

Kod szkolenia: **EEMI**

Tytuł szkolenia: **Migracja starszych aplikacji Java Enterprise do wersji JEE7 – dobre praktyki**

Dni: **3**

Opis:

Adresaci szkolenia:

Szkolenie adresowane jest do programistów oraz architektów platformy JEE, pragnących uzupełnić swoje kwalifikacje o wiedzę o najnowszej wersji 7 tej technologii, różnicach w stosunku do poprzednich wersji oraz przygotować się do realizacji zadania jakim jest migracja istniejącej aplikacji JEE do wersji 7.

Cel szkolenia:

Uczestnicy dowiedzą się jakie zmiany niesie ze sobą wersja 7 platformy JEE. Celem szkolenia jest zdobycie wiadomości i umiejętności implementowania rozwiązań dostarczonych razem z JEE7. W szczególności uczestnicy poznają nowości w specyfikacji EJB3.2, JPA2.1, Servlet3.1, JMS2.0 i inne oraz dowiedzą się jak dokonać migracji istniejącej już aplikacji JEE do wersji 7 tej technologii, jakie są dobre praktyki, czego unikać tak, aby cały proces przeszedł bezproblemowo.

Mocne strony szkolenia:

Program obejmuje całościowo i wyczerpująco zagadnienia nowości jakie zmiany niesie ze sobą wersja 7 platformy JEE. W przeciwieństwie do większości tego typu szkoleń, szkolenie to zawiera część warsztatową, która pozwoli na ugruntowanie wiedzy również w praktycznym jej aspekcie.

Wymagania:

Od uczestników wymagana jest przynajmniej podstawowa umiejętność programowania w języku Java (do poznania na kursie J/JP) oraz wiedza w zakresie architektury JEE.

Parametry szkolenia:

3*8 godzin (3*7 netto) wykładów i warsztatów. Wielkość grupy: maks. 8-10 osób

Program szkolenia:



1. Wstęp
 - I. Wprowadzenie do platformy Java Enterprise Edition
 - II. Historia platformy JEE
 - III. Zarys funkcjonalności w JEE5, JEE6, JEE7
2. Migracja w warstwie logiki biznesowej
 - I. Migracja w warstwie EJB
 - a. Porównanie możliwości beanów EJB 3.0, 3.1, 3.2
 - b. Nowe możliwości:
 - \. Przetwarzanie asynchroniczne
 - \. Rozbudowane możliwości timerów
 - \. Singletony
 - \. Beany lokalne, bezinterfejsowe
 - c. Zmiany w kodzie i adnotacjach
 - d. Zmiany w deskryptorach wdrożenia *-ejb-jar.xml
 - e. Wymiana bibliotek JAR, modułów serwera itd
 - II. Migracja w warstwie CDI
 - a. Porównanie możliwości beanów CDI 1.0, 1.1
 - b. Nowe możliwości
 - \. Praca poza kontenerem JEE
 - \. Statyczne wstrzyknięcia
 - c. Zmiany w kodzie i adnotacjach
 - d. Zmiany w deskryptorach wdrożenia *-beans.xml
 - e. Wymiana bibliotek JAR, modułów serwera itd.
 - f. Migracja z SEAM do WELD
 - III. Migracja w warstwie JPA
 - a. Porównanie możliwości beanów JPA 2.0, 2.1
 - b. Nowe możliwości
 - \. Criteria API
 - \. Wsparcie dla procedur składowanych bazy danych
 - \. Niesynchronizowany kontekst persystencji
 - \. Rozbudowa możliwości JPQL/Criteria – podzapytania arytmetyczne, JOIN ON, TREAT itd.
 - c. Zmiany w kodzie i adnotacjach
 - d. Zmiany w deskryptorach wdrożenia persistence.xml, *-orm.xml
 - e. Wymiana bibliotek JAR, modułów serwera itd
 - IV. Migracja w JNDI
 - a. Standaryzacja przestrzeni nazw JNDI w JEE
 - \. java:comp/
 - \. java:module/
 - \. java:app/
 - \. java:global/
 - b. Przestrzenie nazw wykorzystywane przez serwera aplikacji np.:
java:jboss/
3. Migracja w warstwie prezentacji
 - I. Migracja serwletów
 - a. Porównanie możliwości Servlet 2.5, 3.0, 3.1

- b. Nowe możliwości
 - \. Nieblokujące operacje I/O
 - \. Wsparcie dla HTTP 1.1
 - \. Bezpieczeństwo
 - \. Wsparcie dla WebSocket
 - c. Zmiany w kodzie i adnotacjach
 - d. Zmiany w deskryptorach wdrożenia *-web.xml
 - e. Wymiana bibliotek JAR, modułów serwera itd
- ## II. Migracja JSF
- a. Porównanie możliwości JSF 1.2, 2.0, 2.1, 2.2
 - b. Nowe możliwości
 - \. Wsparcie dla HTML5 i JSON
 - \. Integracja z CDI
 - \. Asynchroniczność
 - \. AJAX
 - \. Szablonowanie
 - \. Bezpieczeństwo
 - c. Problemy z migracją JSF1.x do JSF2.x
 - d. Expression Language
 - e. Zmiany w kodzie, adnotacjach, tagach
 - f. Zmiany w deskryptorach wdrożenia *-faces-config.xml
 - g. Wymiana bibliotek JAR, modułów serwera itd.
- ## 4. Zmiany w warstwie komunikacji
- ### I. Migracja JAX-WS
- a. Zmiany i kompatybilność API JAX-WS 2.0, 2.1, 2.2
 - b. Migracja JAX-RPC do JAX-WS
 - c. Nowe możliwości
 - \. Przetwarzanie asynchroniczne
 - \. Stateful Webservice
 - \. WS-Addressing, WSIT, JAXB
- ### II. Migracja JAX-RS
- a. Zmiany i kompatybilność API JAX-RS 1.0, 2.0
 - b. REST'owe webserwisy oparte o serwlety a JAX-RS
 - c. Nowe możliwości
 - \. Filters & Interceptors
 - \. Przetwarzanie asynchroniczne
 - \. Wsparcie dla hypermedia, negocjacja treści
- ### III. Migracja JMS
- a. Zmiany i kompatybilność API JMS 1.0, 1.1, 2.0
 - b. Nowe możliwości
 - \. Przetwarzanie asynchroniczne
 - c. Zmiany w deskryptorach wdrożenia *-jms.xml
- ## 5. Nowości w JEE7
- ### I. Przetwarzanie wsadowe - Batch Application API 1.0
- a. Wprowadzenie do przetwarzania wsadowego, pojęcie Jobu
 - b. Job Specification Language (JSL)

- c. Tworzenie artefaktów
 - d. Operacje na Jobach
 - II. Współbieżność - Concurrency Utilities API 1.0
 - a. Podstawy współbieżności, wątki i procesy, problemy
 - b. Komponenty: ManagedExecutorService, ManagedScheduledExecutorService, ContextService, ManagedThreadFactory
 - c. Współbieżność a transakcje
 - d. Bezpieczeństwo
 - III. WebSocket 1.0
 - a. Tworzenie i konfigurowanie endpointów
 - b. Wysyłanie i odbieranie wiadomości
 - IV. Wsparcie dla przetwarzania JSON i HTML5
 - a. Parsowanie
 - b. Generowanie
- 6. Deployment aplikacji JEE5, JEE6 na serwerach aplikacji zgodnych z JEE7
 - I. Na co zwrócić uwagę
 - II. Dobre praktyki
 - III. Mieszanie standardów np.: wywołania EJB2.x z EJB3.x
 - IV. Zmiany w konstrukcji serwerów aplikacji na przykładzie JBOSS AS 5/7/8
 - a. Nowa architektura mikrojądra
 - b. Zmiany w modelu classloadingu
 - c. Nowe moduły serwera
 - ` . HornetQ a JBoss Messaging
 - ` . Infinispan a JBoss Cache
 - ` . pozostałe
 - d. Migracja deskryptorów wdrożenia JBOSS
- 7. Wsparcie Maven dla JEE7
 - I. Zmiany w pom.xml
 - II. Dodatkowe pluginy, rozszerzenia

