

## Course duration

- 3 days

## Course Benefits

- Learn all about design patterns including background and key features.
- Learn design pattern strategies.
- Learn testing strategies.
- Learn about different creational patterns and architectural patterns and styles.

### Available Delivery Methods

#### Public Class

Public expert-led online training from the convenience of your home, office or anywhere with an internet connection. Guaranteed to run .

#### Private Class

Private classes are delivered for groups at your offices or a location of your choice.

## Course Outline

1. Dealing with Complexity
  1. Functional Decomposition
  2. Requirements and Inevitable Change
  3. Coupling and Cohesion
  4. Unwanted Side Effects
  5. Perspectives
  6. Responsibilities
2. Object-Oriented Paradigm
  1. Objects and Responsibilities
  2. Single Responsibility Principle (SRP)
  3. Interfaces and Abstract Classes
  4. Encapsulation and Polymorphism
  5. Liskov Substitution Principle (LSP)
  6. Object Construction and Destruction
  7. Classes vs. Structs in .NET
3. Overview of UML
  1. Diagram Types

2. Class Diagrams
3. Sequence Diagrams
4. Introduction to Design Patterns
  1. Origin of Design Patterns
  2. "Gang of Four" Patterns
  3. Key Features of Patterns
  4. Why Study Design Patterns?
  5. Design Strategies
5. Structural Patterns (Part I)
  1. Facade Pattern
  2. Adapter Pattern
  3. Facade vs. Adapter
6. Testability
  1. Introduction
  2. Unit Tests
  3. Integration Tests
  4. Test-Driven Development (TDD)
  5. Dependencies
7. Behavioral Patterns (Part I)
  1. Handling New Requirements
  2. Open-Closed Principle (OCP)
  3. Strategy Pattern
  4. Template Method Pattern
8. Structural Patterns (Part II)
  1. Decorator Pattern
  2. Proxy Pattern
9. Behavioral Patterns (Part II)
  1. Observer Pattern
  2. .NET Events
  3. Command Pattern
  4. WPF and ICommand
10. Creational Patterns
  1. Motivation
  2. Factories
  3. Singleton Pattern
  4. Object Pool Pattern
  5. Factory Method Pattern
  6. Abstract Factory Pattern
11. Model-View-Controller (MVC)
  1. Model-View-Controller (MVC)
  2. Model-View-Presenter (MVP)
  3. Model-View-View-Model (MVVM)
12. Architectural Patterns and Styles
  1. Component-Based Architecture
  2. Layered Architecture
  3. .NET Assemblies and Versioning
  4. N-Tier Architecture

## 5. Service-Oriented Architecture (SOA)

### Class Materials

Each student will receive a comprehensive set of materials, including course notes and all the class examples.

### Class Prerequisites

Experience in the following *is required* for this ASP.NET class:

- Extensive prior experience developing applications for the .NET platform.