

## Autor projektu

Imię i nazwisko: **Hubert Krzemiński**

## Tytuł projektu

Forum dyskusyjne z obsługą postów, komentarzy i obrazów

## Opis projektu

Forum dyskusyjne umożliwiające użytkownikom:

- Tworzenie postów tekstowych z możliwością dodania obrazu.
- Dodawanie komentarzy do postów innych użytkowników.
- Usuwanie własnych postów i komentarzy.
- Zarządzanie kontem użytkownika, w tym jego usunięcie.
- Tylko autorzy postów i komentarzy mogą je edytować lub usuwać, poza administratorami, którzy mają pełny dostęp do zarządzania treściami.

## Technologie użyte w projekcie

### Backend

- Język programowania: **C#**
- Framework: **.NET 8**
- ORM: **Entity Framework Core**
- System plików: **Lokalny**
- Autentykacja: **ASP.NET Identity**

### Frontend

- Framework: **Blazor**
- Stylizacja: **css**

### Baza danych

- Baza: **Microsoft SQL Server**
- Tabele: **Microsoft SQL Server**

## Infrastruktura

- IDE: **Visual Studio 2022**
- System kontroli wersji: **Git**
- Hostowanie: **Self-Host**

## Instrukcje instalacji i uruchomienia

### Wymagania systemowe

- Zainstalowany .NET SDK (wersja 8).
- Zainstalowany Microsoft SQL Server.
- Zainstalowany Visual Studio (wykluczone jest użycie VS-Code).

## Kroki uruchomienia

0. Instalacja wymaganych programów:
  - Instalujemy Visual Studio wraz z .NET SDK 8
1. Otwieramy program Visual Studio i wybieramy opcję klonuj repozytorium.
2. Jako lokalizacja repozytorium podajemy link:  
<https://github.com/YamishoYami/WSB-RadioForum>
3. Wybieramy opcję: klonuj
4. Wchodzimy do Konsoli Menadżera Pakietów i wykonujemy inicjalizację bazy danych komendą: update-database
5. Odpalamy projekt i w przeglądarce powinna otworzyć się strona localhost
6. Wchodzimy w register i tworzymy konto
7. W przypadku potrzeby zalogowania na konto admina dane logowania to:  
[superadmin@gmail.com](mailto:superadmin@gmail.com) 123Pa\$\$word.

## Funkcjonalności

1. Tworzenie konta, logowanie, wylogowanie
2. Stworzenie postu, modyfikacja i usuwanie postu danego użytkownika
3. Dodanie komentarza, usunięcie komentarza danego użytkownika
4. Dodawanie obrazów do postów, usuwanie z postu
5. Konto administratora które jest seedowane do bazy danych, może usuwać posty i komentarze użytkowników

## **Modele wykorzystane w projekcie:**

### **1. UserPost**

**Opis:** Reprezentuje post użytkownika na forum.

#### **Właściwości:**

- Id: Identyfikator posta.
- Title: Tytuł posta (maks. 100 znaków).
- Content: Treść posta (maks. 1024 znaki).
- DateAdded: Data dodania posta (domyślnie bieżąca data).
- UserId: Identyfikator użytkownika, który dodał post.
- ImagePath: Ścieżka do obrazu związanego z postem.
- imageFile: Plik obrazu (nie jest mapowany do bazy danych).
- DeleteImage: Flaga do usunięcia obrazu (nie jest mapowana do bazy danych).
- Comments: Kolekcja komentarzy związanych z postem.

### **2. Comments**

**Opis:** Reprezentuje komentarz do posta użytkownika.

#### **Właściwości:**

- Id: Identyfikator komentarza.
- Content: Treść komentarza.
- DateAdded: Data dodania komentarza.
- UserId: Identyfikator użytkownika, który dodał komentarz.
- UserPostId: Identyfikator posta, do którego komentarz jest przypisany.
- User: Nawigacja do użytkownika, który dodał komentarz.
- UserPost: Nawigacja do posta, do którego komentarz jest przypisany.

### **3. ApplicationUser**

**Opis:** Rozszerza standardowy model użytkownika ASP.NET Identity.

**Właściwości:** Dziedziczy wszystkie właściwości z IdentityUser.

## Kontrolery wykorzystane w projekcie:

### 1. UserPostsController

**Opis:** Zarządza operacjami CRUD dla postów użytkowników.

**Akcje:**

- Index(): Wyświetla listę wszystkich postów.
- Details(int id): Wyświetla szczegóły konkretnego posta.
- Create(): Wyświetla formularz tworzenia nowego posta.
- Create(UserPost userPost): Obsługuje przesłanie formularza tworzenia nowego posta.
- Edit(int id): Wyświetla formularz edycji posta.
- Edit(int id, UserPost userPost): Obsługuje przesłanie formularza edycji posta.
- Delete(int id): Wyświetla formularz usunięcia posta.
- DeleteConfirmed(int id): Obsługuje potwierdzenie usunięcia posta.

### 2. CommentsController

**Opis:** Zarządza operacjami CRUD dla komentarzy.

**Akcje:**

- Create(int userPostId): Wyświetla formularz tworzenia nowego komentarza.
- Create(Comments comment): Obsługuje przesłanie formularza tworzenia nowego komentarza.
- Edit(int id): Wyświetla formularz edycji komentarza.
- Edit(int id, Comments comment): Obsługuje przesłanie formularza edycji komentarza.
- Delete(int id): Wyświetla formularz usunięcia komentarza.
- DeleteConfirmed(int id): Obsługuje potwierdzenie usunięcia komentarza.

### 3. AccountController

**Opis:** Zarządza operacjami związanymi z kontem użytkownika, takimi jak logowanie, rejestracja, resetowanie hasła itp.

**Akcje:** Standardowe akcje związane z zarządzaniem kontem użytkownika, takie jak Login, Register, Logout, ForgotPassword, ResetPassword itp.

### Struktura projektu:

#### Plik projektu WSB-RadioForum.csproj:

- Identyfikator tajemnic: Zdefiniowano UserSecretsId do przechowywania poufnych danych konfiguracyjnych.
- Przestrzeń nazw: Ustawiono główną przestrzeń nazw na WSB\_RadioForum.
- Usunięcie migracji z kompilacji.
- Referencje do pakietów NuGet.
- Foldery: Zdefiniowano folder wwwroot\images\ do przechowywania obrazów przesyłanych przez użytkowników.

#### Widoki Razor:

- Views/UserPosts/Create.cshtml:
  - Formularz tworzenia nowego posta.
  - Pola formularza:
    - Title: tytuł posta.
    - Content: treść posta.
    - imageFile: możliwość przesłania obrazu.
  - Formularz korzysta z metody POST i obsługuje przesyłanie plików (enctype="multipart/form-data").
- Views/UserPosts/Edit.cshtml:
  - Formularz edycji istniejącego posta.
  - Dodatkowe pola:
    - DeleteImage: pole typu checkbox pozwalające na usunięcie istniejącego obrazu.
    - Ukryte pole ImagePath przechowujące ścieżkę do obecnego obrazu.
    - Sekcja Scripts łączy \_ValidationScriptsPartial dla walidacji po stronie klienta.

