

Kod szkolenia: **JSF/ADV**

Tytuł szkolenia: **Zaawansowane mechanizmy tworzenia interfejsu użytkownika w technologii Java Server Faces**

Dni: 4

Opis:

Adresaci szkolenia

Szkolenie adresowane jest do programistów mających praktyczne doświadczenia z Java Server Faces, pragnących poznać bardziej zaawansowane techniki pracy z JSF.

Cel szkolenia

Celem szkolenia jest praktyczne poznanie zaawansowanych technik programistycznych umożliwiających rozwiązywanie bardziej złożonych problemów na jakie napotyka się tworząc aplikacje oparte o JSF.

Kluczowe zagadnienia:

- Tworzenie kodu z wykorzystaniem reguły DRY (w szczególności własne komponenty i ich ponowne użycie)
- Tworzenie własnych walidatorów, konwerterów oraz zdarzeń
- Obsługa walidacji oraz wyświetlania danych w relacjach rodzic-dziecko (posiadanie wielu list wyboru na jednym formularzy, które są zależne od siebie)
- Nawigacja, w szczególności tworzenie zakładek (bookmarkable urls/buttons) i url-i przyjaznych dla użytkownika oraz wyszukiwarek (user-friendly urls, seo-friendly urls)
- Wykorzystanie JavaScript (JavaScript, JQuery, ClientBehavior) do tworzenia bardziej złożonego interfejsu użytkownika
- Praktyczne porównanie bibliotek Javascript min. JQuery, Prototype, Dojo w kontekście użycia w aplikacji JSF
- Tworzenie struktur danych przyjaznych dla JavaScript (JSON) i wykorzystanie ich w JSF

Mocne strony szkolenia

Szkolenie skupia się na zagadnieniach, które bardzo często postrzegane są jako problematyczne, kłopotliwe do zaprogramowania z wykorzystaniem JSF. Każdy programista mający styczność z JSF prędzej czy później orientuje się, że JSF jak każda technologia ma swoje mocne i słabe strony i że nie wszystko da się zrobić niejako automatycznie i z wykorzystaniem podstawowych mechanizmów jakie oferuje JSF. Umiejętność rozwiązywania

pewnej grupy problemów wymaga jednak posiadania głębszej wiedzy na temat funkcjonowania JSF i bieżące szkolenie ma na celu dostarczenie niezbędnych informacji w tym zakresie.

Wymagania:

Od uczestników szkolenia wymagana jest umiejętność programowania w języku Java (do poznania na kursie J/JP), znajomość podstawowych zagadnień związanych z Java Server Faces (do poznania na kursie J/JSF2) oraz podstawowa znajomość Javascript.

Parametry szkolenia

4*8 godzin (4*7 godzin netto) wykładów i warsztatów (z wyraźną przewagą warsztatów).

Wielkość grupy: maks. 8-10 osób.

Program szkolenia:

1. Wstęp

- I. Przypomnienie podstawowych koncepcji JSF
- II. Praktyczne zapoznanie się z aplikacją która będzie punktem wyjścia do ćwiczeń realizowanych w ramach szkolenia
- III. Przypomnienie (z wykorzystaniem aplikacji tam gdzie jest to technicznie możliwe) kluczowych zagadnień i koncepcji związanych z JSF oraz JEE
 - i. Rola JSF w świecie JEE, alternatywne rozwiązania (JSF vs Struts2, JSF vs GWT, JSF vs JSP)
 - ii. Dostępne implementacje (RichFaces, PrimeFaces, IceFaces itd.)
 - iii. MVC, MVP, CDI
 - iv. Model komponentowy JSF
 - v. Cykl życia (Lifecycle)
 - vi. ManagedBeans
 - vii. Nawigacja
 - viii. Walidacja
 - ix. Facelets
- IV. Wprowadzenie do kluczowych zagadnień omawianych na szkoleniu
 - i. Composite Components
 - ii. JavaScript (jQuery, Ajax, JSON, Client Behaviors)
 - iii. Nawigacja

2. Tworzenie komponentów - podstawy

- I. Praktyczne zapoznanie się z kluczowymi koncepcjami, których znajomość jest niezbędna do tworzenia własnych komponentów
 - i. Komponent czy Renderer?
 - ii. Przestrzeń nazw „composite” (composite namespace)
 - iii. Tagi: composite:interface, composite:attribute, composite:implementation
- II. Stworzenie prostych komponentów z wykorzystaniem zdobytej do tej pory

- wiedzy – jeden komponent, wiele możliwości jego wykorzystania
 - i. Komponenty wykorzystujące kontrolki typu “input” (wprowadzanie danych)
 - ii. komponenty wykorzystujące kontrolki typu “select” (wybór danych z list wyboru)
- III. RichFaces CDK
- 3. Tworzenie komponentów – zagadnienia zaawansowane
 - I. Współpraca komponentu z logiką realizowaną po stronie serwera
 - i. Pozyskiwanie przez komponent danych udostępnianych przez backing beany
 - ii. Przekazywanie danych z komponentu do backing beana
 - II. Tworzenie własnych zdarzeń
 - i. Kolejowanie zdarzeń
 - ii. Zdarzenia inicjujące przetwarzanie po stronie serwera (Action listeners)
 - iii. Zdarzenia związane ze skutkami (events) czynności inicjowanych na poziomie interfejsu użytkownika (Event listeners)
 - iv. Zdarzenia związane z fazami cyklu życia JSF - Phase Listeners
 - v. Własne zdarzenia a cykl życia JSF
 - III. Walidacja oraz konwersja
 - i. Tworzenie własnych walidatorów oraz konwerterów
 - ii. Omijanie pełnej walidacji (czynności typu „Cancel”, atrybut immediate i jego wpływ na przetwarzanie żądania, walidacja warunkowa, częściowa)
 - iii. BeanValidation (JSR303) a JSF
 - iv. Standardowy cykl życia JSF a konieczność jego modyfikacji w praktyce
 - IV. Relacje rodzic-dziecko w świecie JSF – zależności pomiędzy listami wyboru (dostępność/niedostępność w zależności od wyboru wartości w liście rodzica/dziecka)
 - V. Inteligentne pobieranie danych (np. paginacja w tabelach czy strukturach złożonych takich jak drzewa)
- 4. Wykorzystanie Javascript, Ajax
 - I. Rozszerzenie możliwości komponentu poprzez wykorzystanie Javascript
 - i. Tworzenie struktur danych przyjaznych dla JavaScript (JSON)
 - ii. JSON w aplikacjach JSF
 - II. Ajax
 - i. Techniki pozyskiwania danych dla strony internetowej (Pull, Comet aka Push/Reverse Ajax)
 - ii. Kontrolowanie kolejności wysyłania żądań (Callbacks, Kolejowanie)
 - iii. Renderowanie fragmentów strony (Partial Rendering, Partial Requests)
 - iv. Wykorzystanie mechanizmów zawartych w dostępnych implementacjach standardu JSF (RichFaces, PrimeFaces)
 - A. Biblioteka „a4j” w RichFaces
 - B. Realizacja zachowań „ajaxowych” z wykorzystaniem PrimeFaces
 - III. Wykorzystanie JQuery – korzyści jakie daje nam wykorzystanie tego rozwiązania

- IV. Wzbogacenie interfejsu użytkownika poprzez wprowadzenie efektów „specjalnych” (np. Fading, Sliding)
5. Tworzenie kodu JavaScript, który może zostać ponownie wykorzystany
 - I. Wykorzystanie ClientBehavior oraz ClientBehaviorHolder do implementacji reużywalnego kodu JavaScript
6. Nawigacja (z wykorzystaniem PrettyFaces)
 - I. Zakładki (bookmarkable urls/buttons)
 - II. Url-e przyjazne dla użytkownika oraz dla SEO
 - III. View Parameters
 - IV. Back button problem
7. Komunikacja w JSF oparta o zdarzenia
 - I. Tradycyjne podejście do komunikacji między obiektami
 - II. Wzorzec Obserwator w implementacji mechanizmów komunikacji
 - III. Wykorzystanie CDI do komunikacji między obiektami (implementacja rozwiązania typu „EventBus”)
8. Podsumowanie

