

Dokumentacja projektu		AI1
Autor	Przemysław Pisaniak, 117828	Data oddania 21.06.2022
Kierunek, rok	Informatyka, II rok, st. stacjonarne (3,5-l)	
Specjalizacja	–	
Grupa	LAB 3	Ocena
Temat projektu	<i>Biuro podróży oferujące wycieczki</i>	

Tematyka projektu

Tematem projektu jest aplikacja internetowa biura podróży. Aplikacja ma za zadanie umożliwić użytkownikowi zarezerwowania sobie dowolnej wycieczki w odpowiednim terminie oraz rezygnację z wycieczki. Projekt miał działać podobnie do istniejących aplikacji biur podróży z tą różnicą, że użytkownik rezerwował sobie wyjazd wakacyjny, nie pobyt w hotelu. Użytkownik ma możliwość przeglądania dostępnych wycieczek, przeszukiwania wycieczek w danym okresie czasu, rezerwacje wycieczek dostępnych, anulowanie swojej wycieczki oraz ogólny przegląd wycieczek kiedykolwiek zarezerwowanych.

Opis technologii

Projekt został zrealizowany w kilku technologiach backendowych oraz frontendowych.

a) Język programowania PHP 8.1 (<https://www.php.net/releases/8.1/en.php>)

b) Framework Laravel 9.17.0 (<https://laravel.com/docs/9.x/releases>)

Biblioteka Bootstrap 5 (<https://getbootstrap.com/docs/5.0/getting-started/introduction/>)

c) Kaskadowe arkusze stylów CSS (<https://developer.mozilla.org/en-US/docs/Web/CSS>)

d) Język HTML5 (<https://developer.mozilla.org/en-US/docs/Web/HTML>)

e) Język JavaScript (<https://developer.mozilla.org/pl/docs/Learn/JavaScript>)

f) Baza danych MariaDB 10.4.22 (<https://mariadb.org/documentation/>)

Opis wykorzystanych narzędzi

a) Środowisko programistyczne PHPStorm (<https://www.jetbrains.com/phpstorm/>)

b) XAMPP 8.1.6 (<https://www.apachefriends.org/pl/download.html>)

c) Composer CLI (<https://getcomposer.org/>)

Instrukcja uruchomienia aplikacji

Do uruchomienia aplikacji potrzebne będzie zainstalowanie PHP w wersji 8.1, zainstalowanie XAMPP, zainstalowanie „composer”, zainstalowanie PHPStorm. Po zainstalowaniu potrzebnych środowisk oraz narzędzi:

1. Uruchomić Panel XAMPP
2. Kliknąć przycisk start w rubryce Apache
3. Kliknąć przycisk start w rubryce MySQL
4. Otworzyć przeglądarkę internetową i przejść pod adres :
(<http://localhost/phpmyadmin/>)
5. Utworzyć nową bazę danych o nazwie „projekt_2”
6. Uruchomić PHPStorm
7. Otworzyć folder z projektem w tym narzędziu
8. Uruchomić terminal
9. Wpisać polecenie: composer update
10. Wpisać polecenie: composer install
11. Wpisać polecenie: php artisan migrate
12. Wpisać polecenie: php artisan db:seed
13. Wpisać polecenie: php artisan serve
14. Przejść pod link wyświetlony po wykonaniu ostatniej instrukcji

Omówienie kodu aplikacji

1. Plik z łączeniem aplikacji z bazą danych to „laravel/.env”

```
DB_CONNECTION=mysql
DB_HOST=127.0.0.1
DB_PORT=3306
DB_DATABASE=projekt_2
DB_USERNAME=root
DB_PASSWORD=
```

2. Pliki z seedarami do bazy danych znajdują się w „laravel/database/seeder”

```
public function run()
{
    $continent = Continent::create([
        'name' => 'Europa'
    ]);
    $continent->save();
    $continent = Continent::create([
        'name' => 'Ameryka Północna'
    ]);
    $continent->save();
    $continent = Continent::create([
        'name' => 'Ameryka Południowa'
    ]);
    $continent->save();
    $continent = Continent::create([
        'name' => 'Azja'
    ]);
}
```

3. Przykładowe funkcjonalności aplikacji (odwołanie rezerwacji)

```
public function cancel($id){
    if(!Auth::check()){
        return redirect()->back()->with('alert', 'Musisz być zalogowany aby przejść');
    }else{
        $trip = Trip::find($id);

        if((strtotime($trip->start_date) - strtotime(date('format: Y-m-d')))/60/60/24 < 5){
            return redirect()->back()->with('alert', 'Nie możesz odwołać rezerwacji mniej niż 5 dni przed wyjazdem');
        }else{
            $user = Auth::user();
            $trip = $user->trips()->where('trip_id', $id)->first();
            if($trip == null){
                return redirect()->back()->with('alert', 'Nie jesteś zapisany do tej wycieczki');
            }else{
                $user->trips()->detach($trip);
                $trip = Trip::find($id);
                $trip->free_space = ($trip->free_space + 1);
                $trip->update();
                return redirect()->back()->with('alert', 'Rezerwacja odwołana pomyślnie');
            }
        }
    }
}
```

4. Przykład uwierzytelnienia i autoryzacji (rejestracja użytkownika)

```
public function register(RegisterRequest $request){

    $user = User::create([
        'name'=>$request['name'],
        'email'=>$request['email'],
        'password'=>$request['password'],
        'role_id'=>2
    ]);

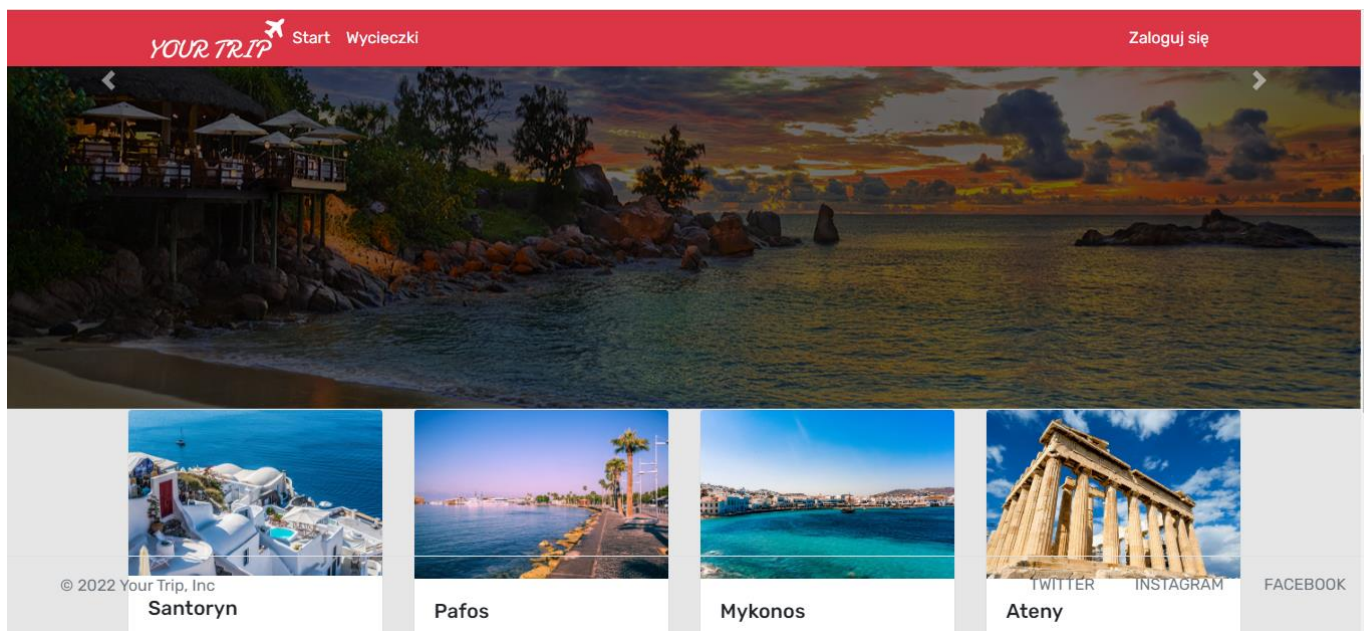
    auth()->login($user);

    return redirect(to: 'home')->with('alert', "Użytkownik zalogowany pomyślnie");
}
```

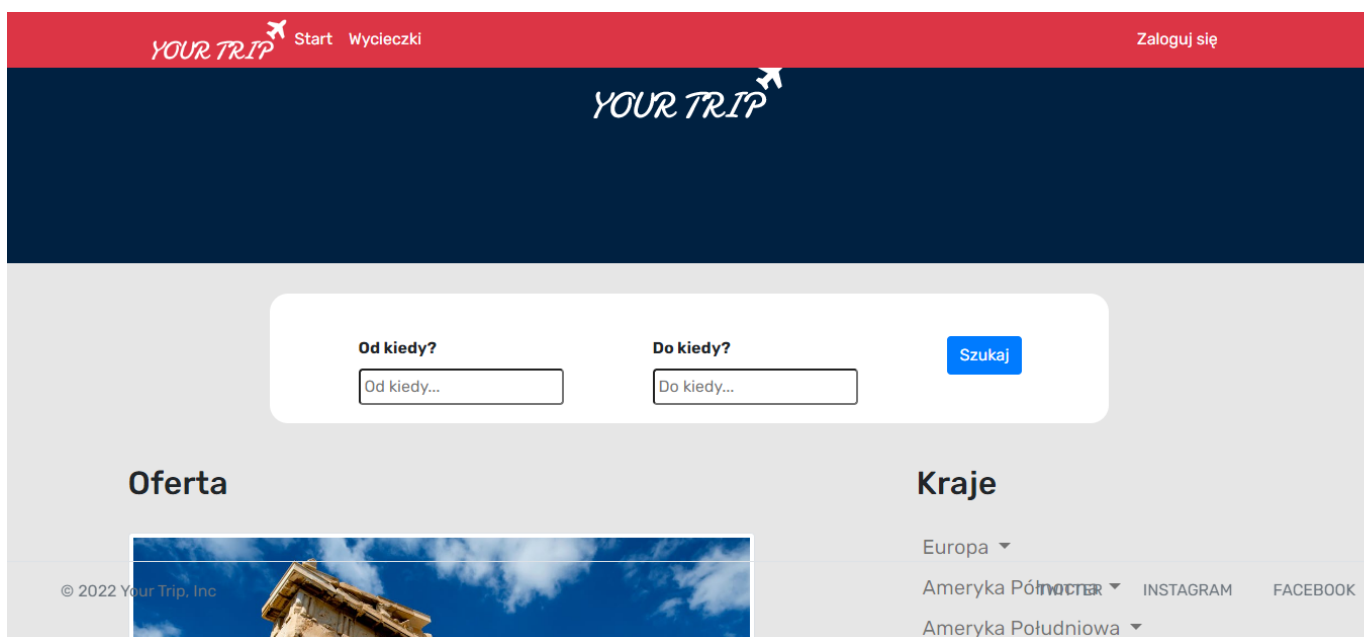
5. Przykład realizacji walidacji danych (walidacja danych rejestracji użytkownika)

```
public function rules()
{
    return [
        'email' => 'required|email:rfc,dns|unique:users,email',
        'name' => 'required',
        'password' => 'required|min:8',
        'password2' => 'required|same:password'
    ];
}
```

6. Zrzuty ekranów podstron aplikacji



(Strona główna) Jest to strona startowa aplikacji. Składa się ona z paska nawigacyjnego który przekierowuje nas do pozostałych podstron aplikacji.



Powyższa strona zawiera informacje o wszystkich wycieczkach w bazie danych. Użytkownik może wyszukać w niej wycieczkę po dacie rozpoczęcia, zakończenia oraz po kraju w którym ona się odbywa. Wszystko odbywa się za pośrednictwem serwera. Oprócz filtrowania i wyszukiwania wycieczek użytkownik ma możliwość przejścia do podstrony z interesującą go wycieczką.

Zaloguj się

Email

Hasło

[Nie masz jeszcze konta? Zarejestruj się!](#)

Zaloguj się!

Powyższa strona jest to podstrona służąca do logowania się użytkownika do aplikacji. W przypadku braku konta użytkownik może przejść do podstrony z rejestracją. W przypadku jeśli użytkownik jest już zalogowany nie ma możliwości odwiedzenia tej strony dzięki zabezpieczeniu od strony serwera.

Zarejestruj się

Email

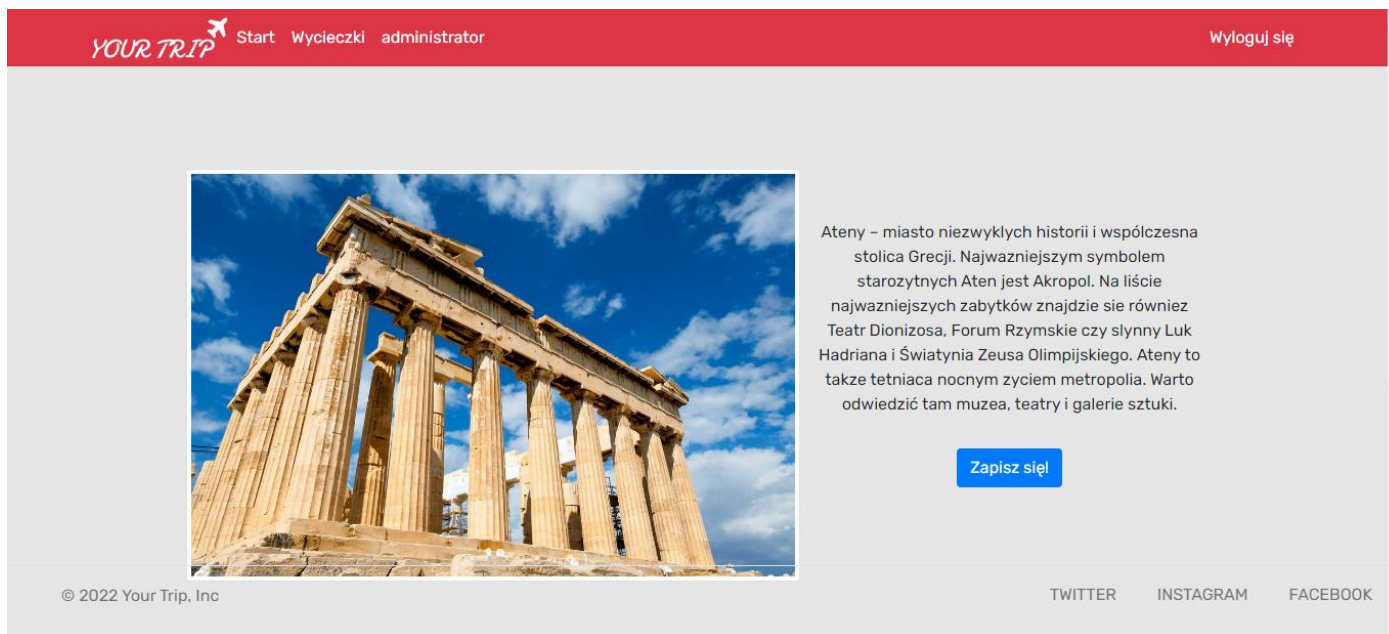
Nazwa użytkownika

Hasło

Powtórz hasło

Zarejestruj się!

Powyższa podstrona to podstrona rejestracji. Użytkownik może się zarejestrować i być pełnoprawnym użytkownikiem aplikacji. Rejestracja na tej podstronie zabezpieczona jest walidacjami takimi, że hasła muszą być takie same, oraz muszą zgadzać się co do długości oraz znaków specjalnych. W przypadku zalogowanego użytkownika nie ma on możliwości odwiedzenia tej strony, również poprzez zabezpieczenie od strony serwera.



Powyższa podstrona jest to podstrona z wycieczką (każda wycieczka takową posiada). Użytkownik na tej stronie ma możliwość zarezerwowania miejsc do tej wycieczki. Jeśli tylko nie jest administratorem, jest zalogowany oraz jest wystarczająca ilość miejsc w tej wycieczce. Wszystko także zostało zabezpieczone po stronie serwera.

Podstrona edytuj wycieczkę/dodaj wycieczkę różni się nagłówkiem oraz tym, że edytowanie wycieczki powoduje wypełnienie pól formularza aktualnymi danymi. Na tej podstronie administrator, no tylko dla niego jest dostępna ta strona może edytować lub dodać nową wycieczkę oczywiście spełniając określone zasady walidacji, czyli np. opis nie może być dłuższy niż 300 znaków, miasto musi występować w bazie danych itp. Wszystko zostało zabezpieczone od strony serwera.

YOUR TRIP

Start Wycieczki administrator

Wyloguj się

Wycieczki

Dodaj wycieczkę

ID	Miasto	Cena	Ilość miejsc	Początek	Koniec	Długość		
1	Ateny	3000.00 zł	20	2022-07-30	2022-08-05	6	Edycja	Usuń
2	Santoryn	3400.00 zł	20	2022-07-31	2022-08-06	7	Edycja	Usuń
3	Mykonos	5200.00 zł	20	2022-08-20	2022-08-29	10	Edycja	Usuń
4	Madryt	2800.00 zł	20	2022-07-18	2022-07-22	4	Edycja	Usuń
5	Barcelona	3000.00 zł	20	2022-07-14	2022-07-22	8	Edycja	Usuń

© 2022 Your Trip, Inc. [Facebook](#)

Powyższa podstrona jest to tzw. Panel sterowania aplikacją. Jest on dostępny tylko i wyłącznie dla administratora. Ma on na nim wgląd do wszystkich jakichkolwiek wycieczek odbytych już jak i nie odbytych. Z tego panelu może on przejść do edycji wycieczek, miast oraz krajów. Może także usunąć te dane z bazy. Wszystkie zabezpieczenia aby zwykły użytkownik nie miał wglądu do tej podstrony są po stronie serwera.

YOUR TRIP

Start Wycieczki użytkownik

Wyloguj się

Wycieczki

ID	Miasto	Cena	Ilość miejsc	Początek	Koniec	Długość	
7	Pafos	7500.00 zł	19	2022-06-23	2022-06-30	7	Odwołaj rezerwacje

Ostatnia podstrona jest to podstrona panelu sterowania ale tylko i wyłącznie dla zalogowanego użytkownika, który nie jest administratorem. Na tej podstronie użytkownik może zarządzać swoją wycieczką. Może zrezygnować z rezerwacji ale tylko i wyłącznie, jeśli termin rozpoczęcia wycieczki nie jest mniejszy niż 5 dni. Wszystko to zostało zabezpieczone po stronie serwera. Oczywiście też odbyta wycieczka nie ma możliwości odwołania rezerwacji, ponieważ została odbyta. Aby przejść do tego panelu trzeba kliknąć zakładkę w nawigacji z nazwą użytkownika, w tym przypadku jest to „użytkownik”.

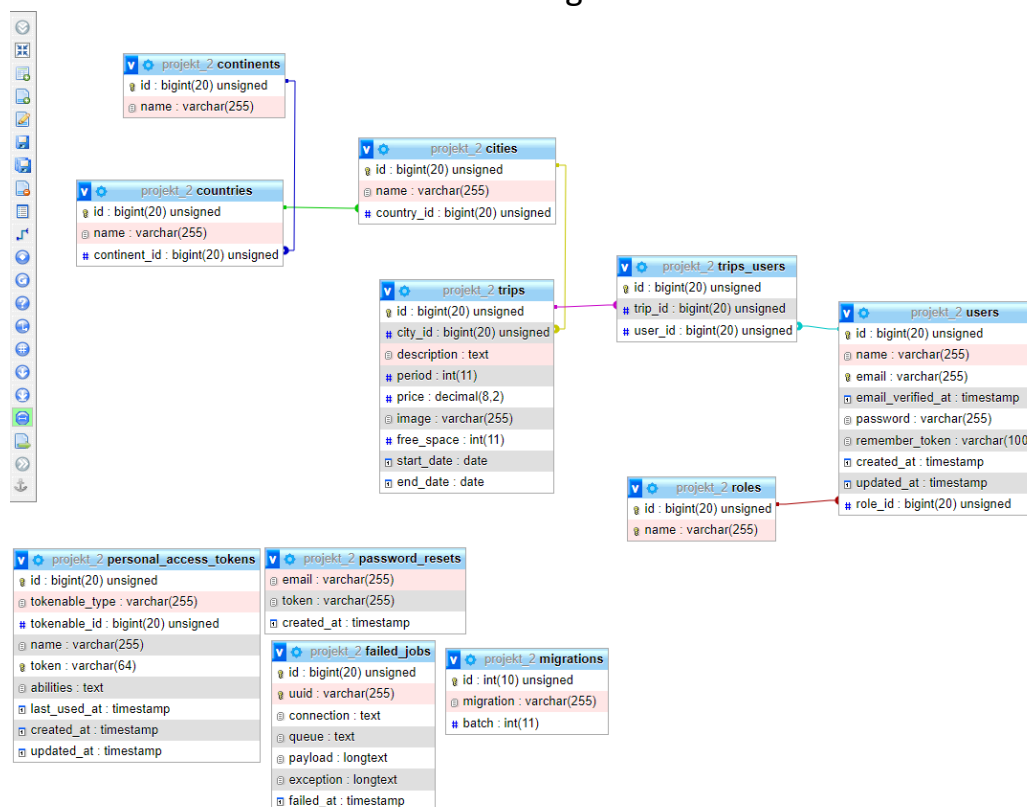
7. Routing

```

GET|HEAD / ..... home > TripController@index
POST _ignition/execute-solution ..... ignition.executeSolution > Spatie\LaravelIgnition > ExecuteSolutionController
GET|HEAD _ignition/health-check ..... ignition.healthCheck > Spatie\LaravelIgnition > HealthCheckController
POST _ignition/update-config ..... ignition.updateConfig > Spatie\LaravelIgnition > UpdateConfigController
GET|HEAD api/user ..... create > TripController@create
GET|HEAD cancel/{id} ..... trips.cancel > UserController@cancel
GET|HEAD create ..... create > TripController@create
PUT create ..... trips.store > TripController@store
GET|HEAD create_city ..... create_city > CityController@create
PUT create_city ..... cities.store > CityController@store
GET|HEAD create_country ..... create_country > CountryController@create
PUT create_country ..... country.store > CountryController@store
DELETE destroy/{id} ..... trips.destroy > TripController@destroy
GET|HEAD offer ..... offer > ContinentController@offer
POST offer ..... offer > ContinentController@offer
GET|HEAD register ..... register > LoginController@register
POST register_ ..... register_ > LoginController@register_
GET|HEAD sanctum/csrf-cookie ..... Laravel\Sanctum > CsrfCookieController@show
GET|HEAD show/{id} ..... show > TripController@show
PUT show/{id}/save ..... save > UserController@save
GET|HEAD trips/{id} ..... trips > TripController@trips
POST trips/{id} ..... trips > TripController@trips

```

8. Diagram ERD



Opis tabel

Tabele: personal_access_tokens, password_resets, failed_jobs oraz migrations zostały wygenerowane automatycznie przez framework laravel.

Tabela continents przechowuje dane o wszystkich kontynentach na których mogą odbyć się wycieczki, składa się z dwóch kolumn: id, name, połączona jest z tabelą countries. Tabela countries przechowuje dane o wszystkich krajach w których mogą odbyć się wycieczki, składa się z trzech kolumn: id, name oraz continent_id, połączona jest z tabelą cities. Tabela cities przechowuje dane o wszystkich miastach w których mogą odbyć się wycieczki, składa się z trzech kolumn: id, name, country_id

połączona jest z tabelą trips. Trzy powyższe tabele służą głównie do przechowywania informacji o miejscach w których mogą się odbywać wycieczki. Kolejna Tabela to tabela trips jest ona najbardziej rozbudowaną tabelą w bazie danych. Zawiera ona najważniejsze informacje na temat głównego przedmiotu projektu czyli wycieczek.

Składa się z 9 kolumn: id, city_id, description, period, price, image, free_space, start_date, end_date, połączona jest z tabelą city tak w łatwy sposób możemy dotrzeć dokładnie do jakiego miejsca jest wycieczka od miasta przez kraj do kontynentu, oraz za pomocą tabeli transferowej z użytkownikami którzy uczestniczą w danej wycieczce.

Informacje które zawiera ta tabela to miejsce gdzie się odbywa wycieczka, opis wycieczki, data początku i końca wycieczki, długość wycieczki, cena, oraz zdjęcie które się wyświetla na stronie związane z wycieczką. Kolejną tabelą jest tabela transferowa która łączy tabelę users i trips. Składa się ona z klucza głównego oraz dwóch kluczy opci odpowiednio z tabeli trips i users. Jest to połączenie wiele do wielu ponieważ każdy użytkownik może być przypisany do wielu wycieczek jak i do jednej wycieczki może być przypisanych wielu użytkowników. Tabela users składa się z dziewięciu kolumn: klucz główny(id), name(nazwa użytkownika), email(email użytkownika), password(zahashowane hasło użytkownika), role_id(klucz obcy z tabeli role przechowująca informacje na temat roli danego użytkownika). Tabela role przechowuje dwie role odpowiednio: admin oraz użytkownik. Dzięki niej aplikacja rozróżnia administratora od zwykłego użytkownika.