Opis wersji aplikacji

Projekt: Aplikacja do wypożyczania samochodów (C#/XAML)

Data: 2025-06-15

Spis treści

[1. Wstęp 2](#_Toc201857062)

[2. Zakres dokumentu 2](#_Toc201857063)

[3. Informacje o wersji/edycji aplikacji 3](#_Toc201857064)

[4. Charakterystyka komponentów/pakietów 3](#_Toc201857065)

[5. Lista pakietów aplikacji 3](#_Toc201857066)

[6. Diagram instalacji UML aplikacji 3](#_Toc201857067)

[7. Zmiany od poprzedniej wersji 3](#_Toc201857068)

[8. Konfiguracja środowiska przed instalacją 3](#_Toc201857069)

[9. Instrukcja instalacji 3](#_Toc201857070)

[10. Problemy i nieusunięte błędy 4](#_Toc201857071)

[11. Pakiety 4](#_Toc201857072)

[11.1 Pakiet: WypożyczalniaSamochodow 4](#_Toc201857073)

[11.2 Lista plików: 4](#_Toc201857074)

[11.3 Instrukcja kompilacji: 4](#_Toc201857075)

[12. Specyfikacja unit testów 4](#_Toc201857076)

[12.1 Lista wykonanych test case’ów: 4](#_Toc201857077)

[12.2 Raport o błędach: 4](#_Toc201857078)

# 1. Wstęp

Celem dokumentu jest przedstawienie opisu wersji aplikacji desktopowej do wypożyczania samochodów, zawierającego informacje o funkcjach, zmianach oraz innych istotnych aspektach.

# 2. Zakres dokumentu

Opis funkcji aplikacji:  
- Przeglądanie listy dostępnych samochodów (marka, model, VIN).  
- Dodawanie nowych samochodów do bazy.  
- Edycja i usuwanie danych samochodów.  
- Tworzenie, edycja i usuwanie rezerwacji (imię i nazwisko klienta, data wypożyczenia/zwrotu).  
- Wyświetlanie listy wszystkich rezerwacji.  
- Brak systemu logowania i rejestracji.  
  
Zależności technologiczne:  
- Wykorzystanie C#, WPF/XAML.  
- Komunikacja z API (lub lokalną bazą) przez klasy Database.cs, Car.cs, Rental.cs.  
  
Jakość kodu:  
- Przejrzysta struktura katalogów: Models, Commands, Views.  
- Zastosowanie wzorca MVVM.

# 3. Informacje o wersji/edycji aplikacji

- Numer wersji: 1.0  
- Data wydania: czerwiec 2025

# 4. Charakterystyka komponentów/pakietów

- Moduł obsługi samochodów: Dodawanie, edytowanie, usuwanie samochodów, prezentacja szczegółów pojazdów.  
- Moduł rezerwacji: Tworzenie nowej rezerwacji na podstawie wybranego auta i danych klienta, modyfikacja oraz usuwanie rezerwacji.  
- Warstwa danych: Klasy Car, Rental, obsługa zapisu/odczytu przez klasę Database.

# 5. Lista pakietów aplikacji

- Models – klasy Car, Rental  
- Commands – obsługa komunikacji z bazą/API  
- Views – okna aplikacji (np. główne, dodawania samochodu, lista rezerwacji)  
- System/Configuration – ustawienia aplikacji

# 6. Diagram instalacji UML aplikacji

A black background with a black square

Description automatically generated with medium confidence

# 7. Zmiany od poprzedniej wersji

Pierwsza publiczna wersja 1.0.

# 8. Konfiguracja środowiska przed instalacją

- System operacyjny: Windows z zainstalowanym .NET Framework.  
- Dostęp do bazy danych lub skonfigurowanego API (np. lokalnie pod adresem https://localhost:7017).

# 9. Instrukcja instalacji

1. Upewnij się, że masz Windows i .NET Framework.  
2. Rozpakuj aplikację do wybranego folderu.  
3. Upewnij się, że baza danych lub API jest uruchomione.  
4. Uruchom plik WypozyczalniaSamochodow.exe (lub z Visual Studio: solucja .sln).

# 10. Problemy i nieusunięte błędy

Brak znanych błędów krytycznych.

# 11. Pakiety

## 11.1 Pakiet: WypożyczalniaSamochodow

## 11.2 Lista plików:

- Car.cs  
- Rental.cs  
- Database.cs  
- App.xaml, MainWindow.xaml  
- ViewModels, Views  
- App.config, packages.config  
- Pliki projektu i zależności (.sln, obj, bin)

## 11.3 Instrukcja kompilacji:

Otwórz solucję w Visual Studio (.sln).  
Skonfiguruj połączenie z bazą/API w pliku konfiguracyjnym.  
Zbuduj i uruchom projekt.

# 12. Specyfikacja unit testów

## 12.1 Lista wykonanych test case’ów:

- Dodawanie samochodu  
- Edycja samochodu  
- Usuwanie samochodu  
- Tworzenie rezerwacji  
- Edycja daty zwrotu rezerwacji  
- Usuwanie rezerwacji

## 12.2 Raport o błędach:

Brak krytycznych błędów zgłoszonych w obecnej wersji.