

Laporan Modul 3: Laravel Controller

Mata Kuliah: Workshop Web Lanjut

Nama: Adha Gusti Harmadhan

NIM: 2024573010009

Kelas: 2B TI

Abstrak

Laporan ini berisi hasil praktikum dari **Modul 3: Laravel Controller** pada mata kuliah Workshop Web Lanjut. Tujuan laporan ini adalah untuk memahami fungsi dan peran controller dalam framework Laravel serta bagaimana controller mengelola logika aplikasi, menangani request, dan mengembalikan response ke view. Praktikum ini meliputi tiga percobaan utama: (1) menangani request dan response melalui controller, (2) menggunakan route grouping, dan (3) menerapkan prefix dan namespace dalam pengelompokan rute. Melalui praktikum ini, penulis belajar bagaimana cara membuat, mengelola, serta mengorganisasi controller agar aplikasi Laravel lebih terstruktur dan mudah dikembangkan.

1. Dasar Teori

- **Apa itu Controller di Laravel**

Controller adalah komponen pada arsitektur MVC (Model–View–Controller) yang berfungsi sebagai penghubung antara *Model* dan *View*. Controller bertugas menangani request dari pengguna melalui route, memproses logika bisnis, berinteraksi dengan data dari model, lalu mengembalikan hasilnya ke view untuk ditampilkan.

- **Jenis-Jenis Controller di Laravel**

1. **Basic Controller:** Controller standar yang dapat berisi beberapa metode untuk berbagai endpoint.
2. **Resource Controller:** Controller yang mengikuti pola RESTful CRUD dengan metode bawaan seperti index, store, update, dan destroy.
3. **Invokable Controller:** Controller dengan satu metode `_invoke()` yang cocok untuk endpoint tunggal.

- **Konsep Routing dengan Controller**

Routing menghubungkan URL dengan metode dalam controller. Misalnya:

```
Route::get('/home', [PageController::class, 'home']);
```

Artinya setiap kali pengguna mengakses `/home`, maka metode `home()` di `PageController` akan dijalankan.

- **Grouping Route dan Prefix**

Laravel memungkinkan pengelompokan rute untuk menjaga kerapian kode. Contohnya:

```
Route::controller(UserController::class)->group(function () {
    Route::get('/users', 'index');
    Route::get('/users/{id}', 'show');
});
```

Selain itu, prefix seperti admin/ dapat digunakan untuk mengelompokkan rute dengan struktur tertentu, misalnya untuk area dashboard admin.

- **Injeksi Request dan Validasi di Controller**

Laravel menyediakan *dependency injection* sehingga objek `Request` dapat langsung digunakan dalam metode controller.

```
public function store(Request $request) {
    $request->validate([
        'name' => 'required',
        'email' => 'required|email'
    ]);
}
```

- **Jenis Response dari Controller**

Controller dapat mengembalikan berbagai jenis response:

- View: `return view('welcome');`
- JSON: `return response()->json(['success' => true]);`
- Redirect: `return redirect()->route('home');`

2. Langkah-Langkah Praktikum

2.1 Praktikum 1 – Menangani Request dan Response View

Langkah-langkah:

1. Buat proyek baru:

```
adhag@Adhagusti MINGW64 /c/web_lanjut/web-lanjut-2024573010009 (main)
$ composer create-project laravel/laravel:^12.0.3 lab-view
```

2. Buat controller:

```
adhag@Adhagusti MINGW64 /c/web_lanjut/web-lanjut-2024573010009/projects/lab-view (main)
$ php artisan make:controller DemoController
INFO Controller [C:\web_lanjut\web-lanjut-2024573010009\projects\lab-view\app\Http\Controllers\DemoController.php] created successfully.
```

3. Tambahkan metode berikut ke `DemoController.php`:

projects > lab-view > app > Http > Controllers > DemoController.php

```

1  <?php
2
3  namespace App\Http\Controllers;
4  use Illuminate\Http\Request;
5
6  class DemoController extends Controller
7  {
8      // Simple data passing
9      public function hello()
10     {
11         $name = 'Laravel Learner';
12         return view('hello', ['name'=> $name]);
13     }
14
15     // Parameterized Route
16     public function greet($name)
17     {
18         return view('greet', ['name'=> ucfirst($name)]);
19     }
20
21     // query string
22     public function search(Request $request)
23     {
24         $keyword = $request->query('q', 'none');
25         return view('search', ['keyword'=> $keyword]);
26     }
27 }
```

4. Tambahkan route pada routes/web.php:

projects > lab-view > routes > web.php

```

1  <?php
2
3  use Illuminate\Support\Facades\Route;
4  use App\Http\Controllers\DemoController;
5
6  Route::get('/hello', [DemoController::class, 'hello']);
7  Route::get('/greet/{name}', [DemoController::class, 'greet']);
8  Route::get('/search', [DemoController::class, 'search']);
```

5. Buat tiga view di resources/views/:

- **hello.blade.php**

```
1  <!DOCTYPE html>
2  <html>
3  <head>
4  |   <title>Hello</title>
5  </head>
6  <body>
7  |   <h1>Hello, {{ $name }}!</h1>
8  </body>
9  </html>
```

- **greet.blade.php**

```
<!DOCTYPE html>
<html>
<head>
|   <title>Greet</title>
</head>
<body>
|   <h1>Nice to meet you, {{ $name }}!</h1>
</body>
</html>
```

- **search.blade.php**

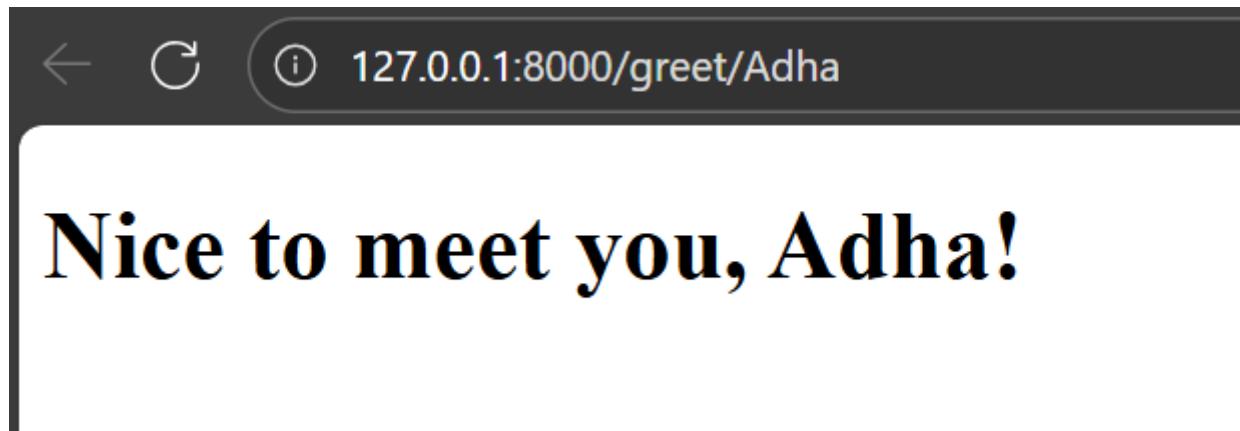
```
<!DOCTYPE html>
<html>
<head>
|   <title>Search</title>
</head>
<body>
|   <h1>You searched for: <strong>{{ $keyword }}</strong></h1>
</body>
</html>
```

Hasil Pengujian:

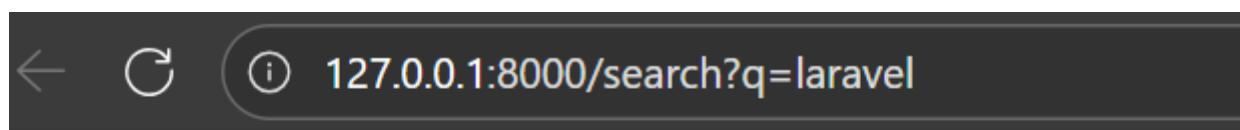
- <http://127.0.0.1:8000/hello> → menampilkan **Hello, Laravel Learner!**



- <http://127.0.0.1:8000/greet/Adha> → menampilkan **Nice to meet you, Adha!**



- <http://127.0.0.1:8000/search?q=laravel> → menampilkan **You searched for: laravel**



You searched for: laravel

2.2 Praktikum 2 – Menggunakan Group Route

Langkah-langkah:

1. Buat proyek baru:

```
adhag@Adhagusti MINGW64 /c/web_lanjut/web-lanjut-2024573010009/projects/lab-group (main)
$ composer create-project laravel/laravel:^12.0.3 lab-group
```

2. Buat controller:

```
adhag@Adhagusti MINGW64 /c/web_lanjut/web-lanjut-2024573010009/projects/lab-group (main)
$ php artisan make:controller PageController

INFO Controller [C:\web_lanjut\web-lanjut-2024573010009\projects\lab-group\app\Http\Controllers\PageController.php] created successfully.
```

3. Tambahkan metode ke `PageController.php`:

```
adhag@Adhagusti MINGW64 /c/web_lanjut/web-lanjut-2024573010009/projects/lab-group (main)
$ php artisan make:controller PageController

INFO Controller [C:\web_lanjut\web-lanjut-2024573010009\projects\lab-group\app\Http\Controllers\PageController.php] created successfully.
```

4. Definisikan route:

```
use App\Http\Controllers\PageController;

Route::controller(PageController::class)->group(function () {
    Route::get('/', 'home')->name('home');
    Route::get('/about', 'about')->name('about');
    Route::get('/contact', 'contact')->name('contact');
});
```

5. Buat folder resources/views/pages/ dengan file:

- **home.blade.php**

```
<!DOCTYPE html>
<html>
<head>
|   <title>Home</title>
</head>
<body>
|   <h1>{{ $message }}</h1>
</body>
</html>
```

- **about.blade.php**

```
lab-group > resources > views > pages >  about.blade.php
1  <!DOCTYPE html>
2  <html>
3  <head>
4  |   <title>About</title>
5  </head>
6  <body>
7  |   <h1>{{ $message }}</h1>
8  </body>
9  </html>
```

- **contact.blade.php**

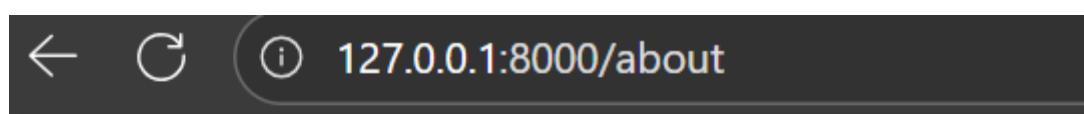
```
<!DOCTYPE html>
<html>
<head>
|   <title>Contact</title>
</head>
<body>
|   <h1>{{ $message }}</h1>
</body>
</html>
```

Hasil Pengujian:

- <http://127.0.0.1:8000/> → menampilkan **Welcome to the homepage.**



- <http://127.0.0.1:8000/about> → menampilkan **This is the about page.**



This is the about page.

- <http://127.0.0.1:8000/contact> → menampilkan **Reach us through the contact page.**



2.3 Praktikum 3 – Prefix dan Namespace Controller

Langkah-langkah:

1. Buat proyek:

```
adhag@Adhagusti MINGW64 /c/web_lanjut/web-lanjut-2024573010009/projects/lab-prefix (main)
$ composer create-project laravel/laravel:^12.0.3 lab-prefix
```

2. Buat controller dalam namespace Admin:

```
adhag@Adhagusti MINGW64 /c/web_lanjut/web-lanjut-2024573010009/projects/lab-prefix (main)
$ php artisan make:controller Admin/DashboardController

INFO Controller [C:\web_lanjut\web-lanjut-2024573010009\projects\lab-prefix\app\Http\Controllers\Admin\DashboardController.php] created successfully.

adhag@Adhagusti MINGW64 /c/web_lanjut/web-lanjut-2024573010009/projects/lab-prefix (main)
$ php artisan make:controller Admin/UserController

INFO Controller [C:\web_lanjut\web-lanjut-2024573010009\projects\lab-prefix\app\Http\Controllers\Admin\UserController.php] created successfully.
```

3. Tambahkan aksi pada controller:

- **DashboardController.php**

```
class DashboardController extends Controller
{
    public function index()
    {
        return view('admin.dashboard',
            ['message'=>'Welcome to Admin Dashboard']);
    }
}
```

- **UserController.php**

```
class UserController extends Controller
{
    public function index()
    {
        $users = ['Adha', 'Ridha', 'PB'];
        return view('admin.users.index', compact('users'));
    }

    public function show($id)
    {
        $user = "User #" . $id;
        return view('admin.users.show', compact('user'));
    }
}
```

4. Tambahkan route:

```
use App\Http\Controllers\Admin\DashboardController;
use App\Http\Controllers\Admin\UserController;

Route::prefix('admin')->group(function () {
    Route::get('/dashboard', [DashboardController::class, 'index'])->name('admin.dashboard');
    Route::get('/users', [UserController::class, 'index'])->name('admin.users');
    Route::get('/users/{id}', [UserController::class, 'show'])->name('admin.users.show');
});
```

5. Buat view di `resources/views/admin/`:

- **dashboard.blade.php**

```
<!DOCTYPE html>
<html>
<head>
    <title>Admin Dashboard</title>
</head>
<body>
    <h1>{{ $message }}</h1>
</body>
</html>
```

- **users/index.blade.php**

```
<!DOCTYPE html>
<html>
<head>
    <title>Users</title>
</head>
<body>
    <h1>User List</h1>
    <ul>
        @foreach ($users as $user)
            <li>{{ $user }}</li>
        @endforeach
    </ul>

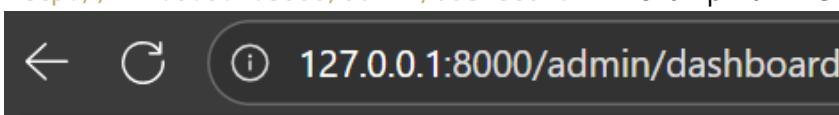
</body>
</html>
```

- **users/show.blade.php**

```
<!DOCTYPE html>
<html>
<head>
    <title>User Detail</title>
</head>
<body>
    <h1>Details for: {{ $user }}</h1>
</body>
</html>
```

Hasil Pengujian:

- <http://127.0.0.1:8000/admin/dashboard> → menampilkan **Welcome to Admin Dashboard**



Welcome to Admin Dashboard

- <http://127.0.0.1:8000/admin/users> → menampilkan **User List**.

The screenshot shows a web browser window with the URL 127.0.0.1:8000/admin/users in the address bar. The main content area displays the title "User List" in large, bold, dark blue font. Below it is a bulleted list of user names: "Adha", "Ridha", and "PB".

- <http://127.0.0.1:8000/admin/users/{id}> → menampilkan **Details for : user #adha**

The screenshot shows a web browser window with the URL 127.0.0.1:8000/admin/users/adha in the address bar. The main content area displays the title "Details for: User #adha" in large, bold, dark blue font.

3. Hasil dan Pembahasan

Dari ketiga praktikum, dapat disimpulkan bahwa:

- Controller mampu memisahkan logika dari tampilan, sehingga kode lebih terorganisir.
- Laravel mendukung berbagai jenis controller untuk fleksibilitas pengembangan.
- Group route dan prefix membuat struktur rute lebih rapi dan efisien.
- Data dapat dikirim dari controller ke view menggunakan array atau fungsi `compact()`.
- Namespace membantu mengelompokkan controller dalam folder berbeda, misalnya untuk area admin.

Keseluruhan praktikum berjalan dengan baik tanpa error, dan hasil tampilan sesuai harapan di setiap URL yang diuji.

4. Kesimpulan

Dari praktikum ini dapat disimpulkan bahwa controller merupakan bagian penting dari Laravel yang berfungsi mengatur alur logika aplikasi. Melalui controller, data bisa diolah dan dikirim ke view dengan cara yang rapi. Praktikum juga menunjukkan pentingnya penggunaan grouping route dan prefix untuk menjaga struktur aplikasi agar tetap mudah dibaca dan dikembangkan ke depannya.

5. Referensi

- Modul 3 - Laravel Controller — <https://hackmd.io/@mohdrzu/B1zwKEK5xe>
- Dokumentasi Resmi Laravel 12 — <https://laravel.com/docs/12.x/controllers>