AKADEMIA GÓRNICZO-HUTNICZA

Dokumentacja do projektu

Baza do zarządzania pojazdami w wynajmowanej flocie

z przedmiotu

Języki Programowania Obiektowego

Elektronika i Telekomunikacja, rok III

Bartłomiej Kozieł

środa 11:30

prowadzący: mgr. Jakub Zimnol

12.01.2026r.

1. Opis projektu

Projekt stanowi konsolowy system informatyczny wspomagający działanie wypożyczalni pojazdów. Umożliwia on zarządzanie flotą pojazdów, obsługę klientów indywidualnych i biznesowych oraz realizację procesu wynajmu i zwrotu pojazdów wraz z automatycznym obliczaniem kosztów. System może znaleźć zastosowanie jako prosta aplikacja wewnętrzna lub baza pod dalszą rozbudowę (np. interfejs graficzny, baza danych).

2. Opis zaimplementowanych klas

Vehicle
Klasa bazowa reprezentująca pojazd. Przechowuje wspólne cechy wszystkich pojazdów, takie jak marka, model, cena wynajmu oraz status dostępności. Stanowi fundament hierarchii dziedziczenia.

CombustionVehicle
Klasa pochodna, rozszerzająca klasę Vehicle. Dodaje właściwości pojazdu typowe dla pojazdu spalinowego takie jak rodzaj paliwa, pojemność silnika czy spalanie.

CombustionCar / ElectricCar / Motorcycle / Truck
Klasy pochodne rozszerzające klasę Vehicle. Reprezentują konkretne typy pojazdów, uwzględniając ich charakterystyczne właściwości (np. rodzaj napędu, ilość drzwi, ładowność). Umożliwiają polimorficzne zarządzanie flotą.

Customer
Klasa opisująca klienta wypożyczalni. Przechowuje dane identyfikacyjne i kontaktowe oraz umożliwia rozróżnienie klientów prywatnych i biznesowych wraz z walidacją danych.

Rental
Klasa odpowiedzialna za logikę pojedynczego wypożyczenia. Łączy klienta z pojazdem, przechowuje informacje o czasie wynajmu i odpowiada za obliczenie końcowego kosztu.

VehicleManager
Centralna klasa zarządzająca systemem. Odpowiada za przechowywanie i koordynację danych dotyczących pojazdów, klientów oraz aktywnych i zakończonych wypożyczeń. Pełni rolę głównego „silnika” aplikacji.

UserInterface
Warstwa interakcji z użytkownikiem. Implementuje menu tekstowe, obsługuje komunikację z użytkownikiem oraz zapewnia walidację danych wejściowych, chroniąc system przed błędnymi danymi.

data.txt
W katalogu projektu znajduje się również plik data.txt, który przechowuje dane na temat bazy pojazdów pomiędzy jej uruchomieniami. Dane są z niego ładowane przy każdym starcie programu. Na razie w tym pliku znajdują się przykładowe pojazdy, klienci i wynajmy.

3. Instrukcja uruchomienia projektu (CMake)

Wymagania systemowe

- System operacyjny Windows 10 lub nowszy
- Visual Studio Build Tools z zainstalowanym workloadem *Programowanie aplikacji klasycznych w języku C++*
- CMake w wersji 3.20 lub nowszej

Instrukcja budowania

1. Uruchomić x64 Native Tools Command Prompt for Visual Studio
2. Przejść do katalogu głównego projektu
3. Utworzyć czysty katalog budowania i przejść do niego:
   ```
   rmdir build /s /q
   mkdir build
   cd build
   ```
4. Wygenerować pliki budowania przy użyciu CMake i generatora NMake:
   ```
   cmake .. -G "NMake Makefiles"
   ```
5. Skompilować projekt:
   ```
   cmake --build .
   ```
6. Można uruchomić program za pomocą “*VehicleRentalSystem.exe*” z terminala lub kliknąć dwukrotnie na ten plik w katalogu build.