

## Program szkolenia:

# Pragmatyczna refaktoryzacja do DDD

### Informacje:

<b>Nazwa:</b>	<b>Pragmatyczna refaktoryzacja do DDD</b>
<b>Kod:</b>	<b>legacy-practical</b>
<b>Kategoria:</b>	Refaktoryzacja kodu legacy
<b>Odbiorcy:</b>	developerzy
<b>Czas trwania:</b>	2 dni
<b>Forma:</b>	100% warsztaty

Podczas warsztatu poznasz kompleksowy i bezpieczny sposób refaktoryzacji kodu legacy do nowego modelu opartego o building blocki z Domain-Driven Design . Poznaj tajniki sztuki refaktoryzacji. Pokonaj Big Ball of Mud!

W trakcie zajęć będziemy wspólnie krok po kroku refaktoryzować typowy odziedziczony projekt. Pokażemy techniki, które stosujemy na co dzień w naszej pracy, aby kod naszych klientów przypominał ten, który widzimy często na konferencjach.

Podczas warsztatu dowiesz się:

- czym różni się technika "small steps" od "blue-green" refactor
- kiedy i gdzie należy je stosować
- jak sprawdzić, że nowy kod działa dokładnie tak samo jak stary
- jak szukać "szwów"
- jak bezpiecznie wycofać zmiany z produkcji gdy coś pójdzie nie tak
- jak powinien wyglądać kod docelowy
- do czego przydaje się Event Storming przy refaktoryzacji
- kiedy refaktoryzować w stronę DDD

Warsztat obfituje w praktyczne ćwiczenia każdego etapu procesu refaktoryzacji.

### Zalety szkolenia:

- Pragmatyczne podejście bez dogmatów
- Przypadki o realnym poziomie złożoności
- Wykorzystanie Event Stormingu i technik DDD

## Szczegółowy program:

### 1. Czym jest refaktoryzacja?

- 1.1. Kiedy refaktoryzować?
- 1.2. Kiedy nie refaktoryzować?
- 1.3. Co refaktoryzować?
- 1.4. Refaktoryzacja a dodawanie nowych funkcjonalności

### 2. Analiza zastanego rozwiązania

- 2.1. Big Picture Event Storming
- 2.2. Testy eksploracyjne
- 2.3. Testy na obecnych danych
- 2.4. Odkrywanie obserwowalnych zachowań systemu
- 2.5. Testy black-box
- 2.6. Zabezpieczenie pożądanych zachowań

### 3. Jak podejść do refaktoryzacji?

- 3.1. Podejście Blue/Green
- 3.2. Podejście Big Bang
- 3.3. Podejście Step by Step

### 4. Budowanie nowego modelu

- 4.1. Design Level Event Storming
- 4.2. Testowanie jednostkowe
- 4.3. Budowanie agregatów i innych building blocków taktycznego Domain-Driven Design

### 5. Wdrożenie zmian na produkcję

- 5.1. Odkrycie "seamów"
- 5.2. Wpięcie Query

5.3. Wpięcie Command

5.4. Rekoncyliacja

5.5. Feature Toggles

5.6. Warstwa ACL