

Tytuł szkolenia: Czysty kod, najlepsze praktyki OOP i wzorce projektowe

Kod szkolenia: czysta-architektura

Wprowadzenie

Adresaci szkolenia

Szkolenie przeznaczone jest dla programistów, którzy chcą wyrobić w sobie nawyk pisania kodu, który będzie czytelny i zrozumiały dla innych.

Wymagania wstępne:

Znajomość języka Java oraz biblioteki JUnit na poziomie podstawowym.

Uwaga:

Przedstawiony ramowy program szkolenia może zostać dopasowany do indywidualnych potrzeb Klienta.

Cel szkolenia

Podczas szkolenia skupiamy się nie tylko na czystym kodzie per se, ale także pokazujemy, jak dobrze pisać programy z wykorzystaniem paradygmatu programowania obiektowego. Wprowadzamy również kilka wzorców projektowych poprawiających czytelność i strukturę kodu.

Czas i forma szkolenia

- 14 godzin (2 dni x 7 godzin), w tym wykłady i warsztaty praktyczne.

Plan szkolenia

1. Dobry kod kontra zły kod
 - Jak definiować i mierzyć jakość kodu
 - Objawy kliniczne złego kodu
 - Przyczyny degradacji kodu
 - Obrona przed pogorszeniem jakości kodu
 - Przegląd kodu
 - Analiza statyczna (SonarQube)
 - Codzienna refaktoryzacja
2. Podstawowe zasady w codziennej pracy programisty
 - Zasada skauta
 - SUCHY
 - Zasada najmniejszego zaskoczenia
 - POCAŁUNEK / YAGNI
 - Wykorzystanie zewnętrznych bibliotek/frameworków
 - Iteracyjne tworzenie kodu
3. Jak stworzyć dobry kod
 - Nazewnictwo zmiennych, metod, klas
 - Formatowanie
 - Tworzenie czytelnych metod
 - Tworzenie zrozumiałych klas
 - Obsługa sytuacji błędów; wyjątki
 - Uwagi
4. Warsztaty - poprawianie jakości istniejącego kodu
 - Tworzenie testów
 - Refaktoryzacja krok po kroku
5. Najlepsze praktyki OOP
 - Wysoka konsystencja
 - Luźne powiązanie
 - Polimorfizm zamiast sprawdzania typów
6. SOLIDNE zasady
 - Odwrócenie zależności
 - Segregacja interfejsów
 - Pojedyncza odpowiedzialność
 - Reguła otwarta (dla rozszerzenia) / zamknięta (dla modyfikacji).
 - Zasada podstawienia Liskowa
 - Zaskakujące efekty z dziedziczeniem
 - Zastąpienie dziedziczenia kompozycją
7. Warsztaty - ulepszanie struktury istniejącego kodu z wykorzystaniem zasad SOLID
8. Wybrane wzorce projektowe w kontekście tworzenia czytelnego kodu
 - Strategia ⇒ zróbmy jedną rzecz na wiele sposobów
 - Dekorator ⇒ zostawmy poboczne sprawy innym
 - State ⇒ przejdźmy przez cały cykl życia naszego obiektu
 - Gość ⇒ zbierzmy różne owoce z jednego drzewa
9. Warsztaty - ulepszanie istniejącego kodu za pomocą wzorców projektowych