

KOENIG SOLUTIONS • COURSE OUTLINE

Architecting Cloud-Native .NET Apps for Azure

Learn to design, build and secure cloud-native .NET applications on Azure, from data and communication patterns to DevOps.

40 Hours

DURATION

10

MODULES

134

TOPICS

COURSE OVERVIEW

This course begins by defining cloud native and introducing a reference application built using cloud-native principles and technologies. Beyond these first two chapters, the rest of the course is broken up into specific chapters focused on topics common to most cloud-native applications. You can learn about cloud-native approaches to: Data and data access, Communication patterns, Scaling and scalability, Application resiliency, Monitoring and health, Identity and security, DevOps

WHO THIS COURSE IS FOR

The audience for this course is mainly developers, development leads, and architects who are interested in learning how to build applications designed for the cloud

A secondary audience is technical decision-makers who plan to choose whether to build their applications using a cloud-native approach

LEARNING OBJECTIVES

- Explain the pillars of cloud-native application design
- Identify Azure services for deploying microservices and containers
- Compare containers and serverless approaches for scaling
- Apply communication patterns such as gRPC and API gateways
- Evaluate relational versus NoSQL data options for cloud-native apps
- Implement resiliency patterns such as retry and circuit breaker
- Configure monitoring and logging using Azure Monitor and Elastic Stack
- Plan DevOps practices including CI/CD pipelines and infrastructure as code

10 Modules, 134 Topics

01

Introduction to Cloud-Native Applications

TOPICS

Introduction to Cloud-native computing · What is Cloud Native?

THE PILLARS OF CLOUD NATIVE

The cloud · Modern design · Microservices · Containers · Backing services · Automation

CANDIDATE APPS FOR CLOUD NATIVE

Modernizing legacy apps

02

Introducing eShopOnContainers Reference App

TOPICS

- Features and requirements
- Overview of the code
- Understanding microservices

MAPPING ESHOPONCONTAINERS TO AZURE SERVICES

- Container orchestration and clustering
- API Gateway
- Data
- Event Bus
- Resiliency

DEPLOYING ESHOPONCONTAINERS TO AZURE

- Azure Kubernetes Service
- Deploying to Azure Kubernetes Service using Helm
- Azure Functions and Logic Apps (Serverless)

CENTRALIZED CONFIGURATION

- Azure App Configuration
- Azure Key Vault
- Configuration in eShop

03

Scaling Cloud-Native Applications

LEVERAGING CONTAINERS AND ORCHESTRATORS

- Challenges with monolithic deployments
- What are the benefits of containers and orchestrators?
- What are the scaling benefits?
- What scenarios are ideal for containers and orchestrators?
- When should you avoid using containers and orchestrators?
- Development resources

LEVERAGING SERVERLESS FUNCTIONS

- What is serverless?
- What challenges are solved by serverless?
- What is the difference between a microservice and a serverless function?
- What scenarios are appropriate for serverless?
- When should you avoid serverless?

- Combining containers and serverless approaches

DEPLOYING CONTAINERS IN AZURE

- Azure Container Registry
- ACR Tasks
- Azure Kubernetes Service
- Azure Bridge to Kubernetes

SCALING CONTAINERS AND SERVERLESS APPLICATIONS

- **The simple solution:** scaling up
- Scaling out cloud-native apps
- Other container deployment options

04

Cloud-Native Communication Patterns

TOPICS

Communication considerations

FRONT-END CLIENT COMMUNICATION

Simple Gateways · Azure Application Gateway · Azure API Management · Real-time communication

SERVICE-TO-SERVICE COMMUNICATION

Queries · Commands · Events

GRPC

What is gRPC? · gRPC Benefits · Protocol Buffers · gRPC support in .NET · gRPC usage · gRPC implementation

Looking ahead

05

Cloud-Native Data Patterns

TOPICS

- Database-per-microservice, why?
- Cross-service queries
- Distributed transactions

HIGH VOLUME DATA

- CQRS
- Event sourcing

RELATIONAL VS. NOSQL DATA

- The CAP theorem
- Considerations for relational vs. NoSQL systems
- Database as a Service
- Azure relational databases
- Azure SQL Database
- Open-source databases in Azure
- NoSQL data in Azure
- Data migration to the cloud

CACHING IN A CLOUD-NATIVE APP

- Why?
- Caching architecture
- Azure Cache for Redis
- Elasticsearch in a cloud-native app

06

Cloud-Native Resiliency

APPLICATION RESILIENCY PATTERNS

Retry pattern · Circuit breaker pattern · Testing for resiliency

AZURE PLATFORM RESILIENCY

Design with resiliency · Design with redundancy · Design for scalability · Built-in retry in services

RESILIENT COMMUNICATIONS

Service mesh · Istio and Envoy · Integration with Azure Kubernetes Services

07

Monitoring and Health

OBSERVABILITY PATTERNS

- When to use logging
- Challenges with detecting and responding to potential app health issues
- Challenges with reacting to critical problems in cloud-native apps

LOGGING WITH ELASTIC STACK

- Elastic Stack
- What are the advantages of Elastic Stack?
- Logstash
- Elasticsearch
- Visualizing information with Kibana web dashboards
- Installing Elastic Stack on Azure

Monitoring and Health (continued)

MONITORING IN AZURE KUBERNETES SERVICES

- Azure Monitor for Containers
- Log.Finalize()

AZURE MONITOR

- Gathering logs and metrics
- Reporting data
- Dashboards
- Alerts
- References

08

Cloud-Native Identity

TOPICS

- Authentication and authorization in cloud-native apps
- Azure Active Directory

IDENTITYSERVER FOR CLOUD-NATIVE APPLICATIONS

- Common web app scenarios
- Getting started
- Configuration
- JavaScript clients

09

Cloud-Native Security

AZURE SECURITY FOR CLOUD-NATIVE APPS

- Threat modeling
- Principle of least privilege
- Penetration testing
- Monitoring

SECURING THE BUILD

- Building secure code

BUILT-IN SECURITY

- Azure network infrastructure
- Role-based access control for restricting access to Azure resources

SECURING SECRETS

- Azure Key Vault
- Kubernetes

ENCRYPTION IN TRANSIT AND AT REST

- In transit
- At rest
- Keeping secure

10

DevOps

TOPICS

Azure DevOps · GitHub Actions

SOURCE CONTROL

Repository per microservice · Single repository ·
Standard directory structure

Task management

CI/CD PIPELINES

Azure Builds · Azure DevOps releases · Everybody gets a
build pipeline · Versioning releases

FEATURE FLAGS

Implementing feature flags

INFRASTRUCTURE AS CODE

Azure Resource Manager templates · Terraform · Azure
CLI Scripts and Tasks

Cloud Native Application Bundles · DevOps Decisions

WHAT IS INCLUDED

Everything You Need to Succeed

- Koenig course material for every module
- Delivery by certified instructors and subject matter experts
- Hands-on exercises throughout the course
- Course completion certificate from Koenig

ABOUT KOENIG

Who We Are

Established in 1993, Koenig Solutions is a globally recognised training provider with 35+ technology partnerships, 5,000+ training programmes and a network of certified instructors and subject matter experts, supporting individuals and enterprise teams worldwide.

FLEXIBLE DELIVERY

Learn the Way That Suits You

Classroom Training

Live Online Classes (ILO)

Fly-Me-A-Trainer (FMAT)

Flexi (Self-Paced)

1-on-1 Training

Ready to Get Started?

Talk to a learning advisor about schedules, group training and customised delivery.

Visit koenig-solutions.com

✉ info@koenig-solutions.com