

AI	Dokumentacja projektu
Autor	Michał Żychowski, 123639
Kierunek, rok	Informatyka i ekonometria, II rok, st. stacjonarne (3,5-l)
Temat projektu	<i>Przychodnia dentystyczna</i>

Spis treści

Wstęp	2
Narzędzia i technologie	2
Baza danych	3
Opis	3
Relacje	4
Migracje.....	5
Seeder	6
Projekt GUI	7
Uruchamianie aplikacji	7
Funkcjonalność Aplikacji	9
Strona główna	9
Informacje o przychodni – zakładka „O nas”	9
Przeglądanie cen.....	10
Umawianie wizyty	10
Logowanie	11
Panel lekarz	11
Panel administratora – dodawanie nowego lekarza, edytowanie danych	12

Wstęp

Aplikacja Przychodni dentystycznej powstała w celu ułatwieniu pracy lekarzom oraz usprawnienia możliwości zapisu na wizytę dla pacjentów.

Widzimy, że osoba chcąca zapisać się na wizytę, może łatwo to zrobić wpisując swoje dane do formularza oraz może od razu wybrać jakie leczenie będzie na wizycie, a lekarz może zaakceptować bądź odrzucić wizytę.

Głównym celem tej aplikacji jest ułatwienie pracy lekarzowi i uproszczenie możliwości zapisu na wizyty.

Narzędzia i technologie

W ramach tego projektu używam następujących narzędzi, technologii, języków programowania, baz danych i frameworków:

- PHP to popularny język programowania używany do tworzenia dynamicznych stron internetowych oraz aplikacji webowych. Jest łatwy w użyciu i obsługuje wiele baz danych i frameworków.
- Laravel (wersja 10.0): jest frameworkiem PHP, który dostarcza zestaw narzędzi i strukturę umożliwiającą szybkie tworzenie aplikacji internetowych. Laravel jest udostępniany na zasadach licencji MIT, co oznacza, że jest bezpłatny do użytku i posiada otwarty kod źródłowy.
- MySQL to popularny system zarządzania relacyjnymi bazami danych (RDBMS), który zapewnia skuteczną i niezawodną obsługę przechowywania, zarządzania i odpytywania danych. Jest szeroko stosowany w aplikacjach internetowych i biznesowych ze względu na swoją wydajność, skalowalność i łatwość użycia.
- Bootstrap (wersja 5.3) popularny framework front-endowy, który umożliwia tworzenie responsywnych i atrakcyjnych interfejsów użytkownika dla stron internetowych i aplikacji. Jest oparty na HTML, CSS i JavaScript i zapewnia zestaw gotowych komponentów, stylów i narzędzi, które ułatwiają projektowanie i rozwijanie interfejsów.
- Visual Studio Code: jest popularnym środowiskiem programistycznym (IDE) przeznaczonym do tworzenia aplikacji webowych. Aby pobrać Visual Studio Code, należy odwiedzić stronę code.visualstudio.com. To oprogramowanie jest udostępniane za darmo i oferuje wsparcie dla wielu języków programowania.

Baza danych

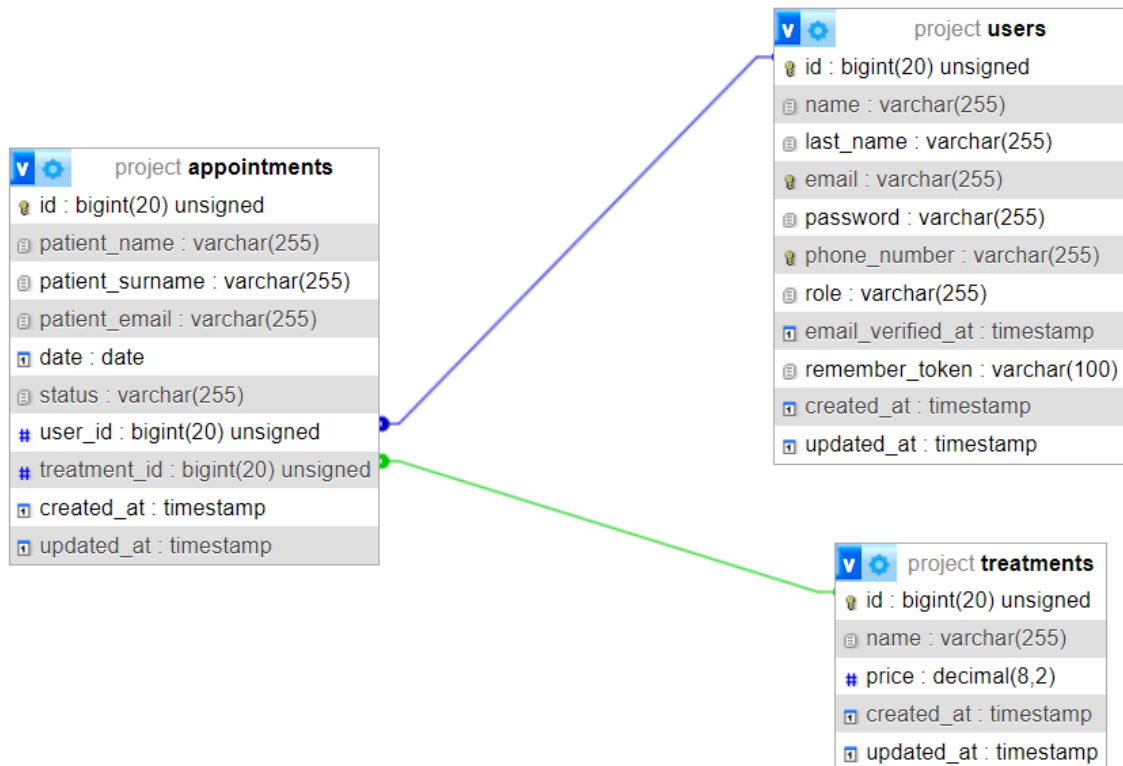


Diagram ERD bazy danych

Opis

Baza danych została zaprojektowana dla przychodni dentystycznej. Głównym zadaniem bazy jest przechowywanie danych na temat lekarzy, zabiegów i wizyt.

Tabela „treatments” zawiera wszystkie dane dotyczące zabiegu lekarskiego jaki ma być przeprowadzony na wizycie. Tabela zawiera nazwę zabiegu oraz jego cenę.

Tabela treatments zawiera kolumny:

- id - unikalny identyfikator zabiegu,
- name – nazwa zabiegu
- price – cena zabiegu
- created_at – data utworzenia w bazie nowego zabiegu
- updated_at – aktualizowanie danych

Tabela „users” jest tabelą lekarzy. Informacja taka jak email jest niezbędna do tego by lekarz mógł się zalogować i sprawdzić zgłoszone wizyty i ewentualnie je usunąć lub zaakceptować. Tabela posiada również dane: imię, nazwisko, numer telefonu i rola.

Tabela users zawiera kolumny:

- id – unikalny identyfikator lekarza,
- name – imię lekarza,
- last_name – nazwisko lekarza,
- email – email lekarza,
- password – hasło,
- phone_number – numer telefonu lekarza,
- role – rola użytkownika (lekarz albo admin),
- created_at – data utworzenia w bazie nowego lekarza,
- updated_at – aktualizowanie danych

Tabela „appointments” zawiera danę na temat wizyty. Dane te to imię i nazwisko pacjenta oraz jego email. Zawiera również ID lekarza, który ma przeprowadzić wizytę i ID zabiegu, który ma być wykonany oraz datę wizyty.

Tabela appointments zawiera kolumny:

- id - unikalny identyfikator wizyty,
- patient_name – imię pacjenta,
- patient_surname – nazwisko pacjenta,
- patient_email – email pacjenta,
- date – data,
- status – status wizyty (oczekująca, zaakceptowana, bądź odrzucona),
- user_id – identyfikator lekarza (klucz obcy, który odnosi się do tabeli users),
- treatment_id – identyfikator zabiegu (klucz obcy, który odnosi się do tabeli treatments),
- created_at – data utworzenia w bazie nowej wizyty
- updated_at – aktualizowanie danych

Relacje

Wszystkie relacje w modelach są „One to Many”, podam przykład relacji appointments z treatments: ta relacja mówi nam, że na konkretnej wizycie może mieć miejsce tylko jeden zabieg. To wszystko dzięki funkcji, która używa belongsTo.

```
public function treatment()  
{  
    return $this->belongsTo(Treatment::class);  
}
```

Migracje

Migracje są wykorzystywane do zarządzania strukturą bazy danych, umożliwiając tworzenie, modyfikację lub usuwanie tabel i kolumn. Na przykład, możemy użyć migracji do zarządzania tabelą o nazwie "treatments".

```
1  <?php      You, 2 days ago • Project backend finished ...
2
3  use Illuminate\Database\Migrations\Migration;
4  use Illuminate\Database\Schema\Blueprint;
5  use Illuminate\Support\Facades\Schema;
6
7  You, 2 days ago | 1 author (You)
8  return new class extends Migration
9  {
10     /**
11      * Run the migrations.
12      */
13     public function up(): void
14     {
15         Schema::create('treatments', function (Blueprint $table) {
16             $table->id();
17             $table->string('name');
18             $table->decimal('price');
19             $table->timestamps();
20         });
21     }
22
23     /**
24      * Reverse the migrations.
25      */
26     public function down(): void
27     {
28         Schema::dropIfExists('treatments');
29     }
30 };
```

W tym przypadku migracja tworzy tabelę „treatments” za pomocą Schema::create(). Tabela będzie zawierać kolumny takie jak: id: unikalne pole identyfikatora; name: pole typu string, przechowuje nazwę leczenia; price: pole typu decimal, przechowuje cenę leczenia; timestamps(): automatycznie generowane pola „created_at” oraz „updated_at”, które przechowują datę utworzenia i aktualizowania danego rekordu.

Seeder

W kodzie są również zaimplementowane seedery które służą do wygenerowania startowych danych. Aplikacja posiada DatabaseSeeder, w którym tworzone są poszczególne rekordy dla danych tabel.

```
0 references | 0 implementations | 1 row, 1 second ago | 1 author (row)
class DatabaseSeeder extends Seeder
{
    /**
     * Seed the application's database.
     */
    0 references | 0 overrides
    public function run()
    {
        // Tworzenie dwóch lekarzy
        $doctor1 = User::create([
            'name' => 'John',
            'last_name' => 'Doe',
            'email' => 'john@example.com',
            'password' => bcrypt('password'),
            'phone_number' => '123456789',
            'role' => 'doctor',
        ]);

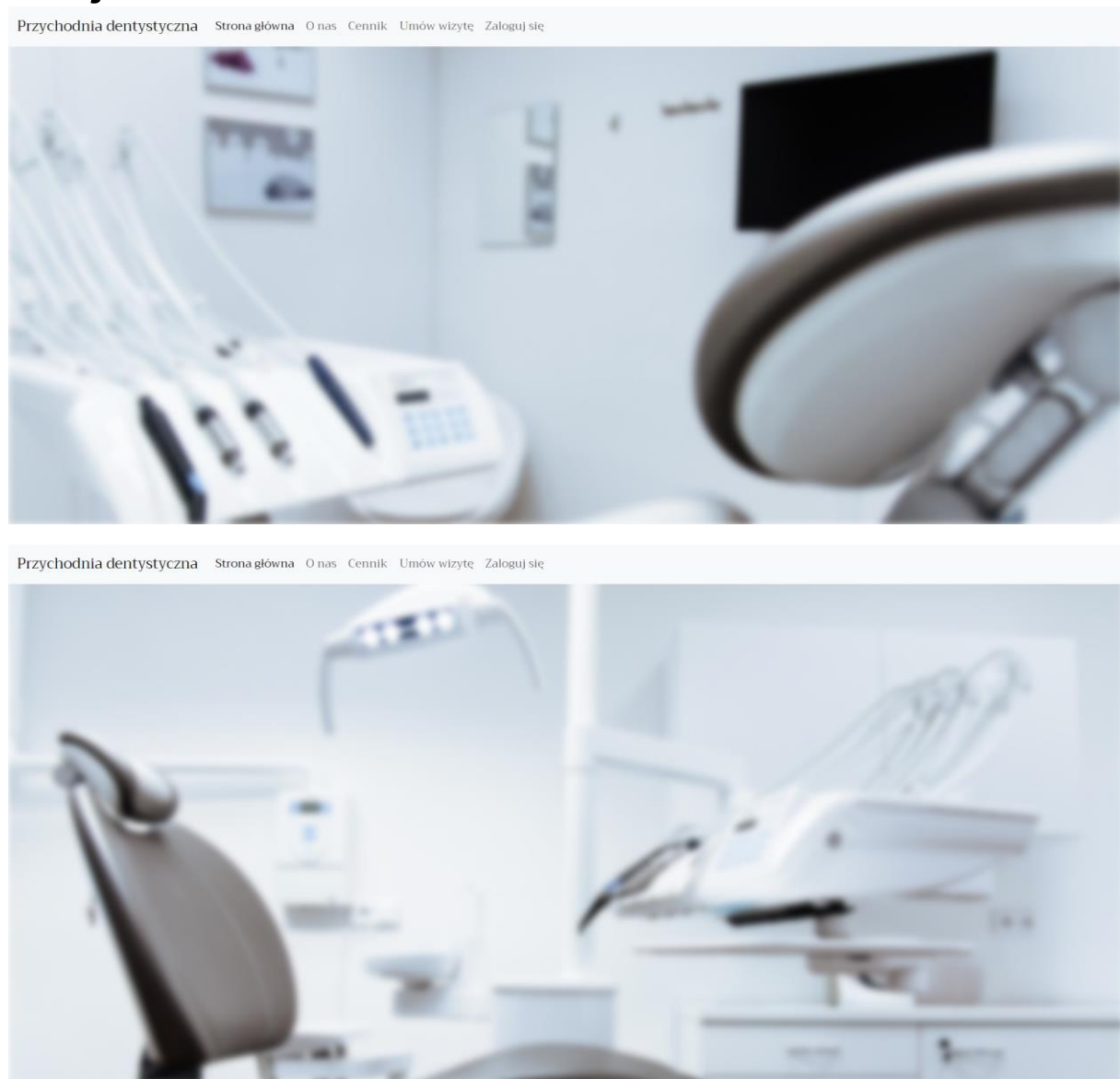
        $admin = User::create([
            'name' => 'Admin',
            'last_name' => 'Smith',
            'email' => 'admin@example.com',
            'password' => bcrypt('password'),
            'phone_number' => '987654321',
            'role' => 'admin',
        ]);

        // Tworzenie zabiegów
        $treatment1 = Treatment::create([
            'name' => 'Badania lekarskie + raport o stanie zębów',
            'price' => 50.00,
        ]);

        $treatment2 = Treatment::create([
            'name' => 'Wizyta adaptacyjna dziecka do 5 lat',
        ]);
    }
}
```

Jak widzimy tworzeni są początkowo dwaj lekarze (w tym admin). Następnie tworzone są zabiegi. Kolejny krok to utworzenie dwóch zabiegów do danej kliniki. Ostatnią czynnością jest utworzenie dwóch wizyt (po jednej dla każdego lekarza) do pierwszych dwóch zabiegów.

Projekt GUI



Uruchamianie aplikacji

Aby uruchomić aplikację należy wykonać następujące czynności:

- Krok 1: Wymagania systemowe

Przed rozpoczęciem upewnij się, że Twój system spełnia minimalne wymagania Laravela. Sprawdź dokumentację Laravela, aby uzyskać aktualne informacje na temat wymagań.

- Krok 2: Środowisko programistyczne

Upewnij się, że masz zainstalowane środowisko programistyczne, takie jak XAMPP, WAMP lub MAMP, w zależności od Twojego systemu operacyjnego. Te narzędzia zapewniają serwer Apache, bazę danych MySQL oraz interpreter PHP.

- Krok 3: Instalacja Composer

Przejdź na stronę <https://getcomposer.org/> i postępuj zgodnie z instrukcjami instalacji Composer na swoim systemie. Composer jest menedżerem pakietów PHP i jest niezbędny do zarządzania zależnościami Laravela.

- Krok 4: Pobierz Laravela

Możesz pobrać najnowszą wersję Laravela z oficjalnej strony internetowej Laravela (<https://laravel.com/>). Kliknij przycisk "Download" i zapisz plik zip na swoim komputerze.

- Krok 5: Rozpakuj Laravela

Po pobraniu pliku zip Laravela rozpakuj go w wybranym miejscu na swoim komputerze.

- Krok 6: Otwórz terminal lub wiersz poleceń

Przejdź do katalogu, w którym rozpakowałeś Laravela, za pomocą terminala lub wiersza poleceń.

- Krok 7: Zainstaluj zależności Laravela

Wpisz polecenie `composer install` w terminalu lub wierszu poleceń, będąc w katalogu Laravela. Composer pobierze i zainstaluje wszystkie zależności frameworka.

- Krok 8: Skonfiguruj plik `.env`

Skopiuj plik `.env.example` i nazwij go jako `.env`. Otwórz plik `.env` i ustaw odpowiednie konfiguracje, takie jak połączenie do bazy danych i inne ustawienia aplikacji.

- Krok 9: Uruchom serwer

Aby uruchomić serwer developerski Laravela, wpisz polecenie `php artisan serve` w terminalu lub wierszu poleceń. Serwer będzie dostępny pod adresem `http://localhost:8000`.

- Krok 10: Sprawdź działanie aplikacji

Otwórz przeglądarkę internetową i przejdź pod adres `http://localhost:8000`. Powinieneś zobaczyć stronę powitalną Laravela, co oznacza, że Twoja aplikacja została pomyślnie uruchomiona.

Po wykonaniu tych wszystkich kroków będzie można swobodnie poruszać się i korzystać ze strony internetowej.

Funkcjonalność Aplikacji

Strona główna



Informacje o przychodni – zakładka „O nas”

W ramach zakładki "O nas" dostępnej na naszej stronie internetowej, klient ma możliwość zapoznania się z krótkim opisem przychodni. Ta sekcja zawiera informacje, które pozwalają klientowi lepiej zrozumieć, kim jesteśmy i czym się zajmujemy..



Przeglądanie cen

W ramach zakładki "Cennik" dostępnej na naszej stronie internetowej, klient ma możliwość przeglądania cen leczenia oraz oferty zabiegów. Ta sekcja zawiera szczegółowe informacje dotyczące kosztów związanych z różnymi usługami medycznymi i zabiegami oferowanymi w naszej przychodni..

Przychodnia dentystyczna Strona główna O nas Cennik Umów wizytę Zaloguj się

Nazwa	Cena
Badania lekarskie + raport o stanie zębów	50.00
Wizyta adaptacyjna dziecka do 5 lat	110.00
Wizyta kontrolna po 6 m-cach	50.00
Konsultacja lekarza specjalisty	100.00
Wypełnienie w zębie stałym	250.00
Wypełnienie w zębie mlecznym	190.00
Odbudowa zęba stałego	400.00
Opatrunek	150.00
Licówka kompozytowa	900.00
Dewitalizacja (zastrucie) zęba	170.00
Leczenie kanałowe zęba 1 kanałowego jednowizytowe	900.00
Leczenie kanałowe zęba 2 kanałowego jednowizytowe	1000.00
Leczenie kanałowe zęba 3 kanałowego jednowizytowe	1100.00
Leczenie kanałowe zęba 4 kanałowego jednowizytowe	1200.00
Dopłata do leczenia u specjalisty	300.00
Reendo zęba 1 kanałowego	1000.00

Umawianie wizyty

W zakładce "Umów wizytę" dostępnej na naszej stronie internetowej, klient ma możliwość wygodnego i szybkiego umówienia się na wizytę u konkretnego lekarza oraz na określony zabieg. Dzięki prostemu i intuicyjnemu formularzowi, klient może łatwo uzupełnić wymagane dane i zarezerwować dogodny termin.

Przychodnia dentystyczna Strona główna O nas Cennik Umów wizytę Zaloguj się

Utwórz nową wizytę

Imię pacjenta

Nazwisko pacjenta

Email pacjenta

Data wizyty

Lekarz

Zabieg

Logowanie

Zalogować może się jedynie lekarz (lub admin, który też jest lekarzem). Po kliknięciu w zakładkę „Zaloguj się” przechodzi do panelu logowania, gdzie może się zalogować używając swojego email oraz hasła.

Przychodnia dentystyczna Strona główna O nas Cennik Umów wizytę Zaloguj się

Logowanie

Email
admin@example.com

Hasło

Zaloguj się

Powrót

Panel lekarz

W panelu lekarza, po zalogowaniu się do systemu, lekarz ma dostęp do specjalnej tabeli, która wyświetla planowane wizyty. Ta tabela zawiera informacje dotyczące pacjentów, daty i godziny wizyty oraz rodzaju planowanego zabiegu.

Przychodnia dentystyczna Strona główna O nas Cennik Wyloguj się

Witaj, John

Wizyty

Imię	Nazwisko	Email	Data	Status	Zabieg	Akcje
Pavel	Stano	stano@gmail.com	2023-06-20	confirmed	Wizyta adaptacyjna dziecka do 5 lat	Usuń
Anna	Kowalska	a.kowalska@pacjent.com	2023-07-02	pending	Reendo zęba 3 kanałowego	Zaakceptuj Odrzuć
Karol	Król	kkrol@pacjent.com	2023-08-05	rejected	Konsultacja lekarza specjalisty	Usuń

Jeśli lekarz naciśnie przycisk „Wyloguj się” wyloguje się. Po tym nastąpi automatyczne przejście do strony głównej.

Panel administratora – dodawanie nowego lekarza, edytowanie danych

Administrator również jest lekarzem, więc posiada te same funkcjonalności, lecz dodatkowo widzi tabelę z lekarzami, gdzie może edytować ich dane lub usunąć lekarza.

Przychodnia dentystyczna Strona główna O nas Cennik Wyloguj się

Witaj, Admin

Wizyty

Imię	Nazwisko	Email	Data	Status	Zabieg	Akcje
Pavel	Doe	doe12@gmail.com	2023-06-21	confirmed	Badania lekarskie + raport o stanie zębów	<button>Usuń</button>

Zarządzaj lekarzami

Imię	Nazwisko	Email	Nr_telefonu	Edytuj	Usuń
John	Doe	john@example.com	123456789	<button>Edytuj</button>	<button>Usuń</button>

Dodaj nowego lekarza do listy

Po kliknięciu przycisku „Edytuj” przy danym lekarzu administrator przechodzi do widoku edycji, gdzie może zmienić dane lekarza.

Przychodnia dentystyczna Strona główna O nas Cennik Wyloguj się

Edytuj dane doktora John

Imię

Nazwisko

Email

Nr telefonu

Zapisz

Po kliknięciu przycisku „Dodaj nowego lekarza do listy” administrator przechodzi do widoku dodawania, gdzie może dodać nowego lekarza wpisując odpowiednie dane.

Utwórz doktora

Imię

Nazwisko

Email

Nr telefonu

Hasło

Zapisz

Jeśli administrator naciśnie przycisk „Wyloguj się” wyloguje się. Po kliknięciu przycisku, administrator zostanie automatycznie wylogowany i przeniesiony do strony głównej.