

## Program szkolenia:

# Twarde narzędzia Lidera technicznego

### Informacje:

<b>Nazwa:</b>	<b>Twarde narzędzia Lidera technicznego</b>
<b>Kod:</b>	<b>career-TL</b>
<b>Kategoria:</b>	Kariera
<b>Czas trwania:</b>	3 dni
<b>Forma:</b>	40% teoria, 60% praktyka

Dziś jako lider techniczny potrzebujesz znacznie szerszych umiejętności niż dawniej. Zwykliśmy myśleć, że liderzy techniczni muszą jedynie posiadać wiedzę techniczną. Jednak posiadanie odpowiednich umiejętności technicznych to tylko czubek góry lodowej. Potrzebujesz również współpracować z biznesem, definiować i oceniać rozwiązania, planować pracę, dowodzić zespołem. Bardzo szeroki zakres umiejętności.

**Szkolenie "Tech Lead" pomoże Ci zdobyć te umiejętności.** Poznasz techniki, które pomogą Ci w codziennej pracy na każdym szczeblu działania. Kompleksowe szkolenie da Ci umiejętności potrzebne do pokonania każdego wyzwania, które stanie na Twojej drodze.

## Po warsztatach

- Nauczysz się współpracować z biznesem wykorzystując drivery biznesowe i architektoniczne
- Określisz propozycje rozwiązań oraz kompromisy i sposoby ich mitygacji
- Opracujesz plan wdrożenia wybranego rozwiązania wraz z zależnościami
- Zwizualizujesz modele architektoniczne
- Poznasz techniki współpracy zespołowej nad projektem architektury i codziennej pracy

### Zalety szkolenia:

- Kompleksowe spojrzenie na rolę Tech Leada w zespole
- Zagadnienia techniczne, organizacyjne i miękkie
- Praca zespołowa nad architekturą
- Określanie planów, ryzyk i ich mitygacji

## Szczegółowy program:

### 1. Lider techniczny - rola i kompetencje

- 1.1. Odpowiedzialności lidera technicznego
- 1.2. Cechy lidera technicznego
- 1.3. Tech Lead a role Team Lead i Staff Engineer

### 2. Drivery biznesowe i architektoniczne

- 2.1. Cele biznesu i klientów
- 2.2. Definicja architektury
- 2.3. Atrybuty jakościowe
- 2.4. Kryteria odbioru architektury
- 2.5. Ograniczenia techniczne i organizacyjne

### 3. Wzorce architektoniczne

- 3.1. Wzorce aplikacyjne
- 3.2. Wzorce integracyjne
- 3.3. Wzorce mikroserwisów
- 3.4. Wzorce procesów rozproszonych
- 3.5. Wzorce przechowywania danych

### 4. Projektowanie systemów

- 4.1. Przypadki użycia i diagramy
- 4.2. Event Storming Process Level
- 4.3. BDD - specyfikacja pracy systemu
- 4.4. Diagramy architektury - model C4
- 4.5. Diagramy sekwencji, kontrakty API
- 4.6. Event Storming Design Level

**5. Praca zespołowa nad architekturą**

- 5.1. Modele pracy zespołowej
- 5.2. Określanie celów projektowania
- 5.3. Delegowanie odpowiedzialności
- 5.4. Różne typy osobowości

**6. Ocena projektu rozwiązania**

- 6.1. Definiowanie różnych propozycji rozwiązania
- 6.2. Określanie kompromisów
- 6.3. Wybór rozwiązania na podstawie driverów
- 6.4. Decyzje architektoniczne

**7. Definiowanie planu dostarczenia**

- 7.1. Podział prac na mniejsze zadania
- 7.2. Ustalanie zależności między zadaniami
- 7.3. Priorytety dostarczania
- 7.4. Eksperymenty

**8. Dokumentowanie projektu rozwiązania**

- 8.1. Request for Comments
- 8.2. Architectural Decision Record
- 8.3. Procesy dookoła dokumentacji

**9. Dbanie o jakość rozwiązania**

- 9.1. Ocena realizacji wymagań na podstawie driverów
- 9.2. Strategia testowania i jakości
- 9.3. Obserwowalność, monitoring, obsługa błędów
- 9.4. Dobre praktyki pracy zespołowej

**10. Zarządzanie procesem dostarczania**

10.1. Lean Development

10.2. Mapowanie strumienia wartości

10.3. Kolejki i tempo dostarczania

10.4. Zarządzanie pracą, a nie osobami

10.5. Praktyki szczupłego dostarczania