



## **COURSE CONTENT**

## COURSE NAME

**Full Stack Angular 18 with C# and ASP.NET Core 8**

## COURSE OVERVIEW

<u>Training</u>	<u>Duration (Days)</u>
Angular 18	3
C#	2
ASP.NET Core Web API	2
Capstone project	3

## DURATION

**10 Days**

## COURSE OUTLINE

### Angular 18

#### **Introduction to Angular 18**

- Overview of Angular and its ecosystem
- What's new in Angular 18, including the latest updates to Angular DevTools
- Setting up the development environment
- Migrating from older versions of Angular

#### **Project Setup and Architecture**

- Using Angular CLI
- Project structure and file organization
- Angular modules and components

#### **Templates and Data Binding**

- Angular templates syntax
- Interpolation, property binding, and event binding
- Two-way data binding with ngModel

#### **Components and Directives**

- Creating and using components
- Component lifecycle hooks

- Built-in directives (ngIf, ngFor, etc.)
- Custom directives

### **Services and Dependency Injection**

- Creating and using services
- Dependency injection in Angular
- Managing application state

### **Routing and Navigation**

- Setting up the Angular router
- Configuring routes and route parameters
- Understanding and implementing new routing features
- Lazy loading and route guards

### **Forms in Angular**

- Template-driven forms
- Reactive forms
- Form validation and error handling

### **HTTP Client and Observables**

- Making HTTP requests with HttpClient
- Handling HTTP responses
- Working with RxJS observables
- Authentication and Authorization

## **C#**

### **Introducing C# and the .NET Framework**

- Introduction to the .NET Framework
- Creating Projects Within Visual Studio 2010
- Writing a C# Application
- Running and Debugging Applications by Using Visual Studio 2010

### **Using C# Programming Constructs**

- Declaring Variables and Assigning Values
- Using Expressions and Operators
- Creating and Using Arrays
- Using Decision Statements

- Using Iteration Statements
- Defining and Invoking Methods
- Lab: Looping, branching and using methods

### **Handling Exceptions**

- Handling Exceptions
- Raising Exceptions
- Lab: Handling Exceptions

### **Creating New Types**

- Creating and Using Enumerations
- Creating and Using Classes
- Creating and Using Structs
- Comparing References to Values
- Lab: Creating New Types

### **Encapsulating Data and Methods**

- Controlling Visibility of Type Members
- Sharing Methods and Data
- Lab: Encapsulating Data and Methods

### **Inheriting From Classes and Implementing Interfaces**

- Using Inheritance to Define New Reference Types
- Defining and Implementing Interfaces
- Defining Abstract Classes
- Lab: Inheriting From Classes and Implementing Interfaces

### **Decoupling Methods and Handling Events**

- Func and Action
- Declaring and Using Delegates
- Using Lambda Expressions
- Lab: Decoupling Methods and Handling Events

### **Using Collections and Building Generic Types**

- Using Collections
- Creating and Using Generic Types
- IEnumerable, ICollection, IList

- List, Dictionary, HashSet
- Lab: Building Generic Types
- Lab: Using Collections

### **Using LINQ to Query Data**

- Using the LINQ Extension Methods and Query Operators
- Building Dynamic LINQ Queries and Expressions
- Lab: Using LINQ to Query Data

## **ASP.NET**

### **ASP.NET Core And REST**

- ASP.NET Core Web API
- Representational State Transfer
- REST and Web API
- HTTP Services Using Web API
- HTTP Testing Tools
- Using Postman
- HTTP Response Codes

### **Basic introduction to ASP.NET Core MVC**

- Introduction to .NET Core and older .NET frameworks
- Configuring for ASP.NET Core MVC
- Rendering Views
- Razor View Engine
- Dynamic Output

### **ASP.NET MVC and ASP.NET Web API Architectures**

- The Controller in ASP.NET MVC
- The View in ASP.NET MVC
- The Model in ASP.NET MVC

### **The Model**

- More Complex Models in MVC Programs
- Microsoft Technologies for Model Persistence
- NuGet Package Manager

## The Controller

- Controller and ControllerBase base classes
- Actions
- Retrieving Data from a Request
- Action Results
- Action Attributes
- Serving Static Files
- Filters

## The View

- View Responsibility
- HTML Helpers
- Validation Attributes

## Routing

- Routing in ASP.NET Core MVC
- Properties of Routes
- Parameters in Routing
- Registering Routes
- Attribute Routing
- Authentication and Authorization

## Capstone Project