

## Program szkolenia:

# Nowoczesna architektura aplikacji i praktyczna implementacja z użyciem Spring Framework

## Informacje:

<b>Nazwa:</b>	<b>Nowoczesna architektura aplikacji i praktyczna implementacja z użyciem Spring Framework</b>
<b>Kod:</b>	<b>Spring -modernApp</b>
<b>Kategoria:</b>	Spring Framework
<b>Odbiorcy:</b>	developerzy
<b>Czas trwania:</b>	3 dni
<b>Forma:</b>	25% wykłady / 75% warsztaty

Szkolenie przedstawia najczęściej spotykane wzorce architektoniczne w nowoczesnych aplikacjach enterprise. Omawiamy przydatne wzorce na wszystkich poziomach architektury zaczynając od wzorców systemowych, poprzez wzorce aplikacyjne, kończąc na wzorcach na poziomie kodu. Wspomagamy się notacją C4 do prezentacji architektury i event stormingiem do analizy procesów i wyłaniania modułów systemu. Wszystko to pokazane w praktyce w kontekście stosu JVM i Spring Framework.

## Zalety szkolenia:

- przekrój wzorców architektonicznych na różnych poziomach abstrakcji
- prezentacja technik implementacji omawianych wzorców - nacisk na stronę praktyczną
- zastosowanie Spring Framework do implementacji omawianych wzorców

## Szczegółowy program:

### 1. Dokumentowanie architektury w podejściu C4.

### 2. Event storming jako narzędzie w planowaniu architektury.

### 3. Modularyzacja systemu

3.1. określanie granic modułów

3.2. bounded context

3.3. wzorce implementacji modułów

3.4. komunikacja pomiędzy modułami

3.4.1. sync

3.4.2. async

3.4.3. zdarzenia

### 4. Architektura aplikacyjna

4.1. modelowanie use case'ów

4.2. architektura warstwowa

4.3. architektura ports and adapters

4.4. cqrs

### 5. Wzorce implementacji logiki biznesowej

5.1. Transaction script

5.2. Rich domain model

5.3. DDD building blocks

5.4. Podejście funkcyjne

5.5. Zdarzenia domenowe

5.6. Event sourcing

### 6. Testowanie strategiczne

6.1. testowanie logiki biznesowej

6.2. testowanie integracyjne

6.3. testowanie e2e

6.4. praktyczne wzorce i techniki

6.4.1. assert object

6.4.2. fixtures

6.4.3. mockowanie obiektów i całych systemów

6.4.4. asynchroność

## **7. Praktyczne techniki implementacji wzorców z użyciem Spring Framework**

7.1. implementacja architektury warstwowej

7.2. implementacja architektury ports and adapters

7.3. implementacja persystencji z użyciem Spring Data

7.4. implementacja logiki aplikacyjnej i biznesowej w różnych odsłonach

7.5. wykorzystanie aspektów

7.6. obsługa błędów

7.7. implementacja restowej komunikacji pomiędzy modułami

7.8. implementacja zdarzeń przy użyciu Apache Kafka lub JMS

7.9. konfiguracja aplikacji w różnych środowiskach

7.10. testowanie integracyjne z użyciem Test Containers

7.11. zastosowanie reaktywności