

OO20

1 day

Introduction to the UML 2

This course introduces the UML 2 and its application to modern software development processes. Topics covered include OO overview and the conversion of Requirements into UML diagrams and documents which directly control the building of successful software product. The course identifies the Business Benefits of the UML approach and shows how UML is used throughout the SDLC, from Requirements capture to Acceptance testing.

Demonstrations include Enterprise Architect and Visual Use Case. The practical use of the UML in working from Requirements Gathering through to Delivery and Test is emphasized, helping participants deliver high-quality surprise-free systems.

Course Objectives

- Explain Object Oriented design and development techniques and terminology
- Describe how the UML works with OO technology to improve the way we build software
- Show how the application of the UML leads to superior products
- Describe Requirements capture and management with Use Cases
- Show how static and dynamic behaviour of programs can be described in the UML
- Show how Design Tools such as Rational Rose® and TogetherSoft® work together with the UML to produce executable code.
- Show how the UML integrates with a project-wide Testing Strategy
- Show how the UML provides a smooth flow from Requirements to Construction and Transition

Audience

- Staff already familiar with OO principles, but need a UML 2.0 grounding.
- Management wanting to understand the project issues of UML
- Technicians wanting to improve their technical performance
- Senior staff needing to examine the potential of UML for their organization
- Anyone concerned with Requirements Gathering and Program Design

Prerequisites

- General knowledge of software development process.
- Keen to provide more professional IT services

Timetable

Register at 09:00 for 09:30 start on Day1

Presentation Style

Lectures, demonstrations, group discussions and exercises

Dates and Venues

Refer to *Course Schedule*.

OO20

1 day

Introduction to OO Analysis and Design using UML 2.0

The course covers:

UML Overview

- 📁 Origins and purpose of UML
- 📁 Versions of UML including 2.0
- 📁 A walkthrough of the UML diagrams
- 📁 Using UML diagrams

OO Analysis and Design

- 📁 Analysis versus design
- 📁 The analysis phase
- 📁 Domain modelling
- 📁 Use Cases in analysis
- 📁 The design phase
- 📁 Interface versus implementation

Use Cases

- 📁 What are Use Cases
- 📁 Constructing Use Case diagrams
- 📁 Requirements and Use Cases
- 📁 Documenting Use Cases
- 📁 Use Cases, Alternate and Exceptions
- 📁 Scenarios
- 📁 Test Cases and Test Case

Static Modelling

- 📁 Classes and interfaces
- 📁 Class diagrams
- 📁 Object diagrams
- 📁 Class relationships
- 📁 Robustness analysis and Class Diagrams
- 📁 Association
- 📁 Aggregation and composition creation from Scenarios

Dynamic Modelling

- 📁 Introduction to UML dynamic modelling
- 📁 Activity diagrams
- 📁 Sequence diagrams
- 📁 Collaboration diagrams
- 📁 Statecharts and Statetables
- 📁 Decision Tables
- 📁 Code generation from tables

Component and Deployment Models

- 📁 Using Packages to organise your system
- 📁 Using components to model physical organisation
- 📁 Modelling hardware Architecture

What are Patterns?

- 📁 Designing with Patterns
- 📁 Frameworks
- 📁 Singleton
- 📁 MVC for GUI
- 📁 JUNIT
- 📁 Pros and Cons of Patterns

Summary and Conclusion

- 📁 A strategy for Beginning
- 📁 Where to go for Information
- 📁 Avoiding mistakes
- 📁 Commonsense approach to introducing the UML