

Course: Object-Oriented Analysis and Design Using UML

Course Outline

Examining Object-Oriented Concepts and Terminology

- Describe the fundamental OO terminology
- Describe the important object-oriented (OO) concepts

Introducing Modeling and the Software Development Process

- Describe the Object-Oriented Software Development (OOSD) process
- Explain the purpose, activities, and artifacts of the following OOSD workflows (disciplines): Requirements Gathering, Requirements Analysis, Architecture, Design, Implementation, Testing & Deployment
- Describe the benefits of modeling software
- Describe how modeling supports the OOSD process

Creating Use Case Diagrams

- Justify the need for a Use Case diagram
- Identify and describe the essential elements in a UML Use Case diagram
- Develop a Use Case diagram for a software system based on the goals of the business owner
- Develop elaborated Use Case diagrams based on the goals of all the stakeholders
- Recognize and document use case dependencies using UML notation for extends, includes, and generalization
- Describe how to manage the complexity of Use Case diagrams by creating UML packaged views

Creating Use Case Scenarios and Forms

- Identify and document scenarios for a use case
- Describe how to reference included and extending use cases.
- Identify the purpose of a Supplementary Specification Document
- Create a Use Case form describing a summary of the scenarios in the main and alternate flows
- Identify and document non-functional requirements (NFRs), business rules, risks, and priorities for a use case

Creating Activity Diagrams

- Model a Use Case flow of events using an Activity diagram
- Identify the essential elements in an Activity diagram

Determining the Key Abstractions

- Identify the key abstractions using CRC analysis
- Identify a set of candidate key abstractions

Constructing the Problem Domain Model

- Identify the essential elements in a UML Class diagram
- Validate the Domain model with one or more Object diagrams
- Construct a Domain model using a Class diagram
- Identify the essential elements in a UML Object diagram

Transitioning from Analysis to Design using Interaction Diagrams

- Create a Sequence diagram view of the Design model
- Identify the essential elements of a UML Sequence diagram
- Create a Communication diagram view of the Design model
- Identify the essential elements of a UML Communication diagram
- Explain the purpose and elements of the Design model

Modeling Object State Using State Machine Diagrams

- Model object state
- Describe the essential elements of a UML State Machine diagram

Applying Design Patterns to the Design Model

- Describe the Composite pattern
- Define the essential elements of a software pattern
- Describe the Strategy pattern
- Describe the Abstract Factory pattern
- Describe the Observer pattern

Introducing Architectural Concepts and Diagrams

- Describe the Architecture workflow
- Select the Architecture type
- Create the Architecture workflow artifacts
- Describe the diagrams of the key architecture views
- Describe tiers, layers, and systemic qualities
- Distinguish between architecture and design

Introducing the Architectural Tiers

- Describe the concepts of the Client and Presentation tiers
- Describe the concepts of the Business tier
- Describe the concepts of the Resource and Integration tiers
- Describe the concepts of the Solution model

Refining the Class Design Model

- Annotate method behavior
- Create components with interfaces
- Refine the attributes of the Domain model
- Declare the constructors of the Domain model
- Refine the relationships of the Domain model
- Refine the methods of the Domain model

Overview of Software Development Processes

- Develop an iteration plan
- Choose a methodology that best suits your project
- Explain the best practices for OOSD methodologies
- Describe the features of several common methodologies

Overview of Frameworks

- Define a framework
- Understand the concept of creating your own business domain frameworks
- Describe the advantages and disadvantages of using frameworks
- Identify several common frameworks