



.NET Core 7 with Angular 16 (5 days)



ในการสร้างเว็บแอปพลิเคชันสมัยใหม่ มีเครื่องมือภาษาและ framework ให้เลือกใช้มากมาย หนึ่งในภาษาที่มีประสิทธิภาพสูงคือ C# ของ Microsoft ด้วยลักษณะตัวภาษาเป็น OOP ตั้งแต่เริ่ม ดังนั้นจึงเหมาะสมกับงานขนาดเล็กไปจนถึงระบบใหญ่ๆ ที่มีความซับซ้อน การทำงานร่วมกันเป็นทีมได้อย่างดี โดยทาง Microsoft ได้พัฒนา Framework ที่ไว้สำหรับทำงานบนเว็บที่ชื่อว่า .NET Core ซึ่งทำงานภายใต้ .NET Framework และในปัจจุบันยังเปิดเป็นโอเพนซอส สามารถ

ทำงานได้ทั้งบน Windows, MacOS และ Linux จึงเปิดกว้างให้นักพัฒนาทุก Platform สามารถเข้ามาใช้งานได้สะดวกขึ้นมาก

เนื้อหาการอบรมในหลักสูตรนี้ จะเป็นการพัฒนาเว็บแอปพลิเคชันแบบ "Full Stack" กล่าวคือจะใช้ Angular 16 เป็นส่วน Frontend และใช้ ASP.NET Core (ปัจจุบันเปลี่ยนชื่อเป็น .NET) ล่าสุดเป็นส่วน Backend โดยเขียนเป็น API ให้ Angular เรียกใช้งาน ร่วมกับฐานข้อมูลยอดนิยม ทั้ง MS SQL Server, MySQL, PostgreSQL และ MongoDB ทำงานผ่าน Entity Framework Core ที่ทาง Microsoft พัฒนามาให้ทำงานกับฐานข้อมูลได้อย่างง่ายดาย

วัตถุประสงค์:

- เข้าใจการทำงาน .NET Core 7 ร่วมกับ Angular 16
- เรียนรู้การใช้ Entity Framework Core เพื่อเชื่อมฐานข้อมูล
- สามารถทำเว็บแอปพลิเคชันแบบ Full Stack ได้

กลุ่มเป้าหมาย:

- นักเรียนนักศึกษา
- ครู อาจารย์ วิทยากรที่สนใจ
- นักวิชาการ นักไอที หรือผู้ดูแลระบบ
- ตลอดจนผู้สนใจทั่วไป

ความรู้พื้นฐาน:

- พื้นฐาน Visual C#
- พื้นฐานการค้นหาข้อมูลจากอินเทอร์เน็ต

ระยะเวลาในการอบรม:

- 30 ชั่วโมง (5 วัน)



ราคาคอร์สอบรม:

- 13,500 บาท / คน (ราคานี้ยังไม่ได้รวมภาษีมูลค่าเพิ่ม)

วิทยากรผู้สอน:

- อาจารย์สนิทวงศ์ กมลภากรณ์

คอร์สที่ควรอบรมก่อนหน้า:

- Basic Visual C#

เนื้อหาการอบรม:

1. Introduction

- What is Angular?
- Why Angular?
- What is a Single Page Application (SPA)?
- How Does the Angular + .NET Core Work?
- Angular vs React

2. Setup and Installation

- Installing Node JS
- Install Angular CLI
- Install VSCode
- Install Extensions VSCode
- Install ASP .NET Core
- Install Git
- Install Postman for API testing
- Configure VSCode
- Guided Tour of VSCode
- Using NPM (Node Package Manager)
- Angular CLI
- New Project



- Exploring the Project Structure
- Running Your WebApp
- Version Control by GitHub

3. Components & Templates

- Introduction to components
- Template Syntax
- Listening to Events
- Displaying Data
- User Input
- Lifecycle Hooks
- Diving Into Structural Directives
- Diving Into Attribute Directives

4. Begin POS Workshop

- Manage resource file (asset)
- Integrate Adminlte Theme
- Typescript Programming
- Create components
- Design Login and Register page
- Header, footer, menu

5. Routing

- Create Routing for Workshop
- Router outlet
- Routing with Parameters

6. Member Workshop

- FormsModule
- ReactiveFormsModule
- TailwindCSS + Angular
- Template-driven forms



- Form Validation
- Introduction to services
- Dependency injection (DI)
- Redirect and Back Location

7. Using the Angular HTTP Client

- HttpClientModule
- Manage Environment
- Create Model
- Mapping JSON to TS
- Observable

8. Stock Workshop

- Integrate Table
- Angular Form and 2-Ways Data Binding
- Angular Regular Expression with Pipe
- Learn Angular Material User Interface
- Typescript Programming

9. ASP .NET CORE WEB API

- What is a RESTful API?
- What is ASP .NET Core?
- Why ASP .NET Core?
- ASP .NET CORE CLI
- New Project
- Exploring the Project Structure
- Generate self cert
- Testing API
- Using Extension Template
- Route
- Request Handler
- Validation Request



- Parser Request (Query String, Path, Form, Json)
- Middleware
- Debug and Logging files

10. Working with SQL Server

- NuGet Package Manager
- Entity Framework Core tools .NET CLI
- Database First Migrations
- Database Context
- Manage Environment
- Service provider and Dependencies injection
- Entity Framework Core
- Using SQL Management on VSCode

11.Member API Workshop

- Repository Pattern
- Password Encryption
- CRUD (Create, Read, Update and Delete) with EFCore
- Using Postman for API Testing

12.Stock API Workshop

- Async and Await
- Upload Files
- Manage Static Files (Images, CSS, Html, etc.)

13.Final API Workshop

- Understanding SPA Authentication
- Protecting Resources with Authorization
- What is JWT?
- Generate and Validate JWT
- Setup CORS (Cross-Origin Resource Sharing)



14. Integrate Angular

- Saving the Token in the Local Storage
- Adding Route Guards
- Http Interceptor
- Using Sweet-Alert
- Adding an Error Dialog
- Adding an Image Preview
- Uploading Files
- Component Communication
- Data Exchange between Components both (Child and Parent)
- Event Emitter and Listener between Components
- Create Service (sharing data and functionality)
- Custom Pipes
- Stylesheet programming
- Finish Workshops and Features

15. Publishing on Docker

- Production Deployment
- Deploy Angular on Docker
- Deploy ASP .NET on Docker