

Kod szkolenia: **DIST**

Tytuł szkolenia: **Zapewnianie wysokiej dostępności aplikacji internetowych w oparciu o Spring, Terracotta, HA-JDBC i otwarte systemy baz danych**

Dni: 2

Opis:

Adresaci Szkolenia:

Szkolenie adresowane jest do projektantów i programistów aplikacji internetowych, którzy chcieliby rozszerzyć swoje umiejętności o budowanie aplikacji o wysokiej dostępności i skalowalności.

Cel szkolenia:

Na szkoleniu zostaną szczegółowo omówione różne architektury systemów rozproszonych. Uczestnicy poznają otwarte technologie umożliwiające budowanie rozproszonych aplikacji odpornych na awarie. Dowiedzą się jak projektować architekturę aplikacji, aby możliwe było łatwe zwiększanie wydajności przez dostawianie kolejnych serwerów. Duży nacisk zostanie położony na aspekty skalowalności baz danych.

W szczególności:

Uczestnicy poznają możliwości technologii Terracotta oraz JGroups do współdzielenia pamięci oraz wymiany komunikatów między węzłami klastra. Dowiedzą się jak integrować powyższe technologie z aplikacjami stworzonymi przy użyciu Spring Framework. Poznają narzędzia służące do równoważenia obciążenia takie jak HA-Proxy. Przećwiczą w praktyce skalowanie poziome bazy danych przy pomocy HA-JDBC oraz Hibernate Shards. Dowiedzą się jak wykorzystać mechanizmy wbudowane w otwarte systemy baz danych do zbudowania klastra master-master lub master-slave.

Mocne strony szkolenia:

Po szkoleniu uczestnicy będą gotowi samodzielnie projektować internetowe aplikacje o wysokiej dostępności oraz modyfikować istniejące aplikacje tak, aby były skalowalne i odporne na awarie.

Wymagania:

Wymagana jest umiejętność tworzenia aplikacji internetowych w Java (do poznania na kursie

J/WEB1 oraz J/WEB2), znajomość Spring-Framework i JPA lub Hibernate.

Parametry szkolenia:

2*8 godzin (2*7 godzin netto) wykładów i warsztatów (z wyraźną przewagą warsztatów).

Wielkość grupy: maks. 8-10 osób.

Program szkolenia:

1. Wprowadzenie - różne architektury klastrów i ich wpływ na niezawodność i skalowalność; twierdzenie CAP
2. Zapewnienie niezawodności i skalowalności warstwy danych:
 - I. Hot-standby oraz replikacja strumieniowa na przykładzie PostgreSQL
 - II. Replikacja multi-master na przykładzie HA-JDBC + PostgreSQL
 - III. Replikacja master-slave na przykładzie Slony-I + PostgreSQL
 - IV. Wprowadzenie do Apache Cassandra
3. Buforowanie danych
 - I. Konfiguracja rozproszonego cache (memcached lub ehcache)
 - II. Buforowanie a tuning GC w JVM, przechowywanie danych poza stertą
 - III. Buforowanie w bazach danych (na przykładzie PostgreSQL i/lub Cassandra)
4. Skalowanie i niezawodność warstwy WEB:
 - I. Load-balancing na przykładzie HA-Proxy
 - II. Replikacja sesji i load-balancing HTTP w Apache Tomcat

