Projekt aplikacji – MovieRental

**1. Wstęp**

Niniejszy dokument opisuje projekt aplikacji MovieRental – systemu do zarządzania wypożyczalnią filmów.

**2. Cel i zakres dokumentu**

Celem dokumentu jest przedstawienie architektury, funkcjonalności oraz diagramów ilustrujących działanie systemu.

**3. Definicja architektury aplikacji**

System oparty jest o architekturę klient-serwer z wykorzystaniem ASP.NET Core WebAPI, WPF (MVVM) oraz bazy danych SQL Server.

**4. Cele i ograniczenia architektury**

System powinien umożliwiać dodawanie, edytowanie, usuwanie oraz raportowanie danych wypożyczeń. Ograniczenia obejmują brak obsługi użytkowników oraz ograniczoną skalowalność.

**5. Obraz logiczny aplikacji**

**5.1 Charakterystyka pakietów**

Pakiety logiczne systemu obejmują: Model, ViewModel, View (WPF), Kontrolery (WebAPI), Repositories, Baza danych.

**5.2 Diagram klas aplikacji**

**5.3 Specyfikacja funkcji i metod aplikacji**

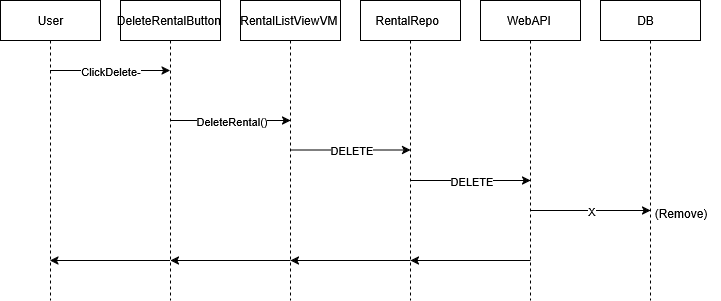
Każdy ViewModel implementuje logikę funkcjonalną. Repozytorium odpowiada za komunikację z WebAPI. Kontrolery WebAPI realizują operacje CRUD.

**6. Dynamiczny obraz modelowanej aplikacji**

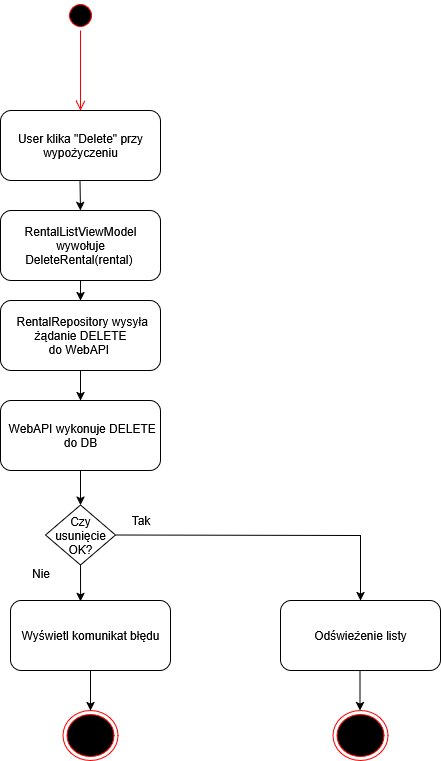
**6.1 Diagram sekwencji UML dla obiektów:**

**6.1.1 Dodawanie wypożyczenia do Bazy danych**

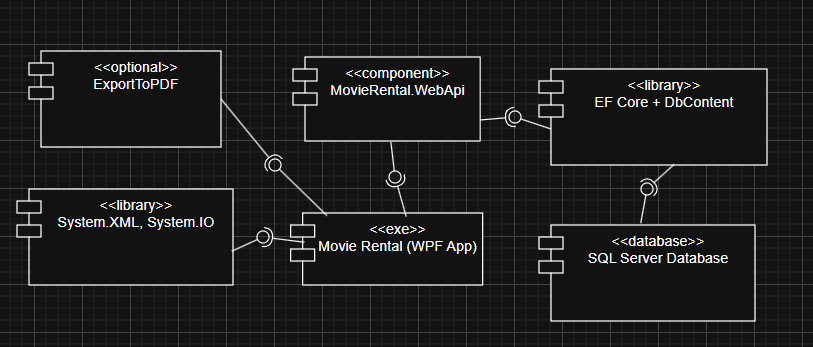
**6.1.2 Usuwanie wypożyczenia z Bazy danych**

****

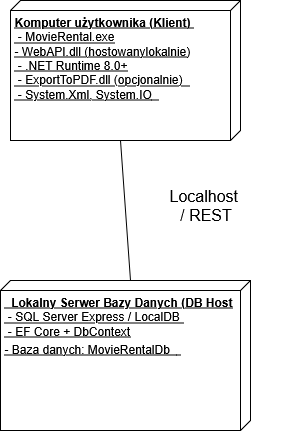
**6.2 Diagram aktywności UML dla obiektów  
6.2.1 Diagram aktywności UML dla usuwania z bazy danych**



**7. Statyczny obraz modelowanej aplikacji**

**7.1 Diagram komponentów UML  
**

**7.2 Diagram instalacji UML**



**8. Projekt bazy danych**

**Specyfikacja XML:**

<ArrayOfRental>

<Rental>

<Id>1</Id>

<CustomerName>Jan</CustomerName>

<CustomerSurname>Kowalski</CustomerSurname>

<MovieTitle>Matrix</MovieTitle>

<RentalDate>2025-06-23T14:00:00</RentalDate>

<ReturnDate>2025-06-25T18:30:00</ReturnDate>

</Rental>

</ArrayOfRental>