

LAPORAN 4
Integratif Programming
CRUD Books



Disusun Oleh :

Nama : Ichsanul Dwi Prayitno

Nim : 4.33.20.0.14

Kelas : TI-4A

PROGRAM STUDI S.TR TEKNOLOGI REKAYASA KOMPUTER

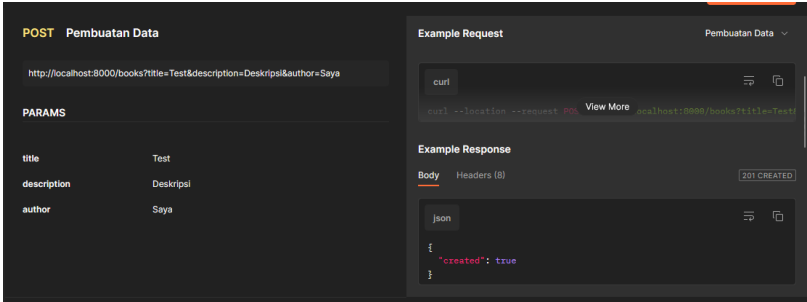
JURUSAN TEKNIK ELEKTRO

POLITEKNIK NEGERI SEMARANG

TAHUN 2024

PRAKTEK

I. Lembar Kerja

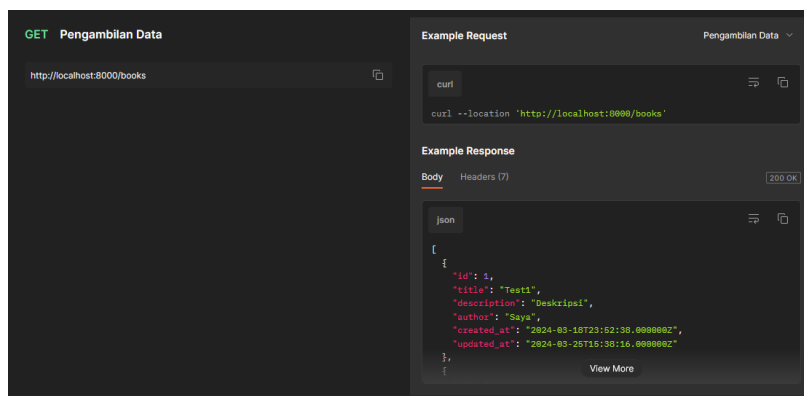
No	Praktik	Hasil Program
1	Create Books	<p>Pada web.php , tambahkan route berikut</p> <pre>1 \$router->post('/books', 'BooksController@store');</pre> <p>Lalu pada models Book.php tambahkan kode identifikasi table seperti berikut</p> <pre>1 protected \$fillable = ['title', 'description', 'author'];</pre> <p>Setelah itu tambahkan controller store</p> <pre>1 public function store(Request \$request) 2 { 3 \$book = Book::create(\$request->all()); 4 return response()->json(['created' => true], 201, [5 'Location' => route('books.show', ['id' => \$book->id]) 6]); 7 }</pre> <p>Testing ke postman</p> 
2	Read & Details	Tambahkan route berikut

```
1 $router->get('/books', 'BooksController@index');
2 $router->get('/books/{id:[\d]+}', [
3     'as' => 'books.show',
4     'uses' => 'BooksController@show'
5 ]);
```

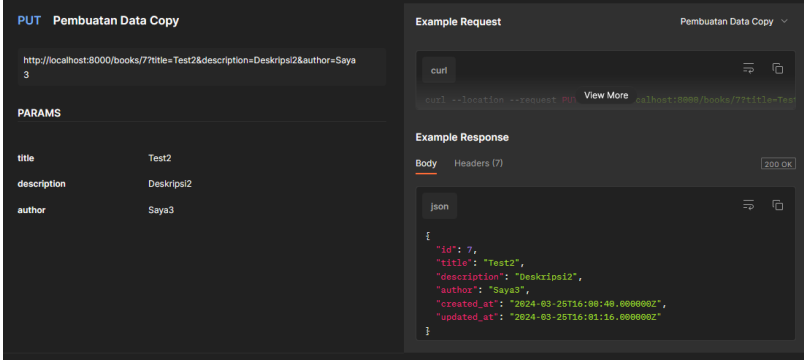

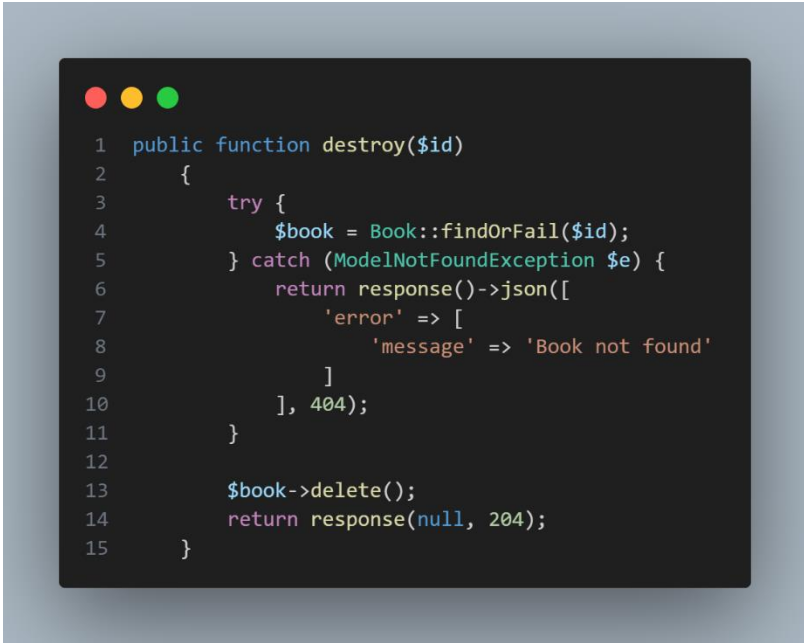
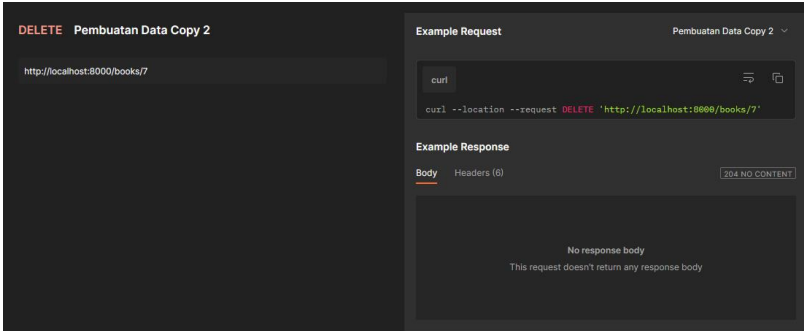
Lalu pada controller

```
1 public function index()
2 {
3     return Book::all();
4 }
5
6 public function show($id)
7 {
8     try {
9         return Book::findOrFail($id);
10    } catch (ModelNotFoundException $e) {
11        return response()->json([
12            'error' => [
13                'message' => 'Book not found'
14            ]
15        ], 404);
16    }
17 }
```

Testing ke postman



		 <p>GET Pengambil Data Details</p> <p>http://localhost:8000/books/7</p> <p>Example Request</p> <pre>curl --location 'http://localhost:8000/books/7'</pre> <p>Example Response</p> <p>Body Headers (7) 200 OK</p> <pre>json { "id": 7, "title": "Test", "description": "Deskripsi", "author": "Saya", "created_at": "2024-03-25T16:08:40.000000Z", "updated_at": "2024-03-25T16:08:40.000000Z" }</pre>
3	Update	<p>Tambahkan route berikut</p>  <pre>1 \$router->put('/books/{id:[\d]+}', 'BooksController@update');</pre> <p>Lalu pada controller tambahkan update</p>  <pre> 1 public function update(Request \$request, \$id) 2 { 3 try { 4 \$book = Book::findOrFail(\$id); 5 } catch (ModelNotFoundException \$e) { 6 return response()->json([7 'error' => [8 'message' => 'Book not found' 9] 10], 404); 11 } 12 13 \$book->fill(\$request->all()); 14 \$book->save(); 15 16 return \$book; 17 }</pre> <p>Testing ke postman</p>

		
4	Delete	<p>Tambahkan route berikut</p>  <p>Lalu pada controller tambahkan destroy</p>  <p>Testing ke postman</p> 

<https://documenter.getpostman.com/view/20942971/2sA35D4Nek>

TEORI

I. Lembar Soal

N/A

II. Lembar Kerja

N/A