

**Kod szkolenia: J/JEE6**

**Tytuł szkolenia: Nowoczesne aplikacje internetowe oparte na JEE6 (JSF2, EJB3.1, JPA2, CDI, Ajax)**

## **Adresaci szkolenia:**

Szkolenie adresowane jest do programistów aplikacji internetowych na platformie JEE, pragnących uzupełnić swoje kwalifikacje o najnowsze technologie, a także dla osób znających język Java i pragnących rozpocząć swoją przygodę z aplikacjami internetowymi od najnowszych technologii.

Szkolenie realizowane jest z użyciem najnowszej odsłony Java Enterprise Edition – wersji 6, która kusi możliwościami lecz nie ma jeszcze pełnego wsparcia u dostawców serwerów aplikacji. Dla osób pragnących pracować z ugruntowanymi technologiami polecamy szkolenie WEB2

## **Cel szkolenia:**

Uczestnicy dowiedzą się jak projektować i implementować aplikacje na platformie JEE z wykorzystaniem technologii będących treścią szkolenia. Celem szkolenia jest zdobycie umiejętności tworzenia skalowanych aplikacji, zgodnie z arkanami sztuki: od komponentów biznesowych po wysoce ergonomiczne i funkcjonalne interfejsy wpisujące się w nurt WEB 2.0.

W szczególności:

Uczestnicy szczegółowo poznają technologię prezentacji JSF2.0 włącznie z tworzeniem własnych walidatorów, konwerterów czy komponentów. Zaznajomią się z popularnymi, bogatymi bibliotekami komponentów, takimi jak RichFaces, IceFaces. Nauczą się projektować i implementować warstwę logiki biznesowej w oparciu EJB 3.1, w połączeniu z mechanizmem mapowania relacyjno-objektowego JPA2. Poznają specyfikację CDI standaryzującą część uznanego Framework Seam.

## Wymagania:

Od uczestników szkolenia wymagana jest umiejętność programowania w języku Java (do poznania na kursie J/JP), podstawy relacyjnych baz danych i SQL.

## Mocne strony szkolenia:

Program obejmuje całościowo i wyczerpująco zagadnienia tworzenia aplikacji internetowych.

W przeciwieństwie do większości szkoleń, szkolenie to nie koncentruje się na jednej technologii, lecz zawiera najważniejsze zagadnienia z wybranych technologii. Dzięki temu, uczestnicy po skończeniu szkolenia, będą mogli tworzyć samodzielnie kompletne aplikacje WWW, uzupełniając już we własnym zakresie wiedzę o pojawiające się nowe potrzeby.

Program jest ciągle uaktualniany, tak, by uwzględniać nowo powstające trendy.

## Parametry szkolenia:

5\*7 godzin wykładów i warsztatów w proporcji 1/3. W trakcie warsztatów, oprócz prostych ćwiczeń, projektowana i implementowana jest prosta kompleksowa aplikacja.

Wielkość grupy: maks. 8-10 osób.

## Polecane szkolenia poprzedzające:

J/JP

## Program szkolenia:

1. Wstęp
  - Wprowadzenie do platformy Java Enterprise Edition
  - Omówienie komponentów i specyfikacji JEE w tym JNDI, JMS, , Servlet, JSP, EJB, webserwisy,
  - Zagadnienia projektowania aplikacji na platformie JEE, doboru technologii, bezpieczeństwa etc.
2. Specyfikacja Managed Beans
3. Warstwa logiki biznesowej – komponenty EJB
  - Typy komponentów EJB 3.x
  - Zasada działania

- Komponenty sesyjne:
  - Stanowe
  - Bezstanowe
  - Komponenty typu Singleton\*
  - cykl życia komponentów, nasłuchiwanie i sterowanie cyklem życia
  - Rodzaje dostępów
    - Lokalny
    - Lokalny bez-interfejsowy\*
    - Zdalny
    - Zdalny przez usługę web service
  - Wywołania asynchroniczne metod komponentów sesyjnych\*
  - Zagadnienia współbieżności w odwoływaniu się do komponentów:
    - Zachowanie kontenera
    - Współbieżność zarządzana przez kontener i współbieżność zarządzana przez komponent \*
    - Komponent stanowy w kontekście współbieżności
- Korzystanie z zasobów przez komponenty EJB
  - Słowo o JCA
  - Przestrzenie nazw JNDI i wyszukiwanie zasobów
  - Inversion of Control w odwoływaniu do zasobów
- Interceptory
- Model wyjątków
- Komponenty Sterowane Wiadomością:
  - Zasada działania
  - Rodzaje komunikacji
- Bezpieczeństwo komponentów
- Transakcje zarządzane przez kontener i komponent
  - Transakcje zarządzane przez kontener (CMT)
    - Atrybuty transakcji w kontekście scenariuszy użycia
    - Transakcje a wyjątki
    - Synchronizacja stanu w komponencie stanowym
  - Transakcje zarządzane przez komponent (BMT) – podstawy JTA
  - Transakcje zarządzane przez klienta
- Usługi czasowe (TimerService)
  - Interfejsy Timer i TimerService,
  - Automatyczne uruchamianie usług w oparciu o kalendarz\*
- Usługi webservice w kontekście EJB
  - Podstawy webservice
  - JAX-WS a EJB
- Przygotowanie, skonfigurowanie i wdrażanie aplikacji
  - Jako samodzielny moduł EJB (jar)
  - Jako składowa aplikacji Enterprise (ear)
  - W obrębie aplikacji Web (war) \*
  - Kontener pełny (full) i lekki (light)\*
  - EJB w kontenerze zaembeddingowanym\*
- Testowanie komponentów EJB
- Wzorce Projektowe warstwy logiki i dostępu do logiki

#### 4. Java Persistence API2

- Koncepcja Mapowania relacyjno-objektowego
- Klasy Encyjne
  - klasy proste
  - klasy zagnieżdżone

- Pola i dostęp do pól
- Pola będące kolekcjami \*
- Asocjacje – z uwzględnieniem krotności i kierunkowości
- Klucze
  - Proste
  - Złożone
  - Automatyczne generowanie kluczy
- Nazewnictwo obiektów w bazie danych
  - Domyślne mapowanie
  - Zmiana domyślnych mapowań
- Kontekst trwałości (PersistentContext):
  - Zarządzany przez kontener i zarządzany przez aplikację
  - Transakcyjny(Transaction) i rozszerzony(Extended)
- Cykl życia encji
- Operacje na danych
- Transakcje
- Zapytania
  - Tworzenie zapytań w JPQL
    - Składnia
    - Query i TypedQuery\*
    - Zapytania nazwane
  - JPA2 Criteria API\*
    - Idea Criteria API, podstawowe składowe
    - MetaModel
      - Koncepcja
      - Klasy MetaModel w postaci kanonicznej i niekanonicznej
      - MetaModel API
    - Tworzenie silnie typowanych zapytań
    - Tworzenie słabo typowanych, dynamicznych zapytań.
    - Edycja zapytań
    - Podejścia alternatywne: LIQUidFORM
  - Typowe scenariusze, w tym serach-within i serach-by-example
  - Zapytania Natywne, mapowanie wyników
- Metody nasłuchiwanie cyklu życia
- Walidacja
  - Bean Validation 1.0 (JSR 303)
    - Koncepcja Bean Validation
    - Definiowanie i nakładanie ograniczeń na typy, pola, metody
    - Parametryzowanie ograniczeń
    - Wbudowane ograniczenia
    - Komponowanie ograniczeń złożonych
    - Tworzenie Walidatorów
    - Proces walidacji
      - Pola, obiekty i grafy obiektów
      - Grupy i sekwencje
      - Blokowanie dostępu do pola oraz kaskadowego dostępu
    - API i wyjątki
  - JPA2 a Bean Validation
    - Konfiguracja walidacji
    - Zasada działania Bean Validation w kontekście JPA
    - Ograniczenia walidujące a generacja obiektów bazy danych
- Modele i implementacja dziedziczenia,

- Zagadnienia spójności danych,
  - optimistic locking
  - pesimistic locking
  - Rozszerzenia w JPA2 \*
- Cache drugiego poziomu
  - Idea cache 2 poziomu,
  - Konfiguracja i tryby pracy cache
  - Konfiguracja encji
  - Interakcja cache <-> baza
  - Cache w oparciu o Hibernate (dla szkolenia bazującego na hibernate)
  - Cache w oparciu o mechanizmy JPA2\*
  - Zagrożenia
- Testowanie komponentów wykorzystujących JPA
- Konfiguracja i deployment
  - Jako aplikacja Java SE / dektop
  - Jako moduł web
  - W obrębie modułu ejb
- JPA a biblioteki dostawców(Hibernate,Toplink,EclipseLink,OpenJPA)
- Zagadnienia specyficzne dla Hibernate (dla szkolenia bazującego na hibernate)
  - Hibernate API
  - Wybrane dodatkowe możliwości
  - Podprojekty Hibernate: Search, Shards, Validator, Envers
- Wzorce Projektowe warstwy dostępu do danych

## 5. Interfejs Webowy

- Wprowadzenie do JSF
  - i. Typowe problemy aplikacji Webowej
  - ii. Wprowadzenie do modelu komponentowo-zdarzeniowego
- Cykl przetwarzania żądania
  - i. Rodzaje żądań
  - ii. Fazy cyklu
  - iii. Przetwarzanie pełne i częściowe
- Komponentowy model interfejsu
- Główne klasy i interfejsy
- Model zarządzania beanami logiki
  - i. Idea i rodzaje zasięgów
  - ii. Konfiguracja
  - iii. Zasięgi „Custom”
- Expression Language
- Definiowanie widoku
  - i. Przegląd dostępnych komponentów standartowych
    - 1. Biblioteka HTML
    - 2. Biblioteka Core
  - ii. View Declaration Language
  - iii. JSF w oparciu o facelets
    - 1. Korzyści wynikające z technologii
    - 2. Mechanizmy szablونowania
    - 3. Wbudowane znaczniki
    - 4. Tworzenie komponentów złożonych (Composite Components)
  - iv. Historia - JSF w oparciu o JSP
  - v. JSF a REST

- Model konwersji
  - i. Konwertery wbudowane
  - ii. Tworzenie konwerterów
- Model walidacji
  - i. Walidatory wbudowane
  - ii. Tworzenie walidatorów
  - iii. Walidator domyślny
  - iv. Bean Validation 1.0 (JSR 303) a JSF
  - v. Zdarzenia związane z walidacją
  - vi. Realizacja nietypowych/złożonych scenariuszy walidacji
- Model zdarzeń
  - i. Zdarzenia i słuchacze – podstawy modelu zdarzeniowego
  - ii. Zdarzenia akcji i zmiany wartości
  - iii. Zdarzenia zmiany faz
  - iv. Zdarzenia systemowe
- Model nawigacji
  - i. Nawigacja „klasyczna”
  - ii. Nawigacja skrócona
  - iii. Nawigacja warunkowa
  - iv. Dynamiczna konfiguracja nawigacji
  - v. Pull i push MVC
- Główne składowe aplikacji JSF i internacjonalizacja
- Typowe problemy przy tworzeniu interfejsu min
  - i. Formularz w postaci tabelki
  - ii. „Inteligentna” paginacja
  - iii. Master-Detail, LOV
- Obsługa błędów
- Zasoby aplikacji
  - i. Żądanie o zasoby (Resource Request) i jego obsługa
  - ii. Zależności od zasobów
  - iii. Znaczniki i obiekty EL
  - iv. Pakowanie i internacjonalizacja zasobów
- Wzbogacanie komponentów – Behaviour
- Tworzenie komponentów „od zera”
  - i. Tworzenie komponentów
  - ii. Tworzenie rendererów
  - iii. Tworzenie znaczników
- Pakowanie i konfiguracja aplikacji, fazy projektu
- Testowanie interfejsu zbudowanego na JSF (JSFUnit/Selenium)
- Zabezpieczanie aplikacji

## 6. Tworzenie aplikacji WWW z użyciem technologii AJAX i JSF

- a. Koncepcja AJAX i modele komunikacji
- b. Przegląd podejść do tworzenia aplikacji AJAX (warstwowe – JSF, klient-serwer – GWT, ...)
- c. Tworzenie aplikacji AJAX „od zera” (XMLHttpRequest)
- d. Podstawy bibliotek Prototype i JQuery w kontekście manipulacji drzewem DOM i komunikacji AJAX
- e. „Natywne” podejście JSF2
  - i. Znacznik f:ajax i Ajax Behaviour
  - ii. Przetwarzanie żądań typu Ajax
    1. Przetwarzanie częściowe (Partial processing)
    2. Selekttywne działania na drzewie komponentów

- iii. Kolejowanie
- iv. obsługa błędów
- f. Ajax4JSF
  - i. Wady i zalety podejścia, porównanie z podejściem natywnym
  - ii. Architektura silnika – zasada działania
  - iii. Przegląd możliwości
  - iv. Zagadnienia związane z wydajnością (kolejkowanie, opóźnianie, unieważnianie żądań, zmniejszanie wielkości żądania)
  - v. Wpływanie na cykl życia JSF
  - vi. Ajax4JSF a JSON
- g. Richfaces
  - i. Przegląd możliwości Richfaces
  - ii. Konfiguracja wyglądu (skórki)
  - iii. Walidacja
  - iv. RichFacej a jquery, *script.aculo.us*

## 7. CDI (webbeans)

- Podstawowe pojęcia
  - Ziarno
  - Kontekst
  - Zasięg
  - Wstrzeliwanie zależności
- Ziarna
  - Managed beans
  - Session means
  - Specjalizacja
  - Aktywne i nieaktywne ziarna
- Zasięgi
  - „klasyczne” zasięgi wbudowane
  - Zasięg konwersacji
  - Zasięg zależny
  - Tworzenie niestandardowych zasięgów
  - Zasięg domyślny
  - Cykl życia obiektów kontekstowych
- Wstrzeliwanie zależności
  - Miejsca docelowe wstrzeliwania
  - Kwalifikatory, Alternatywy, stereotypy
  - Wstrzykiwanie po nazwie
- Metody fabrykujące
- Model zdarzeń i słuchaczy
- Użycie CDI w kontekście aplikacji EJB/JSF