

Program szkolenia:

Aplikacje Web oparte o Spring Framework

Informacje ogólne

Nazwa:	Aplikacje Web oparte o Spring Framework
Kod:	Spring
Kategoria:	Spring
Grupa docelowa:	projektanci, architekci, programiści
Czas trwania:	3 dni
Forma:	50% wykłady / 50% warsztaty

Prezentacja kompleksowego zestawu rozwiązań pozwalających na produktywnie tworzenie profesjonalnych aplikacji webowych.

Koncepcja szkolenia została oparta o zestaw praktycznych problemów jakie stają przed developerem – prezentujemy sprawdzone i najlepsze ich rozwiązania w Spring.

Szkolenie oprócz prezentacji technologii Spring zostało wzbogacone o aspekty: doboru architektury aplikacji, konfiguracji narzędzi developerskich, praktyk projektowych odpowiednich do produktywniej pracy z Spring, istotnych zaawansowanych praktycznych aspektów JPA/Hibernate oraz JSF

W trakcie ćwiczeń praktycznych budowana jest spójna aplikacja web, której wymagania funkcjonalne przekładane są na kolejno prezentowane konstrukcje frameworka.

Zalety szkolenia:

- » Dobór architektury aplikacji
- » Aspekty wydajności JPA
- » Rozszerzony program

Program szkolenia:

1. Warstwowa architektura systemów opartych o Spring

1.1. Wzorce

1.1.1. Wzorce architektoniczne

1.1.2. Wzorce projektowe stosowane w aplikacjach webowych

1.1.3. Wzorce porządkowania logiki biznesowej

1.2. Architektura dla podejścia Domain Driven Design i CqRS(na życzenie)

1.3. Architektura zorientowana na testy

2. Kontener

2.1. Konfiguracja

2.2. Techniki uruchamiania

2.3. Zasada działania – Inversion of Control (w szczególności Dependency Injection)

3. Komponenty

3.1. Deklaracja - adnotacje i XML (dobór właściwego podejścia w zależności od problemu)

3.2. Cykl życia

3.3. Zależności

3.4. Zasięg komponentów – pułapki

3.5. Pre/post – procesory

3.5.1. Wykorzystanie dla technik Domain Driven Design

4. Paradygmat Inversion of Control - wsparcie frameworka dla 3 podejść

4.1. Dependency Injection

4.1.1. Wykorzystanie do zmniejszenia poziomu zależności

4.2. Praktyczne przykłady modularyzacji aplikacji biznesowych - wstrzykiwanie strategii (polityk) biznesowych

4.3. Zdarzenia

4.3.1. Zdarzeniowe architektury otwarte na rozbudowę

4.3.2. Asynchroniczne przetwarzanie zdarzeń jako technika zwiększająca skalowalność

4.3.3. Praktyczne przykłady generowania zdarzeń biznesowych z warstwy logiki

4.4. Techniki Aspect Oriented Programming

4.4.1. Podstawy teoretyczne

4.4.2. Praktyczne przykłady wykorzystania w aplikacjach biznesowych

4.4.3. Zastosowanie w Spring

4.4.4. Zastosowanie do budowy własnych frameletów

5. Funkcjonalności kontenera

5.1. Zarządzanie zasobami

5.2. Język wyrażeń

6. Warstw dostępu do danych

6.1. Transakcje (konfiguracja, poziomy izolacji, warstwa abstrakcji Spring)

6.2. Integracja z JPA/Hibernate (szczegółowy program w module JPA/Hibernate)

6.3. Integracja z JDBC

6.4. Integracja z iBatis

7. Warstwa prezentacji (do wyboru)

7.1. JSP

7.2. Freemarker

7.3. JSF

7.4. Facelets

7.4.1. RichFaces

7.5. AJAX

7.6. Standardowe problemy: Master-detail, listy, Back-button,...

8. Spring Remoting

8.1. Zastosowanie (applety, aplikacje StandAlone)

8.2. Konfiguracja

8.3. Najlepsze praktyki

9. Integracja

9.1. JSF

9.2. EJB

10. Bezpieczeństwo - Spring Security (dawniej ACEGI)

11. Zagadnienia zaawansowane:

11.1. Techniki optymalizacji warstwy dostępu do danych (JPA/Hibernate) oraz unikanie pułapek wydajnościowych:

11.1.1. Cache zapytań

11.1.2. Cache encji

11.1.3. Wydajność zapytań n+1 Select problem - wykrywanie i zapobieganie

11.1.4. Projektowanie wydajnego modelu

12. Testowanie

12.1. Spring - wsparcie dla testów

12.2. Testowanie jednostkowe - techniki mockowania

12.3. Testowanie integracyjne - wsparcie kontenera

13. Narzędzia: Spring Tools Suite