



## **Laboratorium 7**

### **Jakarta Server Faces + Context and Dependency Injection**

Studenci w ramach zajęć laboratoryjnych zapoznają się z zaawansowanymi elementami znanych już specyfikacji Jakarta Server Faces i Context and Dependency Injection.

Należy zrealizować następujące zadania:

1. Konfiguracja autoryzacji. Należy pozostawić zrealizowane wcześniej zabezpieczenia logiki biznesowej i dodać zabezpieczenia w warstwie widoku: (0.5 + 0.5 pkt)
  - próba wejścia na zabezpieczony zasób przekierowuje na formularz logowania,
  - tylko zalogowany użytkownik może wyświetlać listę kategorii,
  - tylko administrator może (widzi przycisk) usunąć wybraną kategorię,
  - zwykły użytkownik widzi tylko swoje elementy w kategorii,
  - administrator widzi wszystkie elementy w kategorii,
  - używając bezpośredniego linku do elementu tylko jego właściciel (lub administrator) może go wyświetlić (zarówno podgląd jak i edycja),
  - w ramach nagłówka należy wyświetlić nazwę zalogowanego użytkownika.
2. Lokalizacja językowa aplikacji. Wszystkie statyczne tekstowe elementy strony powinny być dostępne w dwóch językach wybieranych automatycznie na podstawie nagłówków przesyłanych przez przeglądarkę. Dodatkowo należy zrealizować dwie wersje językowe obrazka wykorzystanego w nagłówku strony. (0.5 + 0.5 pkt)
3. Usuwanie zarówno elementów jak i kategorii powinno być zrealizowane z wykorzystaniem technologii AJAX tak aby zamiast przeładowywania całej strony aktualizowana była tylko wyświetlana lista. (0.5 + 0.5 pkt)
4. Należy dodać przecięcie logujące (za pomocą standardowego mechanizmu logowania zdarzeń) operacje dodania, usunięcia i edycji elementów przez użytkownika. Na logach powinna się znaleźć nazwa użytkownika, nazwa operacji oraz identyfikator zasobu. (0.5 + 0.5 pkt)
5. Implementacja chatu. Należy stworzyć nowy widok reprezentujący chat dla zalogowanych użytkowników. Wysyłanie wiadomości powinno być zrealizowane z wykorzystaniem żądania AJAX, natomiast odbieranie wiadomości powinno być zrealizowane z wykorzystaniem WebSocket. Użytkownik powinien mieć możliwość wysłania wiadomości do wszystkich jak i do jednego wybranego użytkownika. Obsługa wiadomości powinna być zrealizowana z wykorzystaniem mechanizmu zdarzeń CDI (wysyłanie wiadomości powoduje stworzenie zdarzenia, odebranie zdarzenia wysyła wiadomość za pośrednictwem gniazdek). (1 + 1 pkt)



Poprawne wykonanie zadań wymaga m. in. właściwej dekompozycji na komponenty, poprawnego wykorzystania tagów HTML (w tym tagów semantycznych) i stylów CSS, poprawnego zarządzania stanem obiektów zarządzanych oraz poprawnego wykorzystania kwalifikatorów CDI.

**Uwaga:** Podczas sprawdzania zadania będą sprawdzane tylko wywołania w ramach widoków JSF.

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