

# Dokumentacja projektu Blog premium

Autorzy: Maciej Diakowicz, Mateusz Koczorowski

**Blog premium** to platforma internetowa, dedykowana jednej wybranej grupie użytkowników, którzy chcą czerpać z niej wyjątkowe korzyści. System ten posiada szereg kluczowych funkcji, które go charakteryzują:

**1. Logowanie i Rejestracja:** Użytkownicy muszą zalogować się, aby uzyskać dostęp do pełnej funkcjonalności platformy. Dla nowych użytkowników dostępna jest również opcja rejestracji, co umożliwia tworzenie kont.

**2. Wylogowanie:** Zalogowani użytkownicy mogą w dowolnym momencie bezpiecznie wylogować się z systemu, zapewniając prywatność i bezpieczeństwo swojego konta.

**3. Połączenie z Bazą Danych:** Blog premium bazuje na solidnym połączeniu z bazą danych, co umożliwia efektywne przechowywanie i zarządzanie treściami oraz danymi użytkowników.

**4. Uprawnienia Autora i Czytelnika:** W systemie przyznane są różne poziomy uprawnień. Użytkownicy mogą być autorami, co daje im możliwość tworzenia, edycji oraz usuwania własnych artykułów. Pozostali użytkownicy pełnią rolę czytelników, ograniczając ich możliwości publikacyjne, ale pozwalając na dostęp do treści autorów.

**5. System Premium:** Platforma oferuje subskrypcje premium, które dają użytkownikom dodatkowe przywileje. Osoby nie zalogowane oraz zalogowani użytkownicy bez subskrypcji premium widzą tylko pierwsze 100 znaków artykułów, podczas gdy subskrybenci premium mają pełny dostęp do wszystkich treści.

**6. Paginacja Artykułów:** Dla wygody użytkowników, platforma wyposażona jest w funkcję paginacji artykułów. Umożliwia to łatwą nawigację i przeglądanie treści poprzez przewijanie między różnymi stronami.

**7. Tworzenie i Usuwanie Artykułów:** Autorzy mają możliwość tworzenia nowych artykułów, które mogą opublikować na platformie. Dodatkowo, mają pełną kontrolę nad swoimi treściami, umożliwiając usuwanie lub aktualizację artykułów według własnego uznania.

## Użytkownicy i Role

**Customowe Pola Użytkowników:** Użytkownicy w systemie posiadają dodatkowe pola, takie jak FirstName, LastName i BirthDate.

### Role Użytkowników:

- **Admin:** Posiada pełne uprawnienia do zarządzania całą aplikacją, w tym artykułami i subskrypcjami.
- **Author:** Ma uprawnienia do tworzenia, edytowania i usuwania artykułów.
- **Reader:** Może zarządzać swoim kontem i czytać artykuły.

Nowo utworzony użytkownik zawsze posiada rolę Reader.

Role może zmieniać wyłącznie administrator.

Jeżeli podczas uruchomienia systemu, nie istnieje użytkownik o danej roli z emailem zdefiniowanym w pliku .env, w takim wypadku system stworzy użytkowników danych ról. Dane kont definiujemy w pliku .env.

**Automatyczne Tworzenie Użytkowników:** Przy pierwszym uruchomieniu system automatycznie tworzy użytkowników każdego typu z danymi dostępnymi w kodzie.

## Modele

**Articles (Artykuły):** Model artykułów umożliwia zarządzanie treściami publikowanymi na platformie.

**Subscriptions (Subskrypcje):** Model subskrypcji pozwala na zarządzanie subskrypcjami użytkowników, w tym okresami ich ważności.

**User (Użytkownik):** Oprócz standardowych pól Identity, zawiera dodatkowe pola takie jak FirstName, LastName, BirthDate.

## Funkcjonalności i Zabezpieczenia

### Dostęp do Podstron:

- Strona /Articles jest zabezpieczona i dostępna tylko dla użytkowników z rolami Admin i Author. Dodatkowo metody służące do zarządzania artykułami (Create, Delete, Edit i Details) są dostępne tylko dla użytkowników z rolą Admin i Author.
- Strona /Subscriptions jest dostępna wyłącznie dla użytkowników z rolą Admin. Dodatkowo metody służące do zarządzania subskrypcjami (Create, Delete, Edit i Details) są dostępne tylko dla użytkowników z rolą Admin.
- Navbar (Pasek Nawigacyjny):
  - Admin ma dostęp do opcji zarządzania artykułami (manage articles), subskrypcjami (manage subscriptions) oraz zarządzania użytkownikami (manage users).
  - Author ma dostęp do zarządzania artykułami (manage articles).

- User z rolą Reader nie widzi tych opcji i otrzymuje komunikat access denied przy próbie dostępu.
  - Wszyscy użytkownicy mają dostęp do
- Strona Zarządzania Użytkownikami: Dostępna tylko dla Admin, umożliwia zarządzanie kontami użytkowników. Dodatkowo metody służące do zarządzania użytkownikami (Create, Delete, Edit i Details) są dostępne tylko dla użytkowników z rolą Admin.

## Strona Główna i Artykuły

**Paginacja:** Na stronie głównej wyświetlane jest do 10 artykułów na stronę.

**Strona Artykułu:** Zawiera tytuł (title), datę publikacji (release date), imię (first name) i nazwisko (last name) autora, treść dostępną dla wszystkich (free content) oraz treść premium dostępną dla użytkowników z aktywną subskrypcją.

**Obsługa Plików:** Możliwość dodawania jednego obrazu do artykułu.

## Subskrypcje

**Zakup Subskrypcji:** Każda rola użytkownika może kupić subskrypcję. Koszt jednego dnia subskrypcji to 1 zł. Użytkownik wybiera długość subskrypcji, a po dokonaniu płatności w bazie danych aktualizowane są daty rozpoczęcia (start date) i zakończenia (end date) subskrypcji. Potwierdzenie zakupu jest poprzedzone podsumowaniem płatności. Użytkownik ma dostęp do podstrony ze swoimi subskrypcjami (aktualnymi i archiwalnymi). Admin ma dostęp do zarządzania subskrypcjami całego systemu.

## Inne Elementy

**Plik .env:** umożliwia dostosowanie indywidualnych zmiennych środowiskowych

**Zegar w Headerze:** W nagłówku strony umieszczony jest zegar, który wyświetla aktualny czas, co jest przydatną funkcją zwłaszcza na platformie informacyjnej.

## Podsumowanie

Aplikacja "ProjectBlogNews" stanowi kompleksowe rozwiązanie dla platformy blogowej lub informacyjnej, łącząc zarządzanie treścią z zaawansowanymi funkcjami subskrypcji i zarządzania użytkownikami. Dostarcza zróżnicowane role z różnymi poziomami dostępu, co zapewnia elastyczność i bezpieczeństwo zarządzania treściami. Funkcjonalności takie jak zarządzanie artykułami, subskrypcjami i kontami użytkowników, w połączeniu z dodatkowymi funkcjami, takimi jak obsługa obrazów, paginacja i zintegrowany zegar, tworzą bogate i interaktywne środowisko dla użytkowników platformy.

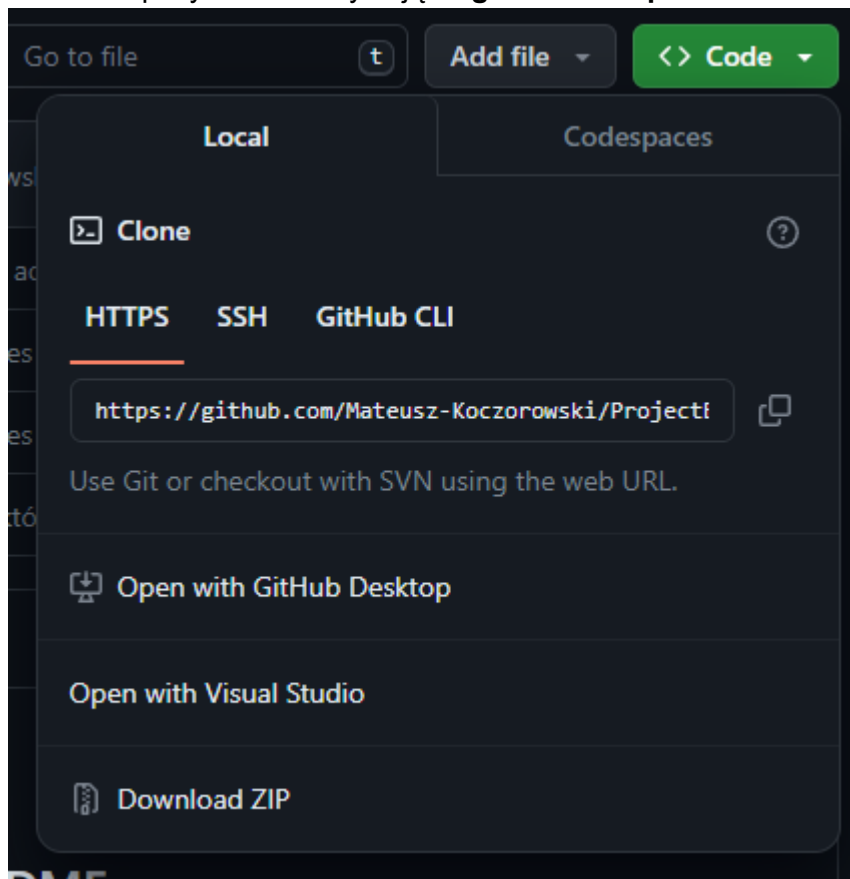
## Uruchomienie aplikacji

Aplikacja powinna zostać pobrana z platformy Github - <https://github.com/Mateusz-Koczorowski/ProjectBlogNews>

***Uwaga!!!***

***Najbardziej aktualny kod znajduje się na branchu master oraz develop  
Proszę używać .NET w wersji 6 oraz bibliotek w wersji 6!***

1. Pobierz repozytorium korzystając z **git clone** lub **pobierz archiwum ZIP**:



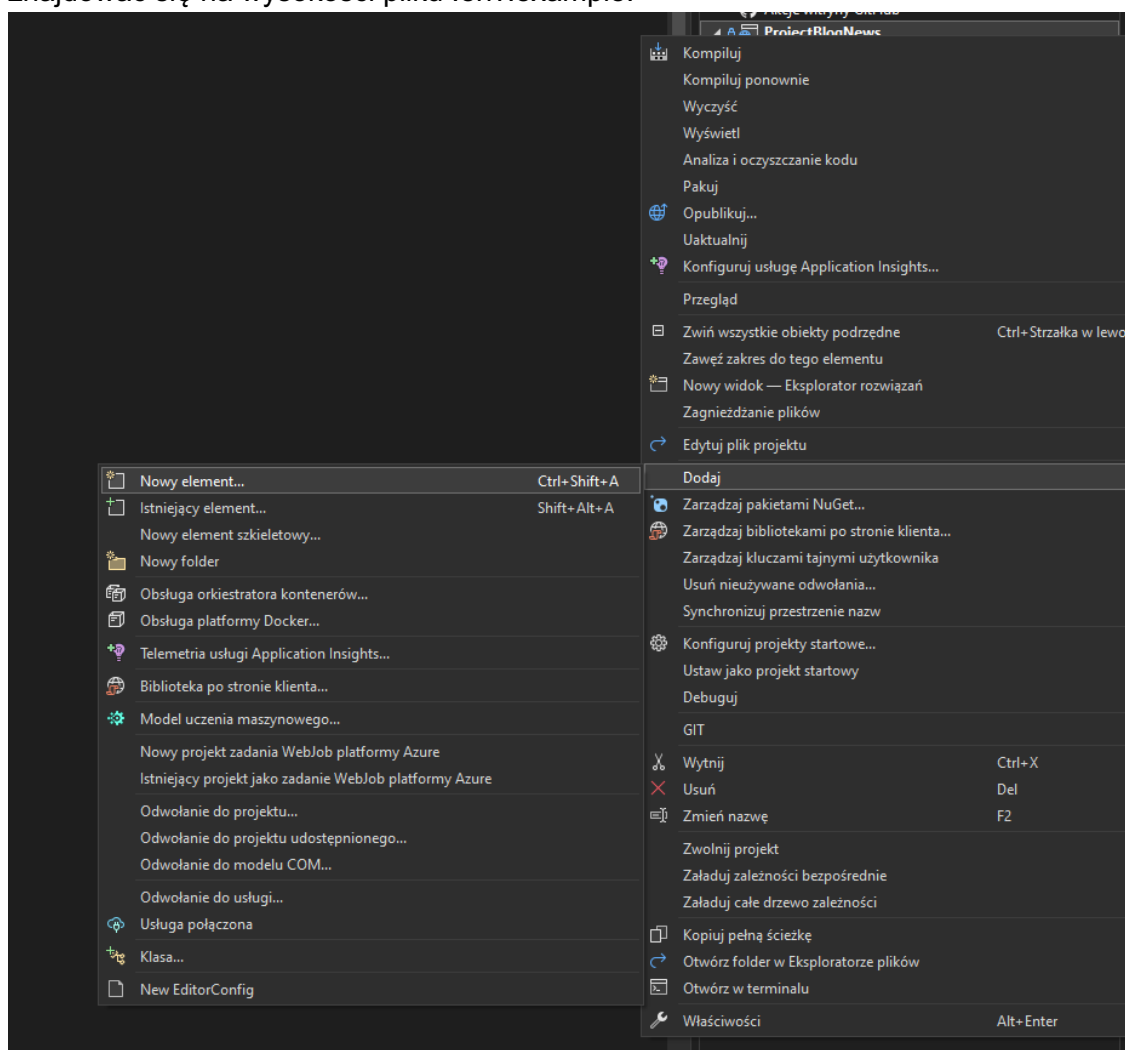
2. Po pobraniu kodu przejdź do folderu w którym znajduje się aplikacja, a następnie uruchom plik **ProjectBlogNews.sln**

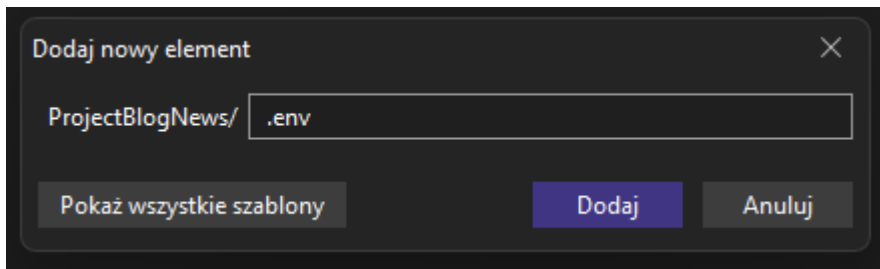
Name	Date modified	Type	Size
.github	03.01.2024 20:33	File folder	
ProjectBlogNews	07.01.2024 21:57	File folder	
.gitattributes	03.01.2024 20:33	GITATTRIBUTES File	3 KB
.gitignore	11.01.2024 20:32	GITIGNORE File	7 KB
ProjectBlogNews.sln	03.01.2024 18:42	Visual Studio Solu...	2 KB

**Wymagane jest posiadanie zainstalowanego programu Visual Studio oraz .NET w wersji 6.**

Struktura projektu powinna zostać uruchomiona w programie Visual Studio.

3. Utwórz plik .env (plik ze zmiennymi środowiskowymi - ze względów bezpieczeństwa zawartość tego pliku nie można być przechowywana w repozytorium). Plik powinien znajdować się na wysokości pliku .env.example.





4. Skopiuj sugerowaną zawartość pliku i wklej w utworzonym pliku .env.

***Uwaga!***

***Aplikacja nie zadziała bez utworzonego i wypełnionego pliku .env!!! Proszę uzupełnić wszystkie pola! Sugerujemy użyć wyżej wymienionej zawartości pliku!***

**Sugerowana zawartość pliku .env:**

*Admin\_Email=admin@admin.com*

*Admin\_Password=Admin123admin!*

*Admin\_FirstName=Admin*

*Admin\_LastName=Adminsky*

*Admin\_BirthDate=1970-01-01*

*Admin\_RoleName=Admin*

*Author\_Email=author@author.com*

*Author\_Password=Author123author!*

*Author\_FirstName=John*

*Author\_LastName=Smith*

*Author\_BirthDate=1970-01-01*

*Author\_RoleName=Author*

*Reader\_Email=reader@reader.com*

*Reader\_Password=Reader123reader!*

*Reader\_FirstName=Lukas*

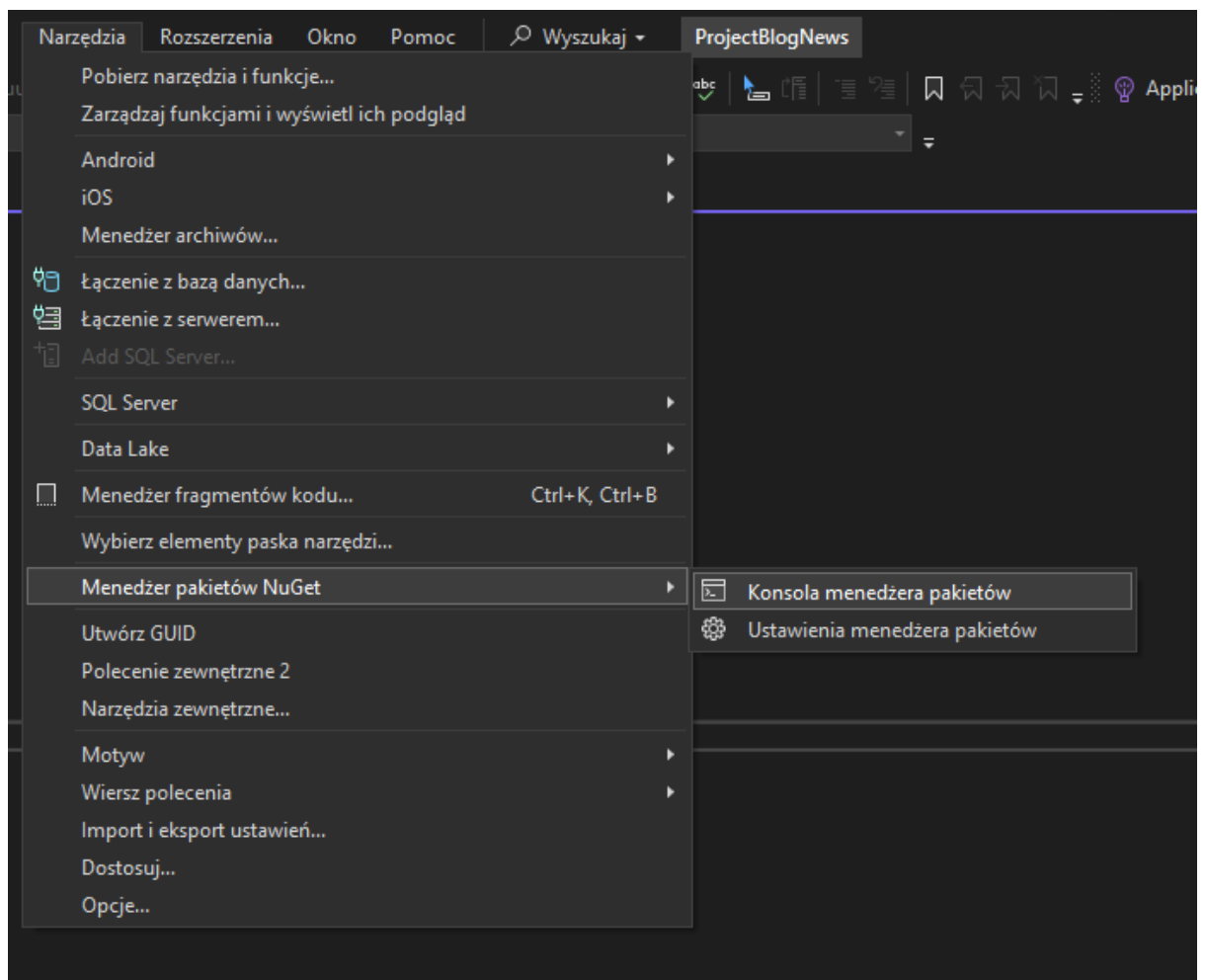
*Reader\_LastName=Kowalsky*

*Reader\_BirthDate=1970-01-01*

*Reader\_RoleName=Reader*

```
Program.cs .env ↵ ✕
1 Admin_Email=admin@admin.com
2 Admin_Password=Admin123admin!
3 Admin_FirstName=Admin
4 Admin_LastName=Admsky
5 Admin_BirthDate=1970-01-01
6 Admin_RoleName=Admin
7
8 Author_Email=author@author.com
9 Author_Password=Author123author!
10 Author_FirstName=John
11 Author_LastName=Smith
12 Author_BirthDate=1970-01-01
13 Author_RoleName=Author
14
15 Reader_Email=reader@reader.com
16 Reader_Password=Reader123reader!
17 Reader_FirstName=Lukas
18 Reader_LastName=Kowalsky
19 Reader_BirthDate=1970-01-01
20 Reader_RoleName=Reader
21
```

5. Następnie otwórz konsolę menedżera pakietów NuGet:



6. W konsoli wykonaj polecenia w celu wykonania migracji:

```
Install-Package Microsoft.AspNetCore.Diagnostics.EntityFrameworkCore  
Add-Migration CreateIdentitySchema  
Update-Database
```

7. Po wykonaniu tego kroku, uruchom aplikację przyciskiem Run:

