

ASP .NET Core con Angular 9

Semanas: 8 (2 meses)

Acerca del Curso

Este curso te permitirá construir la siguiente generación de aplicaciones web y web APIs. Aquí aprenderemos un framework que es más rápido que Node.js y Java Servlets. Este curso cubre buenas prácticas desde cero a un nivel avanzado con laboratorios prácticos y un proyecto final que cubre todos los tópicos aprendidos. Esta es tu oportunidad para sumergirte en Angular 9, ASP .NET Core, MVC, Web API, Entity Framework Core, Unit Testing, Seguridad y patrones de diseño recomendados etc.

Ruta de Aprendizaje

Semana 1 y 2: ASP .NET Core

Semana 3: LinQ y Entity Framework Core

Semana 4: Patrones de Diseño y Unit Testing

Semana 5: Angular y Web API

Semana 6 y 7: Desarrollo del Proyecto Final

Semana 8: Finalización del Proyecto y Entrega de Certificados.

Metodología

- Martes y Jueves se liberan los vídeos de cada clase a las 7:30 PM.
- Soporte vía Slack.joedayz.pe de los instructores.

Objetivos del Curso

Al terminar este curso, los asistentes estarán en la facultad de:

1. Entender arquitectura .NET Core y construir aplicaciones multi-plataforma para Windows, Mac y Linux
2. Uso e implementación de ASP .NET Core como middleware, routing, validaciones, filtros, etc.
3. Usar EF Core Migration e Ingeniería Reversa
4. Entender e implementar un Repository, Unit of Work y patrón de diseño Inyección de Dependencias
5. Escribir y Depurar Unit Test cases usando xUnit, MS Test, Moq

6. Implementar Angular core y conceptos avanzados con ASP .NET Core
7. Manejar y hacer logging de errores en ASP .NET Core usando NLog
8. Desarrollar aplicaciones SPA (Single Page Applications) usando Angular y ASP .NET Core
9. Seguridad de aplicaciones usando ASP .NET Core Identity
10. Crear y asegurar REST API usando token based security
11. Desplegar sus Apps on IIS/Cloud

Pre-requisitos

Conocimientos de Fundamentos de Programación.

Conocimientos del lenguaje de programación C#

Herramientas/IDE

Visual Studio 2019 o Rider

Contenidos del Curso

ASP .NET Core

Introducción a .NET Core

1. Entender .NET Core
2. Características de .NET Core
3. Arquitectura del Framework .NET Core
4. .NET Core 1.0 vs .NET 4.5
5. Soporte .NET Core
6. Ventajas de .NET Core
7. Modelo de una aplicación .NET

.NET CLI

1. Qué es DNX, DNVM, DNU
2. Introducción a .NET Cli
3. Comandos .NET Cli

Introducción a ASP.NET Core

1. Entendiendo ASP .NET Core

2. Características de ASP .NET Core
3. ASP .NET Core 1.0 vs ASP.NET MVC 5 vs ASP.NET WebForms
4. Ventajas de ASP .NET Core

Visual Studio Project Templates

1. Entender los templates de Visual Studio ASP.NET Core
2. Crear un proyecto ASP .NET Core
3. Entender la estructura de directorios de un proyecto ASP .NET Core
4. Entender los archivos de configuración

Modelo, Vista, Controladora y Actions

1. Entender el Modelo, Vista y Controladora
2. Tipos de Vistas
3. Creando Controladoras
4. Entendiendo Actions
5. Actions y Métodos Non-Actions
6. Entendiendo Action Results
7. Comunicación entre Controller y View

Motor de Vista Razor

1. Entendiendo el motor de vista Razor
2. Motor de vista Razor vs Motor Web Form
3. Sintaxis Razor
4. Sentencias Razor, Loops, etc.

Rutas & URLs

1. Introducción a Routing
2. Definir Rutas
3. Atributos de Rutas
4. Necesidad de atributos de rutas

Helpers

1. Entendiendo Html Helpers
2. Tipos de Html Helpers
3. Construir Html Helpers
4. Tag Helpers
5. Inline Helpers

6. Helpers personalizados
7. Url helpers

Recibiendo datos del lado del servidor

1. Parametros Action
2. View Models/Objects
3. FormCollection

Componentes de vista Reutilizables

1. Partial View
2. Componentes View

ViewModel y Validación

1. Creando ViewModel
2. Entendiendo validaciones ASP .NET Core MVC
3. Necesidad de validación del lado del servidor y cliente
4. Validación con anotaciones de Data

Validación personalizada

1. Validación personalizada del lado del cliente
2. Validación personalizada del lado del servidor

Técnicas para pasar datos

1. ViewData
2. ViewBag
3. TempData
4. Session
5. Query String
6. Cookies

Introducción a Web API

1. Introducción a SOA
2. Introducción a REST
3. Principios REST
4. Entendiendo Web API
5. Escoger entre WCF y WebAPI

Web API Controller y Actions

1. Entendiendo Controller y Actions
2. Creando Controller
3. Creando Actions

Operaciones de Base de Datos y Postman

1. Operaciones CRUD usando Web API
2. Usando Postman para probar Web API
3. Consumir Web API usando ASP .NET Core

Entity Framework Core

Introducción a Entity Framework Core

1. Qué es un ORM
2. ORMs usados con .NET
3. EF6 vs EF Core
4. Ventajas de Entity Framework

Modelamiento de Base de DATos

1. Definiendo Mapping usando anotaciones
2. Implementando operaciones CRUD
3. Definiendo Mapping usando Fluent API

Migración de Base de Datos, Procedimientos de BD y Funciones

1. Primeras migraciones con Entity Framework
2. Modificando la base de datos cuando el modelo cambia
3. Llamada a Stored procedures y Functions
4. Generar código con base de datos existente

Patrón de Diseño Repository, Unit of Work

1. Entendiendo Patrones de Diseño Repository y UOW
2. Necesidad del patrón de Diseño Repository
3. Necesidad del patrón de Diseño UOW
4. Implementando Repository y UOW

Inyección de Dependencias

1. Entendiendo Inyección de Dependencias
2. Necesidad de Inyección de Dependencias
3. Implementando ID

Fundamentos de TypeScript

TypeScript

1. Problemas con Plain JavaScript
2. Introducción con TypeScript
3. Ventajas de TypeScript
4. Iniciando con TypeScript

Tipos de Datos y Variables

1. Tipos de Datos de TypeScript
2. Declaración de Variables
3. Var vs let vs const

Funciones y Arrow Function

1. Tipos de Funciones
2. Arrow Function
3. Sobrecarga de Function

Clases e Interfaces

1. Clases
2. Modificadores de Acceso
3. Constructor
4. Herencia

Genéricos y Módulos

1. Genéricos
2. Módulos

Angular 2 a 9

Introducción a Angular

1. Entendiendo Angular
2. Ventajas de Angular
3. Angular 1 vs Angular9
4. Soporte de Navegadores

Iniciando con Angular 2

1. Configurando aplicación Angular
2. Ejecutando aplicación Angular
3. Entendiendo estructura de directorios Angular
4. Proceso de inicialización de Angular
5. Introducción a Componentes
6. Creando componentes
7. Bootstrapping el Componente

Databinding

1. Entendiendo Data Binding
2. Binding de atributos
3. Binding bidireccional
4. Binding de eventos
5. Escogiendo el binding de datos apropiado

Angular CLI

1. Introducción a CLI
2. Creando nuevos project-ng new y ng init
3. Construyendo proyect-ng build y ng serve
4. Opciones de Ng CLI
5. Ng CLI para creación
6. Componentes, servicios, directivas y Pipes, etc.

Directivas y Pipes

1. Construir Directivas ngIf, ngFor, ngSwitch, etc.
2. Creando directivas personalizadas
3. Angular Pipes
4. Creando Pipes personalizados

Routing

1. Entendiendo Routing
2. Definir rutas
3. Rutas anidadas
4. Atachando Rutas a Actions
5. Definiendo Templates

Componentes Avanzados y Directivas

1. Hooks del Ciclo de vida de componentes
2. Componentes anidados
3. Compartiendo datos entre componentes anidados

Herencia

1. Introducción a herencia en Angular
2. Herencia de componentes

Formularios y controles

1. Introducción a Angular Forms
2. Tipos de Angular Forms – Orientados a Template – Orientados a Model
3. Orientados a Template vs Orientado a Model

Orientados a Template

1. Creando formularios orientados a template
2. Controles de entrada
3. Validaciones de formulario
4. Formulario y propiedades de controles

Orientados a Model

1. Creando formularios orientados a Model
2. Controles de entrada
3. Validaciones de formulario
4. Formulario y propiedades de controles

SPA

1. Introducción a SPA
2. SPA vs MPA
3. Los SPA más populares
4. Ventajas del SPA

Servicios

1. Entendiendo servicios Angular
2. Creando Servicios

Operaciones de base de datos con Servicios REST

1. Introducción a REST
2. Operaciones CRUD en un servicio REST

Unit Testing

Introducción a Unit Testing

1. Qué es Unit Testing
2. Ventajas de Unit Testing

Frameworks .NET para Unit Test

1. MS TEST
2. xUnit
3. NUnit

Test Driven Development (TDD)

1. Introducción a TDD
2. Usando TDD en una aplicación
3. Ventajas de TDD

Mocking

1. Introducción a Mocking
2. Frameworks Mocking – Moq, TypeMoq
3. Ventajas de Mocking

Unit Tests para Controller y Actions

1. Escribiendo unit tests para ASP .NET MVC Controller
2. Depurando Unit tests
3. Verificando el resultado actual y esperado

Visual Studio Live Unit Test

1. Introducción a VSLUT
2. Probando unit tests usando VSLUT

Proyecto Final

Crearemos un e-commerce. El sitio web mostrará la lista de productos para Usuario/Cliente. El usuario puede comprar productos y ver la orden. El administrador podrá agregar productos, categorías, y aprobar la orden del cliente. Se usará una arquitectura basada en capas.

La arquitectura del proyecto se hará con patrones de diseño recomendados y buenas prácticas.

