**Dokumentacja techniczna**

**Tytuł projektu**: “MEDBYTE”

**Krotki opis działania projetku**

MedByte to platforma do tworzenia ogłoszeń, gdzie użytkownicy mogą zamieszczać swoje tomografy oraz rezerwować wizyty na badania

**Autorzy projetku**

* Szymon Pikulik,
* Kacper Pawłowski

**Specyfikacja technologii**

Jezyk programowania – C#, html, css

**Lista użytch pakietow NuGet**:

Microsoft.AspNetCore.Diagnostics.EntityFrameworkCore (8.0.3)

Microsoft.AspNetCore.Identity.EntityFrameworkCore(8.0.3)

Microsoft.AspNetCore.Identity.UI(8.0.3)

Microsoft.EntityFrameworkCore.SqlServer(8.0.3)

Microsoft.EntityFrameworkCore.Tools(8.0.7)

Microsoft.VisualStudio.Web.CodeGeneration.Design(8.0.3)

**Instrukcja uruchomienia projektu**

1.Instalujemy wymagane narzedzia z biblioteki z pomocą komendy

dotnet restore

2.Konfigurujemy nasza baze danych wykonując migracje:

update-database

3.Uruchamiamy nasza aplikacje za pomocą przycisku:

F5

Uzytkownicy

Admin:

Login: [admin@medbyte.com](mailto:admin@medbyte.com)

Haslo:Admin@123

Uzytkownik:

Login:user@medbyte.com

Haslo:User@123

Konto zbanowane:

Login:banme@medbyte.com

Haslo:Ban@123

**Modele**

Modele aplikacji znajdują się w katalogu Models. Poniżej znajduje się lista modeli wraz z ich krótkim opisem oraz polami:

Tomograf- model reprezentujący tomograf.

* Id- Identyfikator tomografu, wymagana jest nazwa
* Nazwa - Nazwa tomografu,
* Opis - Opis tomografu,
* Telefon - Numer telefonu, wymagany jest numer telefonu
* Ulica - Ulica, wymagana jest ulica
* Miasto- Miasto, miasto jest wymagane
* KodPocztowy - Kod pocztowy, kod pocztowy jest wymagany w formacie 12-123 gdy format jest nie poprawny wyskoczy bład
* ImagePath - Ścieżka do obrazu,
* UserId - Identyfikator użytkownika.

Rezerwacja - model reprezentujący rezerwację.

* Id- Identyfikator rezerwacji,
* Data - Data rezerwacji, date musi zostać wybrana
* TomografId - Identyfikator tomografu,
* Tomograf - Tomograf,
* UserId - Identyfikator użytkownika,
* Imie - Imię użytkownika, wymagane jest imie
* Nazwisko - Nazwisko użytkownika, wymagane jest nazwisko
* Pesel - Numer PESEL użytkownika. Wymagany jest pesel o długości 11 znakow

ErrorViewModel- model reprezentujący błędy.

* RequestId- Identyfikator żądania,
* ShowRequestId - Flaga wskazująca, czy pokazać identyfikator żądania.

**Kontrolery**

Kontrolery aplikacji znajdują się w katalogu Controllers. Poniżej znajduje się lista kontrolerów wraz z metodami:

**AdminController**

**Metody:**

* Index() - Wyświetla stronę główną panelu administratora. [GET]
* DeleteUser(string id) - Usuwa użytkownika o podanym ID. [POST]
* BanUser(string id) - Banuje użytkownika o podanym ID. [POST]
* UnbanUser(string id) - Odbanowuje użytkownika o podanym ID. [POST]
* DeleteTomograf(int id) - Usuwa tomograf o podanym ID. [POST]

**HomeController**

**Metody:**

* Index() - Wyświetla stronę główną aplikacji. [GET]
* Error() - Wyświetla stronę błędu. [GET]

**RezerwacjaController**

**Metody:**

* Create(int tomografId, DateTime data, string imie, string nazwisko, string pesel) - Tworzy nową rezerwację. [POST]
* MojeTerminy() - Wyświetla listę rezerwacji użytkownika. [GET]
* Odwolaj(int id) - Odwołuje rezerwację o podanym ID. [POST]

**TomografyController**

**Metody:**

* Index() - Wyświetla listę wszystkich tomografów. [GET]
* Details(int? id) - Wyświetla szczegóły wybranego tomografu. [GET]
* Create() - Wyświetla formularz dodawania nowego tomografu. [GET]
* Create([Bind("Id,Nazwa,Opis,Telefon,Ulica,Miasto,KodPocztowy")] Tomograf tomograf, IFormFile imageFile) - Dodaje nowy tomograf do bazy danych. [POST]
* Edit(int? id) - Wyświetla formularz edycji tomografu. [GET]
* Edit(int id, [Bind("Id,Nazwa,Opis,Telefon,Ulica,Miasto,KodPocztowy")] Tomograf tomograf, IFormFile imageFile) - Aktualizuje dane tomografu w bazie danych. [POST]
* Delete(int? id) - Wyświetla formularz usunięcia tomografu. [GET]
* DeleteConfirmed(int id) - Usuwa wybrany tomograf z bazy danych. [POST]
* GetTomografy() - Zwraca listę tomografów w formacie JSON. [GET]

**Widoki**

Poniżej znajduje się lista widoków aplikacji:

**Admin**

* Index.cshtml

**Home**

* Index.cshtml
* Privacy.cshtml

**Rezerwacja**

* MojeTerminy.cshtml

**Shared**

* Layout.cshtml
* LoginPartial.cshtml
* ValidationScriptsPartial.cshtml
* Error.cshtml

**Tomografy**

* ViewImports.cshtml
* ViewStart.cshtml

**O Projekcie**

Nie zalogowany użytkownik może swobodnie przegladac strone jak i sprawdzać szczegóły danych tomografow nie może on nic dodac usunąć czy zarejestrować się na wyzyte

Uzytkownik przegladac strone sprawdzać szczegóły tomografow jak i je dodawać, edytowac czy rejestrować się na wizyte. Może tez rezygnować z wizyt danego dnia jak i usuwac swoje dodane tomografy nie moze on usuwac tomografow czy wizyt innych uzytkownikow.

Admin może on robic wszystko co zwykły użytkownik wraz z mozliwoscia usuania tomografow innych uzytkownikow. Posiada tez on opcje banowania innych uzytkownikow jak i ich odbanowywania, zbanowany użytkownik po zalogowaniu dostanie informacja o ograniczeniu jego konta