# Zaawansowane Metody Wytwarzania Oprogramowania Programowanie Aspektowe

## 1. Przygotowanie do laboratorium:

W celu przygotowania do laboratorium należy zapoznać się z:

- materiałami wykładowymi ZMWO na temat programowania aspektowego
- tutorialem opisującym wykorzystanie AJDT w środowisku Eclipse

Materiały zamieszczono na stronie przedmiotu.

## 2. Treść ćwiczenia laboratoryjnego:

Firma Telecom Inc. jest potentatem na rynku usług telekomunikacyjnych. Cieszy się poważaniem klientów i znacznymi dochodami. Pracowników mocno poruszyły ostatnie wydarzenia w dziale IT. Główny informatyk firmy został zwolniony z pracy w związku z próbą manipulacji bazą danych klientów w celu osiągnięcia korzyści majątkowych. Od tego czasu, systemy informatyczne Telecom zaczęły ulegać niewytłumaczalnym awariom.

Jesteś pracownikiem Aspect Advanced. Twoja firma świadczy usługi w zakresie zastosowań technik projektowania i programowania aspektowego. Jest znana ze skutecznego wspomagania tworzenia oprogramowania, poszerzania funkcjonalności oraz profilowania kodu.

#### Zadanie 1:

Zostałeś wyznaczony przez swojego szefa do zbadania systemu monitorowania rozmów telefonicznych w jednym z oddziałów firmy. Od jakiegoś czasu system nie daje znaku życia. Zaraz po włączeniu zawiesza się, nie wyświetlając żadnego komunikatu. Zaburzenia są prawdopodobnie sprawką niedawno zwolnionego informatyka, który mści się na swoich przełożonych. Symptomy wskazują na użycie technik programowania aspektowego. **Zlokalizuj i napraw wadliwy fragment kodu**, odpowiedzialny za nietypowe zachowanie systemu.

#### Zadanie 2:

Dyrektor Telecom jest bardzo zadowolony z Twojej dotychczasowej pracy. Zleca Aspect Advanced profilowanie kodu w celu optymalizacji. Samej analizy kodu i konkretnych poprawek dokona odpowiedni zespół. Do analizy potrzeba jednak danych.

Zbuduj aspekt śledzący, który dla każdego wywołania metody zapisuje do pliku dziennika:

- klasę obiektu, na rzecz którego została wykonana metoda
- nazwy, typy i wartości argumentów metody
- wynik zwracany przez metodę
- czas wykonania metody podany w milisekundach

Punkty złączeń mają wystąpić tylko w oryginalnym kodzie aplikacji, z pominięciem rozszerzeń aspektowych. Do pomiaru czasu wykonania metod można zastosować przykładową klasę Timer z pakietu aspects.

### Zadanie 3:

Telecom kontynuuje współpracę z Aspect Advanced. Firma telekomunikacyjna pragnie wprowadzić nową funkcjonalność do swojego systemu monitorowania. Ze względu na próby włamania, konieczna jest autentykacja klienta przy wykonywaniu połączeń. Nowa usługa systemu może być zaimplementowana jako aspekt. Właśnie otrzymałeś nowe zadanie.

## Zaimplementuj aspekt, który:

- pobiera hasło od użytkownika
- weryfikuje hasło w momencie wykonywania przez użytkownika połączenia wychodzącego.

Informacje o użytkownikach i hasłach mogą być przechowywane w zwykłym pliku tekstowym lub np. w serializowanym słowniku.

### Zadanie 4:

Telecom przeżywa kolejny kryzys. Do systemu włamuje się haker, który zmienia hasło klienta korzystając z uprzywilejowanych aspektów. Twoim nowym zadaniem jest namierzenie wrogiego kodu. Rolę systemowego "policjanta" ma pełnić specjalny aspekt, który anuluje działania hakera. Zaimplementuj następujące elementy systemu:

Aspekt-haker:

- przechwytuje odpowiednie wywołania metod w aspekcie implementującym autentykację
- na podstawie przechwyconych danych i wiedzy hakera, zmienia hasło użytkownika zaraz po jego wprowadzeniu, czym uniemożliwia wykonywanie połączeń

### Aspekt-policjant:

- anuluje działania hakera, korzystając z wiedzy o wykorzystywanych przez niego aspektach

## Podpowiedź:

Punkty złączeń obejmujące wykonanie rady wyraża punkt przecięcia adviceexecution().

# 3. Sprawozdanie

Należy napisać i przesłać sprawozdanie z realizowanych zadań. W sprawozdaniu należy opisać metodę realizacji poszczególnych zadań, pokazać rezultaty wykonania (np. zrzuty ekranu). Należy również dostarczyć kod źródłowy.