Introduction to Modeling with UML

COURSE DESCRIPTION:

This course offers a highly concentrated look at the major forces, technologies, processes, tools, risks, and benefits associated with adopting object and component technologies. An expert practitioner facilitates the program, bringing real-world experience in strategizing, architecting, designing, and delivering robust object-oriented systems. The course is intentionally broad enough to introduce the major issues, while offering sufficient detail to be directly usable by emerging teams.

COURSE OBJECTIVES:

1. Understand how to identify and classify the objects in business problems
2. Understand how to model business data, behavior, rules and constraints using object-oriented constructs
3. Be introduced to the notation and semantics of class modeling and sequence diagramming
4. Understand the basic concepts of object-oriented software architecture
5. Define the processes used in analysis and design
6. Be introduced to design patterns

TOPICS COVERED:

1. What is an object and what is object-orientation
2. Roadmap of object-oriented analysis process
3. Core analysis and design competencies of class modeling, sequence diagramming, and state modeling
4. Architecture overview including logical and physical software architecture, process and thread architecture, and hardware architecture
5. Roadmap of object-oriented design process
6. Partitioning a system into layers and subsystems
7. Use of patterns

AUDIENCE:

This course is designed for systems analysts, architects, designers, developers, and testers who are directly responsible for developing object-oriented systems. This course also benefits managers, technical leads and software quality assurance personnel who oversee development of UML-driven systems and require an understanding of the process and associated artifacts.

PREREQUISITES:

None

DURATION:

1 day

COURSE OUTLINE

1. INTRODUCTION
    - What are objects and object-orientation?
    - How do we do object-oriented work?

2. REQUIREMENTS: HOW ARE REQUIREMENTS DOCUMENTED?
    - Review of key requirements documents:
        - Stakeholder interviews
        - Vision document
        - Use Case Model
        - Supplemental Specification document

3. ANALYSIS: HOW DO WE ANALYZE THE REQUIREMENTS?
    - Steps in the analysis process
    - Introduction to Class Models, Sequence Diagrams, and State Models

4. ANALYSIS: HOW DO WE IDENTIFY CLASSES?
    - Discussion of what things can be objects
    - Techniques for finding objects

5. ANALYSIS: HOW DO WE DEVELOP CLASS DIAGRAMS?
    - Syntax and semantics: classes, attributes, operations, and associations
    - Process

6. ANALYSIS: HOW DO WE DEVELOP SEQUENCE DIAGRAMS?
    - Syntax and semantics: objects, messages, focus of control, constraints
    - Process

7. ARCHITECTURE: HOW DO WE ARCHITECT A SYSTEM?
    - 4+1 view of architecture
    - Logical View: significant business classes, system partitioning, patterns, etc.
    - Process View: processes and threads
    - Implementation View: mapping of logical entities into physical files
    - Deployment View: hardware layout, connectivity, and communications

8. ARCHITECTURE: HOW DO WE DEFINE SYSTEM PARTITIONING?
    - Define layering and explore common layering schemes
    - Define subsystems and discuss reference architectures

9. DESIGN: HOW DO WE DESIGN A SOLUTION?
    - Steps in the design process

10. DESIGN: HOW DO WE SELECT AND APPLY PATTERNS?
    - Define patterns
    - Explore representative patterns: Composite, Observer, State, Model-View-Controller

Copyright 2002,
IconMedialab International