

## Program szkolenia:

# Szybkie rozpoczęcie pracy w TDD

### Informacje:

<b>Nazwa:</b>	<b>Szybkie rozpoczęcie pracy w TDD</b>
<b>Kod:</b>	<b>dact-tdd</b>
<b>Kategoria:</b>	Testowanie automatyczne
<b>Odbiorcy:</b>	developerzy, testerzy
<b>Czas trwania:</b>	2 dni
<b>Forma:</b>	20% wykłady / 80% warsztaty

---

Szkolenie przeznaczone dla programistów, którzy chcą szybko zacząć pracę w stylu TDD korzystając ze sprawdzonych narzędzi, idiomów i wzorców.

Szkolenie ma formę warsztatową podczas, którego skupiamy się na esencji TDD.

### Zalety szkolenia:

- szybkie praktyczne wprowadzenie
- sprawdzone narzędzia i techniki
- esencja TDD

## Szczegółowy program:

### 1. Jednostkowe testowanie kodu - niezbędny do pisania dobrych testów w TDD

- 1.1. dlaczego warto automatycznie testować kod
- 1.2. podział testu na bloki/sekcje
- 1.3. nazewnictwo testów (metod testowych)
- 1.4. konstrukcje wspierające testowalność kodu (OOP, SOLID)
- 1.5. antywzorce dla testowalnego kodu

### 2. Wybrane narzędzia (zależnie od języka programowania: np Junit + Mockito + AssertJ)

- 2.1. asercje
- 2.2. soft assertions
- 2.3. custom representation
- 2.4. custom conditions
- 2.5. custom assertions
- 2.6. testy parametryzowane
- 2.7. testowanie wyjątków
- 2.8. inicjowanie i sprzątanie w testach
- 2.9. separacja od obiektów współpracujących
- 2.10. testowe zastępniki obiektów współpracujących
- 2.11. pułapki mockowanie

### 3. Testowanie asynchroniczne z biblioteką Awaitility

- 3.1. wprowadzenie do problemu testowania kodu asynchronicznego
- 3.2. zapoznanie z biblioteką Awaitility
- 3.3. assertj integration
- 3.4. initial delay

3.5. testowanie braku efektów ubocznych

3.6. najczęstsze problemy związane z testowaniem asynchronicznym

#### 4. Test Driven Development

4.1. historia TDD

4.2. cykl red-green-refactor

4.3. małe kroki a efektywność pracy

#### 5. TDD i programowanie w parach (ang. pair programming)

5.1. wykorzystanie TDD do pracy nad dobrym designem systemu

5.2. przejrzysta struktura testu

5.3. wybór kolejnych funkcji do zaimplementowania

5.4. sprawne uruchamianie testów z IDE (przydatne wtyczki, skróty klawiaturowe, ciągle uruchamianie testów)

5.5. korzyści ze stosowania TDD

#### 6. TDD Vending Machine Kata

6.1. Pair programming

6.2. Object Calisthenics

6.3. Grupowe code review