

PPSI

Członkowie:

- [Piotr Tekieli](#)
- [Samuel Leonczyk](#)
- [Mariusz Skuza](#)

Projekt aplikacji forum internetowe

Spis treści

- 1) Opis funkcjonalny systemu.
- 2) Wyszczególnione wdrożone kwalifikacje.
- 3) Streszczenie opisu technologicznego.
- 4) Instrukcja lokalnego uruchomienia systemu.
- 5) Wnioski projektowe.

1. Opis funkcjonalny systemu: Jest to aplikacja internetowa w postaci forum, która pozwala tworzyć wątki, edytować je i usuwać oraz dodawać do nich komentarze.

2. Wyszczególnione wdrożone kwalifikacje:

- 1) HTML5 - jest jako niezbędny element do wyświetlania treści na stronie.
- 2) CSS3 - formatowanie treści za pomocą bootstrap 4.
- 3) Formularze - wykorzystane do tworzenia użytkownika, komentarzy oraz samych postów.
- 4) Baza danych - połączenie z lokalną bazą danych, w której przechowywane są użytkownicy, komentarze oraz posty.
- 5) Router - wykorzystanie RouteAttribute to zarządzania ścieżkami dostępu.
- 6) Uwierzytelnianie - rejestracja i logowanie użytkowników za pomocą ASP.NET Identity.
- 7) MVC - wykorzystanie wzorca mvc w celu łatwiejszej lokalizacji elementów projektu systemu internetowego poprzez dzielenie aplikacji na wyspecjalizowane części.

- 8) CRUD - tworzenie komentarzy, postów oraz ich edycja, usuwanie i przeglądanie.
- 9) ORM - wykorzystanie Package Manager Console do zmigrowania modeli systemu do bazy danych oraz używanie programowania obiektowego do manipulowania danych z bazy danych.
- 10) Wystawianie API - umożliwiamy pobranie danych o wątkach i komentarzach.
- 11) Konsumowanie API - jest możliwość wybrania miasta aby sprawdzić jego aktualną temperaturę przy wykorzystaniu API z openweathermap.
- 12) AJAX - asynchroniczne zapytania pozwalają na szybkie wyświetlanie nowych stron komentarzy bez konieczności odświeżania całej strony.
- 13) Mail - Możliwość wysłania maila przez protokół SMTP wykorzystując skonfigurowaną pocztę gmail.
- 14) Lokalizacja - brak.
- 15) RWD - strona poprawnie wyświetla się na wielu różnych rozdzielczościach.
- 16) Logger - prosty logger z biblioteki Microsoft.Extensions.Logging, który pozwala monitorować aktywność na naszej stronie.
- 17) Cache - informacje o wątkach będą pobierane najczęściej co 5 sekund, ponieważ serwer je zapisuje, aby zmniejszyć obciążenie na serwerze.
- 18) System zarządzania zależnościami - brak.
- 19) Automatyzacja - brak.
- 20) SEO - poprawne tytuły, słowa kluczowe i opisy dla stron w formie meta tagów, aby pomóc wyszukiwarkom poprawnie znaleźć i opisać strony.

3. Streszczenie opisu technologicznego Frontend i backend jest oparty na frameworku ASP.NET MVC w wersji 5. Pliki wyświetlające treść na stronie są zapisane za pomocą Razor, który jest połączeniem języków C# i HTML. Pozwala na wykorzystywanie metod i funkcji C# przy renderowaniu strony HTML. Dodatkowo frontend używa biblioteki CSS Bootstrap.

4. Instrukcje lokalnego uruchomienia systemu Aby móc korzystać z aplikacji, należy sklonować repozytorium. Następnie otworzyć projekt w Visual Studio. Aby aplikacja działała poprawnie, należy mieć zainstalowany dodatek:



Opracowywanie zawartości dla platformy ASP.NET i sie...



Twórz aplikacje internetowe dla wielu platform przy użyciu technologii ASP.NET Core, ASP.NET, HTML/JavaScript i k...

oraz NET 5.0 SDK ze strony Microsoft <https://dotnet.microsoft.com/download> Następnie przechodzimy do Package Manager Console i komendą "add-migration [nazwa]" oraz "update-database" i uruchamiamy projekt.

5. Podsumowanie/wnioski projektu Projekt został wykonany we frameworku ASP.NET MVC. Początki były trudne, ponieważ nigdy wcześniej nie mieliśmy kontaktu z takim zadaniem, ale z biegiem czasu praca stawała się coraz przyjemniejsza. Framework ma bardzo dużo wbudowanych funkcjonalności, które przyspieszają pracę. Wcześniej nie mieliśmy również kontaktu z bibliotekami CSS takimi jak bootstrap. Bootstrap okazał się bardzo przydatny i przyjemny w użyciu. Poniżej przykładowy ss z aplikacji:

