

PROJEKT PROGRAMOWANIE OBIEKTOWE C#

Informacje odnośnie realizacji projektu.

- I. Projekt, jest projektem samodzielnym w związku z tym musi być realizowany indywidualnie. Wymagania na poszczególne oceny umieszczone są w Załączniku 1.
- II. Projekt można realizować na dowolny wybrany temat, ale z wykorzystaniem języka C# (projekt konsolowy lub GUI). Przykładowe tematy, które można realizować umieszczono w Załączniku 2. Uwaga tematy nie mogą się powtarzać.
- III. Temat projektu należy podać do **30.11.2024. godz. 23:59** – temat wraz z informacjami z dokumentacji z pkt. 1, 2 3 (punkt 3 w późniejszym czasie może zostać rozszerzony/zmodyfikowany) należy umieścić w zadaniu projektowym na platformie e-Learningowej uczelni. W przypadku gdy student nie zadeklaruje samodzielnie tematu prowadzący sam przypisze temat do realizacji przez studenta. Temat proszę umieścić na forum grupy.
- IV. Projekt można konsultować w wyznaczonych terminach konsultacji w formie stacjonarnej lub on-line oraz w innych terminach po uprzednim kontakcie mailowym z prowadzącym.
- V. Dokumentacja musi być przygotowana zgodnie z regulaminem pisanie pracy dyplomowej, powinna być zredagowana w języku LaTeX, według załączonego szablonu (Dokumentacja_OOP.zip). Dokumentacja powinna zawierać:
 1. Stronę tytułową
 2. Spis treści
 3. Opis założeń projektu/ cele projektu - należy opisać założenia do projektu oraz cele. Zdefiniować wymagania funkcjonalne i нефункционалне.
 4. Opis struktury projektu – wygenerowanie diagramu klas + krótki opis. Opis techniczny projektu – Należy umieścić informacje odnośnie wykorzystywanego języka, narzędzi oraz minimalnych wymagań sprzętowych. Opisać zarządzanie danymi oraz BD.
 5. Harmonogram realizacji projektu – diagram Ganta.
 6. Repozytorium i system kontroli wersji.
 7. Prezentacja warstwy użytkowej projektu - należy przedstawić opis warstwy użytkowej projektu w tym celu należy umieścić opis aplikacji oraz PrtSc o których jest mowa.
 8. Podsumowanie – podsumowanie zrealizowanych prac oraz planowane dalsze prace rozwojowe projektu.
 9. Literatura
- VI. Zadania do zrealizowania przez studenta w ramach zajęć projektowych:

Zadanie 1. Zadeklarowanie tematu w zadaniu projektowym do 30.11.2024r godz. 23:59. **UWAGA!!! Po upływie terminu w/w nie można zmienić tematu.**

Zadanie 2. Zapoznanie się z wymaganiami dotyczącymi przygotowania dokumentacji projektu zgodnie z instrukcją pisanie pracy dyplomowej oraz według metodyki określonej w sprawie zasad organizowania i prowadzenia projektów.

Zadanie 3. Zrealizowanie projektu oraz przygotowanie dokumentacji zgodnie z wymaganiami.

Zadanie 4. Obrona projektu w terminie wskazanym przez prowadzącego (termin zostanie ustalony ze studentami zgodnie z organizacją roku akademickiego).

Zadanie 5. Po obronie projektu należy złożyć pracę w zadaniu projektowym zgodnie z instrukcjami podanymi przez prowadzącego. W przypadku nie złożenia pracy otrzyma się ocenę 2.0.

Załącznik 1.

Wymagania do projektu:

Na ocenę 3.0

Projekt:

1. Język programowania C# (aplikacja konsolowa lub GUI).
2. Pliki źródłowe na repozytorium.
3. Baza danych na plikach tekstowych.
4. Zarządzanie danymi (operacje CRUD) przy wykorzystaniu min. 7 klas, w tym min. 5 ułożonych w sensowną i sensownie wykorzystaną hierarchię.
5. Dokumentacja projektowa.

Na ocenę 4:

Projekt powinien pozwalać dodatkowo:

6. Pobieranie danych z wykorzystaniem interfejsów.
7. Obsługa wyjątków.
8. Walidacja danych.
9. Baza danych np. Sql.

Na ocenę 5:

10. Projekt powinien pozwalać dodatkowo na importowanie oraz deportowanie danych z plików xls lub csv.

Uwaga w przypadku gdy w:

- dokumentacji będą braki ocena końcowa zostanie obniżona,
- dokumentacja będzie przygotowana niezgodnie z wymaganiami formalnymi jak również z zasadami estetyki/wyglądu dokumentu student otrzyma ocenę 2.0.

Załącznik 2.

Przykładowe tematy projektów

- 1 **Aplikacja „Bankomat” konsolowa lub z interfejsem GUI.** Program ma symulować działanie bankomatu. Bankomat po włożeniu karty elektronicznej (karta kredytowa, płatnicza, bankomatowa) oraz weryfikacji posiadacza karty (sprawdzenie PIN), powinien wydawać żadaną kwotę pieniędzy. Program powinien być napisany w taki sposób, aby przed „instalacją” bankomatu można było określić jakie rodzaje kart elektronicznych są akceptowane (obsługiwać między innymi następujące karty: Visa, American Express, Visa Electron, Mastercard).
- 2 **Aplikacja „Automat z napojami” konsolowa lub z interfejsem GUI.** Program ma symulować działanie automatu z napojami, który po wrzuceniu odpowiedniej monety oraz wybraniu dowolnej pozycji wydaje zamówiony napój. Program powinien też mieć możliwość uruchomienia menu administracyjnego które pozwala na zarządzanie asortymentem oraz wyświetlanie listy transakcji.
- 3 **Aplikacja „Biblioteka” konsolowa lub z interfejsem GUI.** Program ma wspierać biblioteki w procesie obsługi czytelników. Powinna być możliwość wypożyczania książki, jej zwrotu oraz sprawdzenia czy książka jest możliwa dla wypożyczenia. Możliwość sprawdzenia aktualnie wypożyczonych książek przez czytelnika. Zarządzanie czytelnikami, dostępnymi książkami. Automatyczne wyliczanie kary za opóźnienie w zwrocie książki.
- 4 **Program „Algorytmy grafowe” konsolowy lub z interfejsem GUI,** program ma za zadanie realizowanie rozwiązywania różnych algorytmów grafowych, powinien posiadać moduł wczytywania i zapisywania danych grafu, wizualizację graficzną rozwiązywania danego problemu.
- 5 **Program do szeregowania zadań na maszynach równoległych** z wykorzystaniem różnych algorytmów (LPT, BPP). Program pozwala na zadanie ilości maszyn, zadań oraz pozwala na wizualizację wyniku.
- 6 **Aplikacja konsolowa lub z interfejsem GUI do zarządzania pojazdami .** Wprowadzanie i usuwanie pojazdu, wyświetlanie danych o pojeździe, odczyt i zapis bazy danych do pliku/bazy. W programie należy utworzyć klasę Pojazd i przynajmniej jej dwie klasy pochodne, np.: Osobowy i Motor. Użyć funkcji wyświetlającej dane wykorzystującej polimorfizm.
- 7 **Aplikacja konsolowa lub z interfejsem GUI do zarządzania sklepem.** Program ma wspierać obsługę sklepu dowolnego rodzaju. Powinna być możliwość przyjęcia towaru do sklepu oraz jego sprzedaż w sposób hurtowy i detaliczny. Dla sprzedaży hurtowej powinna być możliwość rejestrowania stałych klientów. „Sprzedaż” ma polegać na wybieraniu towarów do koszyka, naliczaniu zbiorczej ceny, zapłatę z pieniędzy posiadanych w portfelu oraz usuwanie „sprzedanych” towarów z magazynu.
- 8 **System zarządzania budynkiem.** Kontrola wejścia do budynku, zgłaszanie usterek, prób włamania, monitorowanie temperatury wody i powietrza, wyłączanie/włączanie światła, sieci elektrycznej, zamykanie/otwieranie budynku, itp.
- 9 **Elektroniczny system oceniania** Zarządzanie wstawianiem i wyświetlaniem ocen cząstkowych z różnych przedmiotów. Uprawniony użytkownik może wstawiać oceny, a student może tylko je wyświetlać. Tworzenie raportów i statystyk, itp.
- 10 **System obsługi parkingu.** Parking jest prostokątem zawierającym miejsca w ustalonej liczbie wierszy i kolumn. Przejazdy są co drugi wiersz. System pozwala na wprowadzanie nowych pojazdów (przyjazd), wyświetlenie informacji o wszystkich, oraz na usuwanie ich z listy (odjazd). System obsługuje każdy typ pojazdu. Może nim być:
 - motocykl – zajmuje 1 miejsce
 - samochód osobowy – zajmuje 2 miejsca obok siebie.

- autobus – zajmuje 4 miejsca (w dwóch sąsiednich wierszach i kolumnach)
- W klasie Parking napisz metodę wykonującą wizualizację parkingu (w konsoli znakami tekstowymi). Pojazd musi posiadać informacje o współrzędnych pól na których się znajduje, a także nr rejestracyjny. Można dodać nowy pojazd tylko wtedy, gdy jest puste miejsce na podanych współrzędnych. Dodatkowo każdy pojazd musi wyświetlić informacje o sobie (nr rejestracyjny i pola zajmowane, oraz czym on jest – samochód, motor, autobus). Dodatkowo każdy przyjazd i odjazd jest zapisywany w liście opisującej stan parkingu. W ramach listy – można wyszukać pojazd (wg np. rejestracyjnego lub daty i godziny przyjazdu/odjazdu).

11 Aplikacja konsolowa lub z interfejsem GUI do zarządzania magazynem. Program powinien pozwalać na przechowywanie różnych produktów w określonych terminach. Po przekroczeniu terminu naliczane są kary. Program pozwala na generowanie różnego rodzaju statystyk, np. ilości towaru przechowywanego przez poszczególnych użytkowników, użytkownicy którzy przekroczyli terminy, zyska za przechowywanie towaru w danym miesiącu, możliwość inwentaryzacji, itp.

12 System obsługi biura podróży. Program powinien pozwalać na przyjmowanie, edycje i usuwanie ofert. Dodatkowo powinien umożliwiać przeglądanie ofert, rezerwacja biletów, rozliczanie się z biurem.

13 Wypożyczalnia dowolnego sprzętu. Program powinien mieć możliwość dodawania, usuwania, edycji dowolnego sprzętu, rezerwacji przez użytkownika wybranego sprzętu, naliczanie opłat za wypożyczanie, drukowanie paragonu lub faktury.

14 System rezerwacji sal/podział godzin. Obiekt planu posiada wykaz wszystkich zajęć dla różnych kierunków studiów i różnych form zajęć. Znaczenie ma forma zajęć:

- Laboratorium – odbywa się dla jednej grupy laboratoryjnej – więc posiada nr grupy.
- Wykład odbywa się dla wszystkich grup laboratoryjnych – całego kierunku, nie posiada więc dodatkowych informacji.
- Projekt – odbywa się dla dwóch wskazanych grup laboratoryjnych, który nr posiada.

Każda forma zajęć posiada kierunek (w tym semestr np. 6IIZ), nazwę przedmiotu, nazwisko prowadzącego i nr sali, dzień tygodnia i nr godziny zajęć. Posiada metodę zapytania o grupę – czy są to zajęcia dla grupy o określonym numerze (true/false). Obiekt planu powinien pozwalać na:

- wypisanie podziału dla wybranego dnia, dla wybranej grupy, lub dla wybranej sali.
- dodawanie i usuwanie zajęć
- Powinien też informować o błędzie gdy jest grupa która ma zajęcia w tym czasie, lub sala jest zajęta.

15 Aplikacja GUI do zarządzania hasłami. Aplikacja pozwala na wykonywanie operacji CRUD na przechowywanych hasłach w plikach lub bazie danych. Dane mają być szyfrowane z użyciem algorytmów szyfrowania.

16 Gra 2D lub 3D z Wykorzystaniem Silnika Unity. Kompleksowa gra z grafiką 2D lub 3D, wykorzystująca silnik Unity do renderowania i fizyki. Mechanika gry, zarządzanie zasobami, interakcje użytkownika, AI przeciwników.

17 Inny - Temat zaproponowany przez studenta, po uzgodnieniu z prowadzącym.