

Laporan Modul 2: Laravel Fundamentasl

Mata Kuliah: Workshop Web Lanjut

Nama: Khairatil Aini

NIM: 2024573010031

Kelas: TI 2C

Abstrak

- Laporan ini membahas tentang membangun aplikasi laravel yang meliputi routing, middleware, penanganan request/response, controller, view, dan blade templating, dengan menggunakan MVC.
 - Tujuan laporan ini adalah untuk mendokumentasi proses pembelajaran dan pemahaman terhadap alur kerja laravel dari request ke response dan melatih untuk membangun aplikasi sederhana.
-

1. Dasar Teori

- MVC adalah singkatan dari model-view-controller, sebuah pola arsitektur perangkat lunak yang digunakan untuk memisahkan concern dalam aplikasih web:
 1. Model : Merepresentasikan data dan logika bisnis. Dalam Laravel, biasanya ditangani oleh Eloquent models yang berinteraksi dengan database.
 2. View : Menangani lapisan presentasi — yaitu HTML yang ditampilkan kepada pengguna. Laravel menggunakan Blade sebagai template engine untuk view.
 3. Controller : Mengatur alur antara model dan view. Controller menerima input dari pengguna (melalui route), memprosesnya (seringkali melibatkan model), dan mengembalikan view atau respons yang sesuai.
 - Konsep Routing adalah Laravel yang mendefinisikan bagaimana aplikasi merespons permintaan (request) yang masuk. Routing memetakan URI ke aksi controller atau closure. Route di Laravel didefinisikan di file routes/web.php untuk permintaan web dan di routes/api.php untuk route API.
 - Fungsi Middleware adalah menyediakan mekanisme untuk memfilter permintaan HTTP yang masuk ke aplikasi. Middleware bertindak sebagai lapisan yang berada di antara siklus request dan response.
 - Caranya adalah melalui alur kerja yang terstruktur yang menggunakan konsep HTTP dan MVC, seperti mengakses data request, form request, mengembalikan response.
 - Controller yang mengatur alur dan logika, sedangkan view mengatur untuk hasil yang di tampilkan.
 - Blade merupakan template engine yang disediakan oleh laravel, dan juga memudahkan dalam penulisan kode HTML.
-

2. Langkah-Langkah Praktikum

2.1 Praktikum 1 – Route, Controller, dan Blade View

- Buka projects calculator yang sudah dibuat, lalu buka folder routes dan cari file web.php

```
projects > calculator > routes > web.php
1  <?php
2
3  use Illuminate\Support\Facades\Route;
4  use App\Http\Controllers\WelcomeController;
5
6  Route::get('/calculator', [CalculatorController::class, 'index']);
7  Route::post('/calculator', [CalculatorController::class, 'calculate'])->name('calculator.calculate');
8  Route::get('/welcome', [WelcomeController::class, 'show']);
9
10 Route::get('/', function () {
11     return view('welcome');
12 });
13
```

- Buat controller WelcomeController.

```
acer@DESKTOP-V6N6DBU MINGW64 /d/SEMESTER 3/workshop web lanjut/web-lanjut-2024573010031/projects/calculator (
• $ php artisan make:controller WelcomeController

INFO Controller [D:\SEMESTER 3\workshop web lanjut\web-lanjut-2024573010031\projects\calculator\app\Http\
ers\WelcomeController.php] created successfully.
```

- Buat view mywelcome.blade.php.

```
projects > calculator > resources > views > mywelcome.blade.php
1  <!DOCTYPE html>
2  <html>
3      <head>
4          <title>Welcome to laravel</title>
5      </head>
6      <body>
7          <h1>
8              Welcome to Laravel 12
9          </h1>
10         <p>
11             Hello, {{$message}}
12         </p>
13     </body>
14 </html>
```

- Jalankan aplikasi dan tunjukkan hasil di browser.
- perintah menjalankan nya

```
acer@DESKTOP-V6N6DBU MINGW64 /d/SEMESTER 3/workshop web lanjut/web-lanjut-2024573010031/projects/calculator (main)
$ php artisan serve

INFO Server running on [http://127.0.0.1:8000].
```



- hasil nya

2.2 Praktikum 2 – Membuat Aplikasi Sederhana "Calculator"

- Tambahkan route untuk kalkulator.

```
projects > calculator > routes > web.php
1  <?php
2
3  use Illuminate\Support\Facades\Route;
4  use App\Http\Controllers\WelcomeController;
5  use App\Http\Controllers\CalculatorController;
6
7  Route::get('/calculator', [CalculatorController::class, 'index']);
8  Route::post('/calculator', [CalculatorController::class, 'calculate'])->name('calculator.calculate');
9  Route::get('/welcome', [WelcomeController::class, 'show']);
10
11 Route::get('/', function () {
12     return view('welcome');
13 });
14
```

- Buat controller CalculatorController.

```
acer@DESKTOP-V6N60BU MINGW64 /d/SEMESTER 3/workshop web lanjut/web-lanjut-2024573010031/projects/calculator (main)
$ php artisan make:controller CalculatorController

INFO Controller [D:\SEMESTER 3\workshop web lanjut\web-lanjut-2024573010031\projects\calculator\app\Http\Controllers\CalculatorController.php] created
successfully.
```

- Tambahkan view calculator.blade.php.

```

projects > calculator > resources > views > calculator.blade.php
1  <!DOCTYPE html>
2  <html>
3  <head>
4      <title>Laravel Calculator</title>
5  </head>
6  <body>
7      <h1>Kalkulator Sederhana</h1>
8
9      @if ($errors->any())
10         <div style="color: red;">
11             <ul>
12                 @foreach ($errors->all() as $error)
13                     <li>{{ $error }}</li>
14                 @endforeach
15             </ul>
16         </div>
17     @endif
18
19     <form method="POST" action="{{ route('calculator.calculate') }}">
20         @csrf
21         <input type="number" name="number1"
22             value="{{ old('number1', $number1 ?? '') }}"
23             placeholder="Angka Pertama" required>
24
25         <select name="operator" required>
26             <option value="add" {{ ($operator ?? '') == 'add' ? 'selected' : '' }}></option>
27             <option value="sub" {{ ($operator ?? '') == 'sub' ? 'selected' : '' }}></option>
28             <option value="mul" {{ ($operator ?? '') == 'mul' ? 'selected' : '' }}></option>
29             <option value="div" {{ ($operator ?? '') == 'div' ? 'selected' : '' }}></option>
30         </select>
31
32         <input type="number" name="number2"
33             value="{{ old('number2', $number2 ?? '') }}"
34             placeholder="Angka Kedua" required>
35
36         <button type="submit">Hitung</button>
37     </form>
38
39     @isset($result)
40         <h3>Hasil: {{ $result }}</h3>
41     @endisset
42 </body>
43 </html>
44

```

- Jalankan aplikasi dan coba dengan beberapa input berbeda.

← → ↻ ⓘ 127.0.0.1:8000/calculator

Kalkulator Sederhana

- hasilnya calculator

← → ↻ ⓘ 127.0.0.1:8000/calculator

Kalkulator Sederhana

Hasil: 22

- hasil penjumlahan

← → ↺ 127.0.0.1:8000/calculator

Kalkulator Sederhana

- ▼ Hitung

Hasil: 2

- hasil pengurangan
- hasil perkalian

← → ↺ 127.0.0.1:8000/calculator

Kalkulator Sederhana

* ▼ Hitung

Hasil: 10

Kalkulator Sederhana

/ ▼ Hitung

Hasil: 2

- hasil pembagian

3. Hasil dan Pembahasan

- iya sesuai harapan saya, tapi dari projects ini saya belajar bawah kesalahan bukan bukanlah hal yang mudah dan kita harus benar- benar memahami di mana salah nya.
- Apa yang terjadi jika ada input yang salah? mengalami error.
- mengirimkan data, Menjalankan validasi dan mengatur validasi.
- 1. Route berfungsi sebagai penghubung
 2. controller berfungsi sebgai logika aplikasi
 3. view berfungsi sebagai tampilan yang di lihat langsung oleh pengguna

4. Kesimpulan

Dari praktikum ini yang dapat di ambil adalah laravel dengan konsep MVC yang memudahkan pengembangan aplikasi web terstruktur. Route yang menghubungkan URL dengan logika, controller yang mengatur proses aplikasi, dan view yang menampilkan hasil blade. pembuatan aplikasi calculator sederhana yang membuktikan cara laravel menangani input, validasi, dan menampilkan output.

5. Referensi

1. <https://laravel.com/docs>
 2. <https://www.cloudways.com/blog/routing-in-laravel/>
-