**Szpital**

**Autor – Kacper Błażykowski**

**Opis**

Projekt SzpitalWWW to aplikacja webowa służąca do zarządzania informacjami o szpitalu, w tym aktualnościami, godzinami otwarcia, informacjami o NFZ, oraz ofertą sklepu. Aplikacja umożliwia także użytkownikom przeglądanie historii i dodatkowych informacji związanych ze szpitalem.

**Specyfikacja wykorzystanych technologii**

* .NET 8.0
* ASP.NET Core Identity
* Entity Framework Core
* SQL Server
* jQuery

**Instrukcja pierwszego uruchomienia**

1. Otwórzyć Package Manager Console i wykonać polecenie Update-Database
2. Upewnienie się, że ustawienia w pliku appsettings.json są poprawne
3. Nacisnąć przyciska Start w pasku narzędzi

**Struktura katalogów**

**Controllers** – kontrolery MVC odpowiedzialne za obsługę żądań HTTP.

**Data** – kontekst bazy danych oraz modele Entity Framework.

**Models** – modele danych używane w widokach.

**Views** – widoki Razor Pages.

**wwwroot** – zasoby statyczne (CSS, JavaScript, obrazy).

**Manage** – widoki i logika odpowiedzialne za zarządzanie kontami użytkowników.

**Modele**

**Aktualnosc.cs**

Model reprezentujący aktualności na stronie.

IdAktualnosci: Klucz główny.

LinkTytul: Tytuł odnośnika do aktualności.

Tytul: Tytuł aktualności.

Tresc: Treść aktualności.

Pozycja: Pozycja aktualności na stronie.

**ErrorViewModel.cs**

Model używany do reprezentowania informacji o błędach.

RequestId: Identyfikator żądania.

ShowRequestId: Zwraca true, jeśli RequestId nie jest puste.

Finansowanie.cs

Model reprezentujący informacje o finansowaniu.

Id: Klucz główny.

Nazwa: Nazwa źródła finansowania.

Opis: Opis źródła finansowania.

**Godziny.cs**

Model reprezentujący godziny pracy.

Id: Klucz główny.

Dzien: Dzień tygodnia.

GodzinaOtwarcia: Godzina otwarcia.

GodzinaZamkniecia: Godzina zamknięcia.

**Informacje.cs**

Model reprezentujący informacje ogólne.

Id: Klucz główny.

Tytul: Tytuł informacji.

Opis: Opis informacji.

**Kontakt.cs**

Model reprezentujący dane kontaktowe.

Id: Klucz główny.

EmailAdress: Adres email.

Name: Nazwa kontaktu.

**Platnosci.cs**

Model reprezentujący płatności.

Id: Klucz główny.

TytulPlatnosci: Tytuł płatności.

imieNazw: Imię i nazwisko osoby płacącej.

Data: Data płatności.

Usluga: Rodzaj usługi.

**Prywatnie.cs**

Model reprezentujący prywatne rezerwacje.

Id: Klucz główny.

Imie: Imię osoby rezerwującej.

Nazwisko: Nazwisko osoby rezerwującej.

DataUrodzenia: Data urodzenia osoby rezerwującej.

NrUbezpieczenia: Numer ubezpieczenia osoby rezerwującej.

**Rejestracja.cs**

Model reprezentujący towary do rejestracji.

IdTowaru: Klucz główny.

Kod: Kod towaru.

Nazwa: Nazwa towaru.

Cena: Cena towaru.

FotoURL: URL zdjęcia towaru.

Opis: Opis towaru.

IdRodzaju: Klucz obcy do rodzaju towaru.

Rodzaj: Rodzaj towaru.

Data: Data rejestracji.

Miejscowosc: Miejscowość rejestracji.

nrubez: Numer ubezpieczenia.

**Rezerwacja.cs**

Model reprezentujący rezerwacje.

Id: Klucz główny.

IdRezerwacji: Identyfikator rezerwacji.

DestinationState: Stan docelowy rezerwacji.

Description: Opis rezerwacji.

phoneNumber: Numer telefonu osoby rezerwującej.

**Rodzaj.cs**

Model reprezentujący rodzaje towarów.

IdRodzaju: Klucz główny.

Nazwa: Nazwa rodzaju.

Opis: Opis rodzaju.

**Strona.cs**

Model reprezentujący strony w serwisie.

Id: Klucz główny.

Tytul: Tytuł strony.

Tresc: Treść strony.

Pozycja: Pozycja strony.

**Towar.cs**

Model reprezentujący towary.

Id: Klucz główny.

Kod: Kod towaru.

Nazwa: Nazwa towaru.

Cena: Cena towaru.

FotoURL: URL zdjęcia towaru.

Opis: Opis towaru.

IdRodzaju: Klucz obcy do rodzaju towaru.

Rodzaj: Rodzaj towaru.

**Wizyta.cs**

Model reprezentujący wizyty.

Id: Klucz główny.

DataWizyty: Data wizyty.

Lekarz: Lekarz przyjmujący wizytę.

Pacjent: Pacjent odwiedzający.

**Użytkownicy i role**

**Role w systemie**: Obecnie system wspiera standardowe role użytkowników zarządzane przez ASP.NET Identity.

**Nadanie roli użytkownikowi**: Role można przypisywać użytkownikom za pomocą narzędzi administracyjnych Identity.

**Możliwości użytkowników zalogowanych**: Użytkownicy zalogowani mają dostęp do funkcji dodawania rezerwacji.

**Informacje powiązane z użytkownikiem**: Rezerwacje są powiązane z zalogowanymi użytkownikami, a pozostałe informacje są globalne.

**Najciekawsze funkcjonalności**

**System rezerwacji wizyt:**

Użytkownicy mogą dodawać rezerwacje wizyt, które są przechowywane w bazie danych.

Formularz rejestracji wizyt weryfikuje dane użytkownika przed ich zapisaniem.

**Kontrolery** (nie wiedziałem czy to tak w pełni być, dlatego jest na końcu, bo długie wyszło)

**AktualnoscController.cs**

Index

Metoda HTTP: GET

Parametry: id (int)

Opis: Wyświetla szczegóły wybranej aktualności oraz pobiera listę stron i najnowszych aktualności.

Zwracane dane: Widok ze szczegółami aktualności oraz listą stron i najnowszych aktualności.

**HistoryController.cs**

Index

Metoda HTTP: GET

Parametry: id (int)

Opis: Wyświetla szczegóły strony historycznej oraz pobiera listę stron i najnowszych aktualności.

Zwracane dane: Widok z szczegółami strony historycznej oraz listą stron i najnowszych aktualności.

Kontakt

Metoda HTTP: GET

Parametry: brak

Opis: Wyświetla stronę kontaktową.

Zwracane dane: Widok strony kontaktowej.

Dodaj

Metoda HTTP: GET

Parametry: brak

Opis: Wyświetla formularz dodawania nowego wpisu historycznego.

Zwracane dane: Widok formularza.

OFirmie

Metoda HTTP: GET

Parametry: brak

Opis: Wyświetla stronę "O firmie".

Zwracane dane: Widok strony "O firmie".

Privacy

Metoda HTTP: GET

Parametry: brak

Opis: Wyświetla stronę polityki prywatności.

Zwracane dane: Widok strony polityki prywatności.

Error

Metoda HTTP: GET

Parametry: brak

Opis: Wyświetla stronę błędu.

Zwracane dane: Widok strony błędu.

**HomeController.cs**

Index

Metoda HTTP: GET

Parametry: id (int) opcjonalny

Opis: Wyświetla stronę główną.

Zwracane dane: Widok strony głównej.

Kontakt

Metoda HTTP: GET

Parametry: brak

Opis: Wyświetla stronę kontaktową.

Zwracane dane: Widok strony kontaktowej.

Dodaj

Metoda HTTP: POST (tylko dla zalogowanych)

Parametry: Rezerwacja (model)

Opis: Dodaje nową rezerwację.

Zwracane dane: Widok potwierdzenia dodania rezerwacji.

OFirmie

Metoda HTTP: GET

Parametry: brak

Opis: Wyświetla stronę "O firmie".

Zwracane dane: Widok strony "O firmie".

Privacy

Metoda HTTP: GET

Parametry: brak

Opis: Wyświetla stronę polityki prywatności.

Zwracane dane: Widok strony polityki prywatności.

Error

Metoda HTTP: GET

Parametry: brak

Opis: Wyświetla stronę błędu.

Zwracane dane: Widok strony błędu.

**HourController.cs**

Index

Metoda HTTP: GET

Parametry: id (int)

Opis: Wyświetla szczegóły strony z godzinami pracy oraz pobiera listę stron i najnowszych aktualności.

Zwracane dane: Widok z szczegółami strony z godzinami pracy oraz listą stron i najnowszych aktualności.

Kontakt

Metoda HTTP: GET

Parametry: brak

Opis: Wyświetla stronę kontaktową.

Zwracane dane: Widok strony kontaktowej.

Dodaj

Metoda HTTP: GET

Parametry: brak

Opis: Wyświetla formularz dodawania nowego wpisu dotyczącego godzin pracy.

Zwracane dane: Widok formularza.

OFirmie

Metoda HTTP: GET

Parametry: brak

Opis: Wyświetla stronę "O firmie".

Zwracane dane: Widok strony "O firmie".

Privacy

Metoda HTTP: GET

Parametry: brak

Opis: Wyświetla stronę polityki prywatności.

Zwracane dane: Widok strony polityki prywatności.

Error

Metoda HTTP: GET

Parametry: brak

Opis: Wyświetla stronę błędu.

Zwracane dane: Widok strony błędu

**InfoController.cs**

Index

Metoda HTTP: GET

Parametry: brak

Opis: Wyświetla stronę z ogólnymi informacjami.

Zwracane dane: Widok strony z ogólnymi informacjami.

RezerwcjaPrywatnie

Metoda HTTP: GET

Parametry: brak

Opis: Wyświetla stronę rezerwacji prywatnych.

Zwracane dane: Widok strony rezerwacji prywatnych.

Kontakt

Metoda HTTP: GET

Parametry: brak

Opis: Wyświetla stronę kontaktową.

Zwracane dane: Widok strony kontaktowej.

Dodaj

Metoda HTTP: GET

Parametry: brak

Opis: Wyświetla formularz dodawania nowego wpisu informacyjnego.

Zwracane dane: Widok formularza.

OFirmie

Metoda HTTP: GET

Parametry: brak

Opis: Wyświetla stronę "O firmie".

Zwracane dane: Widok strony "O firmie".

Privacy

Metoda HTTP: GET

Parametry: brak

Opis: Wyświetla stronę polityki prywatności.

Zwracane dane: Widok strony polityki prywatności.

Error

Metoda HTTP: GET

Parametry: brak

Opis: Wyświetla stronę błędu.

Zwracane dane: Widok strony błędu.

**InformationController.cs**

Index

Metoda HTTP: GET

Parametry: id (int)

Opis: Wyświetla szczegóły strony informacyjnej oraz pobiera listę stron i najnowszych aktualności.

Zwracane dane: Widok z szczegółami strony informacyjnej oraz listą stron i najnowszych aktualności.

Kontakt

Metoda HTTP: GET

Parametry: brak

Opis: Wyświetla stronę kontaktową.

Zwracane dane: Widok strony kontaktowej.

Dodaj

Metoda HTTP: GET

Parametry: brak

Opis: Wyświetla formularz dodawania nowego wpisu informacyjnego.

Zwracane dane: Widok formularza.

OFirmie

Metoda HTTP: GET

Parametry: brak

Opis: Wyświetla stronę "O firmie".

Zwracane dane: Widok strony "O firmie".

Privacy

Metoda HTTP: GET

Parametry: brak

Opis: Wyświetla stronę polityki prywatności.

Zwracane dane: Widok strony polityki prywatności.

Error

Metoda HTTP: GET

Parametry: brak

Opis: Wyświetla stronę błędu.

Zwracane dane: Widok strony błędu.

**NFZController.cs**

Index

Metoda HTTP: GET

Parametry: id (int)

Opis: Wyświetla szczegóły strony związanej z NFZ oraz pobiera listę stron i najnowszych aktualności.

Zwracane dane: Widok z szczegółami strony związanej z NFZ oraz listą stron i najnowszych aktualności.

Kontakt

Metoda HTTP: GET

Parametry: brak

Opis: Wyświetla stronę kontaktową.

Zwracane dane: Widok strony kontaktowej.

Dodaj

Metoda HTTP: GET

Parametry: brak

Opis: Wyświetla formularz dodawania nowego wpisu dotyczącego NFZ.

Zwracane dane: Widok formularza.

OFirmie

Metoda HTTP: GET

Parametry: brak

Opis: Wyświetla stronę "O firmie".

Zwracane dane: Widok strony "O firmie".

Privacy

Metoda HTTP: GET

Parametry: brak

Opis: Wyświetla stronę polityki prywatności.

Zwracane dane: Widok strony polityki prywatności.

Error

Metoda HTTP: GET

Parametry: brak

Opis: Wyświetla stronę błędu.

Zwracane dane: Widok strony błędu.