

Introduction to UML for Java Developers

Duration:	3 days
Type:	intermediate

Description

This course teaches delegates how to use the UML to design Java and JEE Applications. In contrast to a generic UML course all examples and exercises are based around the needs of Java developers.

All of the UML diagrams are introduced during the course, including the new features of UML 2.0. In addition to designing standard JSE / JEE systems the course also covers modeling applications which use frameworks such as JSF, Spring and Hibernate.

Prerequisites

This course is designed for experienced Java programmers who have a good understand of the fundamentals of JEE development

List of Modules

Introduction to the UML

- The origins and uses of UML
- The 4 + 1 views of your system
- An overview of each UML diagram
- Review of popular UML tools
- Using UML within Agile development
- Using UML within the Unified Process

Requirements Analysis

- Introducing Actors and Use Cases
- Constructing a Use Case Diagram
- Basic and alternative flows of events
- Writing Use Case Reports
- Using Activity Diagrams to clarify requirements

The Dynamic Logical View

- Why Interaction Diagrams come first
- Allocating responsibilities to classes
- The structure of Sequence and Collaboration Diagrams
- Guidelines for creating clear and useful Interaction Diagrams
- Creating sequence diagrams for a typical Spring/Hibernate based Web Application
- Coping with iteration, selection and concurrency on Sequence Diagrams

The Static Logical View

- The structure of a Class Diagram
- Generating Class Diagrams from Interaction Diagrams
- Adding fields and accessibilities
- Allocating classes to packages
- Creating associations between classes
- Choosing between aggregation and composition
- Creating dependencies between classes
- The importance and implications of navigation and multiplicity
- Using stereotypes to describe class types and JEE specific relationships
- Using Statecharts to clarify the lifecycle of JEE components

Architecture and JEE Design Patterns

- Representing a layered architecture
- Using UML diagrams to describe standard solutions to common problems
- Introducing J2EE Patterns via UML

Designing Advanced Components

- Designing Tag Libraries in UML
- Designing JSF and Spring MVC based Web Apps
- Designing JPA and Hibernate based Data Access Objects
- Adding transactions and security checks to your design

The Component and Deployment Views

- Review of JEE deployment strategies
- Representing JEE Modules using Component Diagrams
- Documenting dependencies between JEE Modules and 3rd party libraries
- Using Deployment Diagrams to describe development, test and production setups