

# Spring 5 Microservices

This course will help you implement the microservice architecture in Spring Framework, Spring Boot, and Spring Cloud. Using the latest specifications of Spring that focuses on Reactive Programming, you'll be able to build modern, internet-scale Java applications in no time. The course starts off with guidelines to implement responsive microservices at scale. You will understand how Spring Boot is used to deploy serverless autonomous services by removing the need to have a heavyweight application server.

You will also learn how to go further by deploying your microservices to Docker and managing them with Mesos. By the end of the book, you will have gained more clarity on the implementation of microservices using Spring Framework and will be able to use them in internet-scale deployments through real-world examples.

## **Intended Audience;**

You should be familiar with [Spring MVC](#)

## **Day 1**

### **Demystifying Microservices**

- Evolution of microservices
- What are Microservices?
- Microservices – The honeycomb analogy
- Principles of microservices
- Characteristics of microservices
- Microservices examples
- Microservices benefits
- Summary

### **Related Architecture Styles and Use Cases**

- Service-Oriented Architecture (SOA)

- Twelve-Factor Apps
- Serverless computing
- Lambda architecture
- DevOps, Cloud, and Containers
- Reactive microservices
- Microservice use cases
- Microservices early adopters – Is there a common theme?
- Microservice frameworks

## **Day 2**

### **Building Microservices with Spring Boot**

- Setting up a development environment
- Spring Boot for building RESTful microservices
- Getting started with Spring Boot
- Developing a Spring Boot microservice
- Developing our first Spring Boot microservice
- HATEOAS-enabled Spring Boot microservice
- Reactive Spring Boot microservices
- Implementing security
- Enabling cross origin for microservices interactions
- Spring Boot actuators for microservices instrumentation
- Documenting microservices
- Putting it all together – Developing a customer registration microservice example

### **Applying Microservices Concepts**

- Microservice design guidelines

### **Microservices Capability Model**

- Microservices capability model
- Core capabilities
- Infrastructure capabilities
- Supporting capabilities
- Process and governance capabilities
- Microservices maturity model

- Entry points for adoption

## **Day 3**

### **Microservices Evolution – A Case Study**

- Understanding the PSS application
- Death of the monolith
- Microservices to the rescue – a planned approach for migration
- Target implementation
- Potential next steps

### **Scale Microservices with Spring Cloud Components**

- What is Spring Cloud?
- Spring Cloud releases
- Setting up the environment for the BrownField PSS
- Spring Cloud Config
- Eureka for registration and discovery
- Zuul proxy as the API Gateway
- Streams for reactive microservices
- Protecting microservices with Spring Cloud Security
- Summarising the BrownField PSS architecture

## **Day 4**

### **Logging and Monitoring Microservices**

- Understanding log management challenges
- Centralized logging solution
- Selection of logging solutions
- Monitoring microservices
- Data analysis using Data Lake

### **Containerizing Microservices with Docker**

- Understanding gaps in the BrownField PSS microservices
- What are containers?
- Difference between VM and containers

- Benefits of containers
- Microservices and containers
- Introduction to Docker
- Deploying microservices into Docker
- Running RabbitMQ on Docker
- Using the Docker registry
- Microservices on Cloud
- Running BrownField services on EC2
- Future of containerization

## **Day 5**

### **Scaling Dockerized Microservices with Mesos and Marathon**

- Scaling microservices
- Container orchestration
- Container orchestration with Mesos and Marathon
- Implementing Mesos and Marathon with DCOS
- Implementing Mesos and Marathon for BrownField microservices
- Preparing BrownField PSS services

### **Microservice Development Life Cycle**

- Practice points for microservice development
- Automating development cycle
- Summary

### **Duration and pricing**

Pricing [Group A](#)

### **Certificate**

- Upon completion of this course we will issue you with attendance certificate to certify your attendance.
- You may sit for our competency assessment test and on passing you will obtain our competency certificate.
- Our competency assessment can be booked and taken by someone who has not attended the course at a cost of

R2950.

### **Bookings**

You can download the course registration form on our home page or by clicking [here](#)

### **Brochure**

You may download a pdf copy of this page by clicking on the pdf icon at the top of the page.

### **Questions**

Please [email us](#)

### **Schedule**

On the calendar below. If your browser doesn't display the calendar below, please click on [this link](#) or try using [Google Chrome](#), alternatively please enquire via our [Contact Us](#) page