

Damian Cygan

Numer indeksu: 12188

Grupa: lab1/BEDS

Wyższa Szkoła Ekonomii i Informatyki w Krakowie

PROJEKT
WYPOŻYCZALNIA SAMOCHODÓW

1. Wstęp.

Niniejszy projekt przedstawia system wypożyczalni samochodów na potrzeby zaliczenia zajęć z przedmiotu „Programowanie obiektowe” w Wyższej Szkole Ekonomii i Informatyki w Krakowie.

2. Charakterystyka.

Celem projektu było skonstruowanie poprawnie działającego systemu wypożyczalni samochodów, symulującego pracę prawdziwej firmy.

System opiera się na stworzonej na potrzeby projektu bazie danych, składającej się z sześciu tabel:

- Samochody
- Pracownicy
- Klienci
- Wynajem
- Cennik
- Numery rejestracyjne

W bazie przechowywane są podstawowe dane klientów którzy wypożyczyli samochód, takie jak: imię, nazwisko, adres, oraz dane pracowników, dział w którym pracują, stanowisko i pensja.

Oprócz tego baza przechowuje dane o samochodach w czterech kategoriach:

- auta luksusowe w cenie – 500zł/dobę,
- auta miejskie w cenie – 100zł/dobę,
- auta rodzinne w cenie – 200zł/dobę,
- auta dostawcze w cenie – 150zł/dobę

W tej wypożyczalni oferowane są samochody z silnikiem diesel oraz benzynowym. Oprócz tego firma oferuje samochody wyprodukowane tylko po dwutysięcznym roku.

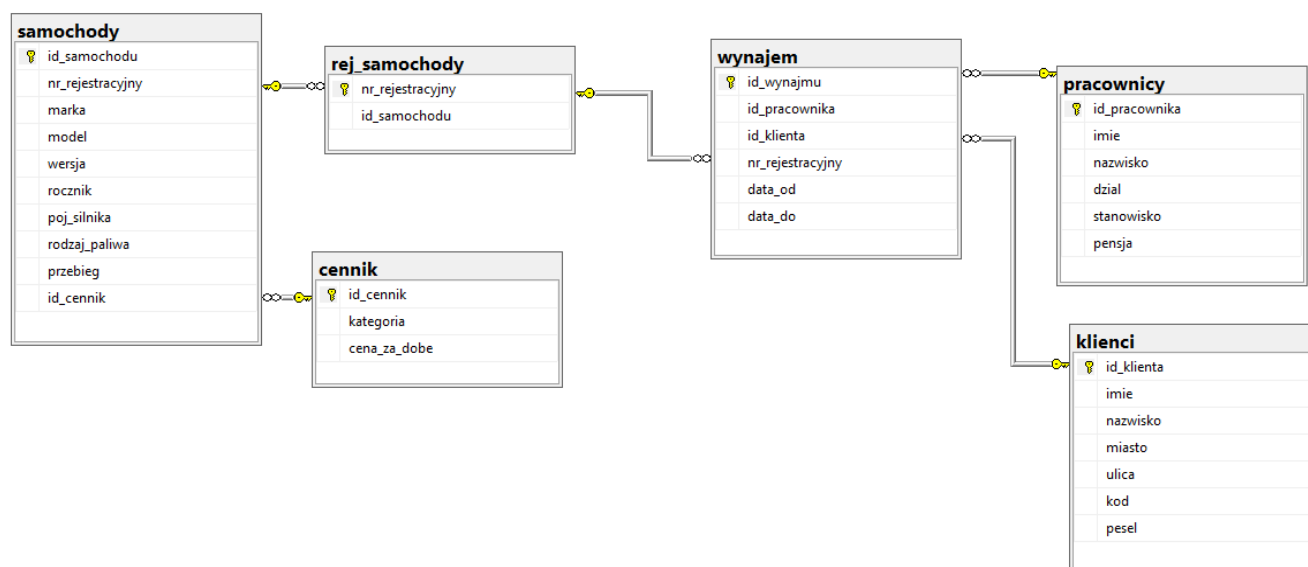
Każde wypożyczenie jest odnotowywane w tabeli „wynajem”, która przechowuje identyfikator klienta, pracownika, datę wypożyczenia i zwrócenia auta, oraz numer rejestracyjny, który z kolei reprezentuje konkretny samochód.

Oprócz tego każdy z klientów może dokonać wielu wypożyczeń, każdy pracownik może wypożyczyć samochody wielu klientom, a każdy samochód posiada swój numer rejestracyjny i może być wypożyczony jednemu klientowi.

3. Projekt systemu.

Głównym założeniem projektu jest to, żeby stworzyć aplikację, która będzie dawała możliwość edycji, dodawania oraz usuwania z bazy danych pracowników, klientów bądź samochodów. Ze względu na złożoność projektu z góry zakładam, że aplikacja będzie używana przez administratora, który jako jedyny będzie miał możliwość edycji pracowników w wypożyczalni.

3.1. Diagram bazy danych



3.2. Szata graficzna.

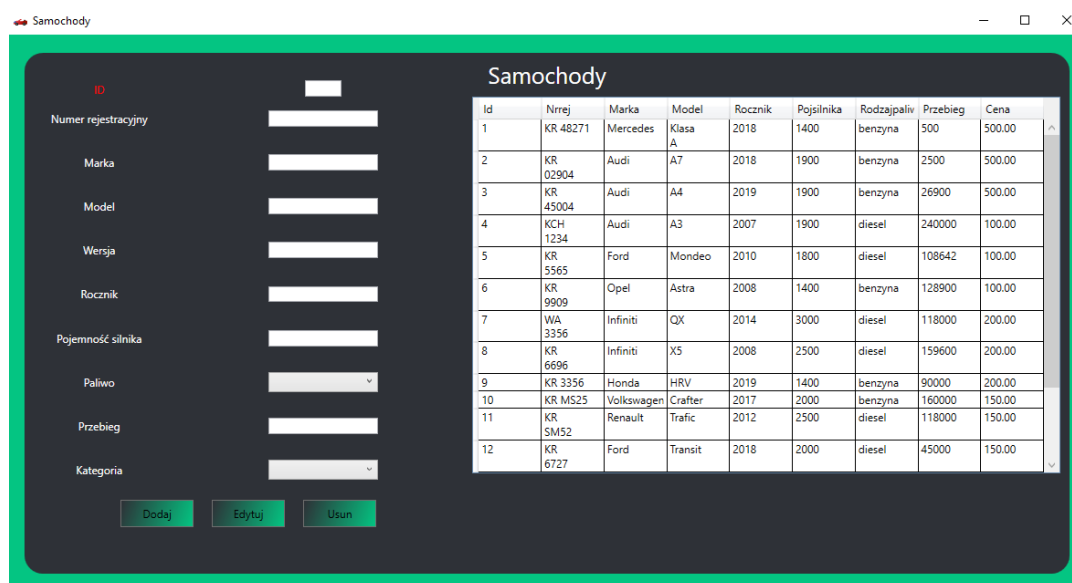
Wizualna część projektu została stworzona z wykorzystaniem struktury interfejsu użytkownika Windows Presentation Foundation (WPF).

Po uruchomieniu aplikacji oczom użytkownika ukazuje się okno główne aplikacji, które pełni rolę nawigatora. Użytkownik może z tego poziomu dostać się do okna: samochodów, wynajmu, pracowników oraz klientów (obraz nr.1).



Obraz nr.1

Okno „Samochody” pozwala na edycję rekordów w bazie danych, dodanie do niej nowych rekordów, bądź usunięcie istniejących. Daje także możliwość podglądu rekordów w tabeli „Samochody”, zawierającej aktualne dane samochodów m.in. takie jak: numer rejestracyjny, marka, model czy cena za dobę (Obraz nr.2).



Obraz nr.2

Okno „Wynajem” pozwala na wynajęcie dla klienta konkretnego modelu samochodu. Daje także możliwość podglądu rekordów w tabeli „Wynajem” zawierającej aktualnie wynajęte samochody (Obraz nr.3). Po kliknięciu w przycisk „Usuń wynajęcie” program przekierowuje nas do strony z formularzem, w którym daje możliwość usunięcia rekordu z wynajmem, bądź też ewentualną zmianą daty danego wynajęcia, w przypadku pomyłki (Obraz nr.4).

Wynajem

Wynajmij Samochód

Pracownik:

Klient:

Nr Rej:

Data od:

Data do:

Idwynajęcia	NrRej	Marka	Model	DataOd	DataDo	IdPracowni	IdKlienta
4	KCH 1234	Audi	A3	2/18/2021 1	2/25/2021 1	1	1
5	KR 5565	Ford	Mondeo	2/27/2021 1	2/28/2021 1	1	1
10	KR 3356	Honda	HRV	2/24/2021 1	2/25/2021 1	3	3
11	WA 3356	Infiniti	QX	2/28/2021 1	3/1/2021 12	1	5

Obraz nr.3

Wynajem

Aktualnie wynajęte

Idwynajęcia	NrRej	Marka	Model	DataOd	DataDo
4	KCH 1234	Audi	A3	2/18/2021 12:00:00	2/25/2021 12:00:00
5	KR 5565	Ford	Mondeo	2/27/2021 12:00:00	2/28/2021 12:00:00
10	KR 3356	Honda	HRV	2/24/2021 12:00:00	2/25/2021 12:00:00
11	WA 3356	Infiniti	QX	2/28/2021 12:00:00	3/1/2021 12:00:00

Obraz nr.4

W oknie „Pracownicy” stworzonego z myślą o funkcji dostępnej tylko dla administratora, możliwe jest usunięcie rekordu zawierającego dane pracownika, bądź też jego edycja lub dodanie nowego całkowicie nowego pracownika (Obraz nr.5).

Pracownicy

ID:

Imię:

Nazwisko:

Dział:

Stanowisko:

Pensja:

Id	Imię	Nazwisko	Dział	Stanowisko	Pensja
1	Jan	Kowalski	Wypożyczenia	Sprzedawca	2500.00
2	Stanisław	Stanisławski	Wypożyczenia	Sprzedawca	2500.00
3	Zbigniew	Dziedzic	Marketing	Menadżer	5000.00
5	Damian	Cygan	Wypożyczenia	Dyrektor	10000.00

Obraz nr.5

W oknie „Klienci” możliwe jest usunięcie rekordu zawierającego dane klienta, bądź też jego edycja lub dodanie nowego klienta (Obraz nr.6).

Id	Imie	Nazwisko	Miasto	Ulica	Kod	Pesel
1	Amian	Awalski	Chrzanów	Chrzanowska	32-500	12245678900
2	Bamian	Bawalski	Kraków	Krakowska	30-042	99090104412
3	Camian	Cawalski	Warszawa	Warszawska	00-014	91022402543
4	Damian	Dawalski	Katowice	Katowicka	40-021	99092504556
5	Emian	Ewalski	Kraków	św. Filipa	30-042	79121267787
8	Famian	Fawalski	Kraków	Krakowska	32-500	99999999999
10	Gamian	Gawalski	Kraków	Polna	32-500	00000000000

Obraz nr.6

3.3. Kod.

Każde z okien ingerujących w bazę danych, dla ułatwienia wprowadzanych danych zostało wyposażone w automatyczne wypełnianie formularza po wpisaniu numeru ID rekordu (Obraz nr.7).

Id	Imie	Nazwisko	Dział	Stanowisko	Pensja
1	Jan	Kowalski	Wypożyczania	Sprzedawca	2500.00
2	Stanisław	Stanisławski	Wypożyczania	Sprzedawca	2500.00
3	Zbigniew	Dziedzic	Marketing	Menadżer	5000.00
5	Damian	Cygan	Wypożyczania	Dyrektor	10000.00

Obraz nr.7

Wykorzystana została do tego funkcja `tb_GotFocus`, która wywoływana jest w pliku XAML w elemencie `TextBox` zawierającym ID danego rekordu. Kiedy tylko w polu ID formularza znajduje się cyfra, za pomocą polecenia LINQ funkcja wyciąga dane z bazy danych dla konkretnego ID i wpisuje je do odpowiednich pól w formularzu programu (Obraz nr.8).

```

private void tb_GotFocus(object sender, TextChangedEventArgs args)
{
    TextBox tb = sender as TextBox;
    if (tb != null && ID.Text.Length != 0)
    {
        var id = int.Parse(ID.Text);
        var editQuery = from item in _db.pracownicy
                        where item.id_pracownika.Equals(id)
                        select new
                        {
                            id = item.id_pracownika,
                            im = item.imie,
                            n = item.nazwisko,
                            dz = item.dzial,
                            stan = item.stanowisko,
                            pen = item.pensja
                        };
        foreach (var item in editQuery)
        {
            Imie.Text = item.im;
            Nazwisko.Text = item.n;
            Combobox_dzial.SelectedItem = item.dz;
            Combobox_stanowisko.SelectedItem = item.stan;
            Pensja.Text = item.pen.ToString();
        }
    }
}

```

Obraz nr.8

Program blokuje także możliwość wpisywania niepożądanych znaków do formularza, aby zapobiec błędnemu wprowadzaniu danych do bazy. Odpowiadająca za to funkcja „Walidacja” wywoływana jest w pliku XAML w tych elementach, w których dane muszą być tylko liczbami (Obraz nr.9).

```

private void Walidacja_numer(System.Object sender, System.Windows.Input.TextCompositionEventArgs e)
{
    e.Handled = IsTextNumeric(e.Text);
}
1 reference
private static bool IsTextNumeric(string str)
{
    System.Text.RegularExpressions.Regex reg = new System.Text.RegularExpressions.Regex("[^0-9]");
    return reg.IsMatch(str);
}

```

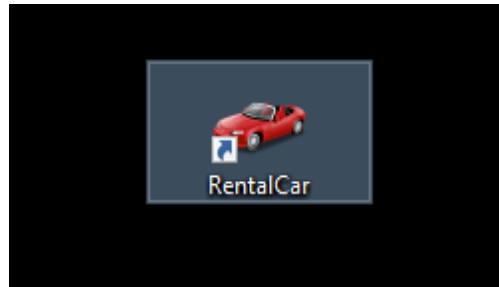
Obraz nr.9

3.4. Język programowania, wykorzystywane narzędzia, biblioteki.

- Język C#,
- Windows Presentation Foundation (WPF),
- Microsoft SQL Server Management,
- Entity Framework,
- LINQ,

3.5. Instalator.

Aplikacja zawiera także plik instalacyjny „Setup.exe” pozwalający na zainstalowanie programu na dowolnym komputerze.



4. Bibliografia.

1. <https://docs.microsoft.com/pl-pl/ef/>
2. https://www.youtube.com/watch?v=mTkh3sgh6tI&t=289s&ab_channel=asmak9