

Microservices .NET

Doelgroep Cursus Microservices .NET

De cursus Microservices .NET is bedoeld voor ervaren .NET developers die schaalbare, gedistribueerde applicaties willen ontwerpen en bouwen volgens de microservices-architectuur.

Voorkennis Cursus Microservices .NET

Grondige kennis van C#, ASP.NET Core en basiskennis van API-ontwikkeling, Docker en gedistribueerde systemen is vereist.

Uitvoering Training Microservices .NET

De training bestaat uit interactieve theoretische sessies, demonstraties en praktische hands-on labs onder begeleiding van een ervaren trainer.

Certificaat Microservices .NET

Deelnemers ontvangen na succesvolle afronding een certificaat van deelname aan de cursus Microservices .NET.

Duur: 4 dagen

Prijs: € 2999

[Open Rooster](#)



Building AI Agents



Inhoud Cursus Microservices .NET

In de cursus Microservices .NET leren deelnemers schaalbare microservices bouwen en beheren met ASP.NET Core, event-driven architecturen, API gateways, containerisatie en deployment naar Kubernetes.

Inleiding Microservices

De cursus start met een overzicht van de microservices-architectuur, inclusief de voordelen, uitdagingen, decompositie-strategieën, domein-gedreven ontwerp (DDD) en het ontwerpen van bounded contexts.

Microservices met ASP.NET Core

In deze module leren deelnemers microservices op te bouwen met ASP.NET Core, inclusief minimal APIs, API-versionering, configuratie via het Options Pattern, en het toepassen van de Clean Architecture.

API Gateway en Communicatie

Deelnemers leren over synchrone en asynchrone communicatie, het bouwen van een API Gateway met YARP, gebruik van gRPC, service discovery en het introduceren van service meshes.

Authenticatie, Autorisatie en Beveiliging

Deze module behandelt beveiliging binnen microservices, OAuth 2.0, OpenID Connect, gebruik van IdentityServer, API policies, distributed identity management en het beveiligen van gateways.

Event-Driven Architectuur

Deelnemers leren de principes van event-driven systemen, message brokers zoals RabbitMQ en Kafka, event sourcing, de outbox pattern, en het coördineren van events via saga's.

Gedecentraliseerd Datamanagement

Deze module gaat in op data-consistentie in gedistribueerde systemen, database per service strategieën, CQRS, eventual consistency en polyglot persistence.

Veerkracht, Monitoring en Logging

Deelnemers leren fouttolerantie patronen zoals circuit breakers en retries met Polly, distributed tracing met OpenTelemetry, gestructureerd loggen en monitoring met Prometheus en Grafana.

Containerisatie en Deployment

De focus ligt op het containeriseren van microservices met Docker, gebruik van Docker Compose, CI/CD pipelines opzetten, deployment naar Kubernetes (AKS) en gebruik van Helm charts.

Geavanceerde Microservices Architectuur

Tot slot verdiepen deelnemers zich in micro frontends, hybride architecturen, serverless integraties, event streaming, service mesh advanced topics en best practices voor grootschalige systemen.

Modules Cursus Microservices .NET

Module 1: Introduction to Microservices	Module 2: Microservices with ASP.NET Core	Module 3: API Gateway and Communication
What are Microservices? Microservices vs Monoliths Benefits and Challenges Domain-Driven Design (DDD) Basics Bounded Contexts Designing Microservices Evolutionary Architecture Microservices Patterns Decomposition Strategies Database per Microservice	Building REST APIs Minimal APIs in ASP.NET Core Versioning APIs Model Validation Content Negotiation Health Checks Swagger and OpenAPI Service Registration and Discovery Service Configuration with Options Pattern Creating Microservices with Clean Architecture	Synchronous Communication Asynchronous Communication Introduction to API Gateway Pattern Building an API Gateway with YARP Routing and Aggregation Authentication at Gateway Level gRPC Basics in .NET gRPC vs REST Service Mesh Concepts Using Dapr for Communication
Module 4: Security	Module 5: Event-Driven Architecture	Module 6: Distributed Data Management
Security in Microservices OAuth 2.0 and OpenID Connect IdentityServer Integration JWT Tokens and Validation Scopes and Claims API Authorization Policies User Authentication and Federation Distributed Identity Management Securing APIs behind API Gateway Handling Security at Service Mesh Level	Event-Driven Architecture Overview Publish-Subscribe Pattern Message Brokers (RabbitMQ, Kafka) Event Sourcing Basics Integration Events Domain Events Idempotent Event Handling Outbox Pattern Saga Pattern Choreography vs Orchestration	Challenges of Distributed Data CAP Theorem Database per Service Strategy Data Consistency Approaches Two-Phase Commit and Sagas CQRS Pattern Read Models and Write Models Eventual Consistency Cross-Service Queries Polyglot Persistence
Module 7: Resilience and Monitoring	Module 8: Containerization and Deployment	Module 9: Advanced Microservices Architecture
Transient Fault Handling Retry Policies with Polly Circuit Breaker Pattern Timeout and Bulkhead Isolation Distributed Tracing (OpenTelemetry) Structured Logging with Serilog Centralized Logging with ELK Stack Monitoring with Prometheus and Grafana Health Checks and Readiness Probes Metrics Aggregation and Alerting	Introduction to Containers Dockerizing .NET Microservices Creating Dockerfiles for Services Multi-Stage Builds Using Docker Compose Container Registry (ACR, DockerHub) Continuous Integration with GitHub Actions Continuous Deployment to Azure Kubernetes Helm Charts for Deployment Rolling Updates and Blue/Green Deployment	Micro Frontends and Microservices Serverless and Microservices Hybrid Architectures Edge Services Event Streaming Architectures Domain-Oriented Microservices Service Mesh Advanced Topics Federated Identity across Services Multi-Cluster Deployments Best Practices for Large Systems