in order to better match the real working environment

SYLLABUS

Introduction

Contents

- General Introduction
- Models Comparison (Waterfall, SCRUM)

From Requirements Elicitation to Coding

Contents

- Functional and Non-Functional (FURPS+) Requirements
- AGILE User Stories
- Feature Driven Development
- Test Driven Development
- Behavior Driven Development

Workshop

- Configuration of the *Eclipse* environment with *JBehave* and basic project sample

Testing

- Unit Testing and TDD.
- Focus on specific Framework
- Integration Testing

Workshop

- Integration of the *Eclipse* environment with *JUnit* and examples of unit testing

Quality Assurance

- General Software Metrics
- Quality Process and Management

Code Management

- Code Reviewing
- Code Versioning
- Focus on tools for Versioning and their IDE integrations

Workshop

- Configuration of a *GitHub* account
- Integration of *Eclipse* and *Git* and basic versioning operations

Bug/Issue Management

- Bugs/Issues Types
- Issue Tracking/Life cycle
- Focus on specific tools
- General Risk and Change Management

Workshop

- Integration between *Mylyn* and *Eclipse*
- Linking *Mylyn* with *GitHub Issue*

Process Automatization

- Tools for automatic test/build/deploy
- Continuous Integration

Workshop

- Building a project with *Ant*
- Configuration of *Maven* and *M2Eclipse*
- Configuration of *Jenkins* CI Server
- Final Workshop : Putting everything together in an Agile Environment