

## **Tematyka zajęć: Projektowanie Obiektowe, I semestr II stopień studiów kierunku Informatyka**

W trakcie zajęć studenci zapoznają się z problematyką projektowania struktury obiektowej programu/systemu/aplikacji. W trakcie zajęć omawiane będą następujące tematy:

1. Diagramy UML przydatne przy modelowaniu obiektowym
2. Zasady projektowania obiektowego
  - (a) SOLID
  - (b) GRASP
3. Wzorce Projektowe (GoF)
4. Projektowanie domenowe
5. Analiza projektu obiektowego – metryki oprogramowania

### **Warunki zaliczenia:**

Na potrzeby uzyskania zaliczenia z przedmiotu student musi przedstawić projekt obiektowej budowy aplikacji. Projekt będzie musiał zawierać:

1. diagram przypadków użycia,
2. diagram klas,
3. co najmniej dwa z diagramów opisujący przebieg procesu(ów)<sup>1</sup> (diagram aktywności, diagram sekwencji, diagram przepływów, diagram stanów).
4. analizę projektu z wykorzystaniem co najmniej dwu metryk omawianych na zajęciach.

Na ocenę 4 lub wyżej powinna zostać wykonana implementacja całości lub chociaż znacznej części aplikacji w dowolnym języku obiektowym (Java, C++, Python, JavaScript/TypeScript...).

Każdy z projektów musi być stworzony indywidualnie, dodatkowo w celu uzyskania oceny student ma obowiązek wyjaśnić wszelkie założenia projektowe jak i wątpliwości w rozmowie z prowadzącym.

### **Propozycje projektów zaliczeniowych w semestrze zimowym 2022/23:**

Proszę o zaprojektowanie struktury obiektowej i stworzenie prostej aplikacji pokazującej działanie dla jednego z poniższych zagadnień:

---

<sup>1</sup>Podkreślono zalecane diagramy

1. Giełda papierów wartościowych (typowe elementy procesu: maklerzy, akcje, obligacje, klienci, ...)
2. Obsługa konta klienta indywidualnego w banku (typowe elementy procesu: personel bankowy, klient, operacje wpłaty wypłaty przelew ...)
3. Zakup elektroniczny biletu do teatru (typowe elementy procesu: spektakl, rezerwacja miejsc, klient, faktura ...)
4. Serwis samochodowy (typowe elementy procesu: klient, samochód, usterka, części, faktura, ...)
5. Inwentaryzacja sprzętu teatralnego (typowe elementy procesu: asortyment, spektakl, reżyser/scenograf, szafa/półka/wieszak, aktor/klient ...)
6. Inwentaryzacja galerii artystycznej (typowe elementy procesu: eksponat, twórca, właściciel, miejsce przechowania, miejsce wystawienia, ...)
7. Obsługa pasażera na lotnisku (typowe elementy procesu: personel lotniska, pasażer, lot, bagaż, karta pokładowa ...)
8. Zarządzanie ZOO (typowe elementy procesu: opiekun, zwierzę, pokarm, terminarz karmienia, miejsca pobytu, ...)
9. System zarządzania nieruchomościami (typowe elementy procesu: właściciel -osoba fizyczna/prawna, nieruchomość, rodzaj budowy, typ budynku, zarządca ...)
10. System informatyczny dla związku łowieckiego (typowe elementy procesu: myśliwi, pozwolenia na odstrzał, tereny łowieckie, zasoby łowne – zwierzyzna łowna, zasoby brońi ...)
11. Zarządzanie cmentarzem miejskim (typowe elementy procesu: kwatery, „denat”, najem i najemca, pogrzeb, personel ...)
12. Zarządzanie firmą serwisującą dźwigi windowe (typowe elementy procesu: personel, uprawnienia, windy, terminy przeglądów, stan, terminarz pracy ...)
13. Obsługa zakładu szwalniczego (typowe elementy procesu: klient, pracownik, zamówienie, stan odbiór, opłata, ...)
14. System wydawania dokumentów paszportowych (typowe elementy procesu: zamówienie, osoba zamawiająca, personel, rejestracja dokumentów procesu, informowanie klientów o odesłaniu paszportu z drukarni, ...)
15. System zarządzania transportem mpk (typowe elementy procesu: kierowcy, pojazdy/wagony, trasy, zmiany i remonty, rozkład jazdy, terminarz pracy, zajezdnie ...)

Każdy projekt powinien zawierać odpowiednie diagramy UML opisujące budowę i działanie aplikacji. Diagram przypadków użycia pokazujący zestaw funkcjonalności systemu, diagramy opisujące działanie np. diagram sekwencji oraz diagram aktywności, strukturę obiektową projektu powinien opisywać diagram klas. W diagramie klas musi zostać zaproponowane co najmniej 7 klas (należy zastosować dziedziczenie i asocjację pomiędzy obiektami). Projekt może być wzbogacony o implementację w wybranym języku programowania. Programy mają na celu pokazanie wykonalności zaproponowanego projektu. Wskazane typowe elementy procesu są tylko wskazówką i nie będą wymagane w razie pominięcia przez studenta. Projekty należy przesyłać, umieszczając w temacie maila skrót przedmiotu **DUIPO** oraz imię i nazwisko, na podany adres mail [zaliczenia\\_podlaski@fis.uni.lodz.pl](mailto:zaliczenia_podlaski@fis.uni.lodz.pl).