



# UML Training Courses from CRaG Systems

sales@cragssystems.co.uk +44 (0)845 003 9358



## Object Oriented Analysis and Design using UML Training Course - OOAD UML - 5 Days

This UML training course is aimed at system architects, analyst/programmers and developers who want to define detailed outside-in system requirements using use cases, a detailed object oriented implementation-free model of the system from those requirements and a layered, component-based, model of system architecture and design in order to maximise the maintainability, re-use and extensibility of the resulting code. The industry best practice modelling techniques are based on the Unified Modelling Language v2.3 and are taught within the context of a [model-driven software development process](#). The modelling employed is applicable to a wide variety of technologies, sufficiently detailed for code generation and traceable from requirements through to code. The advantages that using these techniques has for estimation, test development and management is discussed. Each technique is taught to the level required for competence on a real project. Understanding is tested with exercises based on a real-world project example either [using a suitable UML modelling tool](#) or on paper.

### Delegates will learn:

- The basics and the necessary detail of the Unified Modelling Language
- The basics and the detail of Object Orientation
- How to create a first cut overview of functional requirements with actors and use cases on a use case diagram
- How to write an effective use case description in a way that satisfies both non-technical and technical stakeholders
- How to restructure the use case diagram to handle complex relationships between use cases without bloating the use case model
- How to integrate the use case model with non-functional requirements, data requirements, business rules and screen prototyping
- How to create a detailed model of system data using classes and their relationships
- How to recognise complex data constructs and to use the appropriate syntax to model them
- How to map the functionality of the system requirements onto the object model using sequence diagrams
- How to structure the modelling in the form of a use case implementation
- How to model the dynamics of system data and functionality using statecharts
- How to model at a consistent level of abstraction
- How the modelling performed during system analysis fits into an incremental model-driven development process
- How to develop a flexible system architecture from an object oriented analysis model
- How to develop component and deployment models for the system
- How to model the design of a component using sub-systems and interfaces
- How to build libraries of re-useable classes using generalisation and inheritance
- How to model the use of technology and frameworks as series of design patterns
- How to integrate libraries and frameworks into the design of components
- How to generate frame code and keep the design and code models synchronised
- How to specify, analyse, architect, design, build, test and deliver components as part of an incremental model-driven development process

**Suitable for:**

System Architects, System Designers, Developers and Analyst/Programmers with at least 2 years experience, preferably in a modern object-oriented language. This course is not suitable for those seeking certification as a step towards a qualification. See the [Certification Policy](#) for a detailed discussion.

**Course Logistics:**

Course attendance is limited to 12 students. Courses start at 9.30am on the first day, 9.00am on subsequent days and finish at 5.00pm each day. Students normally use a computer for the exercises, but these can be performed on paper if required. For a discussion on using a UML modelling tool please see [Modelling Tool Use on Courses](#). Printed course manuals for each student with copies of all presentations, exercises and solutions are provided.

**On-Site (In-House) Courses:**

The client is expected to provide an appropriate venue, refreshments, XGA/WXGA projector and screen, whiteboard or flipchart and at least one computer per two students loaded with a suitable UML modelling tool, unless exercises are to be performed on paper. For a full discussion of on-site course issues please see [On-Site Course Logistics](#).

**Scheduled Public Courses:**

This course is currently not available as scheduled public training. Please see the [Scheduled Public Courses](#) page for available courses.

**Pricing:**

On-site (in-house) course pricing is available from the [On-Site Course Price Calculator](#) page. Public course pricing is available on the [Public UML Training Courses in London](#) page. For consultancy pricing please see the [On-Site Consultancy Price Calculator](#).

## Training Course Outline

<b>Day 1</b>	<b>Day 2</b>
<p><b>Introduction</b></p> <p>People - Course Structure - Object Orientation - The Unified Modelling Language - Use Cases and System Analysis - A Process for Modelling</p> <p><b>Specifying Functional Requirements with Use Cases</b></p> <p>System Use Cases and Actors - Primitive Use Cases and the Basic Flow - Writing Effective Use Case Descriptions - Writing Sub-flows and Alternate Flows - 'Include' and 'Extend' Relationships - Modelling Browser-Based Applications <i>System Use Case Workshop</i></p> <p><b>Requirements Gathering</b></p> <p>Collecting Requirements Information - Mapping from the Business Model - Proof of Concept Prototypes - Requirements Documents - Estimating and Traceability - Incremental Development - Gaining Agreement</p>	<p><i>Requirements Gathering Workshop</i></p> <p><b>Objects and Classes</b></p> <p>What is an Object? - Classes and Objects - Attributes - Operations and Methods - Designing Good Classes - Choosing the Right Classes <i>Object and Class Workshop</i></p> <p><b>Object Relationships</b></p> <p>Associations and Links - Navigability and Naming - Multiplicity and Other Adornments - Association Classes and N-arys - Aggregation and Composition <i>Object Relationship Workshop</i></p>

<p><b>Day 3</b></p> <p><b>Interaction Modelling</b></p> <p>Interactions and Messages - Operations and Methods - Sequence Diagrams - Selection and Iteration - Activation - Communication Diagrams <i>Interaction Modelling Workshop</i></p> <p><b>State Modelling</b></p> <p>The Meaning of the State Model - States and Transitions - Events and Conditions - Actions and Activities - Consistency With Other Diagrams <i>State Modelling Workshop</i></p> <p><b>System Analysis</b></p> <p>Creating the Initial Object Model - Reverse Engineering a Data Model - The Analysis Cycle - Iterative Modelling - Prototyping as an Analysis Technique - Completing the Model</p>	<p><b>Day 4</b></p> <p><i>System Analysis Workshop</i></p> <p><b>Architecture, Components and Implementation Diagrams</b></p> <p>Packages and Dependencies - Stereotypes - Control Objects - Layered Architectures - Interfaces, Subsystems and Components - Task Modelling - Component Diagrams - Deployment Diagrams <i>System Architecture Workshop</i></p> <p><b>Class Relationships and Inheritance</b></p> <p>Class Similarities and Differences - Generalisation Syntax - Generalisation Hierarchies - Multiple Inheritance - Polymorphism - Generalisation Versus Interface - Class Dependency <i>Generalisation Workshop</i></p>
--	---

<p><b>Day 5</b></p> <p><b>Design Patterns</b></p> <p>Reflexive Aggregate - Collection Class - Observer - State Machine - Meta-Model - Modelling Patterns <i>Design Patterns Workshop</i></p> <p><b>Detailed Design</b></p> <p>Subsystem Design - Architectural Mechanisms - Linking to Libraries and Frameworks - Visibility and Other Properties - Completing the Model - Incremental Development - Traceability and Review <i>Detailed Design Workshop</i></p>	
--	--

Should the content of this UML training course not fit your exact requirements, then CRaG Systems can create a [custom course](#) for you. Please either email or call us to discuss your particular needs.

CRaG Systems (UK) [sales@cragssystems.co.uk](mailto:sales@cragssystems.co.uk) +44 (0)845 003 9358

Object Oriented Analysis and Design using UML Training Course - OOAD UML