

Autorzy:
Szymon Martyn 149649
Przemysław Radzik 149651
Łukasz Byzdra 151224

API Z WYKORZYSTANIEM LARAVELA 5.3

Technologie użyte do projektu to PHP(Laravel 5.3), JS(jQuery), CSS(Bootstrap), MySQL.

Na potrzeby projektu utworzone zostały:

1) 3 kontrolery:

- Jeden główny (TravelController.php)
- Dwa obsługujące API (Travel1Controller.php, Travel2Controller.php)

2) 2 modele na potrzeby API:

- Travel1.php
- Travel2.php

3) Baza danych MySQL o nazwie 'travel'

W pliku .env podajemy dane do bazy danych

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

4) Migracje do bazy tworzące odpowiednie tabele

Z poziomu CLI wpisując komendę „php artisan migrate” utworzą nam się potrzebne tabele.

5) Seedery, które można uruchomić aby wypełnić bazę losowymi danymi na potrzeby testów API

6) 3 Routingi dla aplikacji i 2 dla API z prefixem „api1” i „api2”.

Nasz główny kontroler służy nam za obsługę aplikacji. W metodzie index() pobieramy z obu API listy do wyświetlenia

```
$a = $this->call('GET', 'api1/travel');
$b = $this->call('GET', 'api2/travel');
```

i następnie scalamy je i wysyłamy w postaci tablicy na widok. Do ścieżki przypisana jest odpowiednia akcja, która zwraca nam pożądany wynik w postaci listy wszystkich rekordów z bazy.

W przypadku wyszukiwania wysyłam requesta również GETem, ale podając już parametr, co zwraca nam obiekty, które zawierają opcję której szukamy z dwóch API.

```
$a = $this->call('GET', "api1/travel/$query");
$b = $this->call('GET', "api2/travel/$query");
```

zmienna \$query to szukany string.

Przy wyświetlaniu strony danego produktu sprawdzamy wartość api i wysyłamy requesta po jeden rekord do konkretnego API

```
$a = '';
if($api == "1")
{
    $a = $this->call('GET', "api1/travel/pages/$id");
}
else
{
    $a = $this->call('GET', "api2/travel/pages/$id");
}
```

W projekcie nie zapisujemy nic w bazie, ale można zauważyć POSTa w seederach, które wykonujemy przy inicjalizacji projektu (jeżeli chcemy).

```
$faker = \Faker\Factory::create();
for($i = 0; $i < 100; $i++)
{
    $table = new Travell();
    $table->o_name = $faker->company;
    $table->o_city = $faker->city;
    $table->o_desc = $faker->text(200);
    $table->o_country = $faker->country;
    $table->o_bprice = $faker->numberBetween(100, 10000);
    $table->o_link = "1";
    $table->save();
}
```

Adres do repozytorium: <https://github.com/kolorowy121/api>