

# MovieDB – społecznościowy serwis dla fanów kina

Przedmiot: Projektowanie Portali Społecznościowych

Patryk Gwizdała, Mateusz Krajnik, Konrad Kubrak

Informatyka III rok inżynierskie

2020

Prowadzący: dr inż. Wojciech Kozioł

# Opis projektu

Projekt zakłada stworzenie aplikacji webowej będącej bazą filmów, gromadzącą społeczność kinomanów (podobne serwisy to na przykład Imdb.com, Filmweb.pl).

Serwis będzie udostępniał rejestrację – zarejestrowani użytkownicy będą mogli komentować oraz oceniać dodane filmy (dodawane przez administratora), niezarejestrowani zaś będą mogli tylko przeglądać zawartość strony bez możliwości kreacji treści.

Administrator będzie miał możliwość dodawania, edytowania oraz usuwania treści ze strony. W przypadku usunięcia filmu z bazy przez administratora, kaskadowo usuwane będą wszystkie dotyczące go oceny oraz komentarze pozostawione przez użytkowników.  
W przypadku usunięcia filmu z bazy, wszystkie dotyczące go komentarze oraz oceny również będą usuwane kaskadowo.

Serwis został wzbogacony również o możliwość „wypożyczania” filmów. Zarejestrowany i zalogowany użytkownik będzie miał możliwość „kupienia” (symulacja procesu transakcji) filmu na określony czas. Przez cały czas trwania wypożyczenia wypożyczający ma możliwość skorzystania z odnośnika prowadzącego do filmu w serwisie wideo (YouTube).

Społecznościowe aspekty serwisu, które pozwalają na interakcję pomiędzy użytkownikami i tworzonymi przez nich treściami to:

* śledzenie użytkownika – pozwala użytkownikowi na „followowanie” innego użytkownika portalu a tym samym śledzenie tworzonych przez niego wpisów, polubień, ocen,
* „polubienia” dla filmów – oprócz oceniania użytkownik może dodać dany film do ulubionych,
* udostępnianie swojego statusu – w podobny sposób jak na Facebook’u, użytkownik na głównej stronie posiada pole tekstowe, w którym może podzielić się szybko swoimi przemyśleniami lub udostępnić jakąś wiadomość. Może ona być komentowana przez innych użytkowników (opcja szybkiej odpowiedzi na stronie głównej lub przejście do postu gdzie zobaczymy cała historię komentarzy),
* profil użytkownika – każdy użytkownik posiada własny profil, który pozwala na interakcję z ostatnimi utworzonymi przez niego treściami (prezentacja kilku ostatnich treści danego typu); tam także można dodać użytkownika do „śledzenia”,
* globalny „feed” – możliwość zobaczenia aktywności wszystkich użytkowników serwisu bez względu na to, czy są przez nas obserwowani czy też nie,

Technologie i narzędzia:

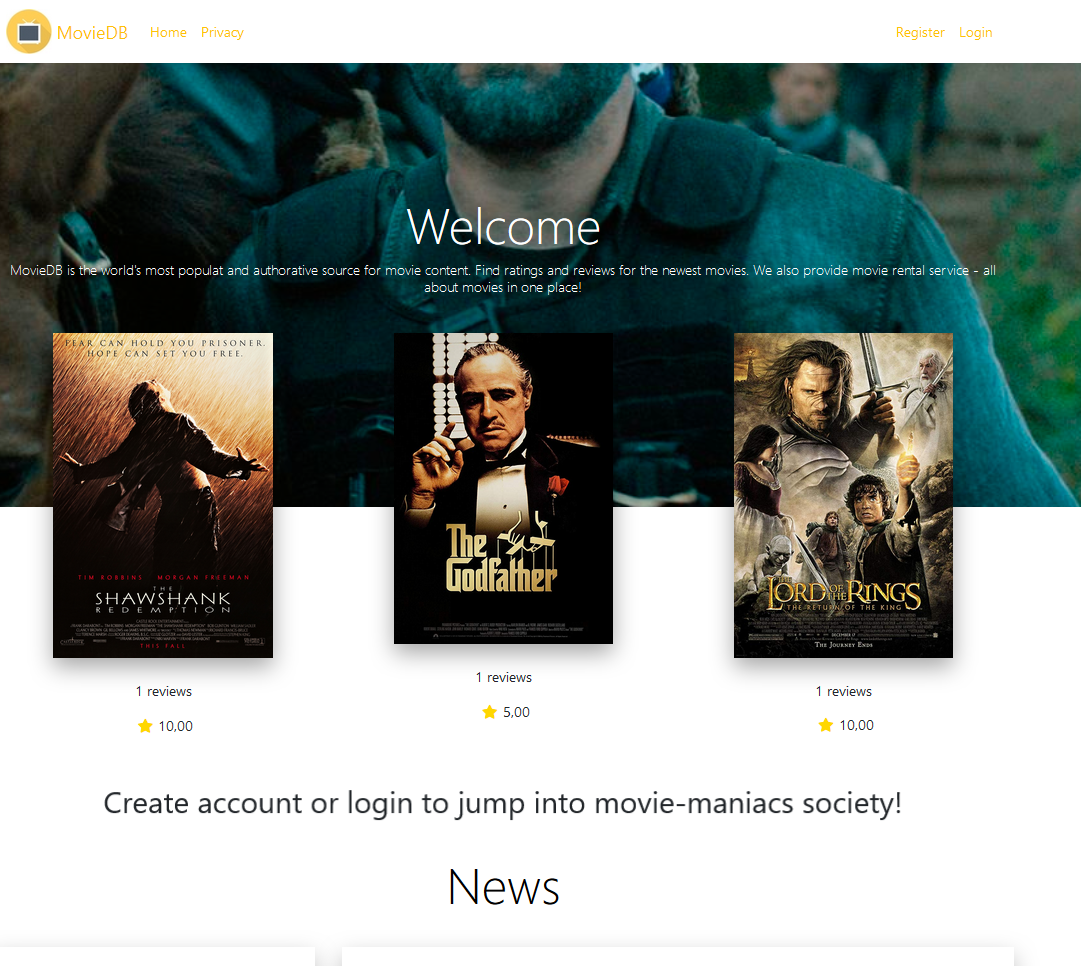
ASP.NET Core, SQL Server, Bootstrap, Entity Framework.

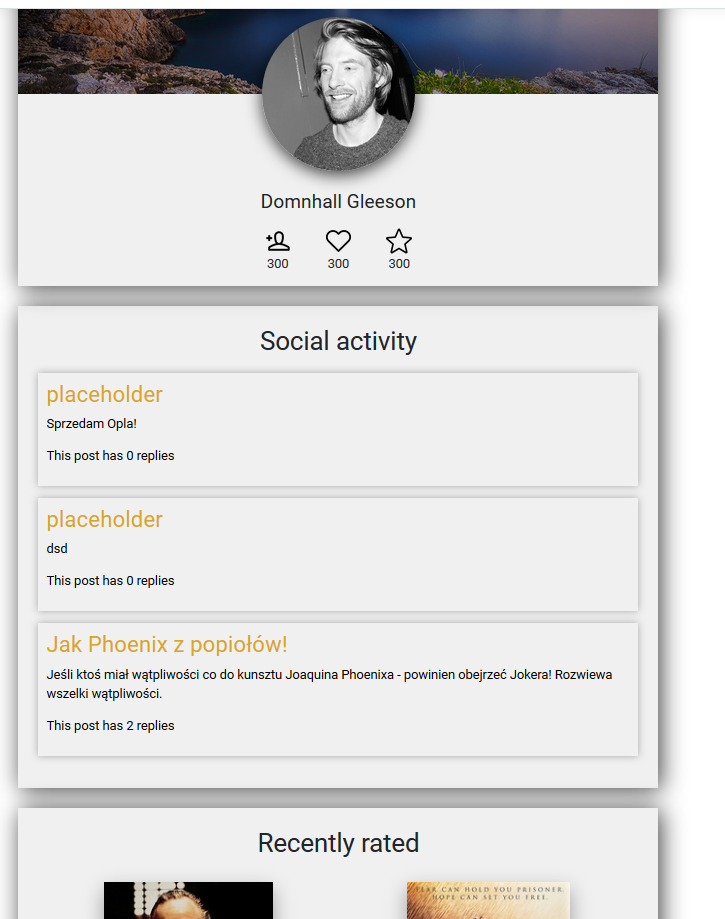
Architektura:

Aplikacja korzysta ze wzorca MVC (Model-View-Controller). Projekt (w nazewnictwie Visual Studio jest to ogólnie „solution”) został rozbity na mniejsze części („projects”) aby zachować odrębność poszczególnych elementów oraz wprowadzić lepsze zarządzanie.

# Projekt interfejsu

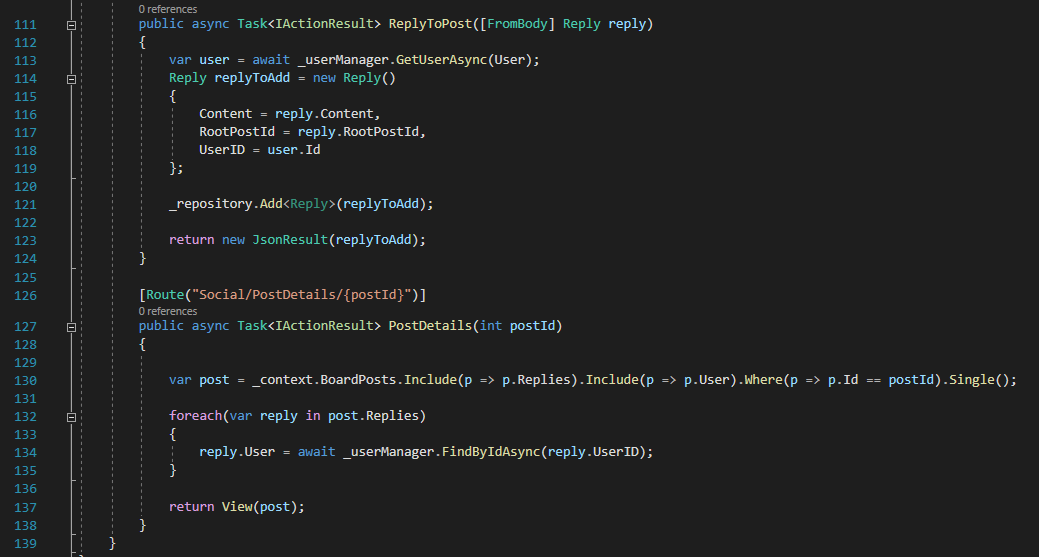
Projekt interfejsu jest dosyć minimalistyczny. Do zaprojektowania został wykorzystany HTML oraz CSS wraz z frameworkiem Bootstrap. Interfejs w zamyśle ma być jak najmniej złożony ale bogaty oraz intuicyjny – użytkownik nie może zastanawiać się za długo, gdzie ma kliknąć. Jego uwaga nie może zostać rozproszona.



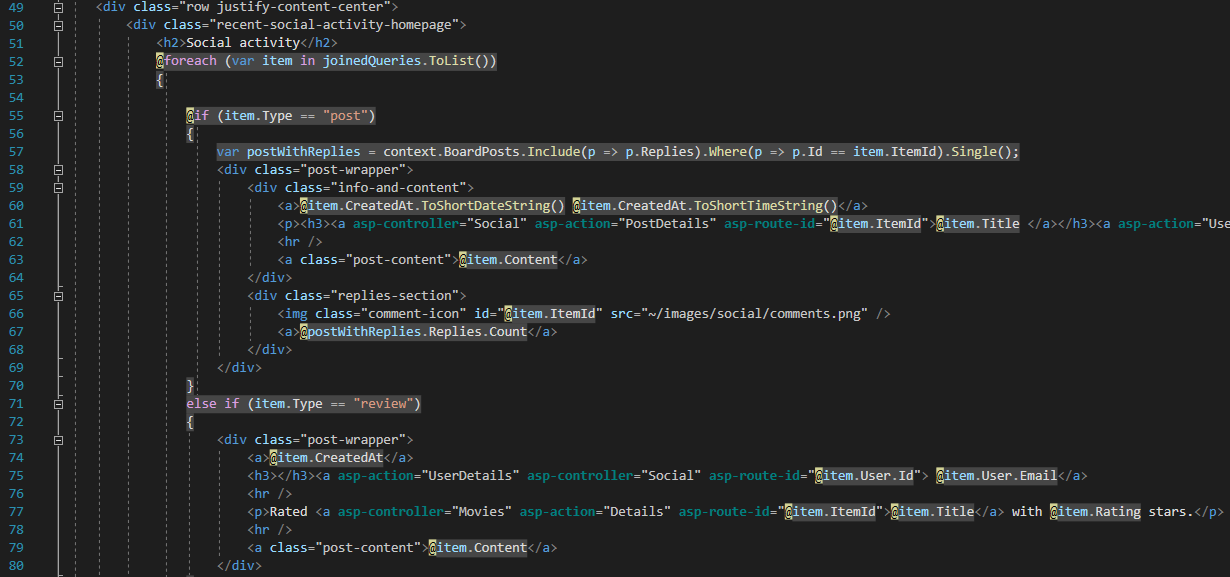


# Przykładowy kod

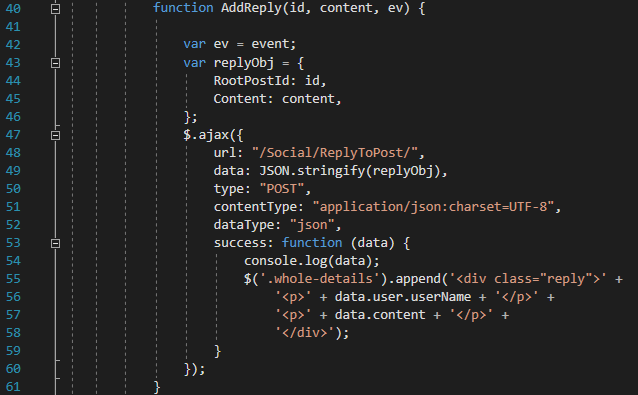
Przykładowy kod prezentujący kolejno: akcje kontrolera, widok oraz przykładowy skrypt



screenshot 1 – Kod kontrolera

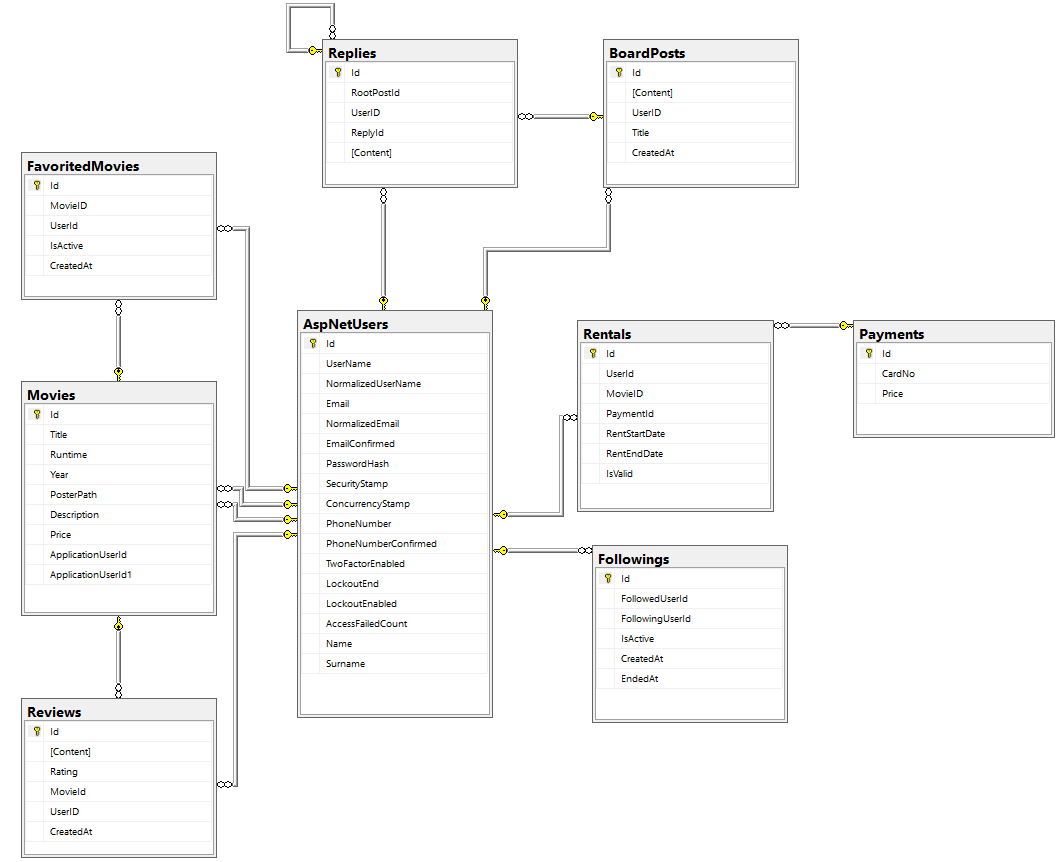


screenshot 2 – Kod widoku



screenshot 3 – Kod skryptu

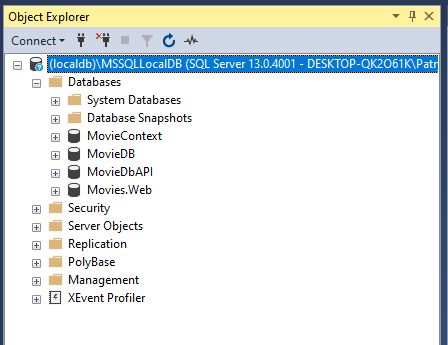
# Diagram ERD



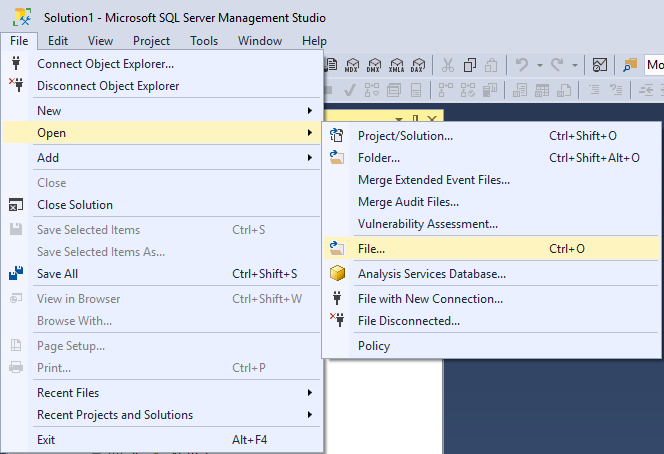
# Konfiguracja i uruchomienie projektu

## 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 nasze aplikacje 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 otworzeniu pliku w nowym oknie zadokowanym na środku okna programu powinien wyświetlić się nam kod źródłowy pliku .sql.

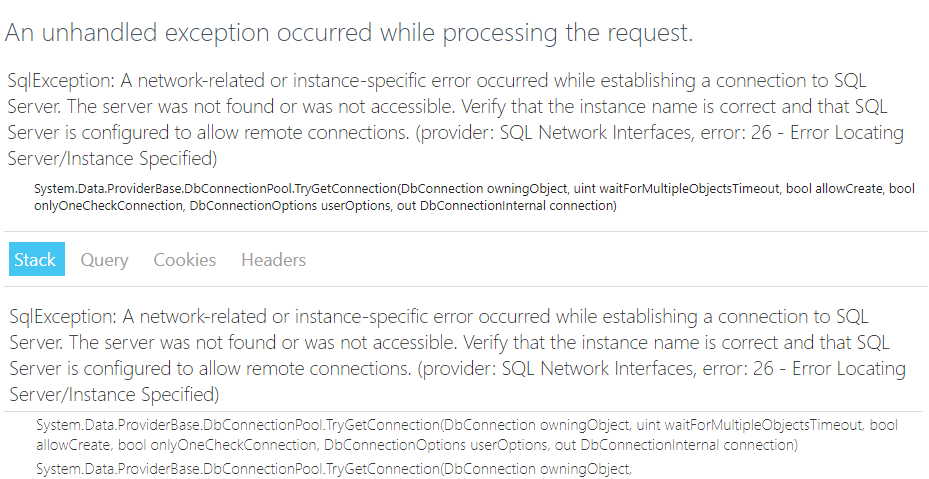
 W tym miejscu należy się na chwilę zatrzymać. Ujęta w nawiasach nazwa [MovieContext] będzie naszą przyszłą referencją do bazy danych. Jeśli w bazie danych nie istnieje baza danych o takiej nazwie lub po prostu chcemy zmienić nazwę na inną to powinniśmy zmienić każde wystąpienie „MovieContext” w tym pliku 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.

Screenshot 3

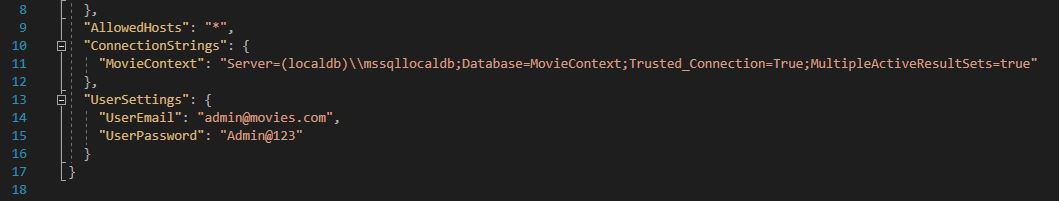
Po sformatowaniu wyszukujemy przycisk Execute, przy którym znajduje się zielona strzałka i spokojnie czekamy na wykonanie skryptu (w zależności od bazy danych może zająć to mgnienie oka lub kilkadziesiąt sekund lub więcej, nasza baza jest niewielka więc powinno wystarczyć kilka sekund) lub wciskamy F5 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 chwilkę 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 chcemy podziałać 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.

Screenshot 4

 Jego treść wygląda następująco:

Screenshot 5

„WebAppContext2” 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 Expres 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.

## 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.

# Wnioski

Sama faza programowania aplikacji, dzięki zastosowaniu dosyć dobrze znanego wzorca MVC oraz przystępnych technologii (C# i .NET, JavaScript oraz HTML i CSS) nie nastręczała trudności. Tym co wymagało największego wkładu w rozwoju projektu była faza projektowania. Powiązania danych musiały być przemyślane i nie mogły pozostawiać miejsca na redundancję lub interpretację inną niż docelowa.

Tworzenie portalu pod kątem gromadzenia społeczności wymaga wkładu pracy głównie w aspekcie społecznościowym. Samo tworzenie aplikacji nie jest wygórowanym wyzwaniem. Aby zatrzymać użytkownika, zainteresować go treścią i sprawić by chciał tworzyć społeczną treść trzeba czegoś więcej. Jest to zadanie bardzo złożone, które spędza sen z powiek wielkim korporacjom tego świata ale też mniejszym graczom. Dzięki temu przedmiotowi i wykonanemu do jego zaliczenia projektowi z pewnością zdobyliśmy wiedzę o projektowaniu portali społecznościowych i będziemy ją wykorzystywać w pracy zawodowej.