

## **Kod szkolenia: J/GWTADV**

## **Tytuł szkolenia: Google Web Toolkit 2.x dla zaawansowanych**

### **Adresaci szkolenia:**

Szkolenie adresowane jest do programistów korzystających już z technologii Google Web Toolkit (GWT), którzy pragną rozszerzyć swoją wiedzę na temat najnowszych wersji tego frameworku oraz poznać dodatkowe biblioteki, ułatwiające pracę programisty.

### **Cel szkolenia:**

Na szkoleniu omówiona zostanie koncepcja architektury Model-View-Presenter, szyny aplikacyjnej oraz wzorca Command w zastosowaniu do aplikacji w GWT. Przedstawiony zostanie paradygmat wstrzykiwania zależności, a także możliwość integracji aplikacji GWT z biblioteką Google Guice, GIN i z JPA. Zaprezentowany zostanie proces tworzenia kompletnej aplikacji internetowej z wykorzystaniem GWT, wraz z aspektami walidacji danych wprowadzanych przez użytkownika, bezpieczeństwa, architektury i sposobów testowania. Przedstawiona zostanie również gama dodatkowych bibliotek wspierających tworzenie aplikacji w GWT.

### **Wymagania:**

Od uczestników szkolenia wymagana jest umiejętność programowania w języku Java (polecamy kurs J/JP) oraz podstawowa znajomość Google Web Toolkit (polecamy kurs J/GWTB lub J/GWT zamiast J/GWTADV).

Dodatkowo znajomość zagadnień tworzenia aplikacji internetowych w Java Enterprise Edition ułatwiłoby postrzeganie omawianej technologii na tle większego zagadnienia, jakim są RIA.

### **Mocne strony szkolenia:**

Szkolenie umożliwia szybką aktualizację wiedzy dotyczącej GWT (upgrade z wersji 1.x do 2.x) oraz zdobycie umiejętności tworzenia złożonych aplikacji z wykorzystaniem tego Framework, uwzględniając kwestie bezpieczeństwa, walidacji danych i integracji z innymi technologiami.

## Parametry szkolenia:

3\*7 godzin wykładów i warsztatów w proporcji 1/3. W trakcie warsztatów, oprócz prostych ćwiczeń, projektowana i implementowana jest aplikacja obejmująca zasięgiem wszystkie omawiane aspekty GWT i technologie dodatkowe.

Wielkość grupy: maks. 8-10 osób.

## Polecane szkolenia poprzedzające:

J/JP

## Program szkolenia:

1. Tworzenie warstwy interfejsu w GWT w podejściu deklaratywnym
  - a. Zalety i różnice w stosunku do podejścia programistycznego
  - b. Edytor graficzny (GWT Designer)
  - c. Tworzenie własnych komponentów
2. Wykorzystanie zasobów zewnętrznych
  - a. Zasoby binarne
  - b. Zasoby graficzne
  - c. Zasoby tekstowe
  - d. Arkusze stylów
3. Wykorzystanie komponentów do prezentacji danych (CellWidgets)
4. Zewnętrzne biblioteki komponentów (przeгляд)
  - Ext GWT (GXT)
  - Smart GWT
  - Vaadin
5. Architektura aplikacji GWT
  - a. Wzorzec Model-View-Presenter
  - b. Event Bus

- c. Wzorzec Command
  - d. Walidacja danych
  - e. Data binding – Editors Framework
6. Zagadnienia zaawansowane i dobre praktyki programistyczne
- a. Logowanie przebiegu działania aplikacji
  - b. Tworzenie niskopoziomowego kodu przy pomocy JSNI
  - c. Wykorzystanie zewnętrznych bibliotek JS w aplikacji GWT
  - d. Komunikacja JSON i XML z serwisami zdalnymi
  - e. Komunikacja z serwerem za pomocą RequestFactory
  - f. Walidacja danych za pomocą BeanValidation (JSR 303)
7. Bezpieczeństwo aplikacji
- a. Uwierzytelnianie użytkownika
  - b. Kontrola dostępu do zasobów
  - c. Obsługa sesji
  - d. Zapobieganie atakom typu XSS
8. Tworzenie rozwiązań portalowych z wykorzystaniem GWT
9. Testowanie i optymalizacja działania aplikacji
- a. Testowanie logiki warstwy klienckiej aplikacji (mock testing)
  - b. Testowanie interfejsu warstwy klienckiej aplikacji (automatyzacja testów w przeglądarce)
  - c. Dzielenie aplikacji na fragmenty pobierane asynchronicznie
  - d. Optymalizacja wielkości aplikacji oraz czasu kompilacji
  - e. Optymalizacja wywołań RPC
  - f. Wykrywanie „wąskich gardeł” aplikacji w warstwie klienckiej (analiza kodu Javascript i wykorzystanie SpeedTracer)
  - g. Testowanie warstwy serwerowej aplikacji

10. Realizacja paradygmatu wstrzykiwania zależności za pomocą Google GIN i Guice

- a. Omówienie koncepcji wstrzykiwania zależności
- b. Konfiguracja zależności
- c. Rodzaje wstrzykiwania zależności
- d. Dobre praktyki programistyczne

11. Przechowywanie stanu aplikacji po stronie serwera z wykorzystaniem JPA

12. Przegląd bibliotek dodatkowych:

- a. Przekazywanie obiektów klas encyjnych za pomocą Gilead
- b. Integracja ze Spring Framework za pomocą GWT-SL
- c. Zapewnienie komunikacji typu ajax-push za pomocą biblioteki gwt-comet

13. Google Web Toolkit a narzędzia budowania aplikacji

- a. Ant
- b. Maven

14. Przegląd możliwych integracji GWT z innymi technologiami

- a. Servlet + EJB
- b. Spring
- c. Seam

15. Wdrożenie aplikacji w usłudze Google App Engine

- a. Konfiguracja Google GIN i Guice
- b. Konfiguracja JPA
- c. Funkcjonalność nie wspierana przez Google App Engine

16. Podsumowanie zaproponowanego sposobu tworzenia aplikacji internetowych