

Kod szkolenia: **J/REST**

Tytuł szkolenia: **Tworzenie RESTful webservice na platformie Java**

Dni: 2

Opis:

Adresaci szkolenia:

Szkolenie adresowane do programistów Java, analityków systemowych oraz architektów pragnących tworzyć zaawansowane RESTowe usługi sieciowe.

Cel szkolenia:

Celem szkolenia jest zapoznanie jego uczestników z zaawansowanymi technikami tworzenia usług RESTowych. Uczestnicy szkolenia poznają szczegóły protokołu http, techniki implementacji usług RESTowych, sposoby ich konsumowania, standardy opisywania i zabezpieczenia.

Szkolenie jest realizowane w kilku wariantach – dla programistów, dla analityków systemowych, architektów. W zależności od grupy szkoleniowej, główny nacisk położony jest na kwestie teoretyczne lub na praktyczne i implementację rozwiązań integracyjnych.

Wymagania:

Od uczestników wymagana jest podstawowa umiejętność programowania w języku Java. Zalecana jest również znajomość podstaw JEE.

Parametry szkolenia:

2*8 godzin (2*7 godzin netto) wykładów i warsztatów (z wyraźną przewagą warsztatów). W trakcie warsztatów implementowane są przykłady obrazujące zaawansowane możliwości i elastyczność usług webservice.

Program szkolenia:

1. Wstęp

- o zarysowanie najważniejszych pojęć i standardów związanych z webserwisami: XML, DTD, XSD, XSLT, SOAP, WSDL, WADL, JAX-RPC, JAX-WS, JAX-RS, JAXP, SAX, StAX, DOM, JAXB, JAXR, UDDI, SAAJ, JSON, AJAX, REST itd.
- o co to właściwie jest REST?
- o usługi RESTowe kontra SOAPowe

2. Warstwa transportowa
 - szczegółowe omówienie protokołu HTTP
 - stanowość w protokole HTTP, realizacja sesji w aplikacjach webowych
 - webserwisy a protokoły transportowe: SOAP over HTTP, JMS, SMTP/POP3, TCP, UDP; XML/JSON/XHTML over HTTP; pozostałe np.: WebSocket, MSMQ
3. Metadane, konfigurowalność i elastyczność
 - WSDL, WADL
 - JAX-R, UDDI, rejestry i repozytoria a REST
4. Bezpieczeństwo
 - problematyka bezpieczeństwa usług REST
 - uwierzytelnienie i autoryzacja na poziomie http, konfiguracja na poziomie kontenera servletów, JAAS, OAuth
 - przegląd standardów: PKI i X.509, SSL, nagłówki http odpowiadające za bezpieczeństwo, XML Digital Signature itd.
5. Niezawodność
 - problematyka niezawodności usług REST
6. Wydajność oraz efektywne przetwarzanie XML i JSON
 - metody zwiększenia wydajności przetwarzania na poziomie protokołu transportowego, przetwarzania protokołu warstwy aplikacji itd.
 - przegląd standardów: MTOM, XOP, MIME, SMIME, JAXP, SAX, StAX, DOM, JAXB, XPath, XSLT itd.
 - JSR 222, czyli Java Api for XML Binding (JAXB) 2.0 w szczegółach
 1. serializacja, externalizacja, marshalling
 2. architektura JAXB
 3. mapowanie klas Java na XML Schema
 4. marshalling, unmarshalling
 5. walidacja
 6. modyfikacja standardowych mapowań za pomocą adnotacji
7. Implementowanie usług REST w oparciu o Servlety
 - JSR 315, czyli Servlet 3.x w szczegółach:
 - I. adnotacje i deskryptor wdrożeniowy, możliwości metody
 - II. bezpieczeństwo
 - III. asynchroniczność
 - IV. pozostałe
 - filtry w aplikacji webowej
 - event listenery w aplikacji webowej
8. Implementowanie usług REST w oparciu o JAX-RS
 - JSR 311, czyli Java API for RESTful Web Services (JAX-RS) w szczegółach
 - I. adnotacje i deskryptor wdrożeniowy, podstawowe możliwości
 - II. praca z parametrami (@*Param)
 - III. praca z @Produces i @Consumes
 - IV. praca z kontekstem i @Context
 - V. pozostałe
 - JAXB i JAX-RS
 - I. praca z różnymi typami MIME i providerami
 - II. marshalling treści i adnotacja @Provider

- ogólnie na temat frameworków implementujących JAX-RS
 - I. Apache CXF
 - II. JBoss RESTEasy
 - III. Apache Wink
 - IV. inne, np.: komercyjne serwery aplikacji zgodne z JEE7
 - szczegółowo na temat JBoss RESTEasy
 - I. zgodność z JAX-RS
 - II. dodatkowe możliwości biblioteki
 - REST a popularne frameworki java
9. Implementowanie klientów usług REST
- JAX-RS Client API
 - Spring RestTemplate
 - AJAX
 - dowolny klient http

