



## **Laboratorium 6**

### **Jakarta Enterprise Beans + Jakarta Security API**

Studenci w ramach zajęć laboratoryjnych zapoznają się ze specyfikacjami Jakarta Enterprise Beans definiującą beany zarządzane EJB oraz oraz Jakarta Security API pozwalającą uwierzytelnianie i autoryzację użytkowników.

Należy zrealizować następujące zadania:

1. Obsługa użytkowników w systemie. Jeśli jeszcze nie zostało to zrobione to należy zdefiniować związek pomiędzy elementem a użytkownikiem. Jeden element może być przypisany tylko do jednego użytkownika. Wystarczy jeśli związek będzie zdefiniowany jako jednokierunkowy po stronie elementu. Do danych testowych ładowanych przy starcie aplikacji należy dodać również użytkowników i ich związki z elementami. Należy zapewnić możliwość rejestrowania nowych użytkowników za pomocą usługi typu REST. (0.5 + 0.5 punktu)
2. Implementacja logiki biznesowej w postaci beanów EJB. Należy zmienić implementacje serwisów logiki biznesowej tak aby były zrealizowane jako bezstanowe lokalne beany EJB. Należy też zapewnić aby repozytoria były niezależne od zasięgu żądania HTTP. (1 + 1 punkty)
3. Konfiguracja domeny bezpieczeństwa. W ramach aplikacji należy skonfigurować domenę bezpieczeństwa, tak aby użytkownicy byli pobieranie z bazy danych (połączenie udostępnione przez serwer) oraz ich uwierzytelnianie odbywało się za pomocą mechanizmu Basic authentication. (0.5 + 0.5 punktu)
4. Konfiguracja autoryzacji. Należy tak zabezpieczyć elementy logiki biznesowej oraz kontrolerów usług typu REST aby:
  - tylko administrator może dodawać i usuwać kategorie,
  - zwykły użytkownik może pobierać wszystkie kategorie,
  - każdy nowy użytkownik może się zarejestrować,
  - administrator może pobrać wszystkie elementy,
  - zwykły użytkownik może pobrać tylko swoje elementy (filtrowanie elementów powinno być realizowane na poziomie bazy danych),
  - administrator może edytować i usuwać wszystkie elementy,
  - zwykły użytkownik może edytować i usuwać tylko swoje elementy,
  - zwykły użytkownik może dodawać elementy, właściciel elementu jest ustawiany automatycznie na poziomie logiki biznesowej,



Poprawne wykonanie zadań wymaga m. in. właściwej dekompozycji na komponenty, poprawnego zarządzania stanem obiektów zarządzanych, konfiguracji połączenia z bazą danych po stronie serwera aplikacji, poprawnego wykorzystania zasięgu obiektu EntityManager oraz poprawnego wykorzystania ról użytkowników.

**Uwaga:** Podczas sprawdzania zadania będą sprawdzane tylko wywołania usług typu REST.

**Uwaga:** Podczas prezentacji rozwiązania należy pokazać, że żądania HTTP wykonują się poprawnie. Przygotowywanie warstwy widoku nie jest tutaj konieczne. Można wykorzystać np.: narzędzie HTTP Client dostępne w IntelliJ IDEA.

**Uwaga:** Wszystkie rozwiązania należy zachować na potrzeby kolejnych zadań, chyba, że w kolejnych instrukcjach będzie napisane inaczej.