

Pełna Dokumentacja Projektu: System Zarządzania Kinem

Wprowadzenie

System zarządzania biletami do kina, stworzony z wykorzystaniem ASP.NET Core, Entity Framework Core oraz Identity Framework, zintegrowany z bazą danych PostgreSQL. Projekt umożliwia zarządzanie filmami, seansami, salami kinowymi i biletami, oferując jednocześnie funkcje uwierzytelniania i autoryzacji użytkowników.

Struktura Projektu

- Modele: Movie, Order, OrderDetails, Room, Seance, Ticket.
- Kontrolery: Obejmują MovieController, OrderController, RoomController, SeanceController, TicketController. Każdy kontroler zarządza operacjami CRUD dla swojego modelu.
- Widoki: Interfejs użytkownika.
- Baza Danych: PostgreSQL do przechowywania i zarządzania danymi.

Funkcjonalności

- CRUD dla Filmów, Seansów, Sal, Zamówień, Biletów: Pełna funkcjonalność tworzenia, odczytu, aktualizacji i usuwania danych.
- Uwierzytelnianie i Autoryzacja: Zarządzanie dostępem do różnych części systemu.
- Devcontainer: Ułatwiona konfiguracja środowiska deweloperskiego dzięki wykorzystaniu devcontainer.

Technologie

- ASP.NET Core: Aplikacje webowe.
- Entity Framework Core: MVC.
- PostgreSQL: Baza danych.
- Identity Framework: Bezpieczeństwo i zarządzanie użytkownikami.

Uruchomienie Projektu

1. Wymagania: Docker (w przypadku Windows w trybie linux containers)
2. Wejście w folder z projektem i uruchomienie projektu przy użyciu docker compose up
3. Domyślny port :5010 (url do projektu <http://localhost:5010>)
4. Default user superadmin@localhost hasło admin

Bezpieczeństwo

- Uwierzytelnianie i Autoryzacja: Identity Framework do zarządzania użytkownikami.
- Zabezpieczenia Hasel: Sposoby hashowania i solenia.
- Kontrola Dostępu: Atrybuty [Authorize] i [AllowAnonymous].

Szczegóły Modeli i Kontrolerów

Modele

1. **Movie**: Przechowuje informacje o filmach, takie jak **Id**, **Title** (tytuł), **Description** (opis), **Duration** (czas trwania) oraz **Image** (ścieżka do obrazu filmu).
2. **Order**: Reprezentuje zamówienia z **Id**, listą **OrderDetailsIds**, powiązonym seansem (**SeanceId**, **Seance**) oraz informacjami takimi jak data utworzenia (**CreateDate**).
3. **OrderDetails**: Zawiera szczegółowe dane o zamówieniach, w tym **Id**, **TicketId**, powiązany bilet (**Ticket**), ilość biletów (**Count**) i obliczaną sumę cenę (**PriceSum()**).
4. **Room**: Definiuje sale kinowe z atrybutami **Id**, **Name** (nazwa) oraz **Capacity** (pojemność).

5. **Seance**: Reprezentuje seanse filmowe, zawierające **Id**, informacje o filmie (**Id_movie**, **Movie**) i sali (**Id_room**, **Room**), datę rozpoczęcia (**Start**) oraz obliczaną datę zakończenia (**End()**).

Kontrolery

Każdy z kontrolerów odpowiada za zarządzanie swoim modelem w systemie:

1. **MovieController**: Zarządza operacjami na modelu **Movie**. Umożliwia wyświetlanie, dodawanie, edycję i usuwanie filmów.
2. **OrderController**: Obsługuje operacje na modelu **Order** oraz powiązanych **OrderDetails**, umożliwiając zarządzanie zamówieniami i ich szczegółami.
3. **RoomController**: Zarządza informacjami o salach kinowych (**Room**), oferując funkcje dodawania, edycji i usuwania sal oraz przeglądania ich szczegółów.
4. **SeanceController**: Umożliwia zarządzanie seansami filmowymi (**Seance**), w tym tworzenie nowych seansów, edycję, usuwanie i wyświetlanie szczegółów.
5. **TicketController**: Odpowiada za operacje na biletach (**Ticket**), w tym ich tworzenie, edycję, usuwanie oraz przeglądanie.

Każdy kontroler w systemie **ProjectPlaczek** wykorzystuje metody **Index**, **Details**, **Create**, **Edit**, i **Delete** do obsługi żądań i interakcji z bazą danych, co zapewnia spójność i efektywność zarządzania poszczególnymi segmentami systemu. Kontrolery te są kluczowe w zapewnieniu funkcjonalności aplikacji i obsłudze logiki biznesowej związanej z zarządzaniem kinem.