

## Program szkolenia:

# Git - zespołowa praca z rozproszoną kontrolą wersji

### Informacje:

<b>Nazwa:</b>	<b>Git - zespołowa praca z rozproszoną kontrolą wersji</b>
<b>Kod:</b>	<b>tools-Git</b>
<b>Kategoria:</b>	Narzędzia
<b>Grupa docelowa:</b>	developerzy
<b>Czas trwania:</b>	1-2 dni
<b>Forma:</b>	30% wykłady / 70% warsztaty

---

Podczas szkolenie uczestnicy nabywają praktyczną umiejętność pracy zespołowej z rozproszonym repozytorium kodu. Uczestnicy przechodzą przez wszystkie etapy pracy (Flow) z projektem poznając najlepsze praktyki dla każdego z etapów.

Podczas warsztatów kładziemy szczególny nacisk na zmianę nawyków powstałych podczas korzystania z repozytoriów starszej generacji.

### Zalety szkolenia:

- Realne przykłady i najlepsze praktyki
- Kontekst całego środowiska developerskiego
- Rozszerzenie o Continuous Delivery

## Szczegółowy program:

### 1. Wstęp do Git-a

- 1.1. Historia
- 1.2. Dlaczego rozproszony system kontroli wersji
- 1.3. Instalacja

### 2. Podstawy

- 2.1. Tworzenie projektu - różne scenariusze
- 2.2. Wersjonowanie zmian
- 2.3. Przeglądnie historii zmian
- 2.4. Cofanie zmian
- 2.5. Tagowanie
- 2.6. Codzienna praca - typowe czynności

### 3. Branching i merging

- 3.1. Podstawowy
- 3.2. Zarządzanie branch-ami
- 3.3. Zdalne branch-e
- 3.4. Rebasing
- 3.5. Jednoczesna praca z master i branch
- 3.6. Najlepsze praktyki

### 4. IDE

- 4.1. Wersjonowanie
- 4.2. Branching i merging
- 4.3. Praca ze zdalnymi repozytoriami

### 5. Workflow

5.1. Git flow

5.1.1. Development

5.1.2. Feature branching

5.1.3. Hotfixes

5.1.4. Release branching

5.1.5. Releasing

5.2. Tips and tricks

5.3. Wstęp do Continuous Delivery

**6. Dzień drugi - doradztwo (opcjonalnie)**

6.1. Praca na konkretnym workflow klienta

6.2. Opracowanie najlepszych praktyk i optymalnego podejścia do wykorzystania Git w kontekście procesu produkcji