

# Microservice Practices and Technical Implementations

## Guided Docker, Kubernetes, Jenkins and Prometheus workflows

FA-0793 · 2 days

DevOps, Cloud & Infrastructure · Software Development

---

### Introduction

Connect microservice design decisions with practical delivery and operations. Use a prepared environment to package and deploy a sample service, explore a Jenkins pipeline and inspect Prometheus metrics. The two-day course provides guided experience with selected workflows, not independent mastery of all four platforms.

### Learning outcomes

- Relate technical practices to business needs and assess microservice trade-offs.
- Package a sample service in a container and deploy it in a prepared Kubernetes cluster.
- Run a guided Jenkins build/test/deployment pipeline.
- Inspect service metrics and discuss visualisation and alert handling.
- Identify improvement opportunities and communicate operational considerations across teams.

### Prerequisites

- Familiarity with software development concepts and practices
- Basic understanding of web services and APIs
- Some exposure to containerization and cloud computing concepts would be beneficial but not mandatory

Access to the prepared compatible container, cluster, Jenkins and monitoring environments.

# Training outline

## 1. Session 1 — Business alignment

- Understanding the relationship between technical practices and business strategy in microservice architecture
- Identifying business justifications and general business value of microservices
- Leveraging microservices for competitive advantage and agility

Competitive advantage and agility are possible goals, not automatic effects of adopting microservices.

## 2. Session 2 — Architecture challenges

- Exploring common challenges and trade-offs in microservice architecture
- Scalability and performance considerations
- Service discoverability and inter-service communication
- Data consistency and synchronization

## 3. Session 3 — Docker and Kubernetes foundation

- Introduction to containerization with Docker
- Managing containers with Docker commands
- Introduction to Kubernetes (K8s) and its benefits
- Deploying and managing applications with Kubernetes

## 4. Session 4 — Container deployment

- Packaging microservices as containers
- Defining container images and Dockerfiles
- Creating and managing deployments with Kubernetes
- Service discovery and load balancing with Kubernetes

## 5. Session 5 — Jenkins CI/CD

- Introduction to Continuous Integration and Continuous Deployment (CI/CD)
- Setting up Jenkins for building and deploying microservices
- Configuring Jenkins pipelines for automated builds and deployments
- Integrating unit tests and quality checks in the CI/CD pipeline

## 6. Session 6 — Prometheus monitoring

- Overview of monitoring and observability in microservices
- Introduction to Prometheus for metrics collection and monitoring
- Setting up Prometheus to monitor microservice health and performance
- Visualizing and alerting on Prometheus metrics

Distinguish metrics collection, graphing and alert delivery; notification handling may use the prepared Alertmanager component.

## 7. Session 7 — Guided exercises

- Hands-on exercises with Docker, Kubernetes, and Jenkins
- Practicing Kubectl commands for Kubernetes management
- Deploying microservices as containers and testing their functionality
- Implementing CI/CD pipelines with Jenkins

## 8. Session 8 — Review and questions

- Reviewing key concepts and learnings from the course
- Addressing participants' questions and clarifications
- Summarizing best practices and recommendations for microservice practices