

**Dokumentacja projektu**

Przedmiot: **Programowanie**

Tytuł projektu: **Program obsługujący wypożyczalnię samochodów**

Prowadzący: Wykonawca:

*Dr Marek Jaszuk*  Aleksandra Krusiec

Grupa: GP01 / BB: 12 / 4 IIZ

Rzeszów, 2019

1. **Opis założeń projektu (wymagania biznesowe)**

Aplikacja ma za zadanie ułatwić i usystematyzować pracę wypożyczalni samochodów. Dzięki niej pracownicy będą mogli skontrolować dostępność pojazdów, dodać/usunąć auto z bazy oraz wypożyczyć i odebrać samochód od klienta. Aplikacja umożliwia również pracownikom zalogowanie się na ich indywidualne konto.

Pracownik wypożyczalni, za pomocą aplikacji, może dodawać do bazy pojazdów nowe samochody, usuwać je, przeglądać lub zmieniać. Każdy pojazd reprezentowany jest przez następujące dane: klasa, marka, model, nr VIN, nr rejestracyjny, rodzaj silnika, napęd, rok produkcji, koszt wynajęcia i status. Pracownik może również wypożyczyć samochód klientowi i odebrać go od niego. Klienta charakteryzują następujące dane: imię, nazwisko, numer prawa jazdy. Klient wypożyczający pojazd jest automatycznie dodawany do bazy danych.

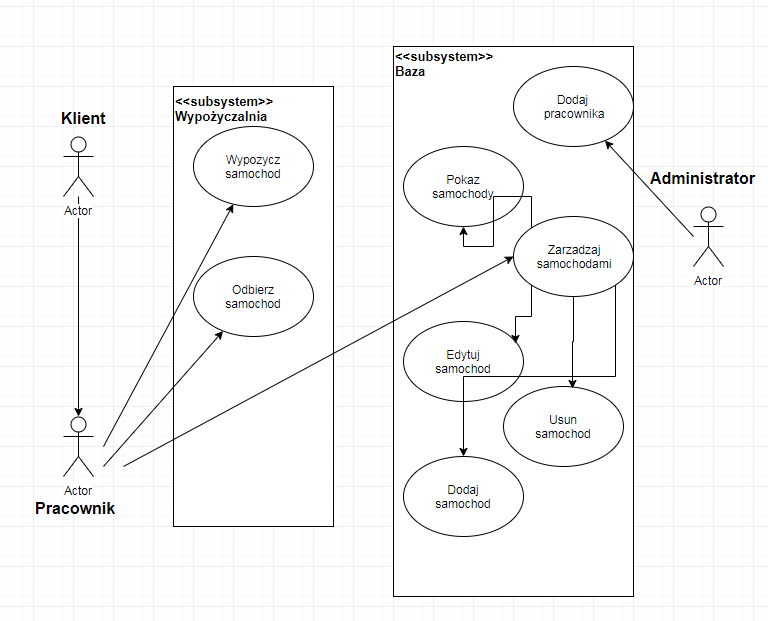
1. **Specyfikacja wymagań**

* Warunki wstępne:
  + Dostęp do narzędzia Visual Studio oraz Windows Forms
  + Dostęp do Microsoft SQL Server
* Wymagania funkcjonalne
  + zalogowanie pracownika na indywidualne konto
  + dodanie pojazdu do bazy
  + usunięcie pojazdu z bazy
  + przejrzenie pojazdów zapisanych w bazie
  + edycja danych zapisanych pojazdów
  + wypożyczenie auta
  + sprawdzenie dostępności samochodów
  + odbiór wypożyczonego pojazdu
  + wyszukanie konkretnego pojazdu w bazie
* Wymagania niefunkcjonalne:

Aplikacja zaprojektowana jest dla systemu Windows. Powinna uruchamiać się natychmiast po włączeniu. Program powinien umożliwiać bezbłędne i szybkie zalogowanie się zapisanego w bazie danych pracownika do aplikacji. Zalogowany pracownik ma mieć możliwość szybkiego przełączania się pomiędzy oknami. Dodawane pojazdy i akcje powinny mieć dokładne odzwierciedlenie w bazie danych, wszystkie podane w aplikacji informacje dotyczące samochodu lub klienta powinny być przeniesione do bazy. Korzystanie z aplikacji powinno być wygodne, funkcje i dostęp do nich ma być przejrzysty i łatwy do opanowania. Jednocześnie na jednym stanowisku pracy może być wykonywana jedna operacja (np. dodanie pojazdu, usunięcie, wypożyczenie).

1. **Diagram przypadków użycia z wyróżnieniem aktorów projektu, oraz przypadków użycia odzwierciedlających wymagania funkcjonalne.**

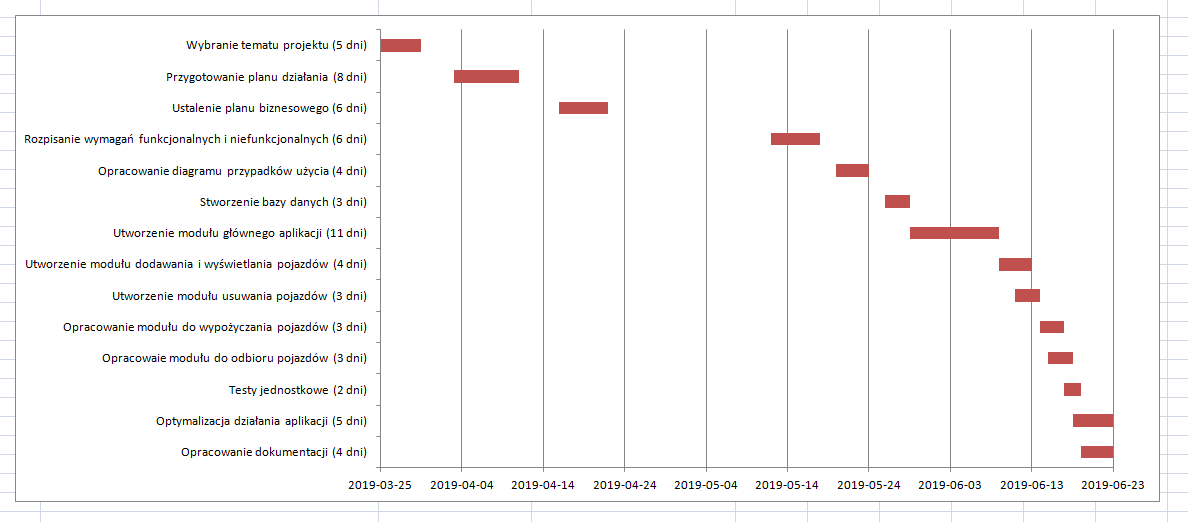
Diagram wykonano za pomocą narzędzia Draw.io (<https://www.draw.io/>)



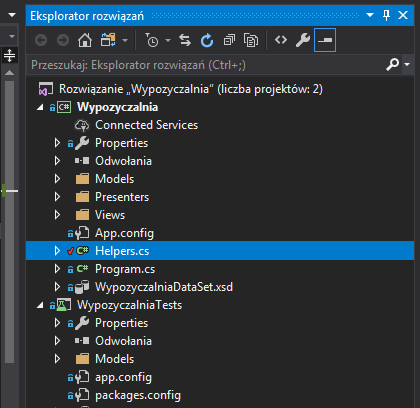
|  |  |  |
| --- | --- | --- |
| **AKTOR** | **OPIS** | **Przypadki użycia** |
| **Administrator** | *Administrator dodaje nowego pracownika do bazy danych. Nowy pracownik może być dodany wyłącznie z poziomu bazy. Nie ma możliwości dodania pracownika poprzez aplikację.* | * Dodaj pracownika |
| **Pracownik wypożyczalni** | *Pracownik loguje się na swoje konto w aplikacji, wypożycza i odbiera pojazdy od klientów. Jest odpowiedzialny za kontrolę i utrzymywanie zasobów wypożyczalni (dodawanie, edytowanie i usuwanie pojazdów z bazy).* | * Zaloguj się * Wypożycz samochód * Odbierz samochód * Zarządzaj samochodami * Dodaj samochód * Edytuj samochód * Usuń samochód |
| **Klient wypożyczalni** | *Klient wypożyczalni nie może bezpośrednie korzystać z systemu. Korzysta z usług pracownika.* | * Wypożycz samochód * Oddaj samochód |

1. **Harmonogram realizacji projektu (diagram Gantta).**

Diagram Gantta został przygotowany w programie Microsoft Excel.



1. **Opis techniczny projektu – w jaki sposób projekt został zrealizowany (struktura kodu programu).**

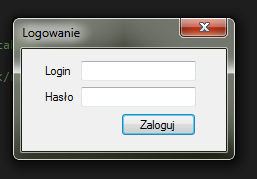
****

Program składa się z jednej solucji. W trakcie tworzenia programu starano się zastosować do wzorca projektowego MVP (Model-View-Presenter). Zgodnie z jego założeniami kod został podzielony na trzy części. W katalogu Views znajdują się klasy odpowiadające za wyświetlanie widoków. W katalogu Models przechowywane są klasy stanowiące odzwierciedlenie tabel zapisanych w bazie danych oraz klasy pozwalające na komunikowanie się z bazą. Katalog Presenters zawiera klasy pośredniczące między widokami a modelami.

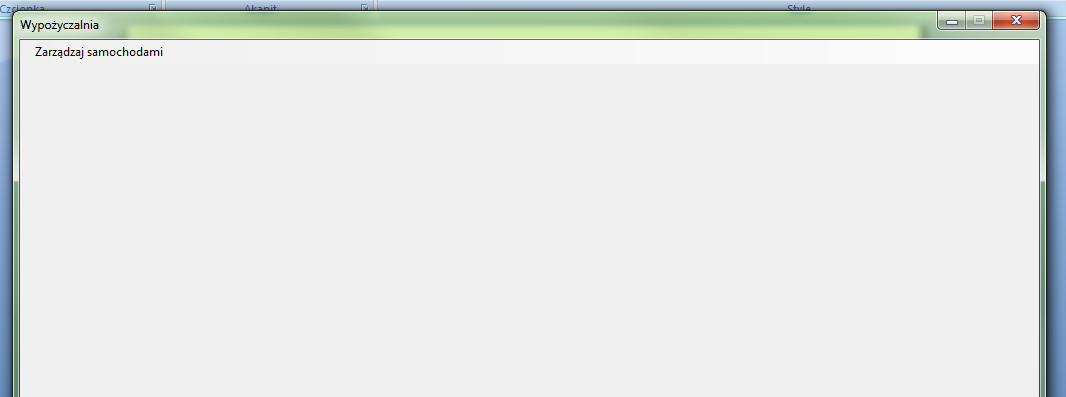
Jako bazę danych wykorzystano Microsoft SQL Server. Widoki napisano z wykorzystaniem technologii Windows Forms.

1. **Prezentacja warstwy użytkowej projektu (widoki ekranu z opisem ich użytkowania).**

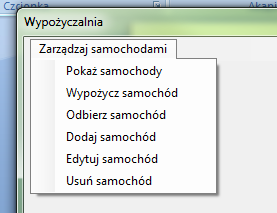
Okno logowania:



Po podaniu prawidłowych danych użytkownika (domyślnie: admin, admin) aplikacja otwiera się. Dostępny jest jeden przycisk – „Zarządzaj samochodami”:

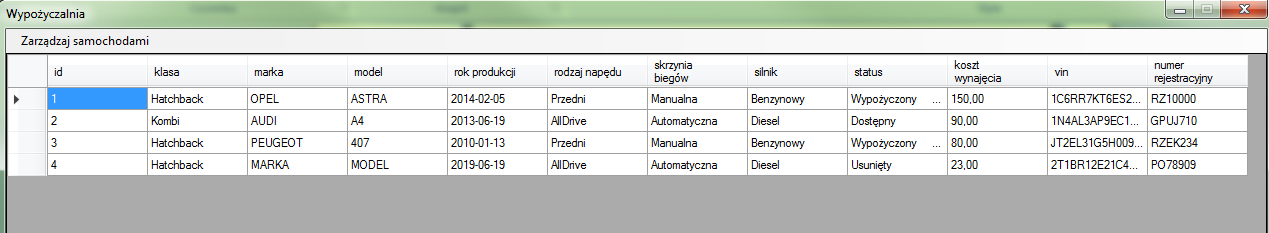


Po rozwinięciu listy użytkownik ma do wyboru następujące opcje:



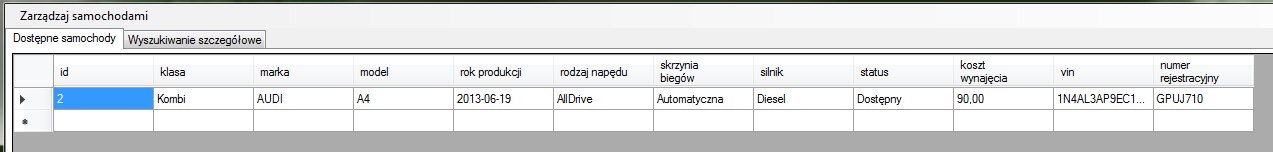
„*Pokaż samochody*”:

Wyświetlane są wszystkie samochody zapisane w bazie, także te usunięte, wraz z ich statusem (dostępny/wypożyczony/usunięty). W tym widoku użytkownik może zobaczyć jakie samochody posiada wypożyczalnia.

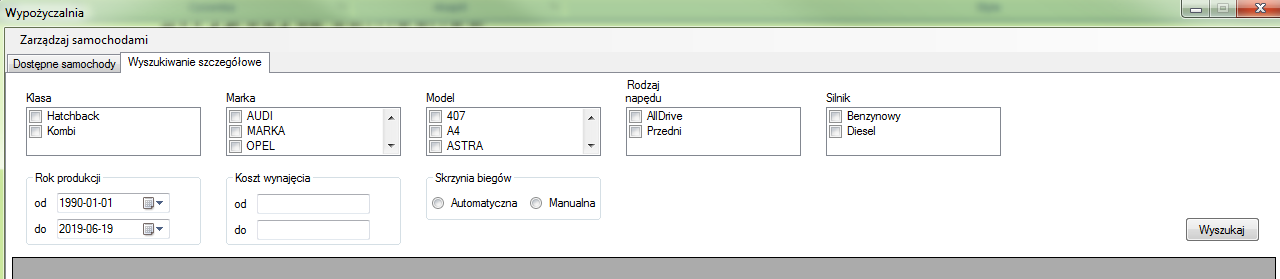


„*Wypożycz samochód”*

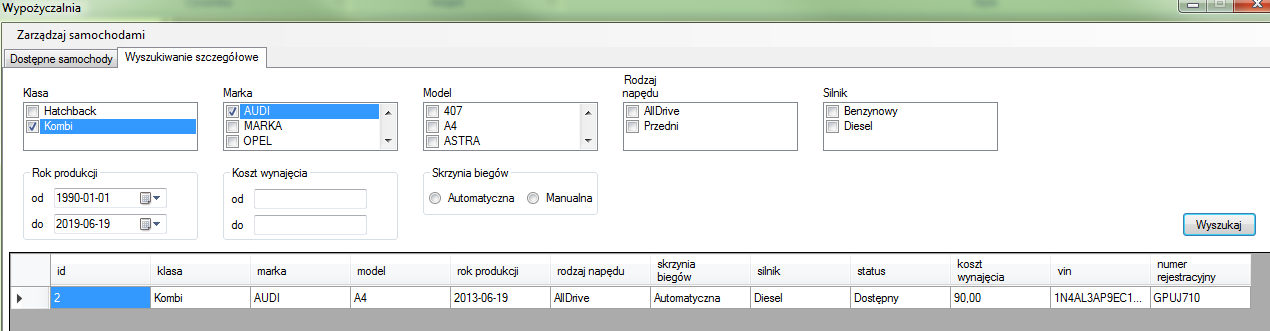
W tym widoku pracownik wypożyczalni może wypożyczyć klientowi pojazd, który jest aktualnie dostępny i nie wypożyczony:



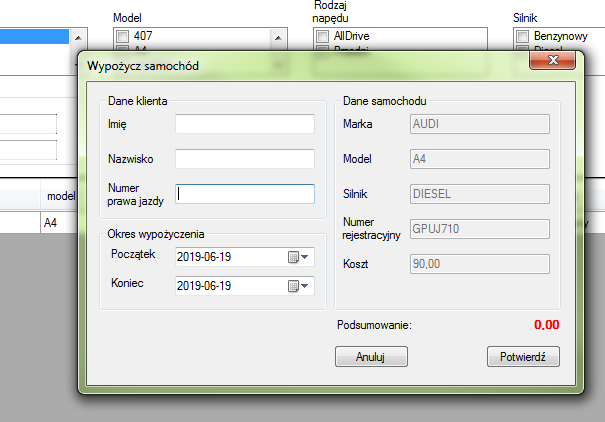
Pracownik jest domyślnie przenoszony do karty z dostępnymi pojazdami, natomiast do wyboru ma jeszcze drugi widok – „wyszukiwanie szczegółowe”:



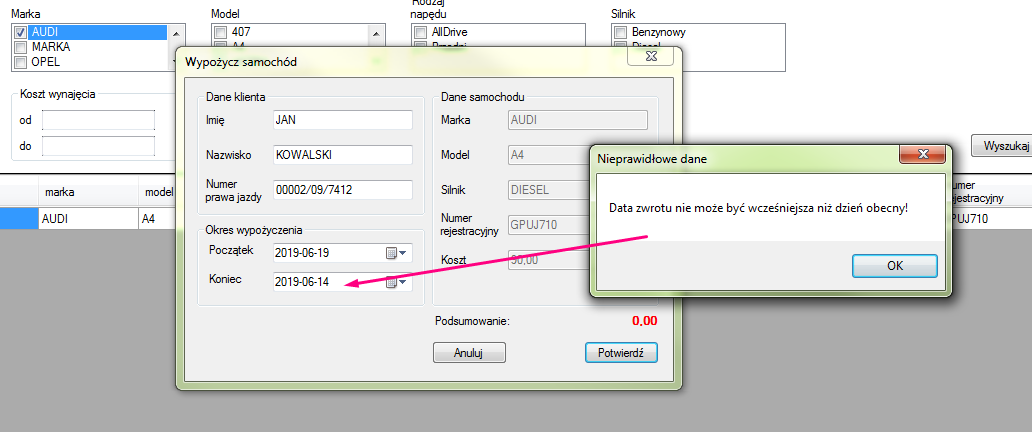
W widoku wyszukiwania szczegółowego użytkownik może określić parametry poszukiwanego pojazdu:



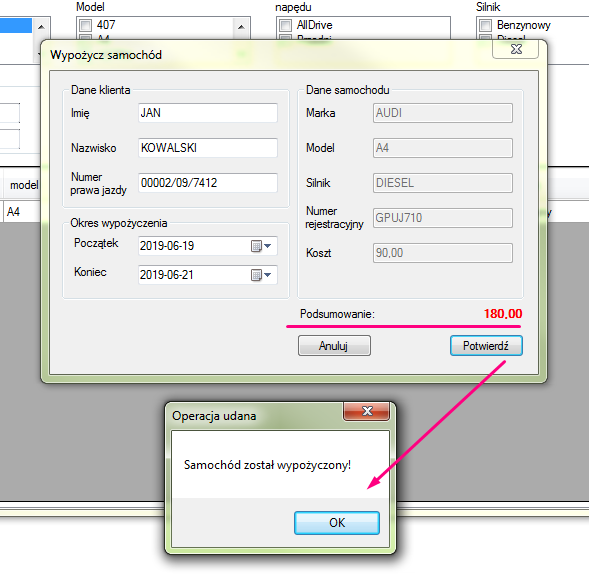
Użytkownik może dostać się do okna wypożyczania pojazdu przez dwuklik na wybranym samochodzie, zarówno z widoku „Dostępne samochody” jak i „Wyszukiwanie szczegółowe”:



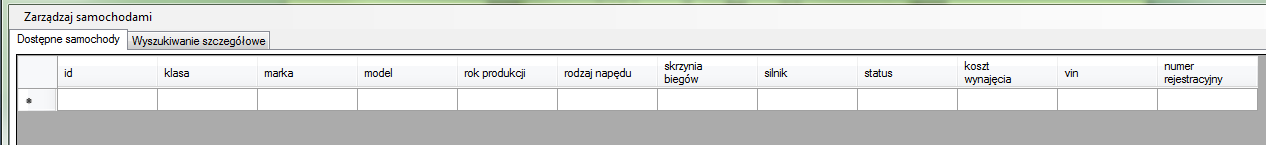
W celu wypożyczenia pojazdu należy wypełnić wszystkie dostępne pola – Imię, Nazwisko, Numer prawa jazdy – oraz wskazać datę początku i końca wypożyczenia. Aplikacja posiada zabezpieczenie przed pozostawieniem pustych pól oraz ustawieniem daty końca wypożyczenia na wcześniejszą niż data wydania pojazdu:



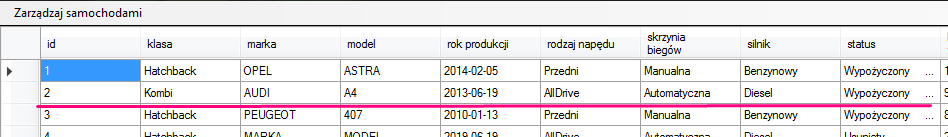
Gdy wszystkie dane zostaną wprowadzone prawidłowo zostanie naliczony koszt wypożyczenia (ilość dni pomnożona przez koszt wynajmu) i po potwierdzeniu transakcji pracownik otrzyma informację, że pojazd został wypożyczony:



Wypożyczony pojazd nie jest wyświetlany na liście dostępnych samochodów:

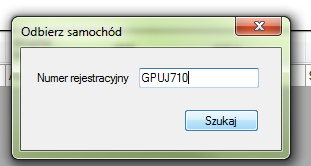


W widoku „Pokaż samochody” otrzymuje on status „wypożyczony”:

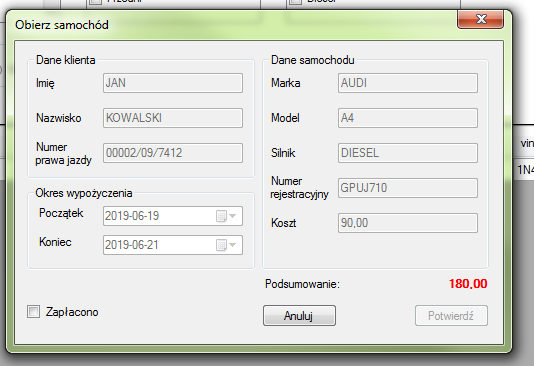


„*Odbierz samochód”*

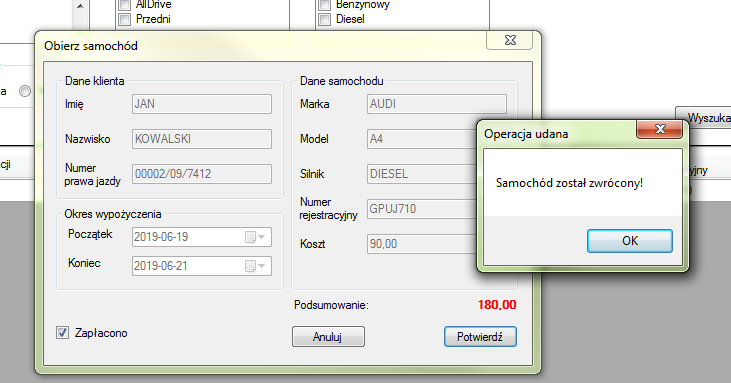
W celu odebrania wypożyczonego pojazdu od klienta pracownik musi podać numer rejestracyjny samochodu:



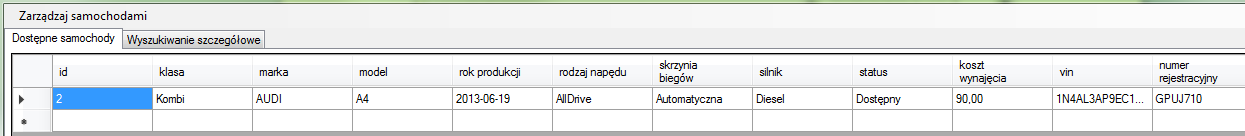
Aplikacja wyszukuje daną transakcję:



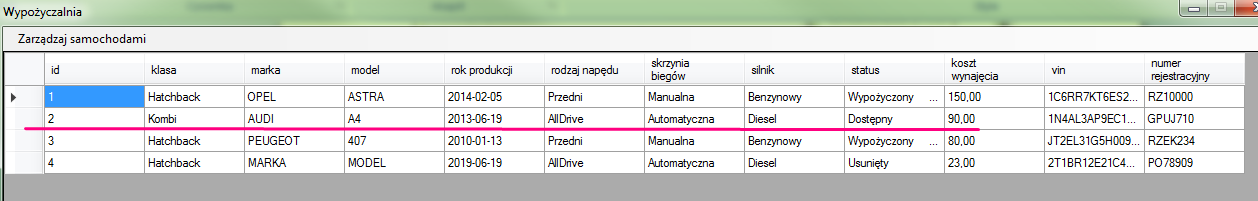
Odbiór pojazdu możliwy jest dopiero po zaznaczeniu checkbox-a „zapłacono”:



Po odebraniu pojazdu jest on ponownie dostępny do wypożyczenia…

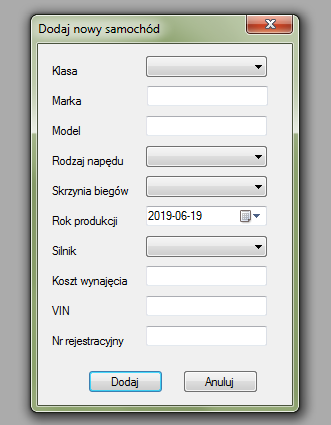


…a w widoku „Pokaż samochody” otrzymuje status „dostępny”

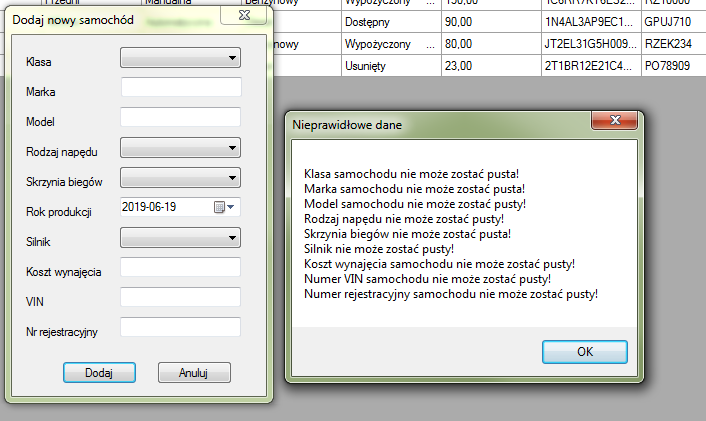


„*Dodaj samochód”*

Wybranie opcji „dodaj samochód” powoduje wyświetlenie następującego okna:

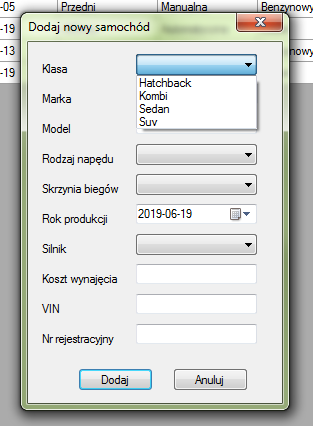
****

Aby samochód został poprawnie dodany do bazy muszą zostać wypełnione wszystkie pola – inaczej program wyrzuci błąd/błędy:

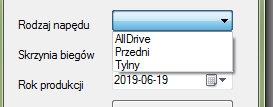


Dodając nowy samochód pracownik może wybierać spośród:

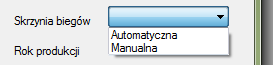
* czterech klas (lista rozwijana):



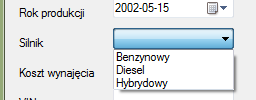
* trzech rodzajów napędów:



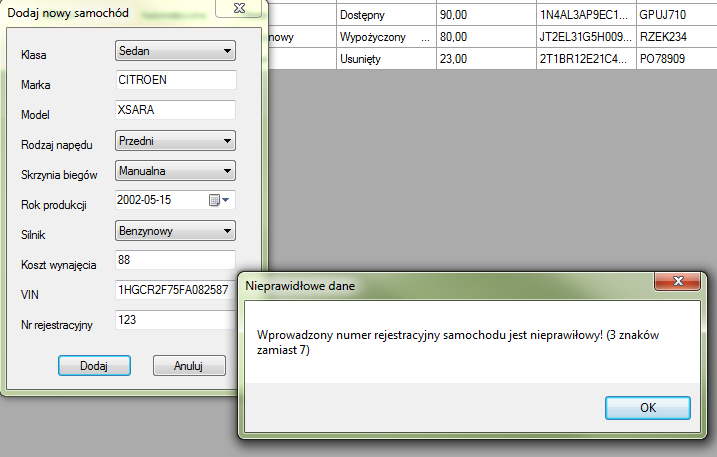
* dwóch typów skrzyni biegów:



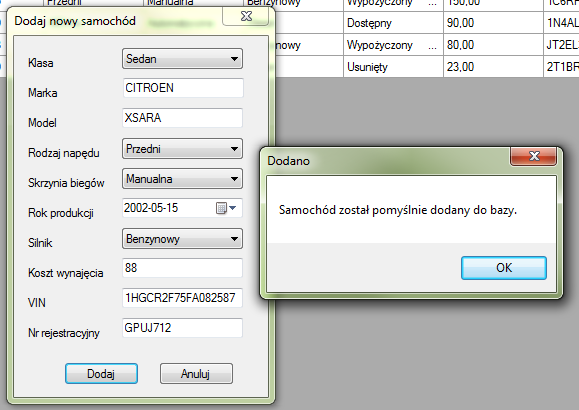
* trzech rodzajów silnika:

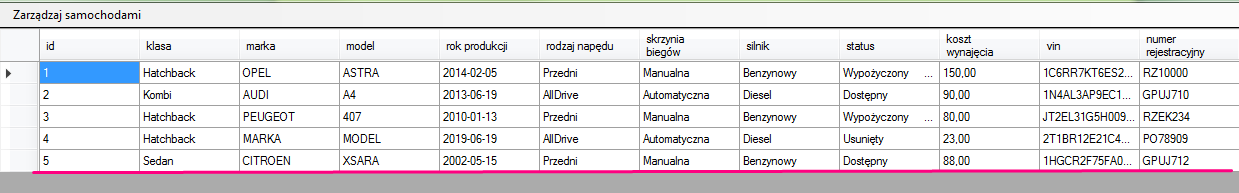


Numer VIN musi posiadać 17 cyfr (bardzo przydatna podczas testów okazała się strona <http://randomvin.com/>), a numer rejestracyjny – 7 cyfr.



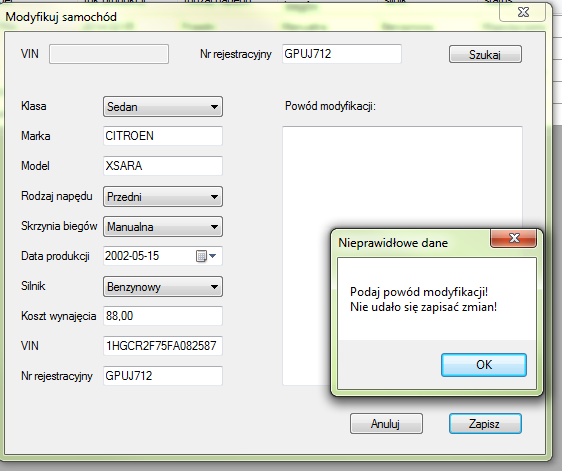
Gdy wszystkie wymagane pola zostały poprawnie wypełnione wówczas samochód pojawił się w bazie:

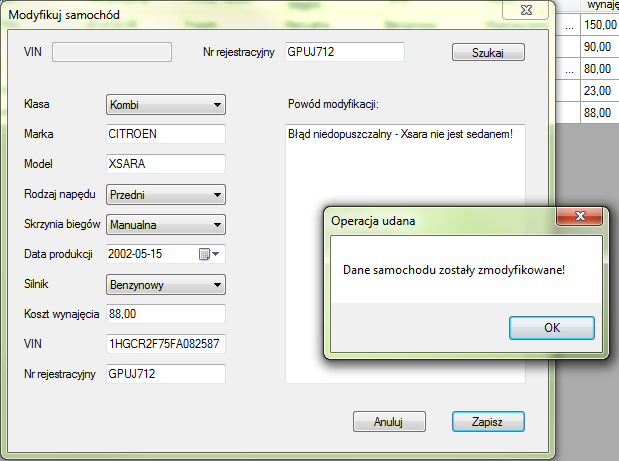




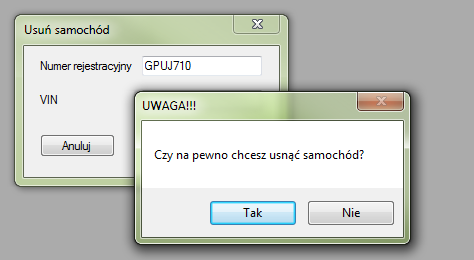
„*Edytuj samochód”*

Jeżeli pracownik błędnie dodał pojazd do katalogu bądź w magiczny sposób zmienił się na przykład rodzaj napędu (lub cokolwiek innego), wówczas możliwa jest edycja samochodu. Pojazd wyszukiwany jest po numerze VIN bądź – co łatwiejsze do zapamiętania – po numerze rejestracyjnym. Ponadto, żeby zapisać samochód po wprowadzeniu zmian, konieczne jest podanie przyczyny:

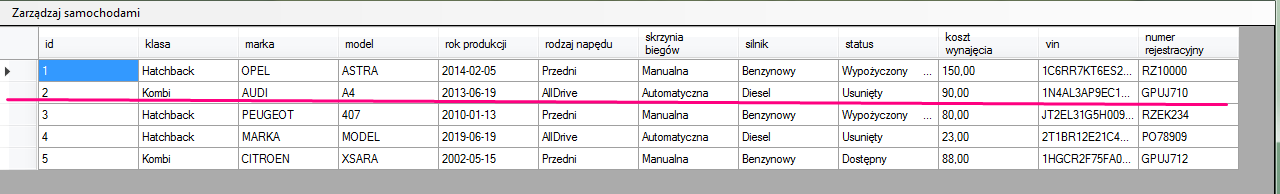


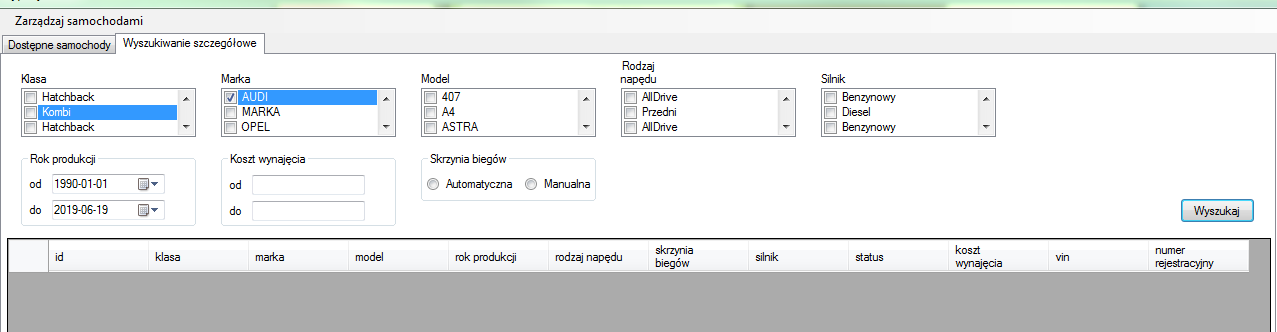


*„Usuń samochód”*

Ostatnią możliwością, jaką aplikacja oferuje pracownikowi, jest usunięcie pojazdu. Podobnie jak w przypadku edycji, samochód można wyszukać po numerze VIN lub numerze rejestracyjnym. Program profilaktycznie zapyta użytkownika, czy aby na pewno chce usunąć pojazd. 

Po podaniu odpowiedzi twierdzącej, dla auta zostanie ustawiony status „usunięty” i nie będzie ono już dostępne do wypożyczenia:

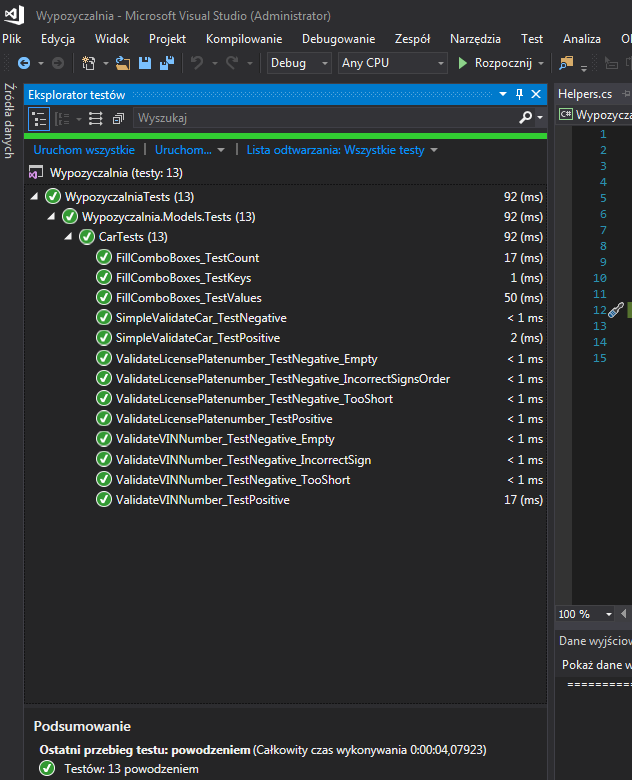
****

****

1. **Projekt powinien być realizowany z wykorzystaniem wybranego systemu kontroli wersji. Należy podać link do repozytorium**

Projekt zrealizowany został przy użyciu rozproszonego systemu kontroli wersji Git. Link do repozytorium:

<https://github.com/gravenimage91/Wypozyczalnia>

1. **Raporty z testów – powinny być przeprowadzone testy jednostkowe. Moduł testowy powinien być w repozytorium. Tutaj umieścić zrzut ekranu z pozytywnie przeprowadzonymi testami.**

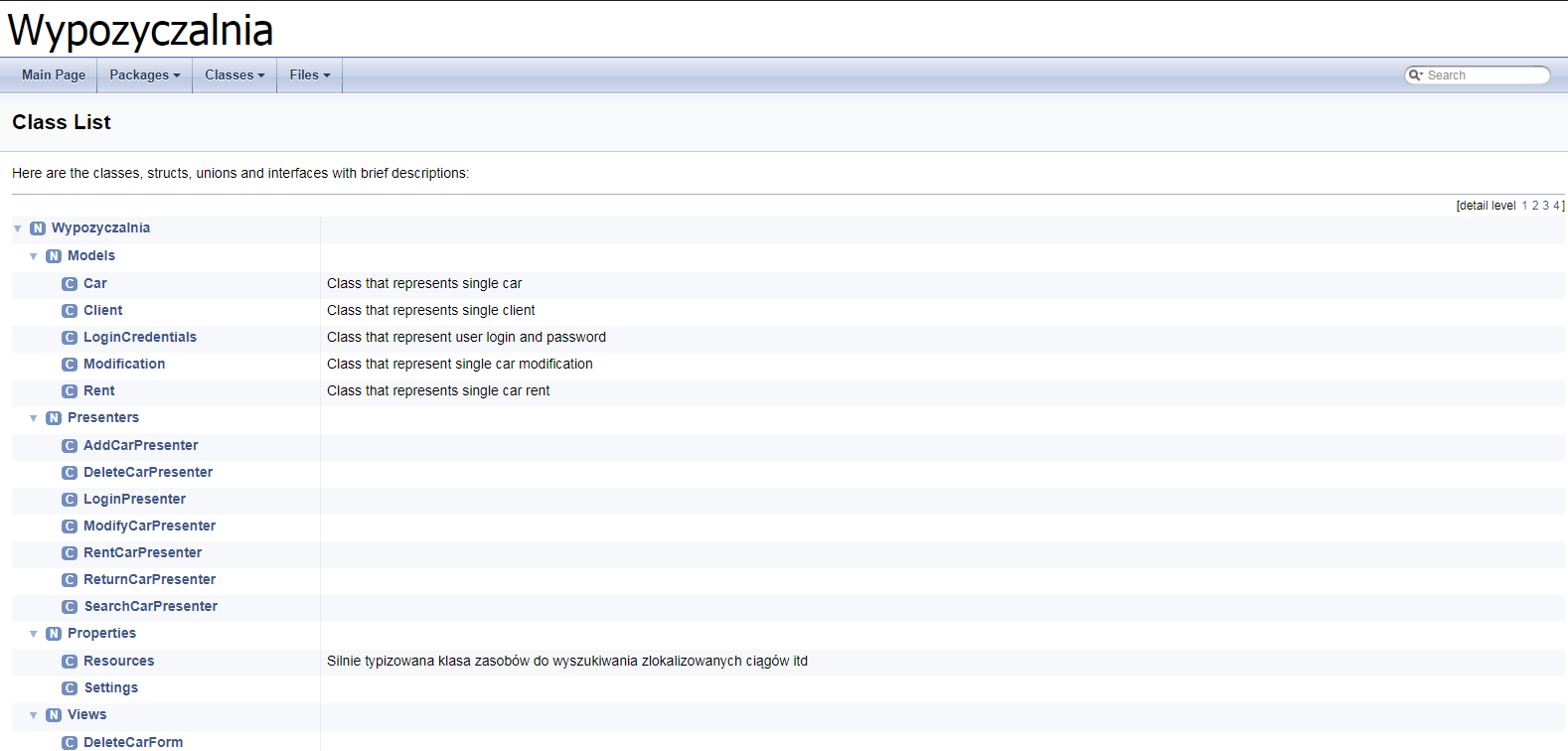
Ze względu na ograniczenia czasowe testy zostały napisane tylko dla klasy Car. Powyższy zrzut ekranu pokazuje, że testy dały wynik pozytywny.

1. **Materiały źródłowe – wskazanie literatury i materiałów źródłowych wykorzystanych przy realizacji projektu (w tym tutoriale internetowe).**
   * „Programowanie C#”, I.Griffiths, M.Adams, J.Liberty, wyd. Helion, 2012r
   * <https://www.c-sharpcorner.com/UploadFile/919746/how-to-export-sql-server-database-as-an-mdf-file/>
   * <https://docs.microsoft.com/pl-pl/dotnet/csharp/programming-guide/>
   * <https://docs.microsoft.com/pl-pl/dotnet/framework/>
   * <https://www.youtube.com/watch?v=-V_vHZPOZfY>
   * <http://randomvin.com/>
   * <https://www.codeproject.com/Articles/32769/Easy-Example-for-Model-View-Presenter-in-ASP-NET-C?fbclid=IwAR1GKlrqczMgglrAscnFulhWxkfrNnRgOVU3AGOJsM-X599n-C_9zlaeyJw>
   * <http://kurs.aspnetmvc.pl/Wzorce/MVP>
2. **Do kodu programu powinna być stworzona dokumentacja poprzez system komentarzy dokumentujących. Załącznikiem dokumentacji projektu powinna być dokumentacja kodu w postaci HTML (wygenerowana przy pomocy programu Doxygen). Wygenerowaną dokumentację należy udostępnić poprzez dysk internetowy podając link w niniejszym dokumencie.**

Link do dokumentacji:

https://drive.google.com/open?id=1n1-ug6e0\_wWApZWHxqw89d75tkKwjtOJ

Zrzut ekranu strony:

****