



Politechnika
Śląska



Przedmiot: Podstawy informatyki

Nr. Ćwiczenia: 4

Imię i Nazwisko: Michał Witczak

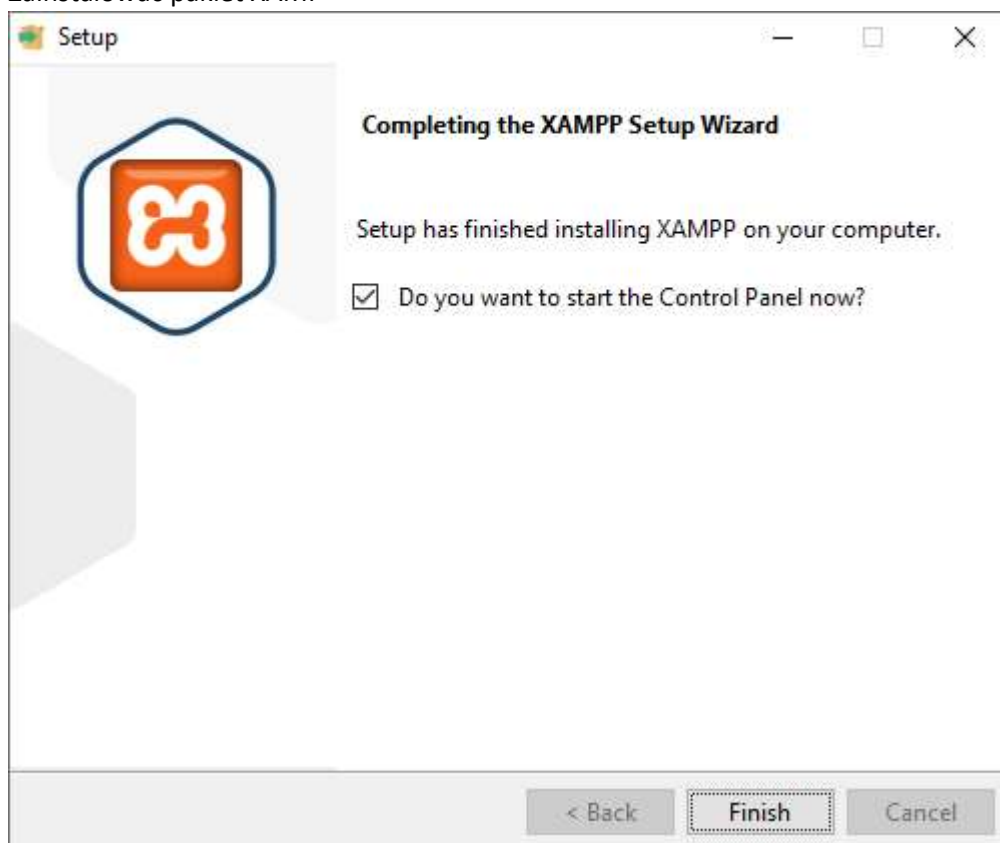
Nr. Albumu 305423

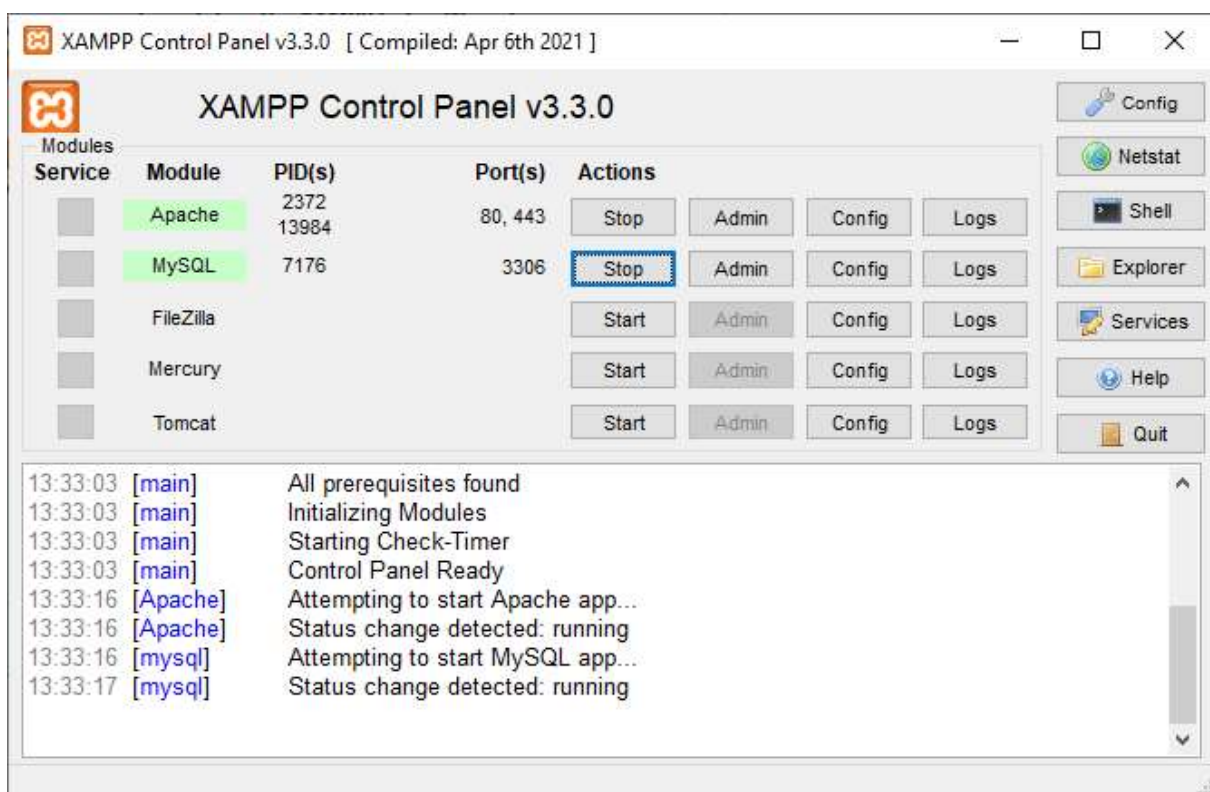
Link do repozytorium:

<https://github.com/MichalWitczakmw/RestpoprawaApi.git>



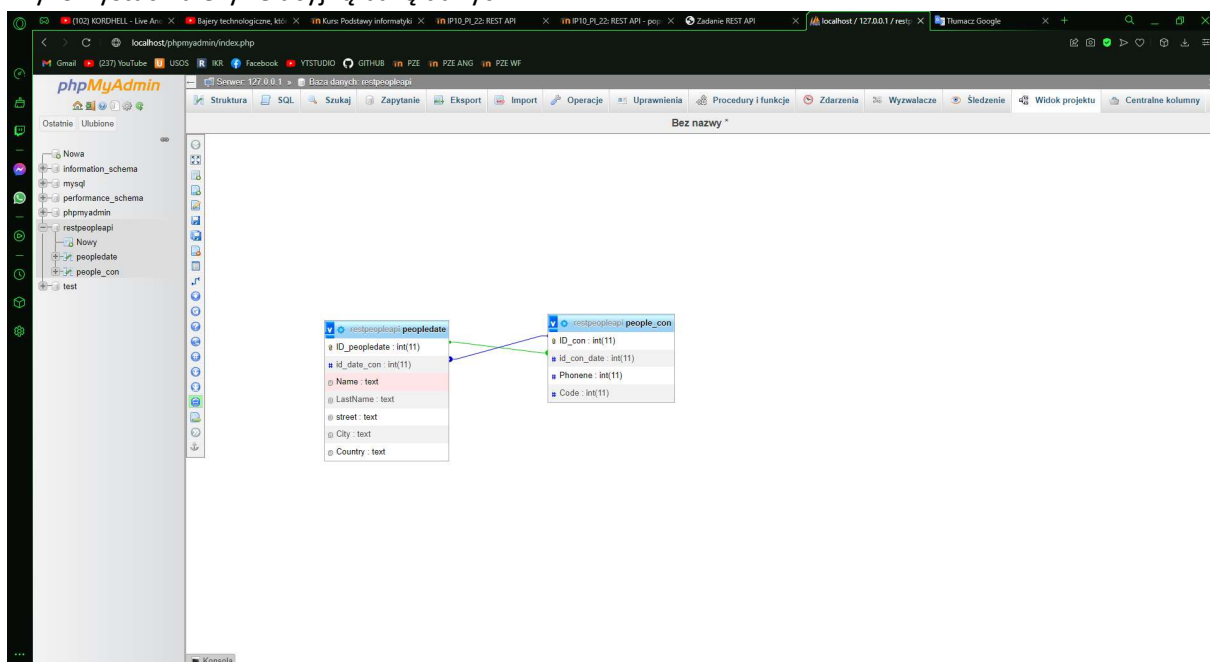
1. Zainstalować pakiet XAMPP





Nie było problemów z instalacją ani z uruchomieniem.

2. Wykorzystać należy relacyjną bazę danych.

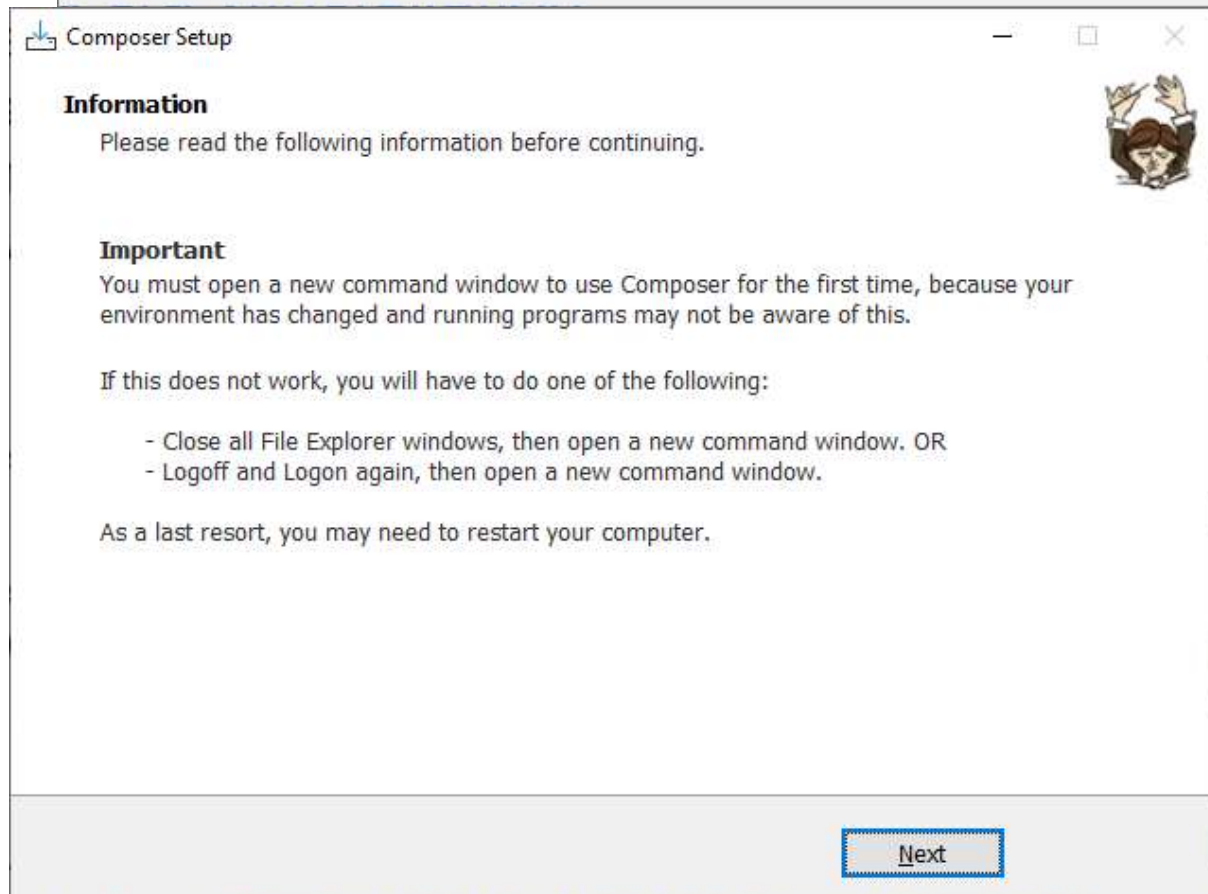
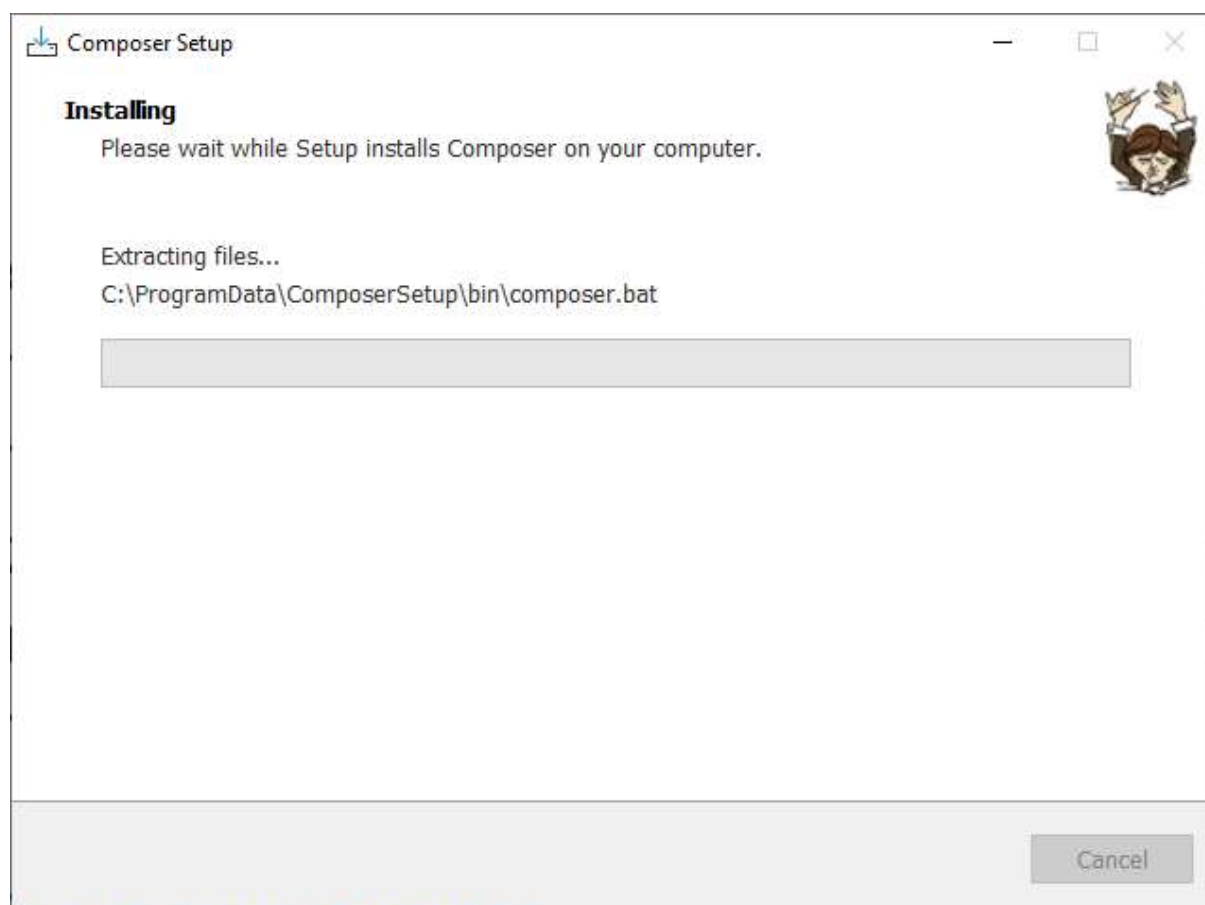




```
1 APP_NAME=Laravel
2 APP_ENV=local
3 APP_KEY=base64:5/GqD5azXxcWGgw3h5Wlk6yQAvIo7kyo2SkQEL1V9xm=
4 APP_DEBUG=true
5 APP_URL=http://localhost
6
7 LOG_CHANNEL=stack
8 LOG_DEPRECATIONS_CHANNEL=null
9 LOG_LEVEL=debug
10
11 DB_CONNECTION=mysql
12 DB_HOST=127.0.0.1
13 DB_PORT=3306
14 DB_DATABASE=restpeopleapi
15 DB_USERNAME=root
16 DB_PASSWORD=
17
18 BROADCAST_DRIVER=log
19 CACHE_DRIVER=file
20 FILESYSTEM_DISK=local
21 QUEUE_CONNECTION=sync
22 SESSION_DRIVER=file
23 SESSION_LIFETIME=120
24
25 MEMCACHED_HOST=127.0.0.1
26
27 REDIS_HOST=127.0.0.1
28 REDIS_PASSWORD=null
29 REDIS_PORT=6379
30
31 MAIL_MAILER=smtp
32 MAIL_HOST=mailhog
33 MAIL_PORT=1025
```

Udało się bez problemów utworzyć szkielet relacyjnej bazy danych plus dodać konto z hasłem i dostępem do niej.

3. Zainstalować program composer.



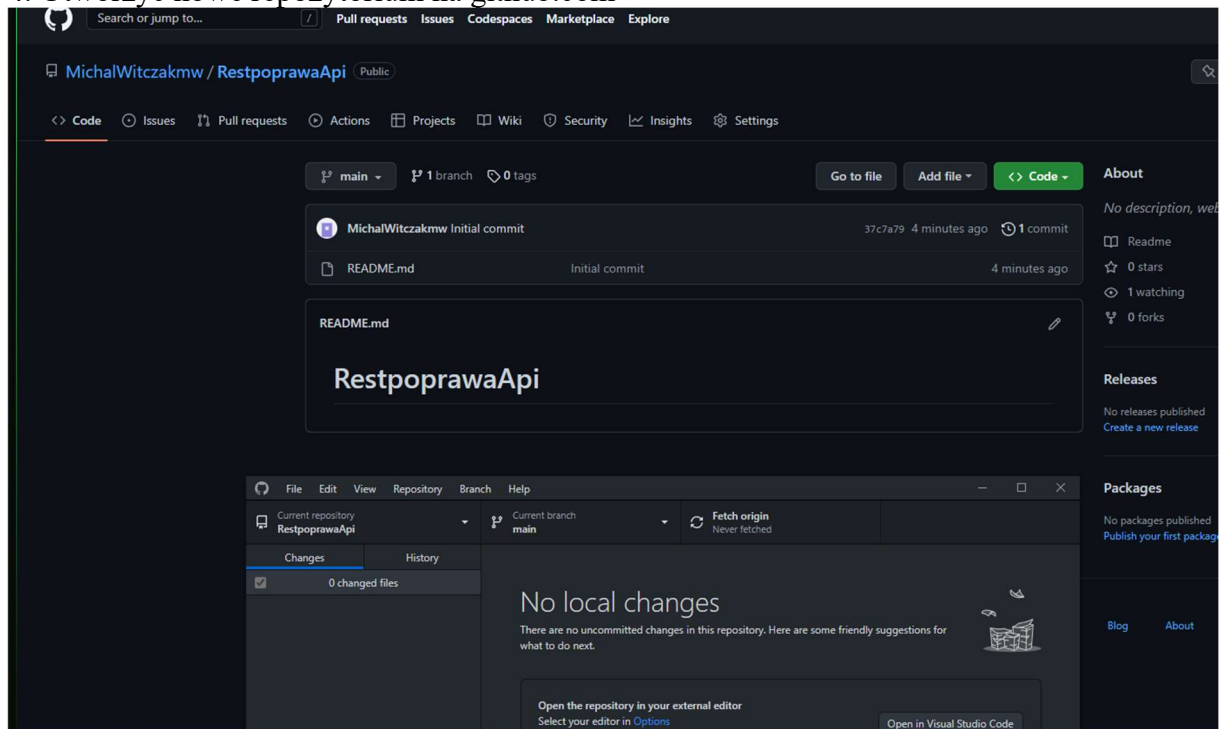


Politechnika
Śląska



Bez problemów.

4. Utworzyć nowe repozytorium na github.com



Bez problemów.

5. Utworzyć nowy projekt oparty na frameworku Laravel

```
Git CMD
- Installing spatie/backtrace (1.2.1): Extracting archive
- Installing spatie/flare-client-php (1.3.2): Extracting archive
- Installing spatie/ignition (1.4.1): Extracting archive
- Installing spatie/laravel-ignition (1.6.4): Extracting archive
72 package suggestions were added by new dependencies, use 'composer suggest' to see details.
Generating optimized autoload files
> Illuminate\Foundation\ComposerScripts::postAutoloadDump
> @php artisan package:discover --ansi

 INFO  Discovering packages.

laravel/sail ..... DONE
laravel/sanctum ..... DONE
laravel/tinker ..... DONE
nesbot/carbon ..... DONE
nunomaduro/collision ..... DONE
nunomaduro/termwind ..... DONE
spatie/laravel-ignition ..... DONE

80 packages you are using are looking for funding.
Use the 'composer fund' command to find out more!
> @php artisan vendor:publish --tag=laravel-assets --ansi --force

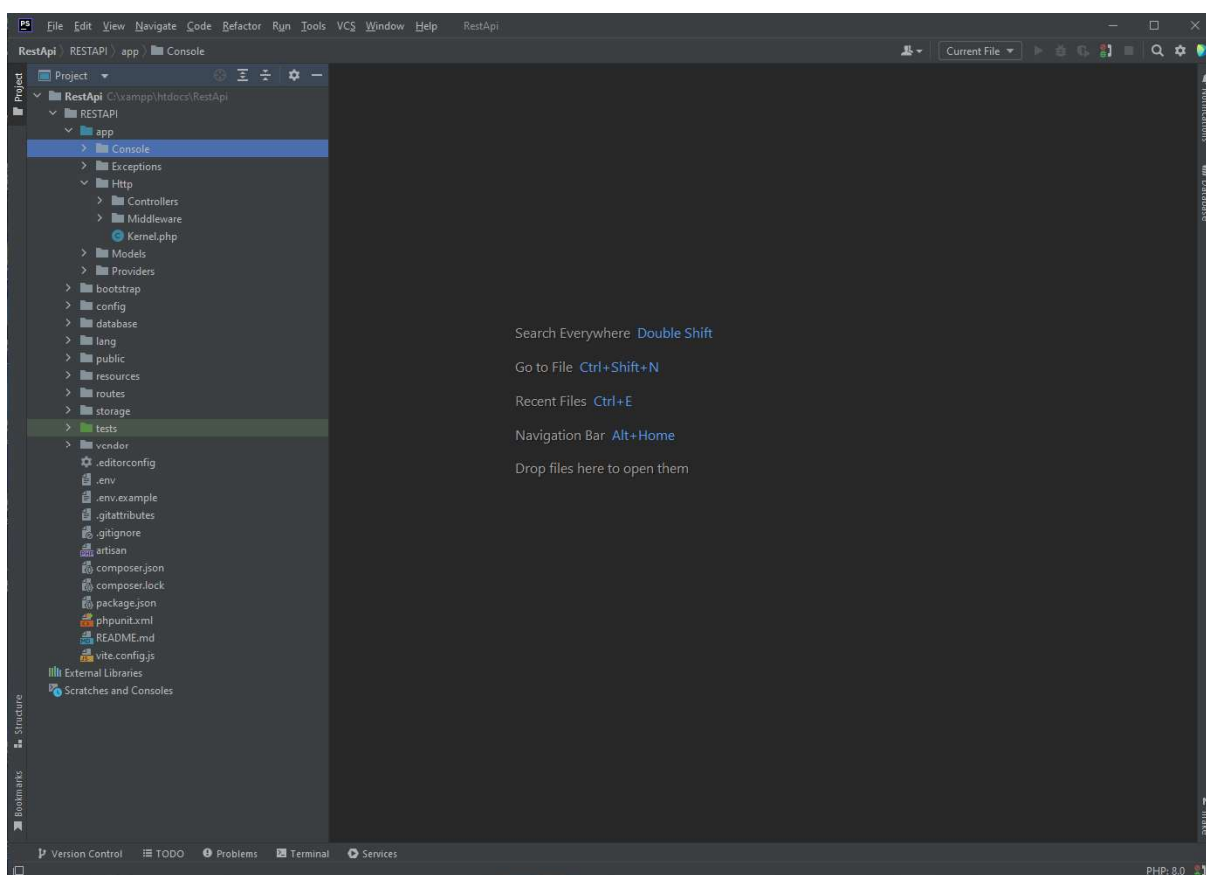
 INFO  No publishable resources for tag [laravel-assets].

No security vulnerability advisories found
> @php artisan key:generate --ansi

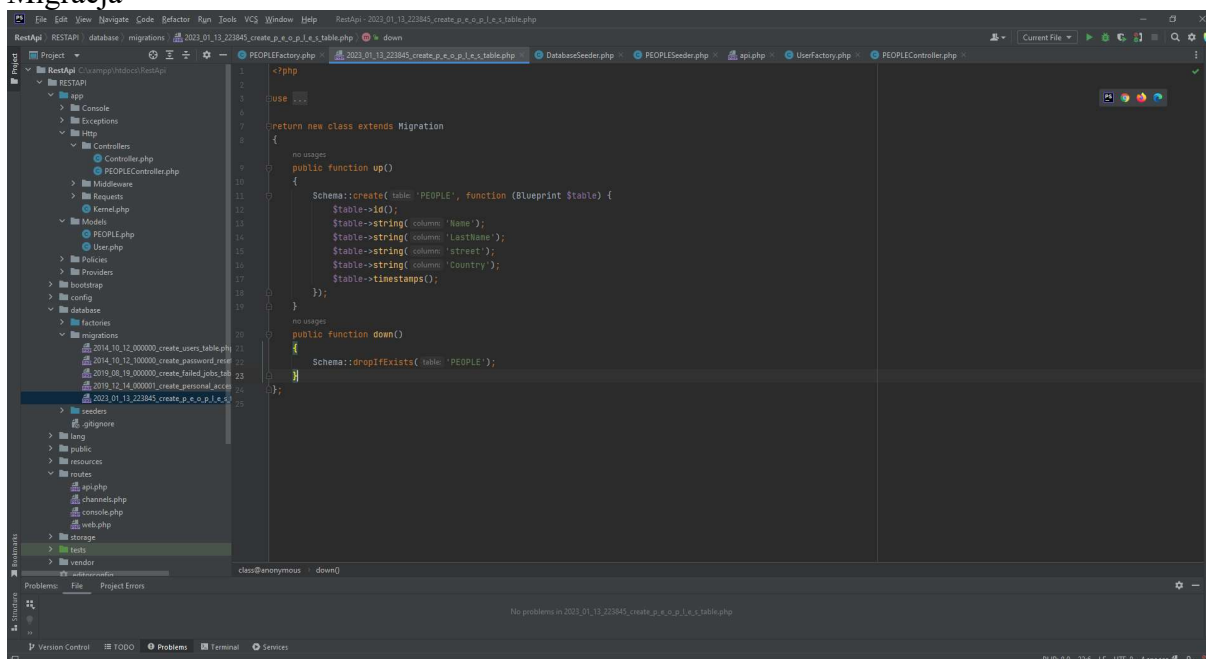
 INFO  Application key set successfully.
```



Politechnika
Śląska



Migracja



6. Poprawnie skonfigurować projekt korzystając z dokumentacji:
<https://laravel.com/docs/9.x/configuration> - należy tutaj m.in. poprawnie skonfigurować bazę danych w projekcie - tak aby połączenie działało i dobrze



wykorzystać pliki konfiguracyjne .env. Tutaj jako pomoc można wykorzystać instrukcje do starszej wersji Laravela -

<https://blog.quickadminpanel.com/how-to-use-laravel-env-example-files/>

```
.env
1 APP_NAME=Laravel
2 APP_ENV=local
3 APP_KEY=base64:615n6b+TKL8ztouY5mgvzRqT8TdcuUkSkudBEPb+
4 APP_DEBUG=true
5 APP_URL=http://localhost
6
7 LOG_CHANNEL=stack
8 LOG_DEPRECATIONS_CHANNEL=null
9 LOG_LEVEL=debug
10
11 DB_CONNECTION=mysql
12 DB_HOST=127.0.0.1
13 DB_PORT=3306
14 DB_DATABASE=restpeopleapi
15 DB_USERNAME=root
16 DB_PASSWORD=
17
18 BROADCAST_DRIVER=log
19 CACHE_DRIVER=file
20 FILESYSTEM_DISK=local
21 QUEUE_CONNECTION=sync
22 SESSION_DRIVER=file
23 SESSION_LIFETIME=120
24
25 MEMCACHED_HOST=127.0.0.1
26
27 REDIS_HOST=127.0.0.1
28 REDIS_PASSWORD=null
29 REDIS_PORT=6379
30
31 MAIL_MAILER=smtp
32 MAIL_HOST=mailhog
33 MAIL_PORT=1025
34 MAIL_USERNAME=null
35 MAIL_PASSWORD=null
36 MAIL_ENCRYPTION=null
37 MAIL_FROM_ADDRESS="hello@example.com"
38 MAIL_FROM_NAME="${APP_NAME}"
39
40 AWS_ACCESS_KEY_ID=
41 AWS_SECRET_ACCESS_KEY=
42 AWS_DEFAULT_REGION=us-east-1
43 AWS_BUCKET=
44 AWS_USE_PATH_STYLE_ENDPOINT=false
45
46 PUSHER_APP_ID=
47 PUSHER_APP_KEY=
48 PUSHER_SECRET=
49 PUSHER_HOST=
```

7. Utworzyć kontroler (<https://laravel.com/docs/9.x/controllers>) wraz z metodami, który będzie obsługiwać wszystkie przychodzące requesty (<https://laravel.com/docs/9.x/requests>).

```
Micha@DESKTOP-H1M9TH MINGW64 /c:/xampp/htdocs/RestApi
$ php artisan serve
could not open input file: artisan

Micha@DESKTOP-H1M9TH MINGW64 /c:/xampp/htdocs/RestApi
$ php artisan serve
could not open input file: artisan

Micha@DESKTOP-H1M9TH MINGW64 /c:/xampp/htdocs/RestApi
$ php artisan serve
could not open input file: artisan

Micha@DESKTOP-H1M9TH MINGW64 /c:/xampp/htdocs/RestApi
$ cd RESTAPI
Micha@DESKTOP-H1M9TH MINGW64 /c:/xampp/htdocs/RestApi/RESTAPI
$ php artisan make:model PEOPLE --all

[INFO] Model [C:\xampp\htdocs\RestApi\RESTAPI\app\Models\PEOPLE.php] created successfully.
[INFO] Factory [C:\xampp\htdocs\RestApi\RESTAPI\database\factories\PEOPLEFactory.php] created successfully.
[INFO] Migration [C:\xampp\htdocs\RestApi\RESTAPI\database\migrations\2023_01_13_223845_create_people_table.php] created successfully.
[INFO] Seeder [C:\xampp\htdocs\RestApi\RESTAPI\database\seeders\PEOPLESeeder.php] created successfully.
[INFO] Request [C:\xampp\htdocs\RestApi\RESTAPI\app\Http\Requests\StorePEOPLERequest.php] created successfully.
[INFO] Request [C:\xampp\htdocs\RestApi\RESTAPI\app\Http\Requests\UpdatePEOPLERequest.php] created successfully.
[INFO] Controller [C:\xampp\htdocs\RestApi\RESTAPI\app\Http\Controllers\PEOPLEController.php] created successfully.
[INFO] Policy [C:\xampp\htdocs\RestApi\RESTAPI\app\Policies\PEOPLEPolicy.php] created successfully.
Micha@DESKTOP-H1M9TH MINGW64 /c:/xampp/htdocs/RestApi/RESTAPI
$
```

Z utworzeniem nie było problemów.

8. Dodać potrzebne wpisy obsługujące routing przychodzących requestów (<https://laravel.com/docs/9.x/responses>) w pliku routes/api.php.



```
<?php
use Illuminate\Http\Request;
use App\Http\Controllers\PEOPLEController;
use Illuminate\Support\Facades\Route;
use App\Models\PEOPLE;

Route::get('/Witczak/305423/people', [PEOPLEController::class, 'index']);
Route::post('/Witczak/305423/people', [PEOPLEController::class, 'create']);
Route::get('/Witczak/305423/people/{id}', [PEOPLEController::class, 'show']);
Route::delete('/Witczak/305423/people/{id}', [PEOPLEController::class, 'delete']);
Route::put('/Witczak/305423/people/{id}', [PEOPLEController::class, 'update']);
Route::middleware('auth:sanctum')->get('/user', function (Request $request) {
    return $request->user();
});
```

9. Przygotować odpowiednio migrację (<https://laravel.com/docs/9.x/migrations>) zawierającą utworzenie 50fake-owych instancji obiektu People. Do zadania można wykorzystać bibliotekę: <https://github.com/fzaninotto/Faker>. Obiekt klasy People ma posiadać oprócz obowiązkowych właściwości (np. id) minimum: imię, nazwisko, numer telefonu, ulica, miasto/kraj. Można dodać również inne swoje. Tworzenie własnego modelu obiektu zostało opisane tutaj: <https://laravel.com/docs/9.x/eloquent>.

```
MINGW64/c:/xampp/htdocs/RestApi/RESTAPI
INFO Seeding database.
DatabaseSeeder: DatabaseSeeder
BadMethodCallException
Call to undefined method App\Models\PEOPLE::setContainer()
at C:\xampp\htdocs\RestApi\RESTAPI\vendor\laravel\framework\src\Illuminate\Support\Traits\ForwardsCalls.php:71
67 * @throws \BadMethodCallException
68 */
69 protected static function throwBadMethodCallException($method)
70 {
71     throw new BadMethodCallException(sprintf(
72         'Call to undefined method %s::%s()', static::class, $method
73     ));
74 }
75 }
i Bad Method Call: Did you mean App\Models\PEOPLE::setConnection() ?
1 C:\xampp\htdocs\RestApi\RESTAPI\vendor\laravel\framework\src\Illuminate\Support\Traits\ForwardsCalls.php:36
Database: Illuminate\Database\Eloquent\Model::throwBadMethodCallException("setContai
```

Pojawił się problem przy użyciu komendy `php artisan migrate:fresh --seed`



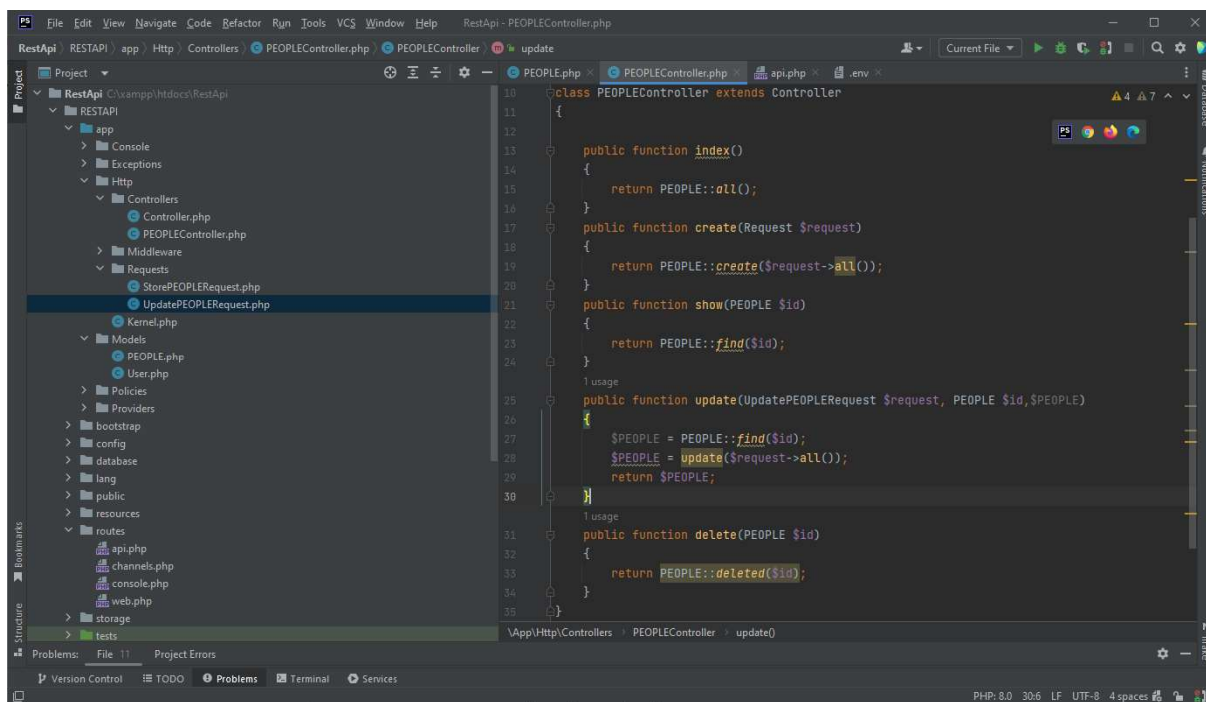
Szukałem w nocy po wszystkich stronach też na stackoverflow i nie znalazłem. Możliwe że nie umiałem zadać prawidłowego pytania.

Nawet po wielu próbach poprawiania samego pliku z tej ścieżki lub samym seeder-ze nie mogłem tego zrealizować.

```
MINGW64: c:\xampp\htdocs\RestApi\RESTAPI seeder.php PEOPLEseeder.php ForwardsCalls.php
INFO Seeding database.
Error Undefined constant Illuminate\Database\Query\Builder::self
at C:\xampp\htdocs\RestApi\RESTAPI\vendor\laravel\framework\src\Illuminate\Support\Traits\ForwardsCalls.php:72
68 | */
69 | protected static function throwBadMethodCallException($method)
70 | {
71 |     throw new BadMethodCallException(sprintf(
72 |         'Call to undefined method %s::%s()', static::self, $method
73 |     ));
74 | }
75 | }
76 |
1 C:\xampp\htdocs\RestApi\RESTAPI\vendor\laravel\framework\src\Illuminate\Database\Query\Builder.php:3782
Illuminate\Database\Query\Builder::throwBadMethodCallException("setContain
er")
```

10. Kontroler ma posiadać metody które umożliwią obsługę następujących funkcjonalności (wszystkie metody CRUD - create, read, update, delete + pokaż wszystko) :

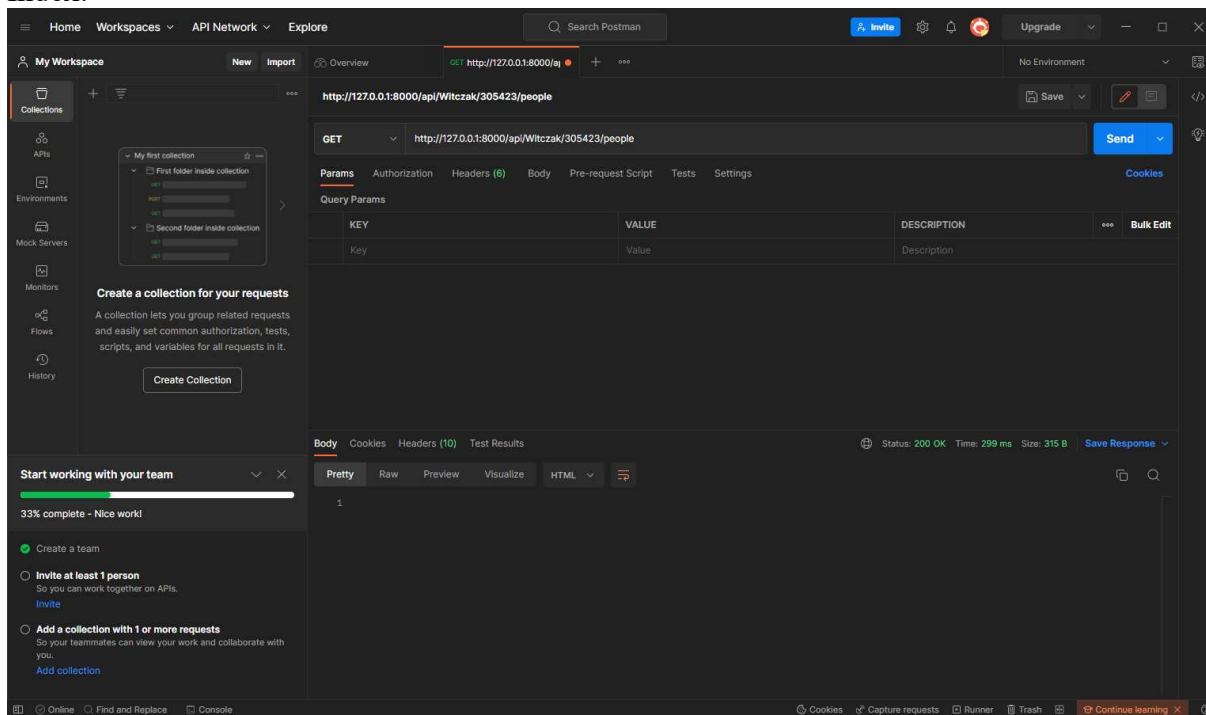
- zwróć wszystkie instancje People przechowywane w bazie
- zwróć pojedynczą instancję People o żądanym ID
- wykonaj aktualizację instancji People o żądanym ID
- usuń instancję People o żądanym ID
- utwórz pojedynczą instancję People o żądanym ID



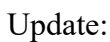
W ten sposób zrobiłem z robiłem korzystając z dokumentacji laravela.

11. Pokazać działanie aplikacji poprzez wykorzystanie programu Postman (<https://www.postman.com/>) - każda operacja ma posiadać swój własny screen z requestem postmana oraz otrzymaną odpowiedź.

Index:



Create:





Politechnika
Śląska



Home Workspaces API Network Explore Search Postman

My Workspace New Import Overview GET http://127.0.0.1:8000/api POST http://127.0.0.1:8000/api PUT http://127.0.0.1:8000/api

PUT http://127.0.0.1:8000/api/Witczak/305423/people/2

Params Authorization Headers (8) Body Pre-request Script Tests Settings

none form-data x-www-form-urlencoded raw binary GraphQL

| KEY | VALUE | DESCRIPTION |
|----------|-----------|-------------|
| Name | andrzej | |
| LastName | karle | |
| street | dasdarnej | |
| Country | gtrgd | |
| Key | Value | Description |

Body Cookies Headers (8) Test Results Status: 404 Not Found Time: 5.00 s Size: 8.19 KB Save Response

Pretty Raw Preview Visualize JSON

```
1 {
2   "message": "The route api/Witczak/305423/people/2 could not be found.",
3   "exception": "Symfony\\Component\\HttpKernel\\Exception\\NotFoundHttpException",
4   "file": "C:\\xampp\\htdocs\\RestApi\\RESTAPI\\vendor\\laravel\\framework\\src\\Illuminate\\Routing\\AbstractRouteCollection.php",
5   "line": 44,
6   "trace": [
7     {
8       "file": "C:\\xampp\\htdocs\\RestApi\\RESTAPI\\vendor\\laravel\\framework\\src\\Illuminate\\Routing\\RouteCollection.php",
9       "line": 162,
10      "function": "handleMatchedRoute",
11      "class": "Illuminate\\Routing\\AbstractRouteCollection",
12      "type": "->"
13    },
14    {}
15  ]
16 }
```

Delete:

Home Workspaces API Network Explore Search Postman

My Workspace New Import Overview GET http://127.0.0.1:8000/api POST http://127.0.0.1:8000/api PUT http://127.0.0.1:8000/api DELETE http://127.0.0.1:8000/api

DELETE http://127.0.0.1:8000/api/Witczak/305423/people/1

Params Authorization Headers (8) Body Pre-request Script Tests Settings

Query Params

| KEY | VALUE | DESCRIPTION |
|-----|-------|-------------|
| Key | Value | Description |

Body Cookies Headers (8) Test Results Status: 404 Not Found Time: 1239 ms Size: 6.71 KB Save Response

Pretty Raw Preview Visualize HTML

```
1 <!DOCTYPE html>
2 <html lang="en">
3
4 <head>
5   <meta charset="utf-8">
6   <meta name="viewport" content="width=device-width, initial-scale=1">
7
8   <title>Not Found</title>
9
10  <style>
11    /* normalize.css v8.0.1 | MIT License | github.com/necolas/normalize.css */
12    html {
13      line-height: 1.15;
14      -webkit-text-size-adjust: 100%;
15    }
16  </style>
17 </head>
18 </html>
```

Niestety Postman prawidłowo nie zadziała dopóki nie będę miał użytkowników w tabeli ale niestety błąd w punkcie 9 przeszkodził bardzo.

12. Należy w programie szczególnie uwzględnić obsługę metod <https://developer.mozilla.org/en-US/docs/Web/HTTP/Methods>), które wykorzystywane



są w operacjach CRUD. Inne metody nie są obowiązkowe.
Nie rozumiałem polecenia.

13. Należy w kontrolerze oraz w postmanie szczególnie zwrócić uwagę na kody zwracanych requestów (<https://developer.mozilla.org/en-US/docs/Web/HTTP/Status>)

Niestety z racji tego błędu w 9 nie mogłem uzyskać poprawnych kodów(nawet błędów).

14. Adresy URL (endpointy) mają zawierać nazwisko oraz numer studenta w swojej pracy, np. 127.0.0.1/kowalski/12345/people/5 etc..

```
Route::get( uri: '/Witczak/305423/people', [PEOPLEController::class, 'index']);
Route::post( uri: '/Witczak/305423/people', [PEOPLEController::class, 'create']);
Route::get( uri: '/Witczak/305423/people/{}', [PEOPLEController::class, 'show']);
Route::delete( uri: '/Witczak/305423/people', [PEOPLEController::class, 'delete']);
Route::put( uri: '/Witczak/305423/people/{}', [PEOPLEController::class, 'update']);
```

Co to jest migracja?

Migration tool to narzędzie w Laravelu, które pozwoli Ci stworzyć nową tabelę w bazie bazując na definicji w kodzie PHP. Wewnątrz metody PHP wskazujesz, jakie kolumny ma zawierać tabela, jakiego typu i jakiej nazwy, po czym narzędzie do migracji stworzy je w bazie danych za Ciebie.

Co to są kody HTTP?

Kod HTTP to informacja zwrócona przez serwer do użytkownika. Każdy posiada odrębne oznaczenie numeryczne. Niektóre z nich są czysto informacyjne, inne potwierdzają powodzenie, lub przekierowanie. Wyróżniamy również błędy HTTP, które strony internetowej bądź serwera.

Co to jest programowanie obiektowe?

Programowanie obiektowe to programowania, w którym programy definiuje się za pomocą obiektów – elementów łączących stan i zachowanie.

Korzystałem z :



Politechnika
Śląska



<https://www.youtube.com/watch?v=MT-GJQIY3EU>

<https://www.youtube.com/watch?v=YGqCZjdJJk>

stackoverflow

<https://laravel.com/docs/9.x>

Wszystkie pliki używane:

The screenshot shows an IDE window with a project named 'RestApi'. The file explorer on the left shows the project structure, including folders like 'app', 'bootstrap', 'config', 'database', 'factories', 'migrations', 'seeds', and 'routes'. The 'factories' folder is expanded, showing files like 'PEOPLEFactory.php', 'UserFactory.php', and 'DatabaseSeeder.php'. The 'seeds' folder is also expanded, showing files like 'DatabaseSeeder.php' and 'PEOPLESeeder.php'. The main editor window displays the content of 'PEOPLE.php', which is a PHP file defining a 'PEOPLE' model. The code is as follows:

```
<?php
namespace App\Models;

use Illuminate\Database\Eloquent\Factories\HasFactory;
use Illuminate\Database\Eloquent\Model;

class PEOPLE extends Model
{
    use HasFactory;

    protected $fillable=[
        'Name',
        'LastName',
        'street',
        'Country',
    ];
}
```




```
PS File Edit View Navigate Code Refactor Run Tools VCS Window Help RestApi - PEOPLEController.php
RestApi RESTAPI app Http Controllers PEOPLEController.php PEOPLEController show
Project Project C:\xampp\htdocs\RestApi
  RESTAPI
    app
    bootstrap
    config
    database
      factories
        PEOPLEFactory.php
        UserFactory.php
      migrations
        2014_10_12_000000_create_user.php
        2014_10_12_100000_create_people.php
        2019_08_19_000000_create_failed_jobs.php
        2019_12_14_000001_create_personal_access_tokens.php
        2023_01_13_223845_create_people.php
      seeders
        DatabaseSeeder.php
        PEOPLESeeder.php
    .gitignore
    lang
    public
    resources
    routes
      api.php
      channels.php
      console.php
      web.php
    storage
    tests
    vendor
      .editorconfig
      .env
      .env.example
      .gitattributes
      .gitignore
      artisan
      composer.json
      composer.lock
      package.json
      phpunit.xml
      README.md
      vite.config.js
    External Libraries
    Scratches and Consoles
  Problems: File 11 Project Errors
  Version Control TODO Problems Terminal Services
PHP: 8.0 22:6 LF UTF-8 4 spaces
```

```
<?php
namespace App\Http\Controllers;

use App\Models\PEOPLE;
use App\Http\Requests\UpdatePEOPLERequest;
use App\Http\Requests\PEOPLEPostRequest;
use GuzzleHttp\Psr7\Request;

class PEOPLEController extends Controller
{
    public function index()
    {
        return PEOPLE::all();
    }

    public function create(Request $request)
    {
        return PEOPLE::create($request->all());
    }

    public function show(PEOPLE $id)
    {
        return PEOPLE::find($id);
    }

    public function update(UpdatePEOPLERequest $request, PEOPLE $id, $PEOPLE)
    {
        $PEOPLE = PEOPLE::find($id);
        $PEOPLE = update($request->all());
        return $PEOPLE;
    }

    public function delete(PEOPLE $id)
    {
        return PEOPLE::deleted($id);
    }
}
```

```
PS File Edit View Navigate Code Refactor Run Tools VCS Window Help RestApi - api.php
RestApi RESTAPI routes api.php
Project Project C:\xampp\htdocs\RestApi
  RESTAPI
    app
    bootstrap
    config
    database
      factories
        PEOPLEFactory.php
        UserFactory.php
      migrations
        2014_10_12_000000_create_user.php
        2014_10_12_100000_create_people.php
        2019_08_19_000000_create_failed_jobs.php
        2019_12_14_000001_create_personal_access_tokens.php
        2023_01_13_223845_create_people.php
      seeders
        DatabaseSeeder.php
        PEOPLESeeder.php
    .gitignore
    lang
    public
    resources
    routes
      api.php
      channels.php
      console.php
      web.php
    storage
    tests
    vendor
      .editorconfig
      .env
      .env.example
      .gitattributes
      .gitignore
      artisan
      composer.json
      composer.lock
      package.json
      phpunit.xml
      README.md
      vite.config.js
    External Libraries
    Scratches and Consoles
  Problems: File 6 Project Errors
  Version Control TODO Problems Terminal Services
PHP: 8.0 15:1 LF UTF-8 4 spaces
```

```
<?php
use Illuminate\Http\Request;
use App\Http\Controllers\PEOPLEController;
use Illuminate\Support\Facades\Route;
use App\Models\PEOPLE;

Route::get('/Witczak/305423/people', [PEOPLEController::class, 'index']);
Route::post('/Witczak/305423/people', [PEOPLEController::class, 'create']);
Route::get('/Witczak/305423/people/{id}', [PEOPLEController::class, 'show']);
Route::delete('/Witczak/305423/people/{id}', [PEOPLEController::class, 'delete']);
Route::put('/Witczak/305423/people/{id}', [PEOPLEController::class, 'update']);
Route::middleware('auth:sanctum')->get('/user', function (Request $request) {
    return $request->user();
});
```



Politechnika
Śląska



```
<?php
1
2
3 namespace App\Http\Requests;
4
5 use Illuminate\Foundation\Http\FormRequest;
6
7
8
9
10
11
12
13
14
15
16
17
18
19
20
21
22
23
24
25
26
27
28
29
30
class StorePEOPLERequest extends FormRequest
{
    /**
     * Determine if the user is authorized to make this request.
     *
     * @return bool
     */
    public function authorize()
    {
        return true;
    }

    /**
     * Get the validation rules that apply to the request.
     *
     * @return array<string, mixed>
     */
    public function rules()
    {
        return [
            //
        ];
    }
}
```



Politechnika
Śląska



```
<?php
1
2
3 namespace App\Http\Requests;
4
5 use Illuminate\Foundation\Http\FormRequest;
6
7 class UpdatePEOPLERequest extends FormRequest
8 {
9     /**
10      * Determine if the user is authorized to make this request.
11      *
12      * @return bool
13      */
14     public function authorize()
15     {
16         return true;
17     }
18
19     /**
20      * Get the validation rules that apply to the request.
21      *
22      * @return array<string, mixed>
23      */
24     public function rules()
25     {
26         return [
27             //
28         ];
29     }
30 }
```



Politechnika
Śląska



```
<?php
1
2
3 namespace Database\Factories;
4
5 use Illuminate\Database\Eloquent\Factories\Factory;
6
7 class PEOPLEFactory extends Factory
8 {
9     no usages
10     public function definition(): array
11     {
12         return [
13             'Name' => $this->faker->name(),
14             'LastName' => $this->faker->lastName(),
15             'street' => $this->faker->streetName(),
16             'City' => $this->faker->city(),
17             'Country' => $this->faker->country()
18         ];
19     }
20 }
```



Politechnika
Śląska



```
<?php
1
2
3 use ...
4
5
6
7 return new class extends Migration
8 {
9     no usages
10    public function up()
11    {
12        Schema::create( table: 'PEOPLE', function (Blueprint $table) {
13            $table->id();
14            $table->string( columns: 'Name');
15            $table->string( columns: 'LastName');
16            $table->string( columns: 'street');
17            $table->string( columns: 'Country');
18            $table->timestamps();
19        });
20    }
21    no usages
22    public function down()
23    {
24        Schema::dropIfExists( table: 'PEOPLE');
25    }
26};

class@anonymous - down()
```

PHP: 8.0 23:6 LF UTF-8 4 spaces



Politechnika
Śląska



```
<?php
1
2
3 namespace Database\Seeders;
4 use App\Models\PEOPLE;
5 use Illuminate\Database\Seeder;
6 class PEOPLESeeder extends Seeder
7 {
8     public function run()
9     {
10         PEOPLE::factory()
11             ->count( count: 50)
12             ->create();
13     }
14 }
15
```

PHP: 8.0 10:9 (65 chars, 2 line breaks) LF UTF-8 4 spaces



The screenshot shows an IDE window with the following components:

- Project Explorer (Left):** Displays the file structure of the 'RestApi' project. The 'seeds' folder is expanded, showing 'DatabaseSeeder.php' and 'PEOPLESeeder.php'. The 'tests' folder is also visible.
- Editor (Center):** Displays the contents of the '.env' file. The file is titled 'RestApi - .env'. The content includes configuration for Laravel, such as database connection details, cache settings, and session settings. The file is 32 lines long.
- Terminal (Bottom):** Shows the status bar at the bottom with the text 'PHP: 8.0 12:18 LF UTF-8 4 spaces'.

The .env file content is as follows:

```
1 APP_NAME=Laravel
2 APP_ENV=local
3 APP_KEY=base64:5/GqD5azXxCWGgw3h5Wk6yQAvIo7kyo2SkQEL1V9xw=
4 APP_DEBUG=true
5 APP_URL=http://localhost
6
7 LOG_CHANNEL=stack
8 LOG_DEPRECATIONS_CHANNEL=null
9 LOG_LEVEL=debug
10
11 DB_CONNECTION=mysql
12 DB_HOST=127.0.0.1
13 DB_PORT=3306
14 DB_DATABASE=restpeopleapi
15 DB_USERNAME=root
16 DB_PASSWORD=
17
18 BROADCAST_DRIVER=log
19 CACHE_DRIVER=file
20 FILESYSTEM_DISK=local
21 QUEUE_CONNECTION=sync
22 SESSION_DRIVER=file
23 SESSION_LIFETIME=120
24
25 MEMCACHED_HOST=127.0.0.1
26
27 REDIS_HOST=127.0.0.1
28 REDIS_PASSWORD=null
29 REDIS_PORT=6379
30
31 MAIL_MAILER=smtp
32 MAIL_HOST=mailhog
33 MAIL_PORT=1025
```