

Kod szkolenia: **SPRING**

Tytuł szkolenia: **Spring framework**

Dni: 5

Opis:

Cel szkolenia

Głównym celem szkolenia jest wprowadzenie i gruntowne poznanie możliwości frameworku oraz zrozumienie idei programowania z użyciem interfejsów, wstrzykiwania zależności oraz aspektów. Szkolenie ma formę warsztatową - uczestnicy przy wsparciu prowadzącego rozwijają projekt/aplikację, poznając praktycznie kolejne elementy frameworku.

W czasie zajęć uczestnicy dowiedzą się między innymi: * jakie są możliwości i zastosowanie frameworku Spring * na czym polega idea inwersji kontroli / wstrzykiwania zależności oraz jakie płyną z niej korzyści * jak tworzyć i konfigurować własne komponenty, a także wykorzystywać usługi dostarczone w ramach kontenera * w jaki sposób za pomocą aspektów separować logikę biznesową od funkcjonalności pobocznych takich jak logowanie, bezpieczeństwo czy obsługa transakcji * jak testować aplikacje na poziomie jednostkowym i integracyjnym * jak integrować aplikacje Spring z istniejącymi rozwiązaniami JEE

Mocne strony szkolenia

Warsztatowy charakter zajęć Duża dawka życiowej wiedzy, przydatnej w realnych projektach

Grupa docelowa

programista/tester

Adresaci szkolenia

- Szkolenie adresowane do programistów języka Java chcących tworzyć skalowalne i łatwe w utrzymaniu aplikacje z wykorzystaniem najnowszej wersji Spring framework
- Programiści przygotowujący się do certyfikacji Spring, chcący powtórzyć i uporządkować posiadaną wiedzę
- Testerzy chcący poznać metody testowania jednostkowego na poziomie usług, repozytoriów i kontrolerów oraz tworzenia testów integracyjnych dla aplikacji opartych o Spring

Wymagania dla uczestników

Od uczestników szkolenia wymagana jest umiejętność programowania w języku Java na poziomie średnio zaawansowanym, elementarna wiedza z zakresu technologii wchodzących w skład platformy JEE oraz podstawowa znajomość SQL i relacyjnych baz danych.

Parametry szkolenia

5*7 godzin wykładów i warsztatów (z wyraźną przewagą warsztatów).

Program szkolenia:

1. Wprowadzenie

- Charakterystyka frameworku
- Programowanie z użyciem interfejsów
- Wstrzykiwanie zależności
- Separacja odpowiedzialności z wykorzystaniem AOP
- Konfigurowanie projektu
- Schemat budowy aplikacji

2. Kontener Spring

- Przegląd dostępnych implementacji
- Konfiguracja (XML, adnotacje, JavaConfig)
- Wstrzykiwanie zależności
- Zasięg i cykl życia beanów
- Rozwiązywanie konfliktów
- Customizacja / rozszerzanie możliwości kontenerów (postprocesory, konwertery, zasoby)
- Event bus i programowanie przez zdarzenia
- Wprowadzenie do Spring Expression Language
- Konfiguracja projektu z użyciem Spring Boot

3. Programowanie aspektowe

- Koncepcje i najważniejsze mechanizmy AOP
- Tworzenie, konfigurowanie i stosowanie aspektów
- Przechwytywanie argumentów, rezultatów działania oraz wyjątków
- Kontekst i kolejność wykonywania aspektów
- Zaawansowane wykorzystanie aspektów

4. Warstwa dostępu do danych

- Konfigurowanie połączenia do bazy (standalone, lokalna pula połączeń, jndi)
- Szablony JDBC
- Menadżer transakcji
- Konfigurowanie mechanizmu transakcyjnego
- Zatwierdzanie i wycofywanie transakcji
- Wprowadzenie do mapowania obiektowo-relacyjnego i standardu JPA

- Integracja z JPA i frameworkiem Hibernate
- Tworzenie warstwy utrwalania z wykorzystaniem Spring Data
- 5. Warstwa prezentacji
 - Wzorzec Model-View-Controller
 - Wprowadzenie do Spring MVC
 - Cykl obsługi żądania
 - Konfiguracja aplikacji webowej (xml, adnotacje, JavaConfig)
 - Budowa i mapowanie kontrolerów
 - Obsługa formularzy
 - Walidacja
 - Internacjonalizacja
 - Integracja z innymi frameworkami webowymi
- 6. Usługi oparte o architekturę REST
 - Tworzenie i mapowanie kontrolerów
 - Obsługa nagłówków i ciasteczek
 - Statusy odpowiedzi
 - Mapowanie i obsługa wyjątków
 - Konwertery i obiekty transferowe
- 7. Wprowadzenie do Spring Security
 - Architektura, komponenty i podstawowe usługi mechanizmu bezpieczeństwa
 - Konfiguracja bezpieczeństwa (uwierzytelnianie, autoryzacja dostępu, wylogowanie, remember-me, csrf filter)
 - Zarządzanie sesją
 - Customizacja wybranych komponentów mechanizmu bezpieczeństwa
 - Integracja z Spring OAuth2
- 8. Integracja i testowanie
 - Integracja z elementami Java EE (JMS,EJB,JNDI,JTA)
 - Zarządzanie zadaniami
 - Wykorzystanie pamięci podręcznej cache
 - Testy jednostkowe i integracyjne z użyciem JUnit oraz Mockito

