

## Program szkolenia:

# AI-Assisted SDLC - fundamenty i techniki zaawansowane

### Informacje:

<b>Nazwa:</b>	<b>AI-Assisted SDLC - fundamenty i techniki zaawansowane</b>
<b>Kod:</b>	<b>prog-assisted</b>
<b>Kategoria:</b>	programowanie
<b>Odbiorcy:</b>	developerzy, architekci, liderzy zespołów, testerzy, liderzy techniczni
<b>Czas trwania:</b>	2 dni
<b>Forma:</b>	60% praktyki / 40% teorii

Opanuj zaawansowane techniki programowania wspomaganego przez AI i zmień sposób tworzenia oprogramowania. Używanie AI tylko do generowania kodu miało sens na etapie kiedy ta technologia raczkowała. Teraz, w dobie agentowego AI, pokażemy Ci, jak wykorzystać je jako partnera w całym procesie wytwarzania oprogramowania – od analizy wymagań i komunikacji z biznesem, przez implementację, testowanie i code review, aż po dokumentację.

W trakcie szkolenia:

- nauczysz się, jak sprawić, by AI bez zająknięcia ogarniało to, czego developerzy nie znoszą: opisy tasków, testy, logi, dokumentację
- poznasz praktyczne techniki pracy z agentami AI (takimi jak Claude Code)
- zobaczysz, jak efektywnie zarządzać kontekstem i integrować AI z codziennymi narzędziami
- krok po kroku przejdziesz od podstawowej konfiguracji do realnych workflowów, które przyspieszają pracę całego zespołu
- otrzymasz wprowadzenie do zaawansowanych tematów takich jak sub-agents do delegowania zadań, MCP do rozszerzania funkcjonalności, parallel processing mindset i version control z AII,

**To nie jest kolejny kurs o promptach. To zmiana sposobu pracy – AI staje się Twoim dodatkowym członkiem zespołu, który nigdy nie marudzi i zawsze dowozi.**

Szkolenie, które nauczy Cię wykorzystywać AI jako partnera w procesie wytwarzania oprogramowania. Od podstawowej instalacji po zaawansowane techniki zarządzania kontekstem - wszystko czego potrzebujesz, aby zacząć efektywnie pracować agentami AI na przykładzie Claude Code.

### Zalety szkolenia:

- Rzeczywiste przypadki użycia - praca na prawdziwych projektach
- Hands-on z Claude Code - natychmiastowe zastosowanie wiedzy
- Nauka przez działanie

## Szczegółowy program:

### 1. Wprowadzenie do paradygmatu AI-Assisted

- 1.1. Ewolucja wspomagania AI w programowaniu
- 1.2. Modele mentalne i zmiana myślenia
- 1.3. Pierwsze kroki z Claude Code
- 1.4. @notation - precyzyjne odniesienia do plików

### 2. CLAUDE.md Supremacy - kontekst jako fundament

- 2.1. Filozofia i moc instrukcji systemowych
- 2.2. Warsztat: Analiza skutecznych CLAUDE.md
- 2.3. Warsztat: Tworzenie własnego CLAUDE.md
- 2.4. Context poisoning i jak go unikać

### 3. Plan Mode, kontekst i dokumentacja pracy

- 3.1. Plan Mode - myśl zanim działasz
- 3.2. Context Window Management
- 3.3. STATUS.md i dokumentacja kontekstowa
- 3.4. Todo Lists i tracking postępu
- 3.5. Warsztat: Workflow z pełną dokumentacją

### 4. Debugging i tight feedback loops

- 4.1. AI jako debugger
- 4.2. Tight feedback loops - pisanie wykonanie iteracja
- 4.3. Warsztat: Kompleksowa sesja debugowania
- 4.4. Context-sized chunking

### 5. Refactoring i praca z kodem legacy

- 5.1. Strategia podejścia do legacy code

5.2. Batch operations z CLAUDE.md

5.3. Warsztat: Refactoring z wykorzystaniem CLAUDE.md

5.4. Explicit pattern adherence

## 6. Testowanie z AI i weryfikacja

6.1. Test-Driven Development z AI

6.2. Poziomy testowania i strategia

6.3. Warsztat: Kompleksowe testowanie

6.4. Bash automation dla test runners

## 7. Podstawy subagentów i MCP

7.1. Task/Agent Tools - delegowanie specjalizowanych zadań

7.2. MCP (Model Context Protocol) - podstawy

7.3. Version control z AI

7.4. Warsztat: End-to-end workflow z subagentami

## 8. Best practices i optymalizacja

8.1. Effective Prompting Mastery

8.2. Optymalizacja wydajności pracy

8.3. Budowanie kultury AI-Assisted w zespole