

**UNIWERSYTET RZESZOWSKI**

**Kolegium Nauk Przyrodniczych**

Kacper Roda

113764

Informatyka

**Warsztat samochodowy**

**Projekt zaliczeniowy z przedmiotu Programowanie Obiektowe**

Prowadzący:

mgr inż. Marcin Mrukowicz

Rzeszów, 26.01.2021

**Spis Treści:**

Strona 1 – Strona tytułowa

Strona 2 – Spis treści

Strona 3 – Wprowadzenie

Strona 4 – Panel logowania

Strona 5 – Panel resetu hasła

Strona 6 – Panel rejestracji

Strona 7 – Strona główna

Strona 8 – Usługa

Strona 9 – Naprawa

Strona 10 – Właściciel

Strona 11 – Panel dodawania

Strona 12 – Panel dodawania – Usługa

Strona 13 – Panel dodawania – Właściciel

Strona 14 – Panel dodawania – Naprawa

Strona 15 – Panel usuwania

Strona 16 – Panel usuwania – Usługa

Strona 17 – Panel usuwania – Właściciel

Strona 18 – Panel usuwania – Naprawa

Strona 19 – Panel aktualizacji

Strona 20 – Panel aktualizacji – Usługa

Strona 21 – Panel aktualizacji – Właściciel

Strona 22 – Panel aktualizacji – Naprawa

Strona 23 – Wykorzystane technologie

Strona 24 – Wykorzystane technologie cd.

Strona 25 – Baza danych

**Wprowadzenie**

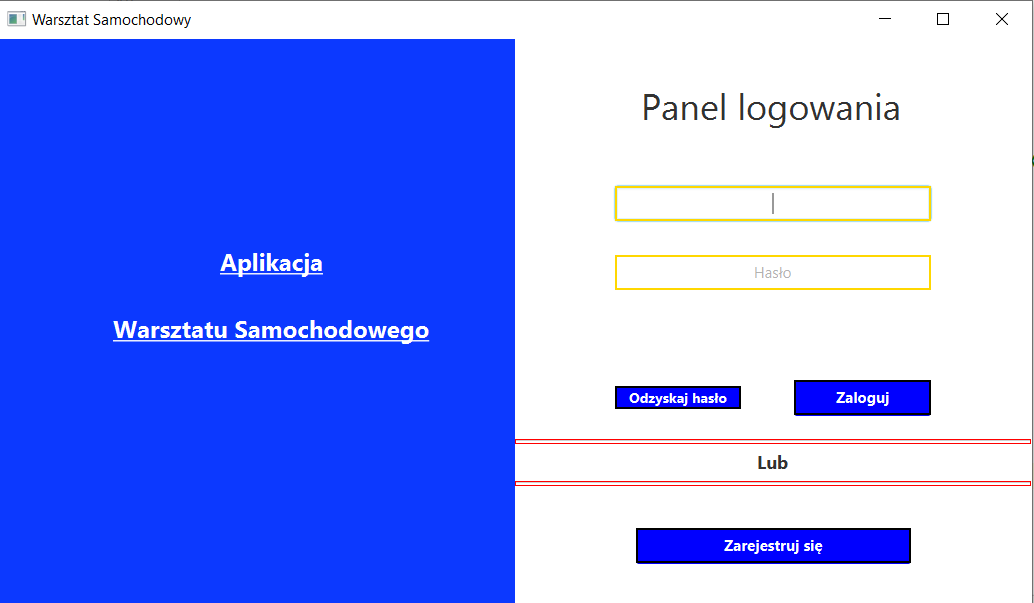
Aplikacja warsztatu samochodowego to aplikacja okienkowa, która pozwala wykonywać operacje na bazie danych. Za jej pomocą możemy dodawać, usuwać, aktualizować, a także wyszukiwać rekordy w bazie. Aplikacja składa się z czterech tabel: „Użytkownik”, „Naprawa”, „Właściciel”, „Usługa”.

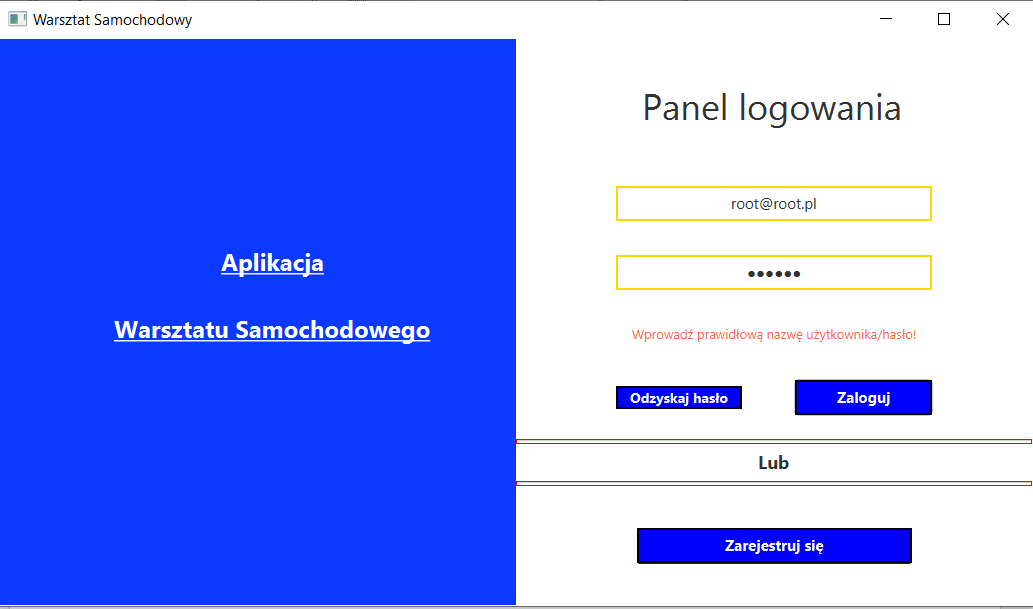
W aplikacji został użyty niekomercyjny obrazek:

https://cdn.pixabay.com/photo/2013/07/12/13/21/sports-car-146873\_960\_720.png

**Panel logowania**

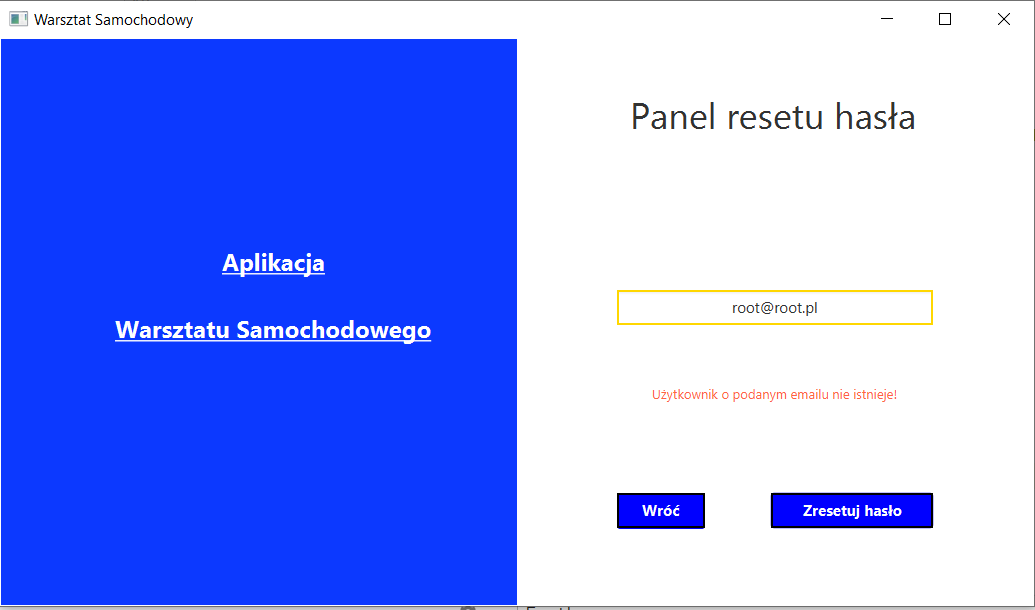
Panel logowania, to pierwsza scena, którą widzimy po uruchomieniu aplikacji. Aplikacja oczekuje od nas wprowadzenia nazwy użytkownika i hasła, aby móc wyświetlić strone główną aplikacji. Program sprawdza wpisane przez użytkownika dane, czy istnieją one w bazie i czy ogólnie są poprawne składniowo. Z panelu logowania możemy także wybrać się do panelu rejestracji lub panelu odzyskania hasła.





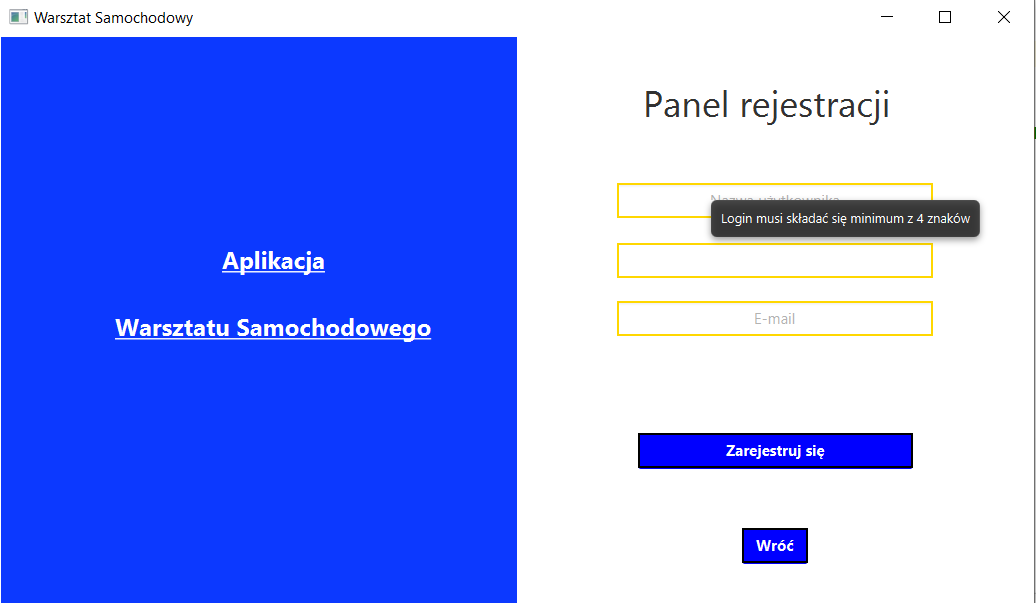
**Panel resetu hasła**

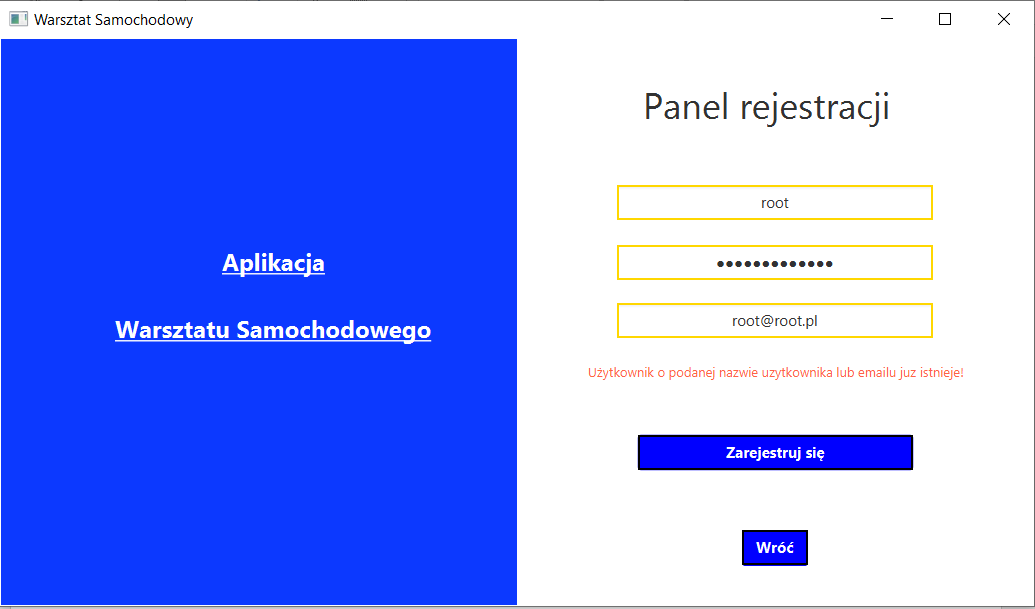
Jeśli użytkownik zdecyduje się zresetować hasło, ma obowiązek wtedy podać adres e-mail, który znajduje się w bazie i w dodatku jest poprwany składniowo w przeciwnym wypadku zostanie o tym poinformowany odpowiednim komunikatem.



**Panel rejestracji**

W tym panelu możemy dokonać rejestracji, poprzez podanie nazwy użytkownika, hasła i adres e-mail. Zaznajomienie się z komunikatami poprawnie wpisanych danych, pozwolą na poprawne zarejestrowanie się.

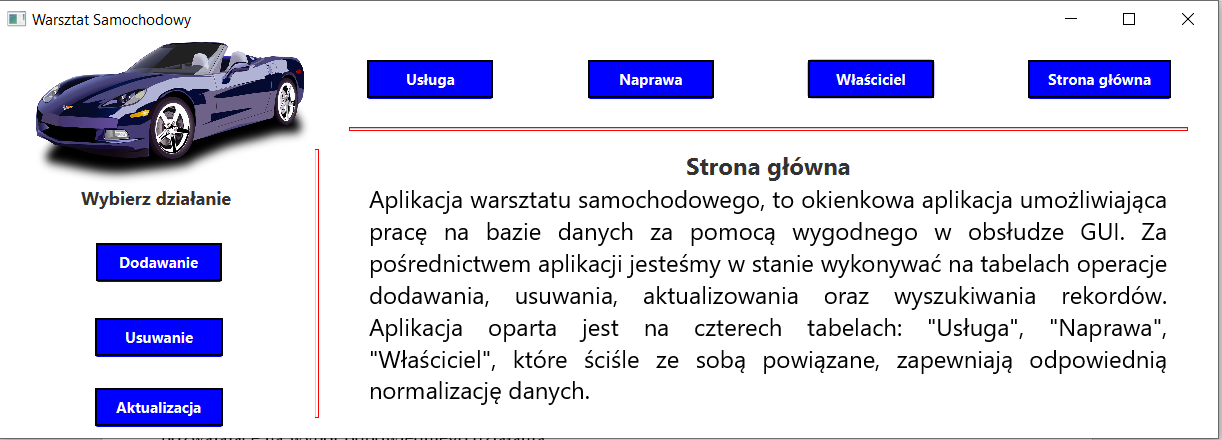
****



**Ad: Dla panelu logowania, resetu hasła oraz rejestracji aplikacja sprawdza, czy istnieje połączenie z bazą danych. Brak połączenia lub jego przerwania spowoduje wyłączenie możliwości edycji odpowiedniego panelu.**

**Strona główna**

Jeśli logowanie zakończy się sukcesem naszym oczom ukaże się strona główna. Strona główna zawiera krótkie wprowadzenie informaując użytkownika na czym polega działanie aplikacji, a także przyciski pozwalające na wybór odpowiedniego działania.



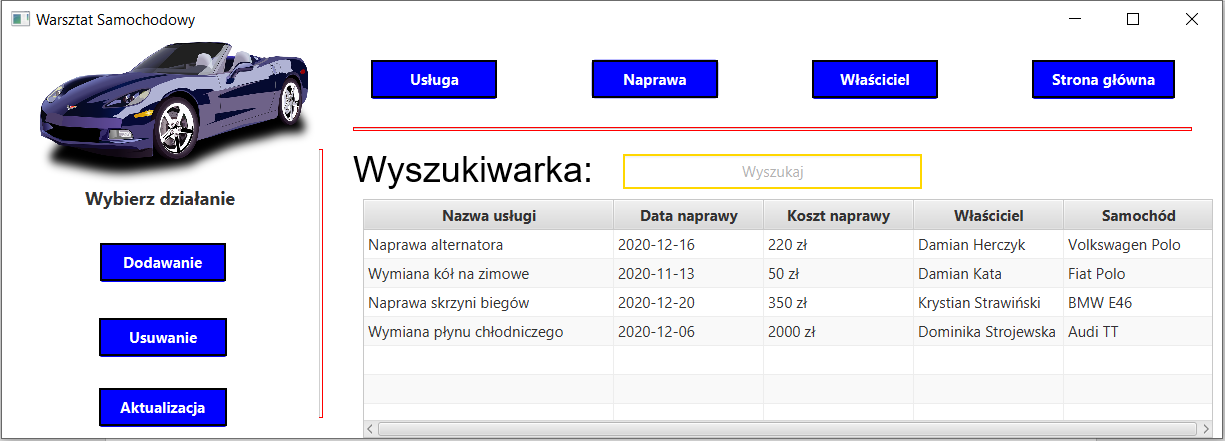
**Usługa**

Po kliknięciu przycisku „Usługa” naszym oczom ukaże się odpowiednia tabela z bazy danych.



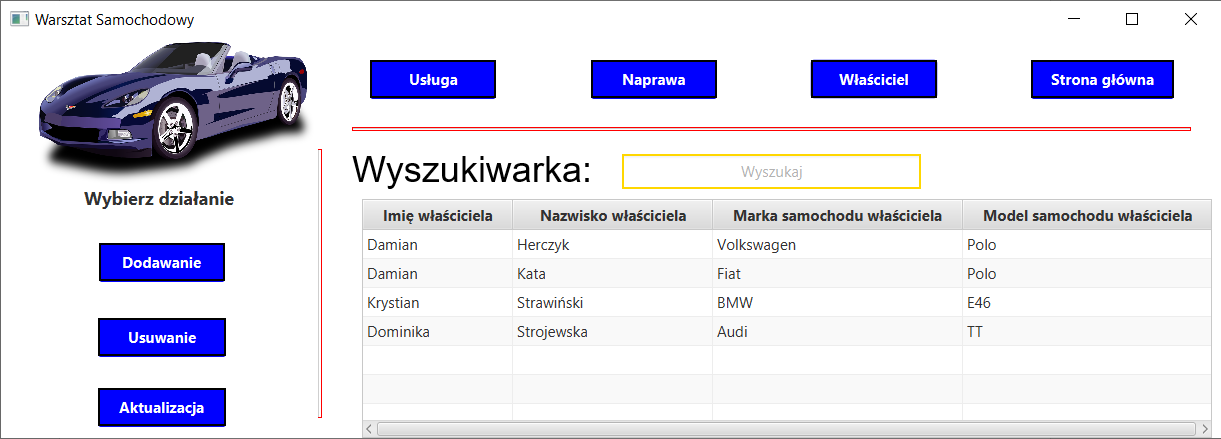
**Naprawa**

Po kliknięciu przycisku „Naprawa” naszym oczom ukaże się odpowiednia tabela z bazy danych.



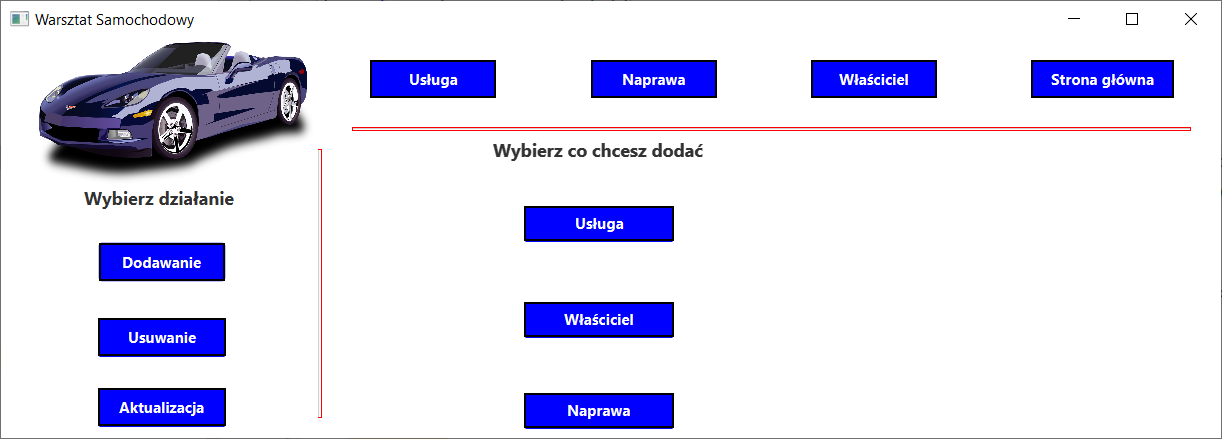
**Właściciel**

Po kliknięciu przycisku „Właściciel” naszym oczom ukaże się odpowiednia tabela z bazy danych.



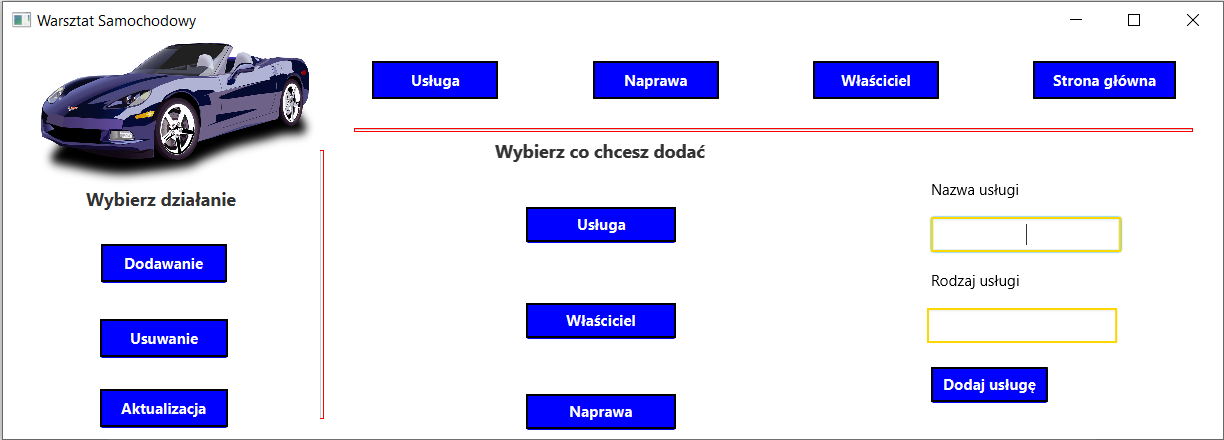
**Panel dodawania**

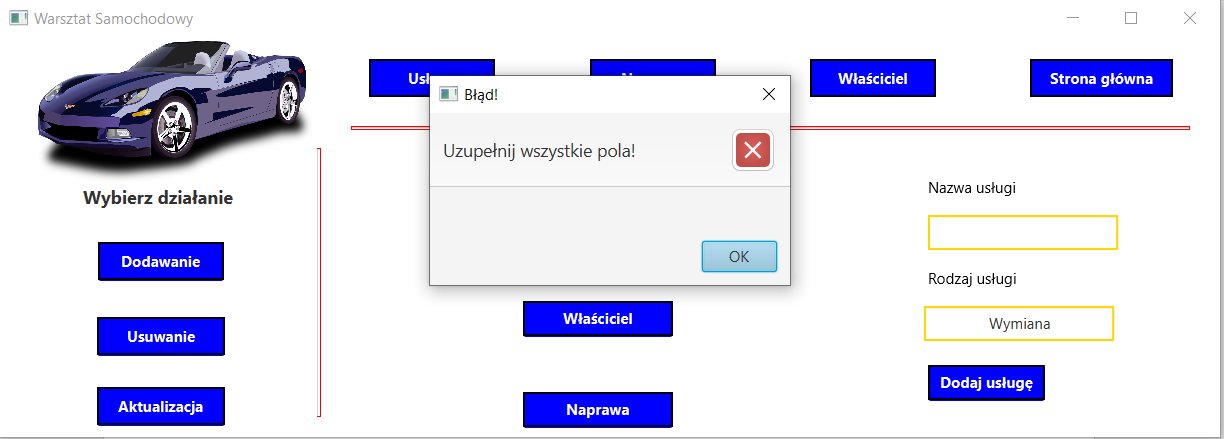
Po kliknięciu przycisku „Dodawanie” wyświetli się panel, w którym wybieramy do jakiej tabeli chcemy dodać rekord.



**Panel dodawania – Usługa**

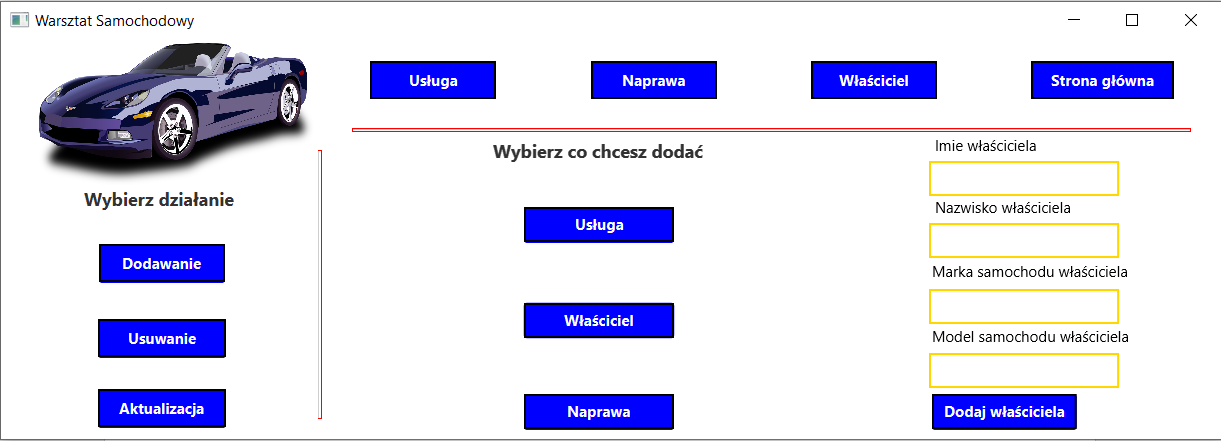
Po kliknięciu przycisku „Usługa” wyświetla się panel z okienkami w celu dodania usługi z odpowiednią walidacją danych.

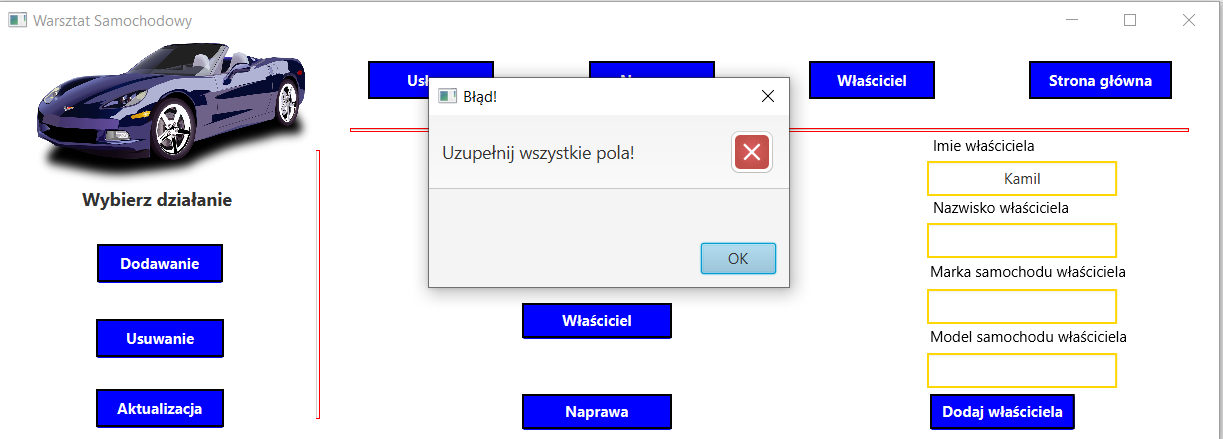




**Panel dodawania – Właściciel**

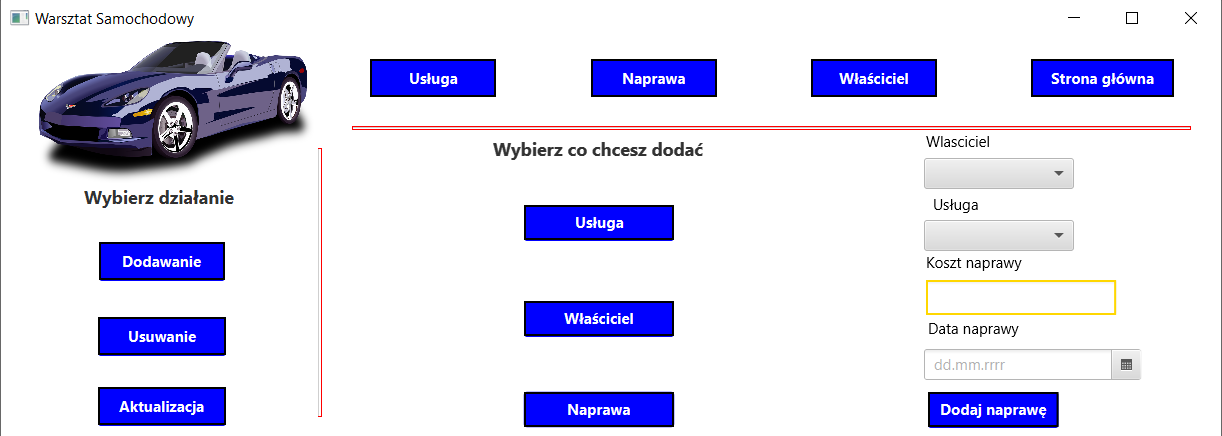
Po kliknięciu przycisku „Właściciel” wyświetla się panel z okienkami w celu dodania właściciela z odpowiednią walidacją danych.

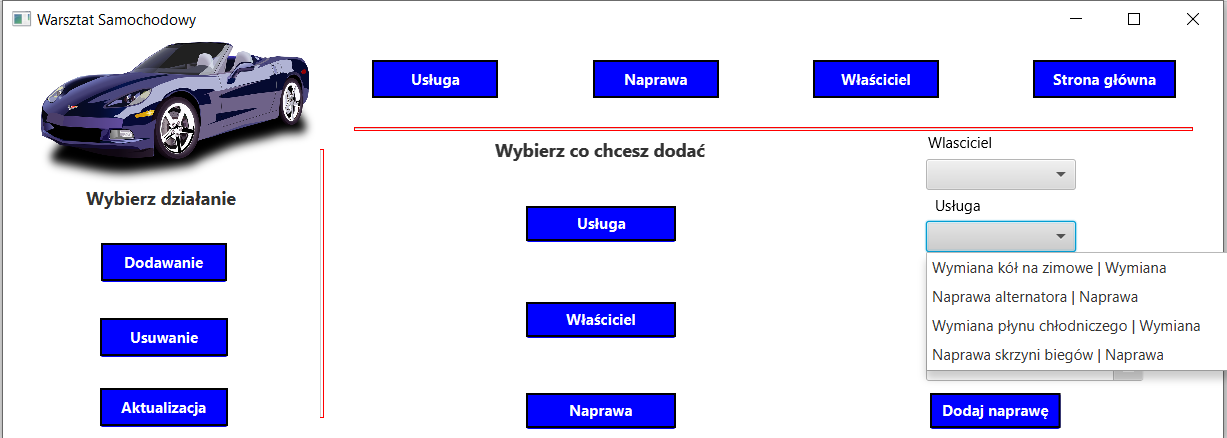


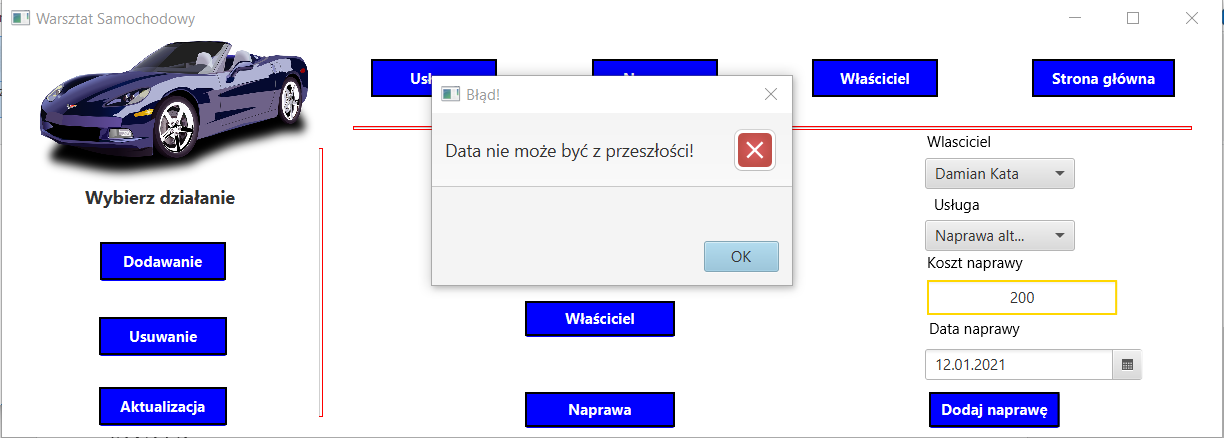


**Panel dodawania – Naprawa**

Po kliknięciu przycisku „Naprawa” wyświetla się panel z okienkami w celu dodania naprawy z odpowiednią walidacją danych.

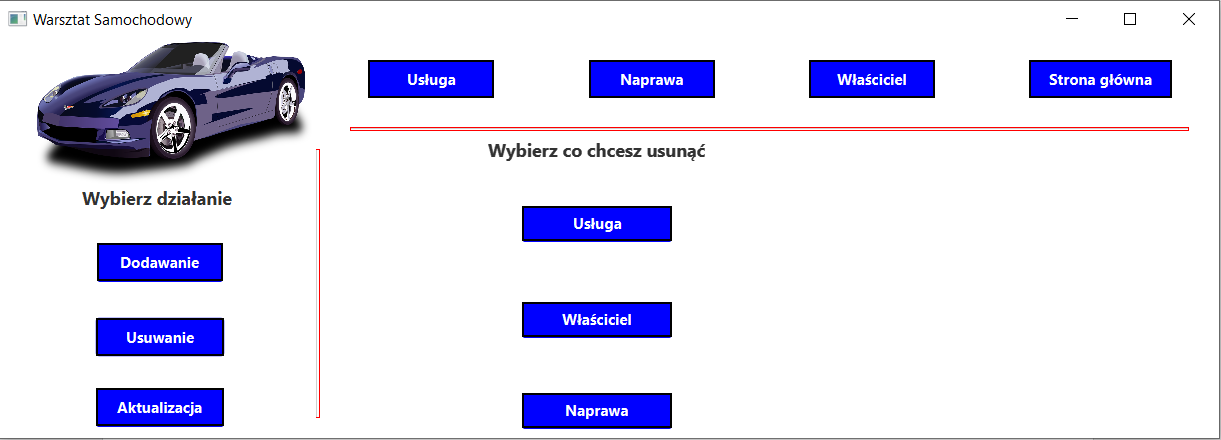


****



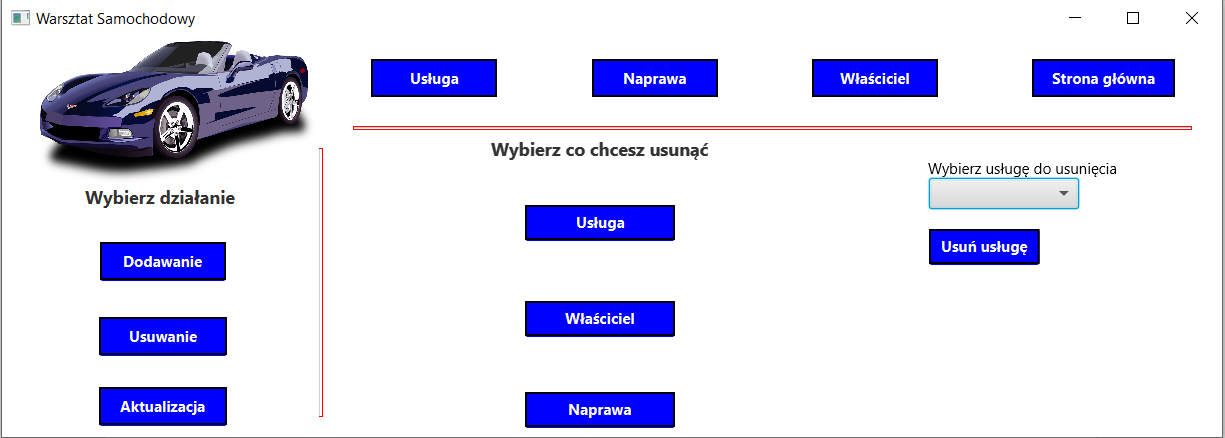
**Panel usuwania**

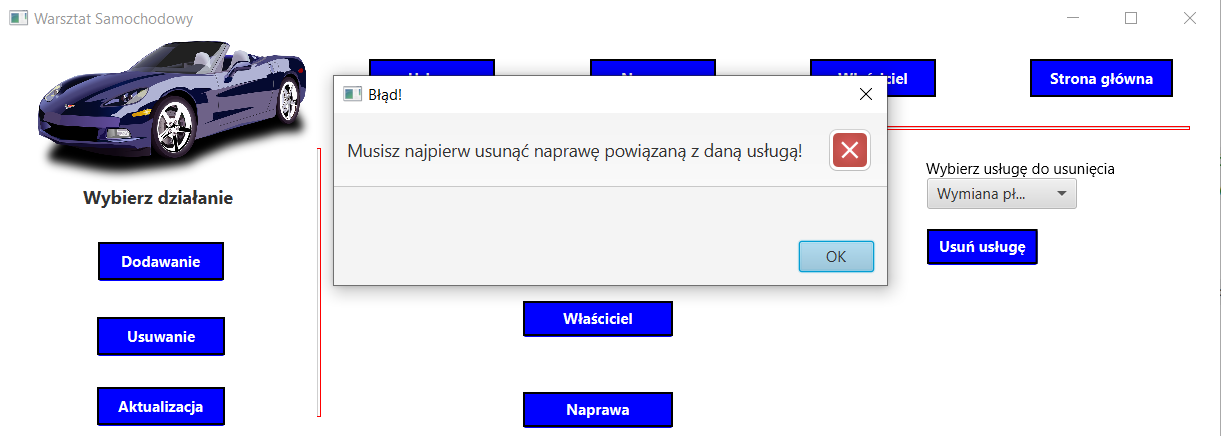
Po kliknięciu przycisku „Usuwanie” wyświetli się nam panel z wyborem tabeli w celu usunięcia konkretnego rekordu.



**Panel usuwania – Usługa**

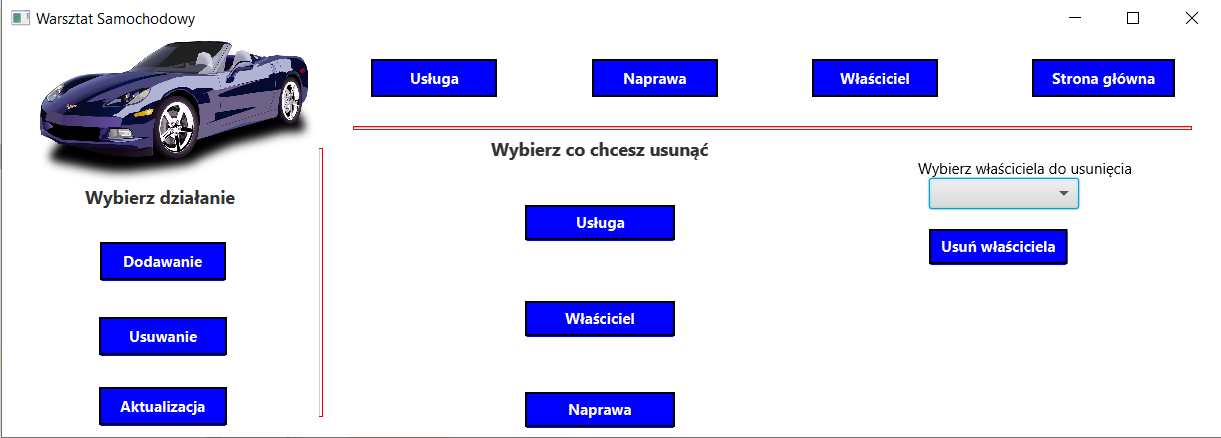
Po kliknięciu przycisku „Usługa” wyświetli nam się „ComboBox” w którym wybieramy usługę do usunięcia. Musimy jednak pamiętać, że jeśli istnieje naprawa związana z usługą, którą chcemy usunąć operacja nie powiedzie się.

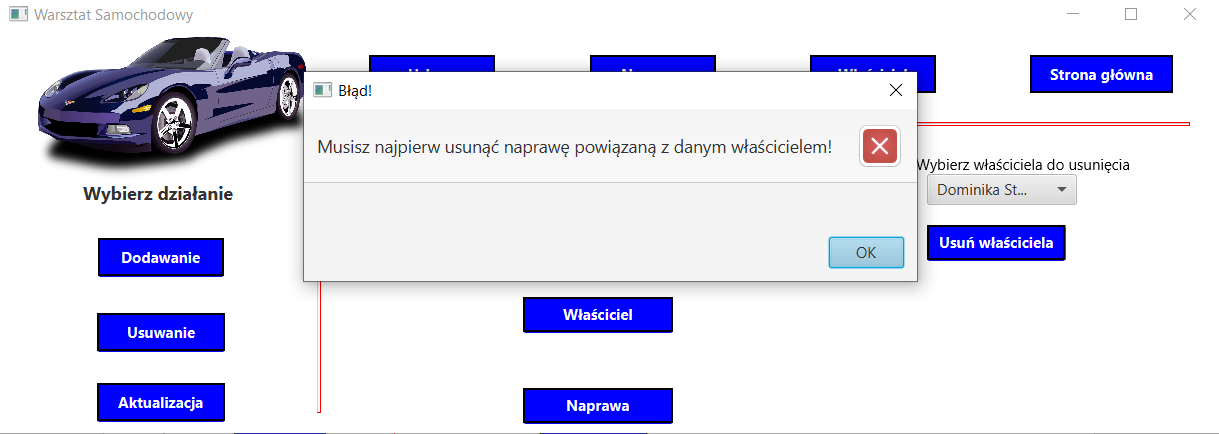




**Panel usuwania – Właściciel**

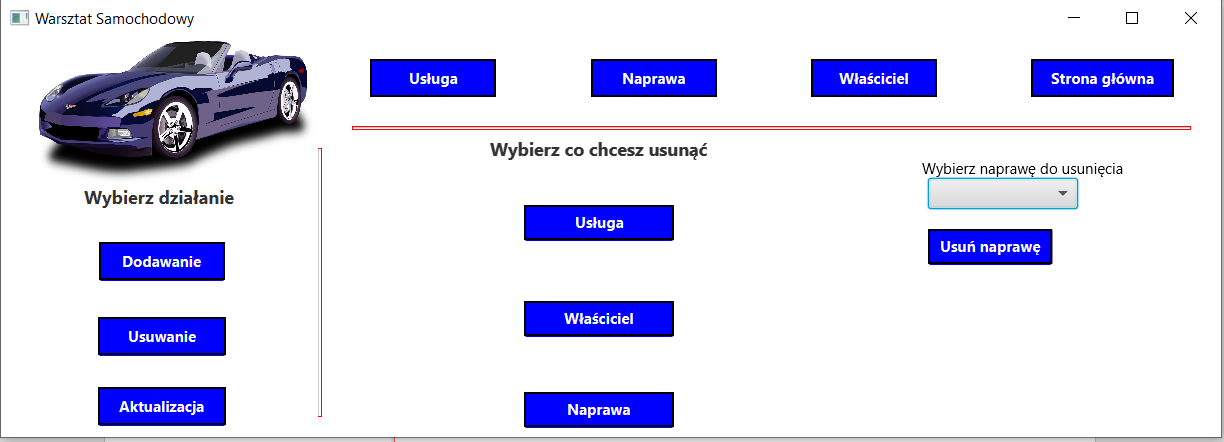
Po kliknięciu przycisku „Właściciel” wyświetli nam się „ComboBox” w którym wybieramy właściciela do usunięcia. Musimy jednak pamiętać, że jeśli istnieje naprawa związana z właścicielem, którego chcemy usunąć operacja nie powiedzie się.

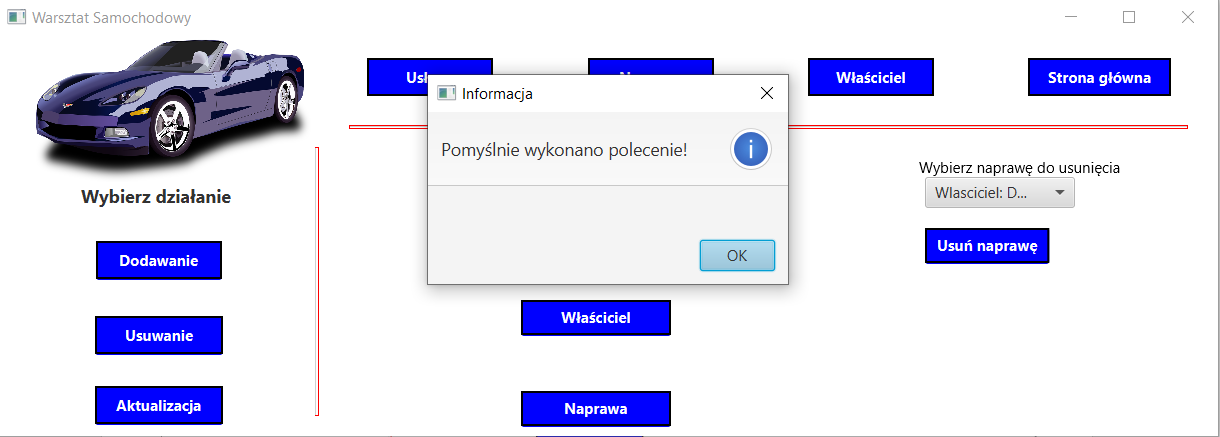




**Panel usuwania – Naprawa**

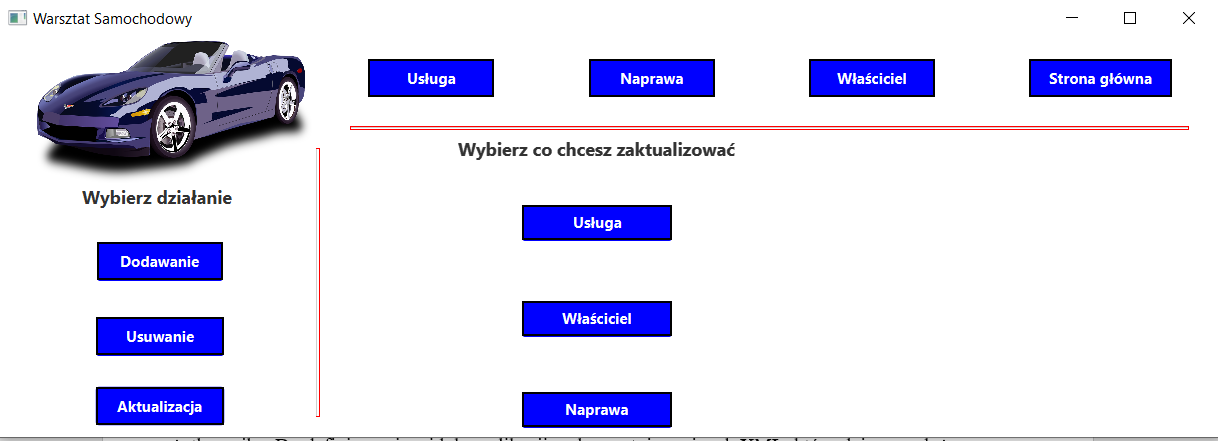
Po kliknięciu przycisku „Naprawa” wyświetli nam się „ComboBox” w którym wybieramy naprawę do usunięcia. Po poprawnym usunięciu otrzymamy odpowiedni komunikat.





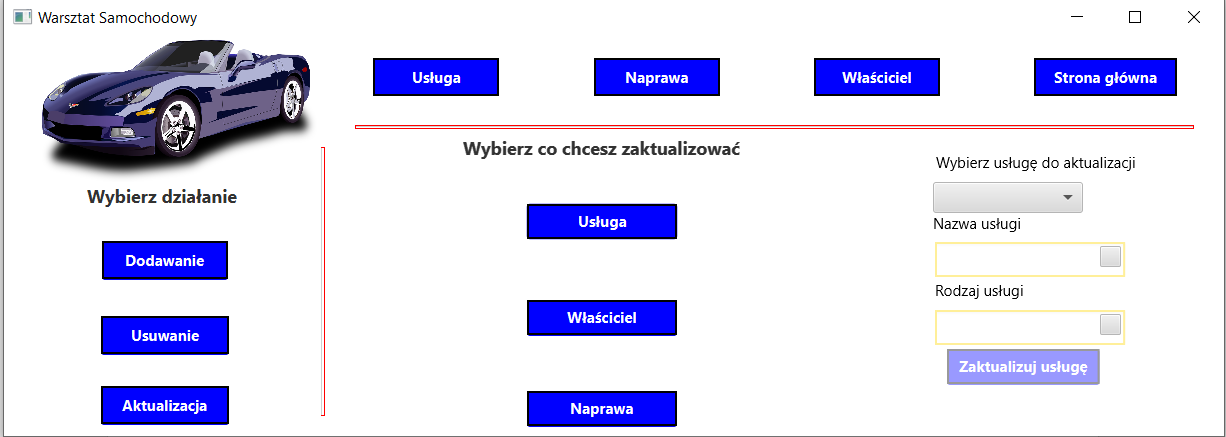
**Panel akualizacji**

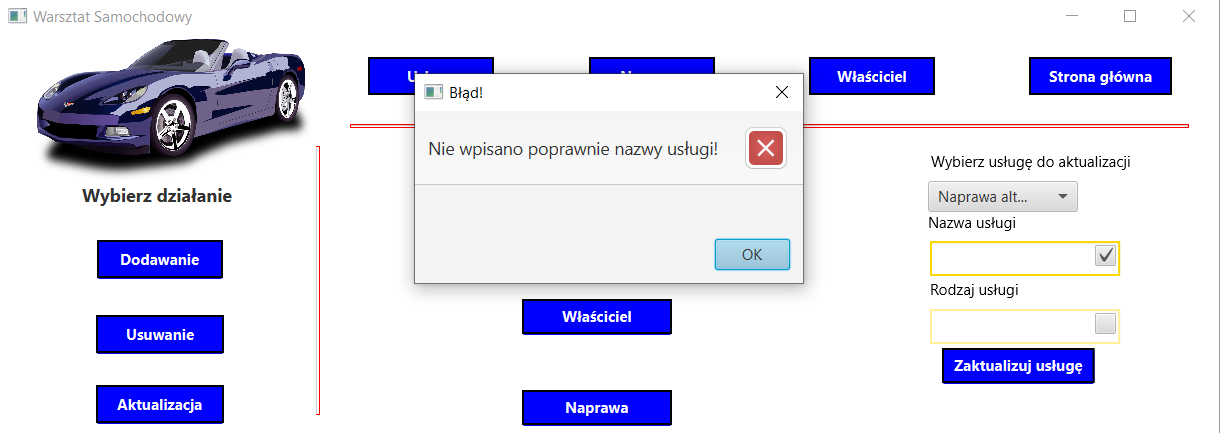
Po kliknięciu przycisku „Aktualizacja” wyświetli się panel, w którym wybieramy w jakiej tabeli chcemy zaktualizować rekord.



**Panel aktualizacji – Usługa**

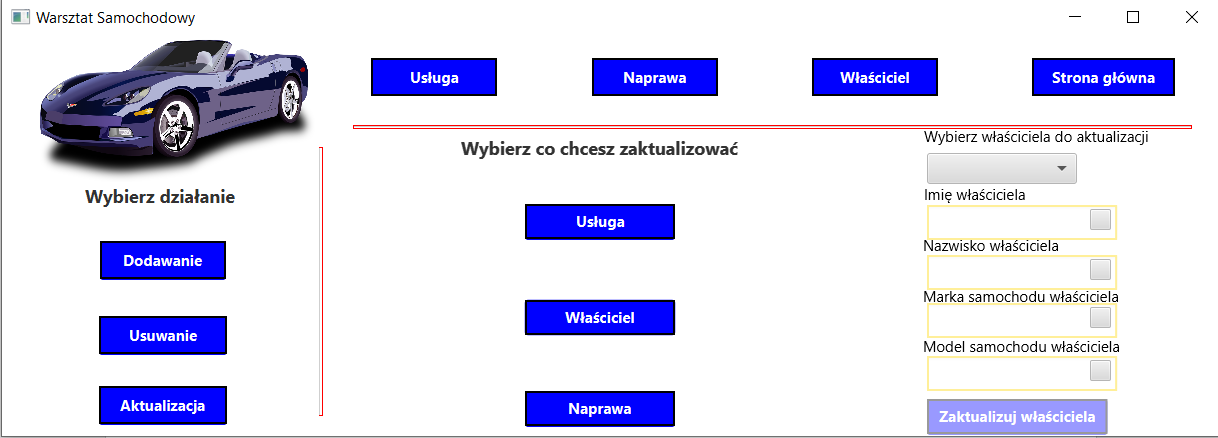
Po kliknięciu przycisku „Usługa” wyświetli się panel, w którym wybieramy z listy rozwijanej usługę, która chcemy zaktualizować. Następnie zaznaczamy „checkboxy” dla odpowiedniych okienek tekstowych. Jeśli dane wpisane przez użytkownika będą błędne, wyświetli się odpowiedni komunikat.

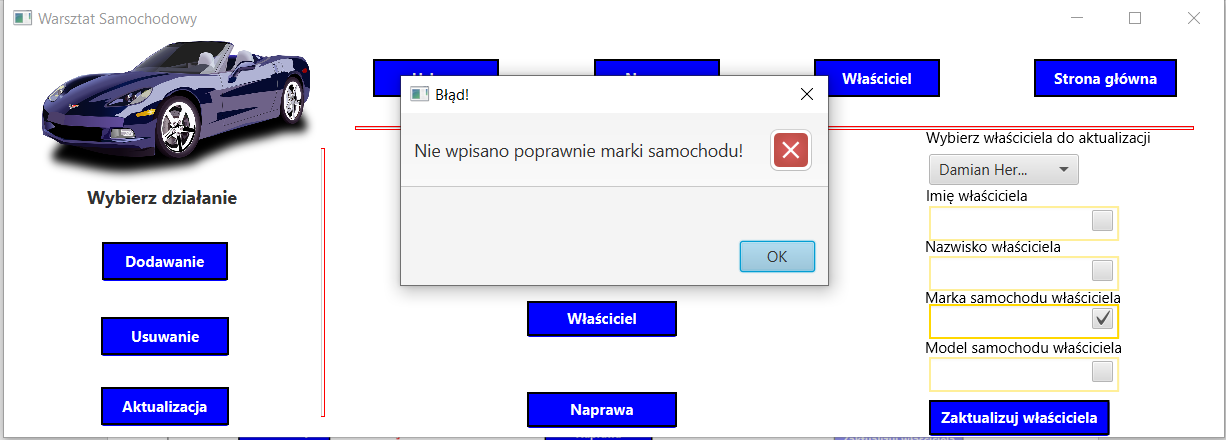




**Panel aktualizacji – Właściciel**

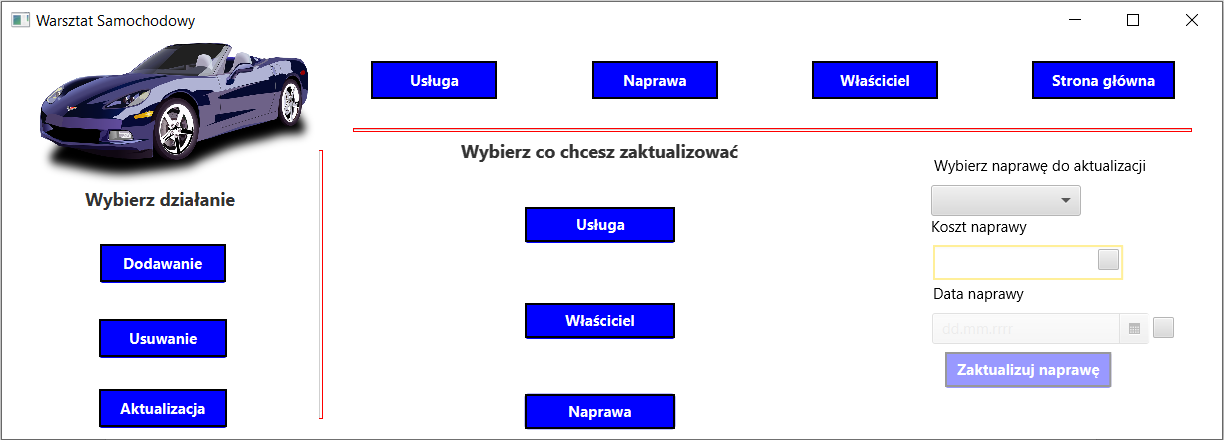
Po kliknięciu przycisku „Właściciel” wyświetli się panel, w którym wybieramy z listy rozwijanej właściciela, którego chcemy zaktualizować. Następnie zaznaczamy „checkboxy” dla odpowiedniych okienek tekstowych. Jeśli dane wpisane przez użytkownika będą błędne, wyświetli się odpowiedni komunikat.

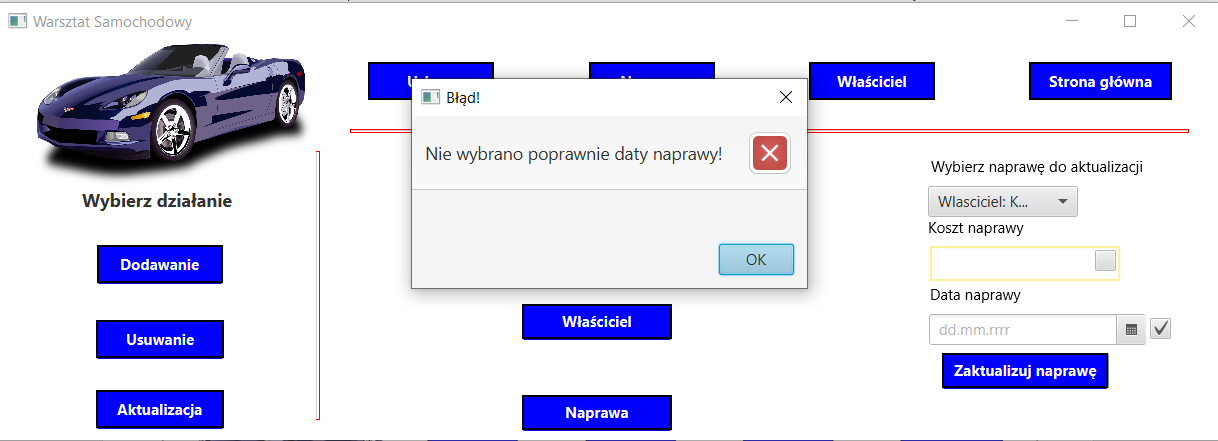




**Panel aktualizacji – Naprawa**

Po kliknięciu przycisku „Naprawa” wyświetli się panel, w którym wybieramy z listy rozwijanej naprawę, którą chcemy zaktualizować. Następnie zaznaczamy „checkboxy” dla odpowiedniych okienek tekstowych. Jeśli dane wpisane przez użytkownika będą błędne, wyświetli się odpowiedni komunikat.





**Wykorzystane technologie:**

**JAVA FX**

 JavaFX jest pakietem umożliwiającym tworzenie interfejsów graficznych w języku Java. Prace z JavaFX można porównać do teatru. W każdym teatrze jest conajmniej jedna scena. W JavaFX rolę teatru pełni **Stage** (z angielskiego można tłumaczyć jako teatr albo obszar roboczy) oraz **Scene** (z angielskeigo scena). Każda aplikacja musi zawierać conajmniej jednen obiekt **Stage** i **Scene**.

Stage można uważać za pojemnik dla Scene a Scene jako pojemnik dla elementów, które się znajdują w niej. Tworząc interfejs graficzny należy pamiętać, że do obiektu Stage dodaje się przynajmniej jeden obiekt Scene.Klasa Stage to pojemnik na najwyższym poziomie. Aplikacje mają zawsze dostęp do głównego obiektu Stage, który powstaje podczas uruchamiania aplikacji napisanej z użyciem JavaFX.

Klasa Scene to pojemnik do którego wklada się wszystkie elementy stanowiące scenę tj. kontrolki.Elementy sceny nazywa się **węzłami** (z angielskiego node). Istnieje główny węzeł zwany **korzeniem**.Każdy węzeł, poza korzeniem ma rodzica. Węzły mogą posiadać węzły potomne. Węzeł, który nie ma potomka nazywa się **liściem**.Węzłęm może być kontrolka ale również grupa węzłów. Do definiowania widoku aplikacji wykorzystujemy język XML, który daje nam dużą dowolność działania. Dodatkowo mamy możliwość modyfikacji wyglądu kontrolek przy pomocy stylów CSS (takich jak na stronach internetowych).

Podstawowym narzędziem do tworzenia aplikacji w Javie FX jest Scene Builder. Narzędzie to pozwala definiować jak nasza aplikacja będzie wyglądała. Jest niesamowicie wygodne, większość rzeczy możemy stworzyć korzystając z mechanizmu drag and drop, dodatkowo wpisując odpowiednie wartości dla pól takich jak szerokość, czy wysokość elementów. W efekcie Scene Builder wygeneruje dla nas gotowy kod XML, który będzie w bardzo czytelnej formie i bez problemu będziemy go mogli edytować z poziomu naszego IDE.

**JDBC**

JDBC to interfejs programowania stworzony w 1996 r. przez firmę Sun Microsystems, umożliwiający łączenie się z różnymi bazami danych z aplikacji napisanych w języku Java. Interfejs ten jest analogiczny do intefejsu ODBC (Open DataBase Connectivity).

JDBC to API niezależne od systemu operacyjnego oraz bazy danych. Systemy do zarządzania relacjami z kontrahentami stworzone w języku Java praktycznie zawsze korzystają ze sterowników JDBC w celu połączenia z bazą danych.

**MySQL**

MySQL jest szybkim i solidnym systemem zarządzania relacyjnymi bazami danych (Relational Database Management System - RDBMS). Bazy danych pozwalają na efektywne przechowywanie, wyszukiwanie, sortowanie oraz pozyskiwanie informacji.

Serwer MySQL steruje dostępem do danych, udostępnia je równocześnie dla wielu użytkowników oraz zapewnia możliwość jak najszybszego z nich korzystania. Ponadto serwer steruje uwierzytelnianiem użytkowników, które pozwala operować na danych jedynie osobom upoważnionym. Jest to system wieloużytkownikowy i wielowątkowy korzystający ze standardu - języka SQL, dzięki któremu możliwe jest zadawanie zapytań do bazy danych.

MySQL został oficjalnie opublikowany w roku 1996. Jest również dostępny na licencji Open Source, lecz jeśli jest to konieczne, można zakupić jego wersję komercyjną.

**Baza danych:**

Stworzona baza danych składa się z 4 tabel. Tabele są znormalizowane do takiego stopnia aby nie występowała redundancja danych.

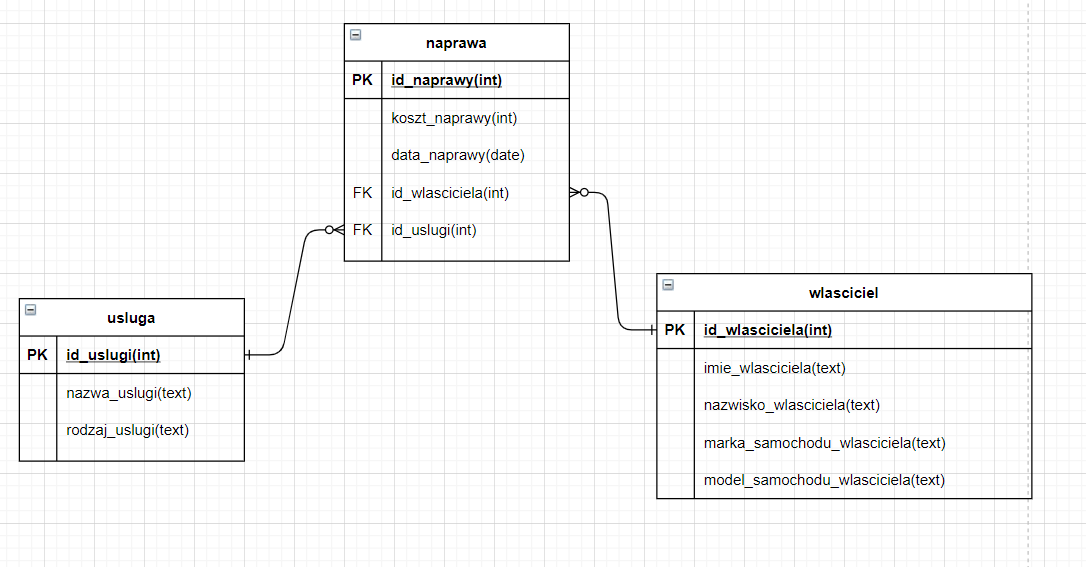


Diagram związków encji - ERD