

MovieDB – serwis filmowy z możliwością komentowania oraz wypożyczania

O projekcie

Aplikacja MovieDB oparta jest na pomysśle, który został zrealizowany w postaci serwisów Filmweb.pl oraz IMDb.com. Posiada ona podobne funkcjonalności, które możemy znaleźć we wcześniej wymienionych serwisach. Oprócz podstawowej możliwości przeglądania katalogu już dodanych filmów (jest to możliwe zarówno dla użytkowników nie zalogowanych jak i tych, którzy posiadają już swoje konto) zaimplementowana została również możliwość administrowania katalogiem oraz jego modyfikacji (w postaci konta administratora o rozszerzonych uprawnieniach). Dla zalogowanych użytkowników udostępniona została również możliwość komentowania oraz oceniania poszczególnych wprowadzonych do katalogu filmów a także możliwość „wypożyczania” poszczególnej pozycji.

Zastosowane technologie

Aplikacja została stworzona w edytorze Visual Studio 2019 z zastosowaniem technologii ASP.NET Core MVC, EntityFramework. Technologia ta odpowiada za całą biznesową warstwę aplikacji oraz połączenie z serwerem bazodanowym, również produkcji Microsoftu – MS SQL Server 2017. Warstwa prezentacji została wykonana za pomocą składni Razor, gdzie „mieszane” ze sobą są składnia języka C# oraz znaczniki HTML stylowane za pomocą CSS. W warstwie prezentacji został również zastosowany framework Bootstrap, oraz biblioteka JavaScript jQuery (m.in. skrypt walidujący pole do wpisania karty kredytowej po stronie klienta).

Wymagania uruchomieniowe

Do uruchomienia aplikacji wymagane jest zainstalowanie:

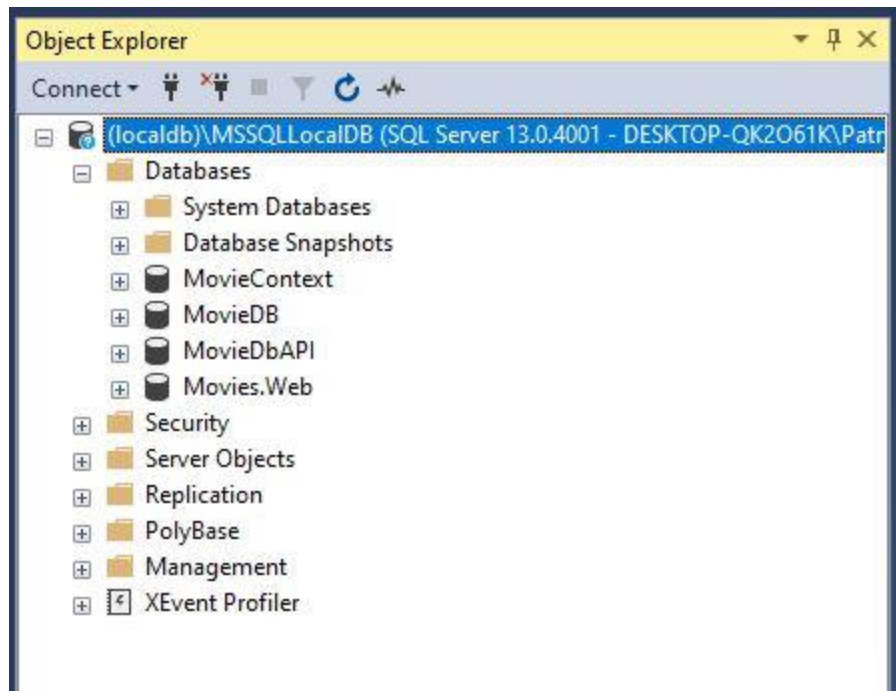
- .NET Core SDK w wersji co najmniej 2.2 (środowisko uruchomieniowe)
- Visual Studio 2019 lub Visual Studio Code
- Microsoft SQL Server w wersji co najmniej 2017

Sposób uruchomienia aplikacji

Aby uruchomić aplikację konieczne jest zainstalowanie wyżej określonych programów oraz środowiska uruchomieniowego. Następnie potrzebna będzie odpowiednio skonfigurowana baza danych. Najpierw zachodzi potrzeba zainstalowania Microsoft SQL Servera oraz narzędzia do zarządzania bazami danych MSSQL czyli Microsoft SQL Server Management Studio (dalej będziemy posługiwać się skrótem SSMS).

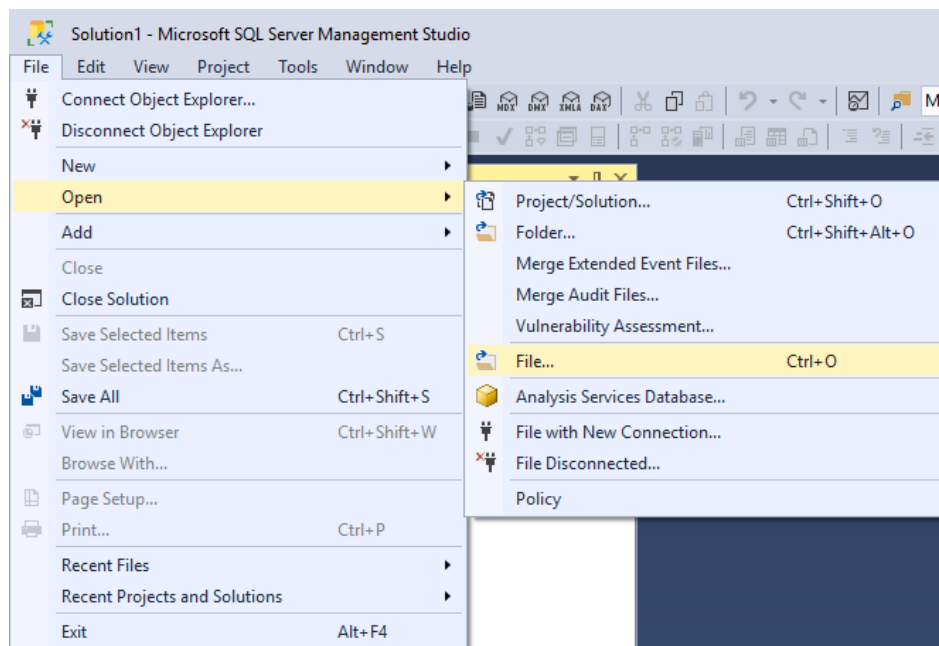
W katalogu DATABASE_EXPORT znajduje się kilka plików bazy danych – są to pliki, które pozwolą odtworzyć strukturę bazy danych. Aby zaimportować plik DATABASE.sql (zawiera on kompletną strukturę bazy danych) do naszego serwera bazodanowego otwieramy SSMS. Jeśli odpowiednio skonfigurowaliśmy naszą aplikację możemy przystąpić do

importu pliku .sql w SSMS. Po połączeniu się z lokalnym serwerem SQL przechodzimy do lewej sekcji aplikacji „Object Explorer”.



Screenshot 1

Powinniśmy zobaczyć zbliżony układ – rozwinięty folder Databases po pierwszej instalacji nie będzie najprawdopodobniej zawierał żadnej bazy danych. Aby zaimportować bazę danych z pliku DATABASE.sql rozwijamy menu kontekstowe File > Open > File i wybieramy plik DATABASE.sql



Screenshot 2

Po otwarciu pliku w nowym oknie zadokowanym na środku okna programu powinien wyświetlić się nam kod źródłowy pliku .sql.

```
DATABASE.sql - (lo...K2O61K\Patryk (54)) SQLQuery1.sql - (lo...K2O61K\Patryk (53)) Object Explorer Details
USE [master]
GO
/***** Object: Database [MovieContext] Script Date: 2020-01-11 17:23:56 *****/
CREATE DATABASE [MovieContext]
CONTAINMENT = NONE
ON PRIMARY
( NAME = N'MovieContext', FILENAME = N'C:\Users\Patryk\MovieContext.mdf' , SIZE = 73728KB , MAXSIZE = UNLIMITED, FILEGROWTH = 65536KB )
LOG ON
( NAME = N'MovieContext_log', FILENAME = N'C:\Users\Patryk\MovieContext_log.ldf' , SIZE = 8192KB , MAXSIZE = 2048GB , FILEGROWTH = 65536KB )
GO
ALTER DATABASE [MovieContext] SET COMPATIBILITY_LEVEL = 130
GO
IF (1 = FULLTEXTSERVICEPROPERTY('IsFullTextInstalled'))
begin
EXEC [MovieContext].[dbo].[sp_fulltext_database] @action = 'enable'
end
GO
ALTER DATABASE [MovieContext] SET ANSI_NULL_DEFAULT OFF
GO
ALTER DATABASE [MovieContext] SET ANSI_NULLS OFF
GO
ALTER DATABASE [MovieContext] SET ANSI_PADDING OFF
GO
ALTER DATABASE [MovieContext] SET ANSI_WARNINGS OFF
GO
ALTER DATABASE [MovieContext] SET ARITHABORT OFF
```

Screenshot 3

W tym miejscu należy się na chwilę zatrzymać. Ujęta w nawiasach nazwa [MovieContext] będzie naszą przyszłą referencją do bazy danych. Jeśli w bazie danych nie istnieje baza danych o takiej nazwie lub po prostu chcemy zmienić nazwę na inną to powinniśmy zmienić każde wystąpienie „MovieContext” w tym pliku na żądane przez nas. Można tego dokonać w dowolnym edytorze tekstowym. Podana na czerwono ścieżka kieruje nas do pliku .mdf bazy danych – jest to kolejna możliwość przeniesienia (eksportu i importu) bazy danych – niech pozostanie niezmieniona.

Po sformatowaniu wyszukujemy przycisk Execute, przy którym znajduje się zielona strzałka i spokojnie czekamy na wykonanie skryptu (w zależności od bazy danych może zająć to mgnienie oka lub kilkadziesiąt sekund lub więcej, nasza baza jest niewielka więc powinno wystarczyć kilka sekund) lub wciskamy F5 co da taki sam efekt. Po wczytaniu skryptu nasza baza danych jest gotowa lecz pusta. Aplikacja została na tę ewentualność przygotowana i nie powinna sprawić problemu gdy nie jest domyślnie wypełniona danymi.

Jeśli wszystko przebiegło pomyślnie, przejdźmy do Visual Studio 2019 lub Visual Studio Code aby otworzyć nasze rozwiązanie i odpowiednio je skonfigurować aby móc zapewnić połączenie aplikacji z wcześniej utworzoną bazą danych. Z mojej strony zalecam używanie Visual Studio 2019 zamiast Visual Studio Code choć są to podobne programy, VS 2019 daje pełniejsze doświadczenie z projektem

A więc aby otworzyć nasz projekt w Visual Studio 2019 (program powinniśmy mieć zainstalowany na komputerze) przechodzimy do katalogu MovieDB i uruchamiamy plik WebApp1.sln. Automatycznie powinien uruchomić się VS 2019 i otworzyć cały projekt. Warto przed rozpoczęciem edycji projektu poczekać chwilę na ewentualne doczytanie się brakujących pakietów, pobranie zależności. Następnie warto zbudować projekt aby wyłapać ewentualne błędy już na starcie (przeniesienie aplikacji na inną maszynę może wywołać problemy, jednak odpowiednie i zgodne z instrukcją skonfigurowanie komputera czyli instalacja odpowiednich bibliotek nie powinna dać takiemu zdarzeniu szansy). Aby to zrobić przechodzimy do menu kontekstowego Build i najpierw wybieramy > Clean solution a następnie > Build solution. Jeżeli w tym momencie włączymy aplikację bez uprzedniego skonfigurowania połączenia z bazą danych najprawdopodobniej otrzymamy w oknie przeglądarki komunikat o wystąpieniu nieobsłużonego wyjątku:

Aby uniknąć takich nieprzyjemnych widoków skonfigurujmy połączenie bazy danych we właściwy sposób (w końcu

An unhandled exception occurred while processing the request.

SqlException: A network-related or instance-specific error occurred while establishing a connection to SQL Server. The server was not found or was not accessible. Verify that the instance name is correct and that SQL Server is configured to allow remote connections. (provider: SQL Network Interfaces, error: 26 - Error Locating Server/Instance Specified)

```
System.Data.ProviderBase.DbConnectionPool.TryGetConnection(DbConnection owningObject, UInt32 waitForMultipleObjectsTimeout, bool allowCreate, bool onlyOneCheckConnection, DbConnectionOptions userOptions, out DbConnectionInternal connection)
```

Stack

Query

Cookies

Headers

SqlException: A network-related or instance-specific error occurred while establishing a connection to SQL Server. The server was not found or was not accessible. Verify that the instance name is correct and that SQL Server is configured to allow remote connections. (provider: SQL Network Interfaces, error: 26 - Error Locating Server/Instance Specified)

```
System.Data.ProviderBase.DbConnectionPool.TryGetConnection(DbConnection owningObject, UInt32 waitForMultipleObjectsTimeout, bool allowCreate, bool onlyOneCheckConnection, DbConnectionOptions userOptions, out DbConnectionInternal connection)
```

```
System.Data.ProviderBase.DbConnectionPool.TryGetConnection(DbConnection owningObject,
```

Screenshot 4

chcemy podzielać co nieco w samej aplikacji). Z prawej strony powinniśmy dostrzec zadokowane okno o nazwie Solution Explorer. Przyjrzyjmy się dobrze i odnajdźmy w nim plik „appsettings.json” – jest to nasz plik konfiguracyjny dla połączenia z bazą danych.

Jego treść wygląda następująco:


```
8  },
9  "AllowedHosts": "*",
10 "ConnectionStrings": {
11   "MovieContext": "Server=(localdb)\\mssqllocaldb;Database=MovieContext;Trusted_Connection=True;MultipleActiveResultSets=true"
12 },
13 "UserSettings": {
14   "UserEmail": "admin@movies.com",
15   "UserPassword": "Admin@123"
16 }
17 }
18 }
```

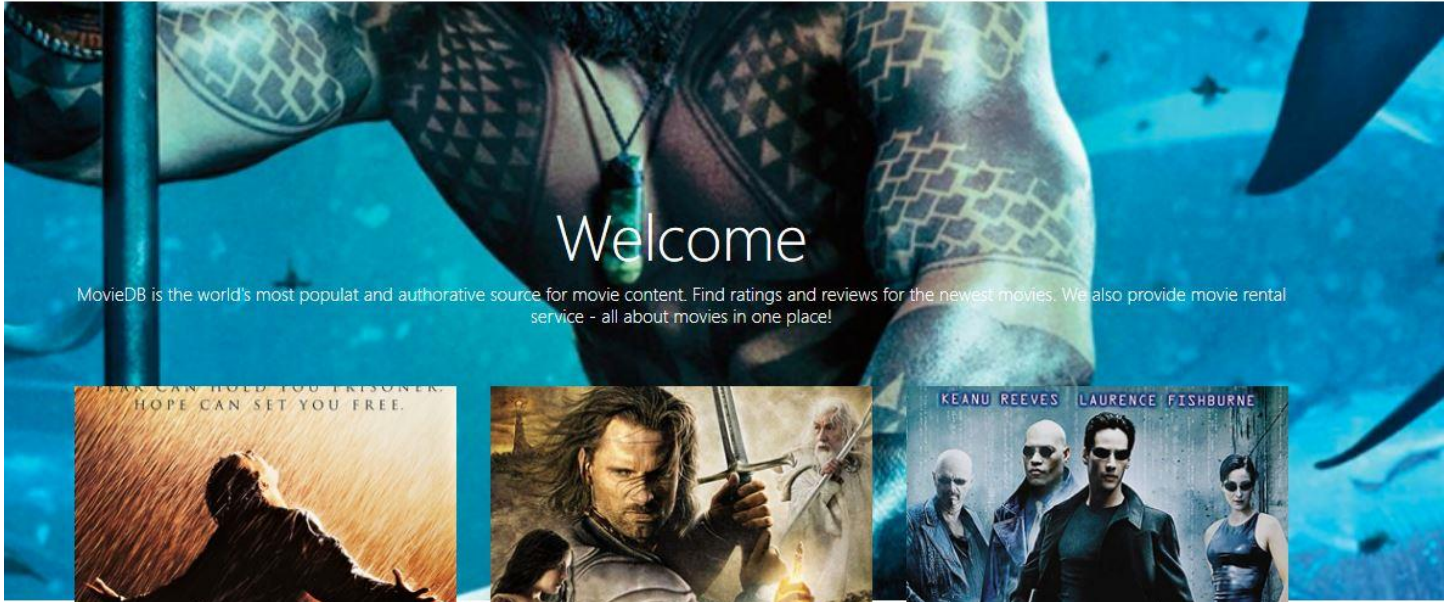
Screenshot 5

„WebApplicationContext2” jest aktualnie wykorzystywaną perspektywą do połączenia z bazą danych. W miejscu Server, po znaku = wpisujemy nazwę naszego serwera. Jest to nazwa, jaką możemy zobaczyć w SSMS po lewej stronie, nad aktualnie posiadanymi bazami danych (patrz wyżej, screenshot 1). W tym przypadku aktualnie jest To **(localdb)\\mssqllocaldb** – tę wartość ustaw na taką jaką widzisz u siebie. Kolejny atrybut, który musimy poprawić to „Database”. Jego wartość ustawiamy zgodnie z wybraną przez nas nazwą bazy danych. Jeśli nie zmieniliśmy nic, powinna znajdować się tam wartość z pliku DATABASE.sql. Jeśli wszystkie dotychczasowe kroki wykonaliśmy zgodnie z niniejszą dokumentacją i nie napotkaliśmy po drodze żadnych błędów możemy w końcu kliknąć przycisk IIS Express obok zielonej strzałki i uruchomić naszą skonfigurowaną aplikację.

Pierwsze wczytanie może zająć nieco dłużej czasu z powodu cache’owania się plików projektu – przy następnych uruchomieniach (o ile nie wyczyścimy pamięci podręcznej) aplikacja uruchomi się szybciej.

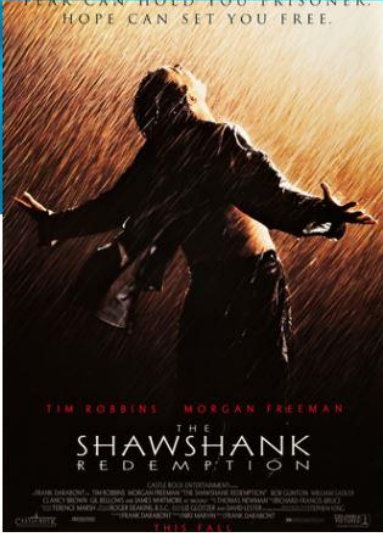
Strona domowa aplikacji:

 [MovieDB](#) [Home](#) [Privacy](#) [Register](#) [Login](#)




Welcome

MovieDB is the world's most popular and authoritative source for movie content. Find ratings and reviews for the newest movies. We also provide movie rental service - all about movies in one place!




9 reviews

★ 8.00



1 reviews

★ 9.00



1 reviews

★ 8.00

News

Screenshot 6

Ze strony domowej możemy przejść do katalogu filmów poprzez odnośnik „Movies”. Możemy także zapoznać się z polityką prywatności przechodząc do sekcji „Privacy”. Aby utworzyć konto nowego użytkownika (zwykłego użytkownika z uprawnieniami do przeglądania filmów, oceniania oraz komentowania) z wybranymi przez nas danymi osobowymi, datą urodzenia oraz loginem i hasłem przechodzimy do sekcji „Register”. Po utworzeniu konta zostajemy automatycznie zalogowani, wówczas obok pojawi się przycisk „Logout”, który pozwala nam zakończyć sesję danego użytkownika. Aby zalogować się na utworzone wcześniej konto przechodzimy do sekcji „Login”.



Register.

Create a new account.

Email

Password

Confirm password

Register

Screenshot 7

Aby administrować aplikacją potrzebne jest konto administratora. Taka funkcjonalność została zakodowana w klasie Startup.cs naszej aplikacji i przy pierwszym uruchomieniu zostaje utworzone konto administratora z następującymi danymi:

Login: admin@movies.com

Hasło: Admin@123



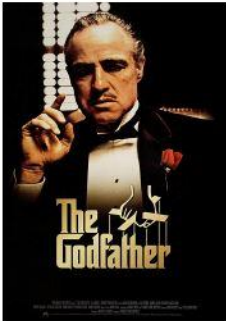

Po zalogowaniu na to konto możemy dodawać filmy do bazy, edytować je oraz usuwać. Wygląd interfejsu dla administratora różni się nieco od interfejsu zwykłego użytkownika – są to różnice wynikające z zakresu uprawnień i przyporządkowania do roli.

Widok administratora po zalogowaniu i przejściu do „Movies”:

MovieDB

Create New

Search:

Title	Runtime	Year	Administracja
 <div><p>The Shawshank Redemption</p><p>Two imprisoned men bond over a number of years, finding solace and eventual redemption through acts of common decency.</p></div>	142min	 1994	Edit Details Delete
 <div><p>The Godfather</p><p>The aging patriarch of an organized crime dynasty transfers control of his clandestine empire to his reluctant son.</p></div>	175min	 1972	Edit Details Delete

Screenshot 8

Jak widzimy, w bazie istnieje już kilka dodanych wcześniej rekordów – dla jasności procesu tworzenia stwórzmy nowy film. Klikamy w „Create New”. W tym momencie przechodzimy do widoku dodawania filmów. Jest on dostępny tylko po zalogowaniu na konto administratora. Próba przejścia do tej lokalizacji na innym koncie (bez roli Admin) skończy się wyświetleniem komunikatu „Access denied”. Pola wypełniamy zgodnie z ich przeznaczeniem, próba wypełnienia pola niedozwolonymi znakami zostanie natychmiast zgłoszona przez walidację – informacja zwrotna będzie zawierać właściwy sposób wypełnienia danego pola.

Widok poprawnie wypełnionego formularza dodania filmu. Plakat dodajemy poprzez wybranie odpowiedniego pliku, do którego przechodzimy klikając „Browse” w polu „Choose image...”:

Create

Movie

Title

Runtime

Year

Poster

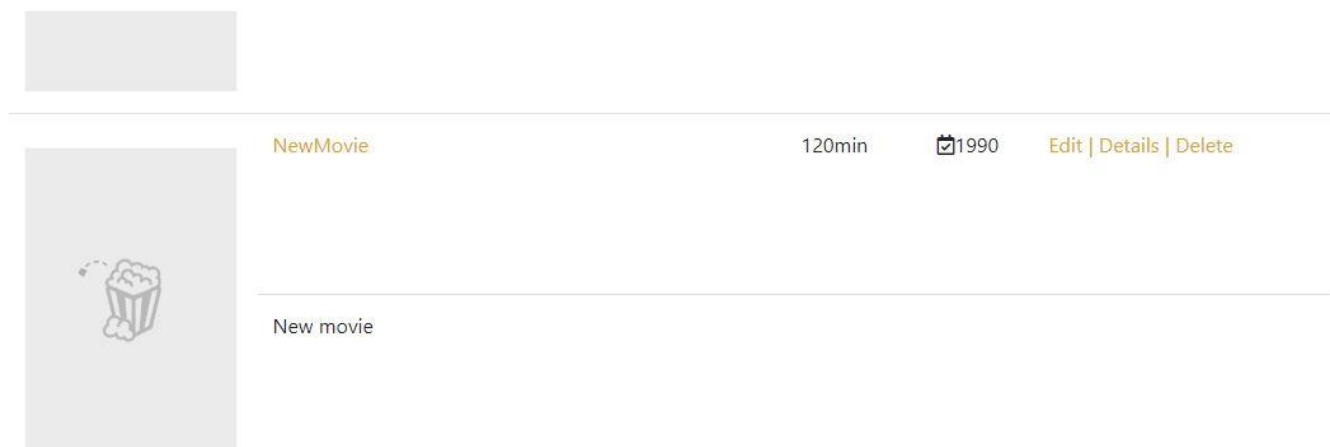
Description

[Back to List](#)

Screenshot 9

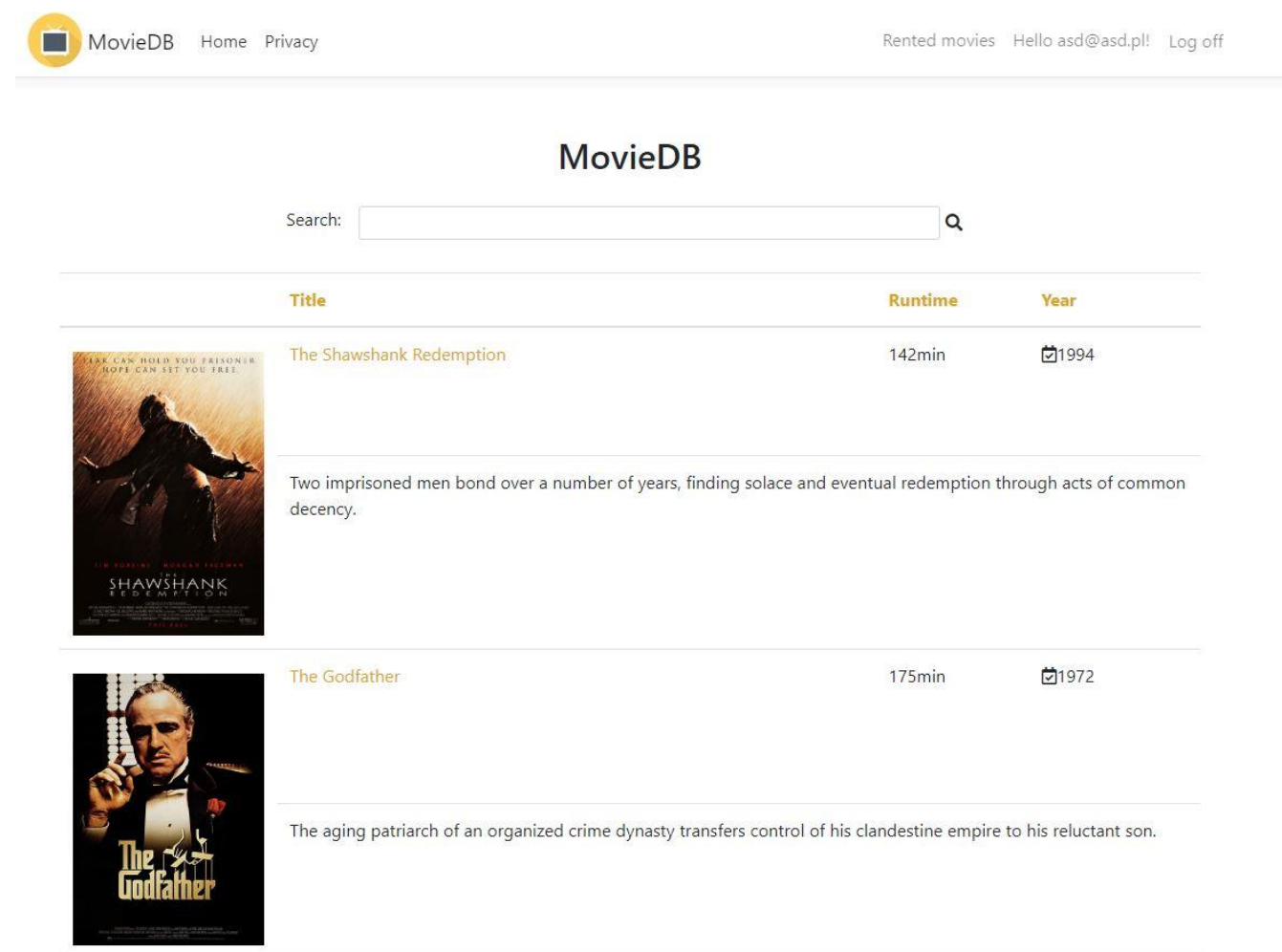
Po dodaniu, zostaniemy przekierowani na stronę „Movie” gdzie możemy zobaczyć efekt naszego działania. Film został dodany, możemy zobaczyć dodane przez nas informacje oraz plik z plakatem w formie obrazka. W widoku administratora widzimy dodatkowo możliwość edycji oraz usunięcia wpisu. W widoku użytkownika możliwości te nie są widoczne, nie są też w jakikolwiek sposób dostępne.

Widok „Movies” po dodaniu filmu:



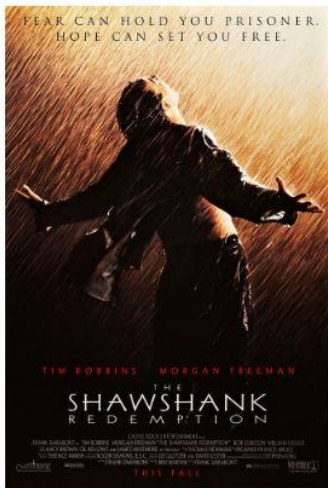
Screenshot 10

Tak natomiast prezentuje się widok „Movies” po zalogowaniu na profil zwykłego użytkownika:



Screenshot 11

Przejdźmy do widoku „Details” gdzie będziemy mogli zobaczyć pole do wpisania komentarza oraz slider do wyboru oceny:



FEAR CAN HOLD YOU PRISONER.
HOPE CAN SET YOU FREE.


TIM ROBBINS MORGAN FREEMAN

THE SHAWSHANK REDEMPTION

THIS FALL

The Shawshank Redemption

Rent this movie

Average rating: 8.00
Based on  ratings

Year: 1994

Runtime: 142

Plot: Two imprisoned men bond over a number of years, finding solace and eventual redemption through acts of common decency.

5

Content

Add review

Screenshot 12

Po dodaniu komentarza, wybraniu oceny oraz zatwierdzeniu, strona odświeży się pokazując nowododany komentarz. Przedstawia on login autora komentarza, datę dodania oraz ocenę a także treść samego komentarza. W trakcie kończenia projektu pojawiło się pytanie – co stanie się jeśli konto użytkownika zostanie usunięte? Co stanie z dodanymi przez niego komentarzami? Otóż zostaną one usunięte kaskadowo, nie zostanie po nich śladu. Tak samo sprawa ma się z filmami – po usunięciu filmu kaskadowo usuwane są wszystkie dotyczące go komentarze i oceny.

Kolejną funkcjonalnością systemu jest możliwość wypożyczenia filmu. Po zalogowaniu się i przejściu do szczegółowego widoku filmu dostrzec możemy przycisk „Rent this movie”. Po kliknięciu zostaniemy przekierowani do ekranu potwierdzenia chęci przeprowadzenia symulacji płatności.

Confirm that you are willing to rent this movie:

"The Shawshank Redemption"

Price: 4.99\$ (one day)

Confirm

CardNo

Create

Screenshot 13

Po wpisaniu poprawnego numeru karty bankowej (np. 4179592160025006) i potwierdzeniu, zostaniemy przeniesieni do podsumowania. Stąd możemy przejść do „obejrzenia” filmu lub z powrotem do jego karty. Po zalogowaniu obok loginu widoczna jest sekcja „Rented movies”. Po przejściu do niej pojawi się lista z aktualnie wypożyczonymi przez nas filmami (tylko aktywne wypożyczenia).

4



The Shawshank Redemption

2.99\$

2020-01-11 16:16:06

2020-01-16 16:16:06

Screenshot 14

Uruchamianie aplikacji w Visual Studio Code

Aby uruchomić aplikację w programie Visual Studio Code zastosujemy następujące kroki:

1. Otwieramy program VS Code.
2. Otwieramy folder z naszą aplikacją w programie VS Code: File > Open folder...
3. Przechodzimy do terminala wybierając Terminal > New terminal... lub wybieramy kombinację klawiszy klawiatury Ctrl + ~.
4. Wpisujemy komendy `dotnet restore` a następnie po jej wykonaniu `dotnet run`
5. Jeśli wszystko poszło tak jak powinno, na samym dole terminala powinniśmy odnaleźć linijkę brzmiącą następująco (bardzo podobnie): Now listening on: <http://localhost:51198> (zamiast 51198 możemy zobaczyć inny port, jest to przykład)
6. Wpisujemy w przeglądarkę ten adres i widzimy uruchomioną aplikację
7. Kombinacja Ctrl + C zamyka działanie serwera.