LAPORAN 5

Integratif Programming
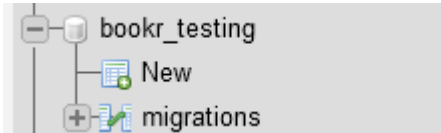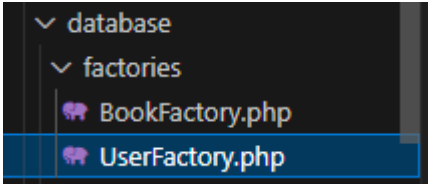
Respons Json

Disusun Oleh :

Nama : Ichsanul Dwi Prayitno

Nim : 4.33.20.0.14

Kelas : TI-4A

PROGRAM STUDI S.TR TEKNOLOGI REKAYASA KOMPUTER

JURUSAN TEKNIK ELEKTRO

POLITEKNIK NEGERI SEMARANG

TAHUN 2024

PRAKTEK

I. Lembar Kerja

No	Praktik	Hasil Program
| 1 | Test Database | Configurasi php unit untuk penggatian database saat testing dijalankan , buka file phpunit.xml dan tambahkan kode berikut<br><br>```
1  <!-- Test Database -->
2        <env name="DB_DATABASE" value="bookr_testing"/>
```<br><br>Setelah itu buatlah database baru dengan nama seperti diatas<br><br>```
bookr_testing
    New
    migrations
```<br><br>Lalu pada folder factories buatlah BookFactory.php . ini bisa di duplikat dari UserFactory.php<br><br>```
∨ database
  ∨ factories
     BookFactory.php
     UserFactory.php
```<br><br>Setelah terbuat gantilah bagian return dengan kode berikut, hal ini dibuat sebagai format testing data bohongan<br><br>```
1  public function definition()
2      {
3          $title = $this->faker->sentence(rand(3, 10));
4          return [
5              'title' => substr($title, 0, strlen($title) - 1),
6              'author' => $this->faker->name,
7              'description' => $this->faker->text,
8          ];
9      }
```|

Buka file BooksControllerTest dan tambahkan kode berikut untuk penggunaan fungsi migration pada saat testing dijalankan

```php
use Laravel\Lumen\Testing\DatabaseMigrations;

class BooksControllerTest extends TestCase
{
    use DatabaseMigrations;

```

Selanjutnya testing phpunit apakah sudah bekerja dengan baik atau belum. Jalankan comand

./VENDOR/BIN/PHPUNIT --

FILTER=INDEX_SHOULD_RETURN_A_COLLECTION_OF_RECORDS

```
PS D:\laragon\www\Intergrative Programming\Project> ./vendor/bin/phpunit --filter=index_should_return_a_collection_of_records
PHPUnit 10.5.11 by Sebastian Bergmann and contributors.

Runtime:       PHP 8.1.10
Configuration: D:\laragon\www\Intergrative Programming\Project\phpunit.xml

.                                                                   1 / 1 (100%)

Time: 00:04.649, Memory: 26.00 MB

OK (1 test, 2 assertions)
```

Lalu berikut penggatian metode test dengan menggunaakan factory yang telah dibuat

```php
public function index_should_return_a_collection_of_records()
{
    $books = Book::factory(2)->create();
    $this->get('/books');
    foreach ($books as $book) {
        $this->seeJson(['title' => $book->title]);
    }
}
```

2		

```php
public function show_should_return_a_valid_book()
{
    $book = Book::factory()->create();
    $this
        ->get("/books/{$book->id}")
        ->seeStatusCode(200)
        ->seeJson([
            'id' => $book->id,
            'title' => $book->title,
            'description' => $book->description,
            'author' => $book->author
        ]);

    $data = json_decode($this->response->getContent(), true);
    $this->assertArrayHasKey('created_at', $data);
    $this->assertArrayHasKey('updated_at', $data);
}
```

```php
public function update_should_only_change_fillable_fields()
{
    $book = Book::factory()->create([
        'title' => 'War of the Worlds',
        'description' => 'A science fiction masterpiece about Martians invading London',
        'author' => 'H. G. Wells',
    ]);

    $this->put("/books/{$book->id}", [
        'id' => 5,
        'title' => 'The War of the Worlds',
        'description' => 'The book is way better than the movie.',
        'author' => 'Wells, H. G.'
    ]);

    $this
        ->seeStatusCode(200)
        ->seeJson([
            'id' => 1,
            'title' => 'The War of the Worlds',
            'description' => 'The book is way better than the movie.',
            'author' => 'Wells, H. G.'
        ])
        ->seeInDatabase('books', [
            'title' => 'The War of the Worlds'
        ]);
}
```

```php
public function destroy_should_remove_a_valid_book()
{
    $book = Book::factory()->create();
    $this
        ->delete("/books/{$book->id}")
        ->seeStatusCode(204)
        ->isEmpty();

    $this->notSeeInDatabase('books', ['id' => $book->id]);
}
```

2	Better Error Response	Pada file handler.php import library Response agar tidak mendapatkan error

```php
use Symfony\Component\HttpFoundation\Response;
```

Lalu pada function render tambahkan kode berikut

```php
public function render($request, Throwable $exception)
{
    if ($request->wantsJson() && !($exception instanceof ValidationException)) {
        $response = [
            'message' => (string) $exception->getMessage(),
            'status' => 400
        ];
        if ($exception instanceof HttpException) {
            $response['message'] = Response::$statusTexts[$exception->getStatusCode()];
            $response['status'] = $exception->getStatusCode();
        }
        if ($this->isDebugMode()) {
            $response['debug'] = [
                'exception' => get_class($exception),
                'trace' => $exception->getTrace()
            ];
        }

        return response()->json(['error' => $response], $response['status']);
    }
    return parent::render($request, $exception);
}
```

Setelah itu tambahkan function baru

```php
public function isDebugMode()
{
    return (bool) env('APP_DEBUG');
}
```

Testing respon yang didapatkan dengan menjalankan comand berikut di terminal

CURL -H "ACCEPT:APPLICATION/JSON"

HTTP://LOCALHOST:8000/BOOKS/5555

```
C:\Users\LENOVO>CURL -H "ACCEPT:APPLICATION/JSON" HTTP://LOCALHOST:8000/FOO/BAR
{"error":{"message":"Not Found","status":404}}
C:\Users\LENOVO>
```

Selanjutnya install mockery dengan menjalankan comand

REQUIRE --DEV MOCKERY/MOCKERY

```
PS D:\laragon\www\Intergrative Programming\Project> composer require --dev mockery/mockery
./composer.json has been updated
Running composer update mockery/mockery
Loading composer repositories with package information
Updating dependencies
Lock file operations: 0 installs, 1 update, 0 removals
  - Upgrading mockery/mockery (1.6.7 => 1.6.11)
Writing lock file
Installing dependencies from lock file (including require-dev)
Package operations: 0 installs, 1 update, 0 removals
  - Downloading mockery/mockery (1.6.11)
  - Upgrading mockery/mockery (1.6.7 => 1.6.11): Extracting archive
Generating optimized autoload files
64 packages you are using are looking for funding.
Use the `composer fund` command to find out more!
No security vulnerability advisories found.
Using version ^1.6 for mockery/mockery
PS D:\laragon\www\Intergrative Programming\Project>
```

Buatlah file HandlerTest.php lalu tambahkan kode berikut pada file tersebut

```php
public function it_responds_with_html_when_json_is_not_accepted()
{
    // Make the mock a partial, we only want to mock the `isDebugMode` method
    $subject = m::mock(Handler::class)->makePartial();
    $subject->shouldNotReceive('isDebugMode');
    // Mock the interaction with the Request
    $request = m::mock(Request::class);
    $request->shouldReceive('wantsJson')->andReturn(false);
    // Set expectation for expectsJson method
    $request->shouldReceive('expectsJson')->andReturn(false); // Assuming JSON is not expected
    // Mock the interaction with the exception
    $exception = m::mock(\Exception::class, ['Error!']);
    $exception->shouldNotReceive('getStatusCode');
    $exception->shouldNotReceive('getTrace');
    $exception->shouldNotReceive('getMessage');
    // Call the method under test, this is not a mocked method.
    $result = $subject->render($request, $exception);

    // Assert that `render` does not return a JsonResponse

    $this->assertNotInstanceOf(JsonResponse::class, $result);
}
```

Lalu test function yang telah dibuat dengan menjalankan comand berikut

./VENDOR/BIN/PHPUNIT --
FILTER=IT_RESPONDS_WITH_HTML_WHEN_JSON_IS_NOT_ACCEPTED

```
PS D:\laragon\www\Intergrative Programming\Project> ./vendor/bin/phpunit --filter=it_responds_with_html_when_json_is_not_accepted
PHPUnit 10.5.11 by Sebastian Bergmann and contributors.

Runtime:       PHP 8.1.10
Configuration: D:\laragon\www\Intergrative Programming\Project\phpunit.xml

.                                                                   1 / 1 (100%)

Time: 00:01.103, Memory: 16.00 MB

OK (1 test, 1 assertion)
PS D:\laragon\www\Intergrative Programming\Project>
```

Pada TestCase.php tambahkan library Mockery agar Assertion bertambah

```
use Mockery\Adapter\Phpunit\MockeryPHPUnitIntegration;

abstract class TestCase extends BaseTestCase
{
    use MockeryPHPUnitIntegration;
```

Lalu test ulang

```
PS D:\laragon\www\Intergrative Prog
isplay_errors=On
PHPUnit 10.5.11 by Sebastian Bergma

Runtime:        PHP 8.1.10
Configuration: D:\laragon\www\Inter

.

Time: 00:01.061, Memory: 20.00 MB

OK (1 test, 5 assertions)
```

Lalu tambahkan function berikut

```
public function it_responds_with_json_for_json_consumers()
{
    $subject = m::mock(Handler::class)->makePartial();
    $subject
        ->shouldReceive('isDebugMode')
        ->andReturn(false);

    $request = m::mock(Request::class);
    $request
        ->shouldReceive('wantsJson')
        ->andReturn(true);

    $exception = m::mock(\Exception::class, ['Doh!']);
    $exception
        ->shouldReceive('getMessage')
        ->andReturn('Doh!');

    /** @var JsonResponse $result */
    $result = $subject->render($request, $exception);
    $data = $result->getData();

    $this->assertInstanceOf(JsonResponse::class, $result);
    // $this->assertObjectHasAttribute('error', $data);
    $this->assertEquals('Doh!', $data->error->message);
    $this->assertEquals(400, $data->error->status);
}
```

```
1   public function it_provides_json_responses_for_http_exceptions()
2       {
3           $subject = m::mock(Handler::class)->makePartial();
4           $subject
5               ->shouldReceive('isDebugMode')
6               ->andReturn(false);
7
8           $request = m::mock(Request::class);
9           $request->shouldReceive('wantsJson')->andReturn(true);
10
11          $examples = [
12              [
13                  'mock' => NotFoundHttpException::class,
14                  'status' => 404,
15                  'message' => 'Not Found'
16              ],
17              [
18                  'mock' => AccessDeniedHttpException::class,
19                  'status' => 403,
20                  'message' => 'Forbidden'
21              ],
22              [
23                  'mock' => ModelNotFoundException::class,
24                  'status' => 404,
25                  'message' => 'Not Found'
26              ]
27          ];
28
29          foreach ($examples as $e) {
30              $exception = m::mock($e['mock']);
31              $exception->shouldReceive('getMessage')->andReturn(null);
32              $exception->shouldReceive('getStatusCode')->andReturn($e['status']);
33
34              /** @var JsonResponse $result */
35              $result = $subject->render($request, $exception);
36              $data = $result->getData();
37
38              $this->assertEquals($e['status'], $result->getStatusCode());
39              $this->assertEquals($e['message'], $data->error->message);
40              $this->assertEquals($e['status'], $data->error->status);
41          }
42      }
```

Dan jalankan test file tersebut dengan menggunakan comand

./vendor/bin/phpunit --filter=HandlerTest


```
PS D:\laragon\www\Intergrative Prog
PHPUnit 10.5.11 by Sebastian Bergmar

Runtime:       PHP 8.1.10
Configuration: D:\laragon\www\Interg

...

Time: 00:01.027, Memory: 20.00 MB

OK (3 tests, 17 assertions)
PS D:\laragon\www\Intergrative Prog
```

Pada file BookController.php gantilah method show menjadi kode seperti berikut

```php
1   public function show($id)
2       {
3               return Book::findOrFail($id);
4       }
```

Lalu lakukan uji coba menggunakan comand

```
C:\Users\LENOVO>curl -i -H"Accept: application/json" http://localhost:8000/books/5555
HTTP/1.1 400 Bad Request
Host: localhost:8000
Date: Thu, 04 Apr 2024 14:54:25 GMT
Connection: close
X-Powered-By: PHP/8.1.10
Cache-Control: no-cache, private
Date: Thu, 04 Apr 2024 14:54:25 GMT
Content-Type: application/json

{"error":{"message":"No query results for model [App\\Models\\Book] 5555","status":400}}
C:\Users\LENOVO>
```

Pada file handler.php tambahkan kode elseif berikut untuk mengubah respon error yang diberikan

```php
1
2   if ($exception instanceof HttpException) {
3       $response['message'] = Response::$statusTexts[$exception->getStatusCode()];
4       $response['status'] = $exception->getStatusCode();
5   } else if ($exception instanceof ModelNotFoundException) {
6       $response['message'] = Response::$statusTexts[Response::HTTP_NOT_FOUND];
7       $response['status'] = Response::HTTP_NOT_FOUND;
8   }
```

Lakukan uji coba kembali

```
C:\Users\LENOVO>curl -i -H"Accept: application/json" http://localhost:8000/books/5555
HTTP/1.1 404 Not Found
Host: localhost:8000
Date: Thu, 04 Apr 2024 14:56:04 GMT
Connection: close
X-Powered-By: PHP/8.1.10
Cache-Control: no-cache, private
Date: Thu, 04 Apr 2024 14:56:04 GMT
Content-Type: application/json

{"error":{"message":"Not Found","status":404}}
C:\Users\LENOVO>
```

Jalankan phpunit untuk pengecekan

```
Runtime:       PHP 8.1.10
Configuration: D:\laragon\www\Intergrative Programming\Project\phpunit.xml

.......E........                                              16 / 16 (100%)

Time: 00:13.887, Memory: 32.00 MB

There was 1 error:

1) Tests\Feature\BooksControllerTest::show_should_fail_when_the_book_id_does_not_exist
Symfony\Component\HttpKernel\Exception\NotFoundHttpException: No query results for model [App\Models\Book] 99999
```

Untuk memperbaikinya buka file BookControllerTest

Lalu ubahlah function show_should_fail_when_the_book_id_does_not_exist menjadi seperti berikut

```php
public function show_should_fail_when_the_book_id_does_not_exist()
    {
        $this
            ->get('/books/99999', ['Accept' => 'application/json'])
            ->seeStatusCode(404)
            ->seeJson([
                'message' => 'Not Found',
                'status' => 404
            ]);
    }
```

Lalu jalankan phpunit lagi

```
PS D:\laragon\www\Intergrative Programming\Project> ./vendor/bin/phpunit
PHPUnit 10.5.11 by Sebastian Bergmann and contributors.

Runtime:       PHP 8.1.10
Configuration: D:\laragon\www\Intergrative Programming\Project\phpunit.xml

...............                                                  15 / 15 (100%)

Time: 00:11.402, Memory: 32.00 MB

OK (15 tests, 39 assertions)
```

Selanjutnya pada HandlerTest tambahkan function berikut

```
 1   public function it_provides_json_responses_for_http_exceptions()
 2   {
 3       $subject = m::mock(Handler::class)->makePartial();
 4       $subject
 5           ->shouldReceive('isDebugMode')
 6           ->andReturn(false);
 7
 8       $request = m::mock(Request::class);
 9       $request->shouldReceive('wantsJson')->andReturn(true);
10
11       $examples = [
12           [
13               'mock' => NotFoundHttpException::class,
14               'status' => 404,
15               'message' => 'Not Found'
16           ],
17           [
18               'mock' => AccessDeniedHttpException::class,
19               'status' => 403,
20               'message' => 'Forbidden'
21           ],
22           [
23               'mock' => ModelNotFoundException::class,
24               'status' => 404,
25               'message' => 'Not Found'
26           ]
27       ];
28
29       foreach ($examples as $e) {
30           $exception = m::mock($e['mock']);
31           $exception->shouldReceive('getMessage')->andReturn(null);
32           $exception->shouldReceive('getStatusCode')->andReturn($e['status']);
33
34           /** @var JsonResponse $result */
35           $result = $subject->render($request, $exception);
36           $data = $result->getData();
37
38           $this->assertEquals($e['status'], $result->getStatusCode());
39           $this->assertEquals($e['message'], $data->error->message);
40           $this->assertEquals($e['status'], $data->error->status);
41       }
42   }
```

Dan import librarynya

```
1   use Illuminate\Database\Eloquent\ModelNotFoundException;
```

Lalu jalankan phpunit

```
PS D:\laragon\www\Intergrative Programming\Project> ./vendor/bin/phpunit
PHPUnit 10.5.11 by Sebastian Bergmann and contributors.

Runtime:       PHP 8.1.10
Configuration: D:\laragon\www\Intergrative Programming\Project\phpunit.xml

................                                            16 / 16 (100%)

Time: 00:17.741, Memory: 32.00 MB

OK (16 tests, 48 assertions)
```

TEORI

I. Lembar Soal

N/A

II. Lembar Kerja

N/A