

Containerization of Applications using Docker**Introduction to Containers and Docker**

- Understanding VM's and Containers.
- What is Docker?
- Docker Benefits.
- Docker Architecture and Docker Taxonomy.

Developing Docker Custom Images using .NET Core

- Docker Images for .NET Core.
- Executing .NET Core applications in Containers.
- Inspecting the Image Architecture
- Developing and Publishing .NET Core Applications
- Dockerfile and Building Docker Images
- Breaking down and understanding dockerfile indepth

Multi-Stage Builds

- Multiple stages in dockerfile
- Hosting ASP.NET Applications in Docker

State and Data in Docker Applications

- Purpose of using Data Volumes
- Access Data in Docker Containers
- Creating Container with Volumes
- Data volume containers

Docker Compose

- Overview
- Docker compose features
- Building docker-compose.yml file
- Docker-compose command
- Working with multiple images in a single application
- Environment Variables and Configuration File

Microservices Architecture – Get perfect in concepts and practical**Understanding Microservices**

- Understanding Monolithic Architecture
 - What are Monolithic Applications

- Deploying
- Containerizing using Docker
- Scaling Applications
- Managing State and Data
- Benefits and Drawbacks of Monolithic Architecture
- Microservices Architecture
 - What are Microservices
 - Monolithic vs Microservices Architecture
 - Characteristics of Microservices Architecture
 - Benefits of using Microservices Architecture
 - Microservices Design Principles.
- SOA vs. Microservice
- Handling Data in Microservices
- Communication between Microservices
 - Synchronous Communication across Microservices.
 - Asynchronous communication across Microservices.
- Microservices Patterns
 - Domain Driven Design
 - Command and Query Responsibility Separation (CQRS)
 - Event Sourcing
 - API Gateway Pattern
- Creating Composite UI with Microservices
- Drawbacks of Microservices

Setup Microservice Based Application

- Creating a Solution and Project Layout
- Implementing a CRUD microservice
- Writing Domain Classes and Controllers
- Data Context Class and Data Seeding
- Using Repository Classes
- Swagger and SwashBuckle Integration
- Practical Demonstration using eStoreApplication
 - Product Catalog Service with SQL Server
 - Invoking both services using Swagger UI

Build UI Service

- Adding ASP.NET MVC Project
- Writing Model Classes
- Writing Service Classes
- Building Web Controller and Views
- Practical Demonstration using eStoreApplication
 - a. Writing Backend For Frontend (BFF)
 - b. UI Microservice to display Product Catalog

Hosting Microservices using Docker Containers

- Adding Docker Support to the Microservice Application
- Creating a Dockerfile file
- Designing and Developing Multi-Container Microservices
- Database Connection string and environment variables in Docker containers
- Handling Configuration Data
- Use a database server running as a container
- Practical Demonstration using eStoreApplication
 - Creating an SQL Server database container
 - Create a docker images for Product Catalog Microservices
 - Create a docker image for UI Microservice
 - Writing a YAML for deploying and executing the application.

Securing Microservices

- The Big Picture
- Authentication and Authorization
- Basic Authenticate with JWT Bearer Token
- Understanding OAuth2 and OpenIdConnect
- Using IdentityServer4 Authorization Server
- Authorization Grant Types
 - Authorization Code
 - Implicit
 - Client Credentials
 - Resource Owner Password Credentials (ROPC)
- Refresh Tokens
- Authentication between Microservices

- Implementing Role based and Policy based Authorization
- Practical Demonstration using eStoreApplication
 - Adding Authentication Microservice to Solution
 - Securing ProductCatalog Microservice
 - Accessing secure Product Catalog in Swagger UI
 - Accessing secure Product Catalog in Client Application

Microservices Communication

- Synchronous Communication using REST API
- Asynchronous Communication using Service Bus
- Practical Demonstration using eStoreApplication
 - Adding Shopping Cart Service with Redis Cache
 - Adding Product to Shopping Cart Service for logged-in client
 - Show Shopping Cart using ViewComponent
 - Create a docker images for Product Catalog Microservices
 - Asynchronous Communicating with Notification Service and Service Bus Queue

Domain Driven Design and CQRS

- Understanding DDD
- Understanding Command Pattern
- What is CQRS and CQS
- Comparing CQRS with traditional CRUD approach
- Apply CQRS and CQS approach in DDD microservice
- Implement Command and Command Handler Classes
- Event Sourcing Pattern
- Practical Demonstration using eStoreApplication
 - a. Build an Order Service
 - b. Update Shopping Cart if Product Price changes.
 - c. Update Product Catalog inventory if order is placed.

Handling Failures

- Handle Partial failure
- Implement retries and exponential backoff
- Using Polly policies
- Circuit Breaker Pattern

- Practical Demonstration using eStoreApplication
 - a. Handle Order Service failure
 - b. Handling temporary downtime of sql database.

Microservices and Kubernetes

- What is Monolithic Application
- What are Microservices
- Kubernetes Architecture
- Azure Kubernetes Service (AKS)
- Practical Demonstration using eStoreApplication
 - a. Deploying the above service in Azure Kubernetes Service.