

TRAINING COURSE SUMMARY

ESSENTIAL PRACTICES

Agile

Agile 101 (1 day)

Provides an understanding of the fundamental principles and practices of the three main-stream agile methodologies (XP, Scrum and FDD). These are reviewed and compared to exemplify the agile principles and to understand key agile techniques. Five real-world case-studies and three workshop exercises provide further insight and practical appreciation of applied use.

Scrum (1 day)

The principles, workflow, output and team structure of Scrum are discussed and then the top five Scrum practices (Product Backlog, Sprints, etc) are explored in more detail. Three hands-on group activities plus four real-world case-studies aim to cement this understanding of Scrum.

eXtreme Programming (1 day)

Provides a pragmatic insight into the rules and practices of XP, covering all development aspects (manage, design, test & code) the XP way. Three real-world case studies and five workshop exercises ensure a practical appreciation of XP.

Agile Analysis: JAD to Prototype (1 day)

This course focuses on the key Agile techniques that facilitate true collaboration to ensure the system actually meets the user's requirements. A practical mini-JAD (Joint Application Development) session is conducted, as well as workshop exercises in Domain Object Modelling, User Stories and Behaviour Driven (BDD).

Agile Testing: TDD and more (1 day)

Practical guidance and real-world examples on key testing techniques, roles and environments to work Agile. The techniques covered show how to use "tests" as executable specifications through to building an automated test suite for continuous regression testing. Test-Driven Development (TDD) is the core focus, complimented with activities and case studies on the Given-When-Then and Decision Table approaches to ATDD as well as Test Patterns, Architectural Spikes, Automated Builds and Peer Review techniques.

Code Inspection (half day)

A very hands-on training workshop, with review of the different forms of peer review, followed by a practical code inspection (in pairs) of the attendees' code they've brought to the workshop.

Object-Oriented

OO Principles and Patterns (1 day)

Two fundamental aspects of designing object-oriented software are explained – OO Design Principles are the underlying guidelines for all OO designs; and Design Patterns are the essence of reuse. A range of principles and patterns are presented with two workshop exercises undertaken to cement this understanding.

Software Architecture (2 days)

An advanced workshop giving practical insights into "architecturally centric" software development. Real world and analogous examples are reviewed as well as eight group-based workshop exercises undertaken.

UML in a Nutshell (1 day)

A comprehensive introduction to the Unified Modelling Language (UML). Several real-world examples are examined and there are four workshop exercises in the key UML notations: Use Case, Class, Sequence and State Machine.

