## Міністерство освіти і науки України

## Запорізький національний технічний університет

кафедра програмних засобів

# звіт

з лабораторної роботи № 5

з дисципліни «Якість програмного забезпечення» на тему:

# «Проведення тестів»

Виконав:

студент групи КНТ-115 Л.І. Кулик

Прийняли:

Н.О. Миронова

2018

## Мета роботи

Реалізувати тести із використанням сучасної мови програмування.

## Виконання роботи

Для реалізації тестів була обрана мова програмування PHP та фреймворк для проведення тестів PHPUnit. PHPUnit є представником сімейства фреймворків xUnit.

Далі наведено файл з тестами, який перевіряє позитивні та негативі сценарії виконання програми. Програма виконує CRUD операції.

Код файлу RoomType.php:

<?php

namespace Tests\Feature;

use App\User;

use Laravel\Passport\Passport;

use Tests\TestCase;

use Illuminate\Foundation\Testing\WithFaker;

use Illuminate\Foundation\Testing\RefreshDatabase;

class updateRoomTypeTest extends TestCase

{

/\*\*

\* A basic test example.

\*

\* @return void

\*/

public function testUpdateRoomType()

{

Passport::actingAs(

User::find(1)

);

$response = $this->put('/api/hotels/1/room-types/1', [

"id" => 1,

"hotel\_id" => 1,

"type" => "BEST TYPE EVER",

"photo" => "full/photoNAME.jpg",

]);

$response

->assertStatus(200)

->assertJson([

"id" => 1,

"hotel\_id" => 1,

"type" => "BEST TYPE EVER",

"photo" => "full/photoNAME.jpg",

]);

}

public function testUpdateWithoutToken()

{

$response = $this->put('/api/hotels/1/room-types/1', [

"id" => 1,

"hotel\_id" => 1,

"type" => "BEST TYPE EVER",

"photo" => "full/photoNAME.jpg",

]);

$response

->assertStatus(401);

}

public function testUpdateRoomTypeWithInvalidData()

{

Passport::actingAs(

User::find(1)

);

$response = $this->put('/api/hotels/1/room-types/1', [

"id" => '',

"type" => "BEST TYPE EVER",

"photo" => "full/photoNAME.jpg",

]);

$response

->assertStatus(422);

}

public function testUpdateRoomTypeWithNonexistentId()

{

Passport::actingAs(

User::find(1)

);

$response = $this->put('/api/hotels/1/room-types/0', [

"id" => 1,

"hotel\_id" => 1,

"type" => "BEST TYPE EVER",

"photo" => "full/photoNAME.jpg",

]);

$response

->assertStatus(404);

}

public function testUpdateRoomTypeWithNonexistentHotelId()

{

Passport::actingAs(

User::find(1)

);

$response = $this->put('/api/hotels/0/room-types/1', [

"id" => 1,

"hotel\_id" => 1,

"type" => "BEST TYPE EVER",

"photo" => "full/photoNAME.jpg",

]);

$response

->assertStatus(404);

}

public function testUpdateRoomTypeWithMismatchingIdInUrlAndRequest()

{

Passport::actingAs(

User::find(1)

);

$response = $this->put('/api/hotels/1/room-types/1', [

"id" => 3,

"hotel\_id" => 1,

"type" => "BEST TYPE EVER",

"photo" => "full/photoNAME.jpg",

]);

$response

->assertStatus(200)

->assertJson([

"id" => 1,

]);

}

public function testUpdateForeignRoomType()

{

Passport::actingAs(

factory(User::class)->create()

);

$response = $this->put('/api/hotels/1/room-types/1', [

"id" => 1,

"hotel\_id" => 1,

"type" => "BEST TYPE EVER",

"photo" => "full/photoNAME.jpg",

]);

$response

->assertStatus(401);

}

}

Результати тестування зображені на рисунку 1. З результатів видно більшість тестів пройдено успішно. А дві функції testBasicTest і testStartParser, які не пройшли тест, потребують перегляду та виправлення помилок.

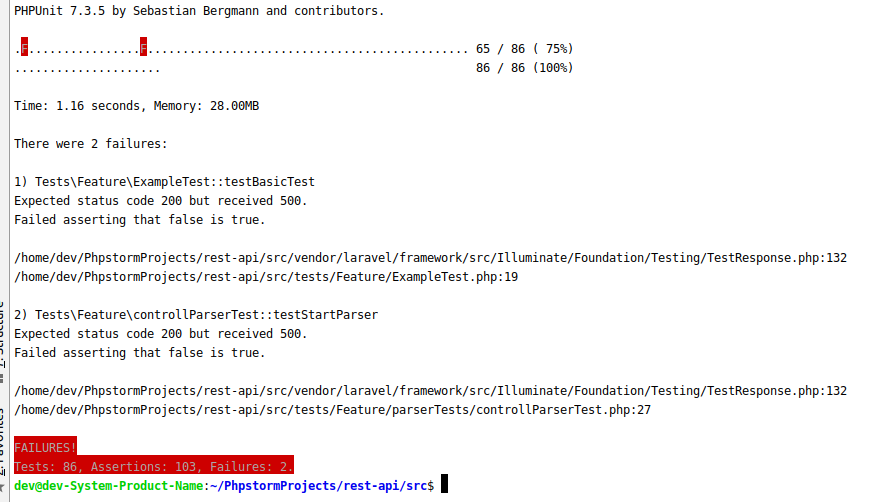


Рисунок 1 – Результати тестування

## Висновки

В ході виконання лабораторної роботи, було вивчено можливості PHP, щодо проведення тестування та здобуто навички написання тестів.