

## Program szkolenia:

# Microservices - architektura aplikacji, systemu i wdrożenia na platformie Java

### Informacje:

<b>Nazwa:</b>	<b>Microservices - architektura aplikacji, systemu i wdrożenia na platformie Java</b>
<b>Kod:</b>	<b>arch-ms-Microservices</b>
<b>Kategoria:</b>	Microservices
<b>Grupa docelowa:</b>	developerzy architekci admini
<b>Czas trwania:</b>	3 dni
<b>Forma:</b>	30% wykłady / 70% warsztaty

---

Szkolenie przedstawia gotowe i kompleksowe przepisy na zaprojektowanie architektury aplikacji i systemu opartej o styl Microservices. Podchodzimy do architektury całościowo, począwszy od infrastruktury i CI/CD, przez DDD i CQRS po testowanie.

Założeniem warsztatu jest przygotowanie prototypu środowiska przeznaczonego do uruchamiania mikroserwisów. Środowisko produkcyjne jest budowane na szkoleniu [Microservices - architektura infrastruktury dla devops](#).

### Zalety szkolenia:

- Kompleksowe podejście poprzez integrację: CQRS, DDD, CI/CD, Testowania, Spring Boot, REST, Microservices
- Najlepsze praktyki i wzorce
- Dostęp do ekspertów posiadających doświadczenie w projektach dużej skali

## Szczegółowy program:

### 1. Wprowadzenie teoretyczne

- 1.1. Prawo Conwaya
- 1.2. Charakterystyka mikroserwisów
- 1.3. Kryteria stosowalności
- 1.4. Określanie granic serwisów
- 1.5. Antywzorce i pułapki

### 2. Koncepcja DevOps

- 2.1. CI/CD
- 2.2. Infrastructure as a Code
  - 2.2.1. Jenkins DSL
  - 2.2.2. Ansible
  - 2.2.3. Wykorzystanie GIT
- 2.3. Wirtualizacja i konteneryzacja
- 2.4. Wykorzystanie chmury
- 2.5. Service discovery
- 2.6. Zarządzanie konfiguracją
- 2.7. Wdrożenia w środowisku rozproszonym

### 3. Komunikacja

- 3.1. Synchroniczna vs asynchroniczna
- 3.2. Consumer Driven Contract
- 3.3. Dokumentacja
- 3.4. Zagadnienia zmian wstecznie niekompatybilnych
- 3.5. Circuit breakers

## 4. Implementacja

4.1. Wykorzystanie Spring Boot i Spring Cloud

4.2. Strategie refaktoryzacji systemów monolitycznych

4.3. Zapewnienie jakości systemów rozproszonych

## 5. Utrzymanie

5.1. Centralne logowanie

5.2. Śledzenie przepływu komunikacji

5.3. Metryki techniczne i biznesowe

5.4. Skalowanie systemów rozproszonych

5.5. Podejście "safe to fail"