

## Course duration

- 2 days

## Course Benefits

- Confidently use the stack outlined in the course.
- Understand the various key components.
- Apply the knowledge to migrate applications to microservice architected solutions with containerization for Microsoft development and deployment environments.

## Course Outline

1. Microservice development
  1. The Microservices Architecture Design Principles
  2. Decentralized Processing
  3. Business Domain-Centric Design
  4. Crossing Process Boundary is Expensive!
  5. Designing for Failure
  6. Fault Injection During System Testing
  7. Distributed Transactions
  8. Managing Distributed Services with Docker & Kubernetes (OpenShift)
  9. Microservices and their relationship to the front-end
  10. Rich Client Applications
  11. Single Page Applications (SPA)
  12. The Building Blocks of a Fault-tolerant Application
  13. Example of Microservices in Their Purest Form: AWS Lambdas
  14. Example of Traditional Enterprise Application Architecture
  15. Example of Microservices Architecture
  16. Summary
2. REST services
  1. Many Flavors of Services
  2. Understanding REST
  3. Principles of RESTful Services
  4. Create
  5. Retrieve
  6. Update
  7. Delete
  8. Client Generated ID
  9. SOAP Equivalent Examples
  10. JSON
  11. REST vs SOAP Communication

12. Restful API Design
13. Versioning
14. Documentation
15. Security
16. Additional Resources
17. Summary
3. Getting started with asp.net core
  1. Getting Started with ASP.NET Core
  2. Configuration and Middleware Pipeline
  3. Controllers and Views
  4. RESTful Services with Web API
  5. Data Access with Entity Framework Core
  6. Summary
4. Serverless architecture with Azure
  1. What is a Serverless Architecture?
  2. Azure Functions
  3. WebJobs SDK
  4. WebJobs Core
  5. WebJobs Extensions
  6. Azure App Service
  7. Azure Resource Manager
  8. Understanding Azure Functions usage
  9. Azure Container Service
  10. Azure Service Fabric
  11. Microsoft Application Insights
  12. Microsoft Operations Management Suite
  13. Summary
5. Azure container service w/ Kubernetes
  1. What is Kubernetes?
  2. What Is a Container?
  3. Microservices and Orchestration
  4. Microservices and Infrastructure-as-Code
  5. Kubernetes Container Networking
  6. Summary
6. Kubernetes: from the firehose
  1. Masters
  2. Nodes
  3. Pods
  4. Namespaces
  5. Resource Quota
  6. Authentication and Authorization
  7. Routing
  8. Registry
  9. Storage Volumes
  10. Microservices, Linking, and Catalogs
  11. Summary
7. Docker: Bring on the Whale

1. What is Docker
2. Docker ecosystem
3. Docker concepts
4. Docker Architecture
5. Microservice encapsulation
6. Secure microservices using Docker
7. Agility, Portability, and Control
8. .NET Core or .NET Framework for Docker
9. Summary
8. Microsoft Docker Tooling
  1. Using Visual Studio Tools for Docker
  2. Configuring your local environment
  3. Using Docker Tools in Visual Studio 2015
  4. Using Docker Tools in Visual Studio 2017
  5. Using Windows PowerShell commands in a DockerFile to set up Windows Containers
  6. DevOps workflow for a Docker application
  7. Inner-loop development workflow
  8. Source-Code Control integration and management with Visual Studio Team Services and Git
  9. Build, Continuous Integration, and Test with Visual Studio Team Services and Docker
  10. Continuous Delivery, Deploy
  11. Run and manage
  12. Monitor and diagnose
  13. Summary
9. Operational Readiness
  1. Monitoring, Logging, and Resource Management
  2. Monitoring Compute Resources
  3. Checking Pod Health
  4. Logging
  5. Secrets & Vaulting
  6. Summary
10. Application Modernization
  1. What is Application Modernization?
  2. Typical App Modernization Projects
  3. Why Modernize?
  4. Goals for Modernization
  5. Twelve-factor Application Microservices
  6. Maintaining State
  7. Cloud Service Fabric
  8. Summary
11. Security in microservices
  1. Role-based
  2. Claim-based
  3. Session Cookie
  4. HTTP Basic

- 5. JWT
- 6. Summary

## 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 Microservices class:

- A desire to learn how this Microservices toolchain can improve your organization effectiveness, build and release processes, application architecture and development, and business continuity for greenfield and application modernization.