

LEMBAR KERJA MAHASISWA
MODUL 11

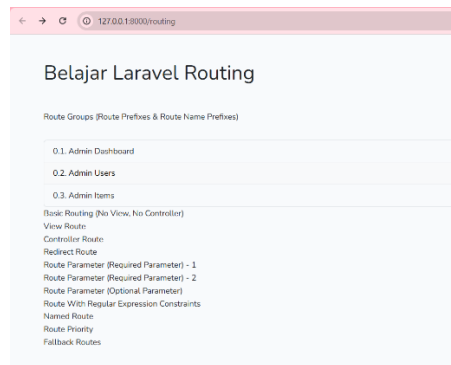


Oleh :

Azrabelva Juventia (IS-06-02 – 1204230083)

PROGRAM STUDI SISTEM INFORMASI
FAKULTAS REKAYASA INDUSTRI
TELKOM UNIVERSITY SURABAYA
2024

1. Generate Laravel Project dan Laravel UI



```
- Installing spatie/flare-client-php (1.8.0): Extracting archive
- Installing spatie/ignition (1.15.0): Extracting archive
- Installing spatie/laravel-ignition (2.8.0): Extracting archive
64 package suggestions were added by new dependencies, use `composer suggest` to see details.
Generating optimized autoload files
> illuminate\Foundation\ComposerScripts::postAutoloadDump
> @php artisan package:discover --ansi

INFO Discovering packages.

laravel/sail .....
... DONE
laravel/sanctum .....
... DONE
laravel/tinker .....
... DONE
nesbot/carbon .....
... DONE
nunomaduro/collision .....

INFO Discovering packages.

laravel/sail .....
... DONE
laravel/sanctum .....
... DONE
laravel/tinker .....
... DONE
laravel/ui .....
... DONE
nesbot/carbon .....
... DONE
nunomaduro/collision .....
... DONE
nunomaduro/termwind .....
... DONE
spatie/laravel-ignition .....
... DONE
```

1. Bundling Asset dengan Vite

```
PS C:\Users\ASUS\OneDrive\Documents\project laravel\Modul11-app> npm install

added 37 packages, and audited 38 packages in 14s

11 packages are looking for funding
  run `npm fund` for details

found 0 vulnerabilities
```

Npm sendiri adalah suatu alat yang kita gunakan untuk mengelola file seperti java script dan CSS. Pada perinah ini kita akan mengunduh semua hal yang akan dibutuhkan bootstrap. Dalam penginstalan npm ini juga dapat membaca file konfigurasi Bernama package.json yang akan berguna untuk megetahui daftar Pustaka yang akan diunduh

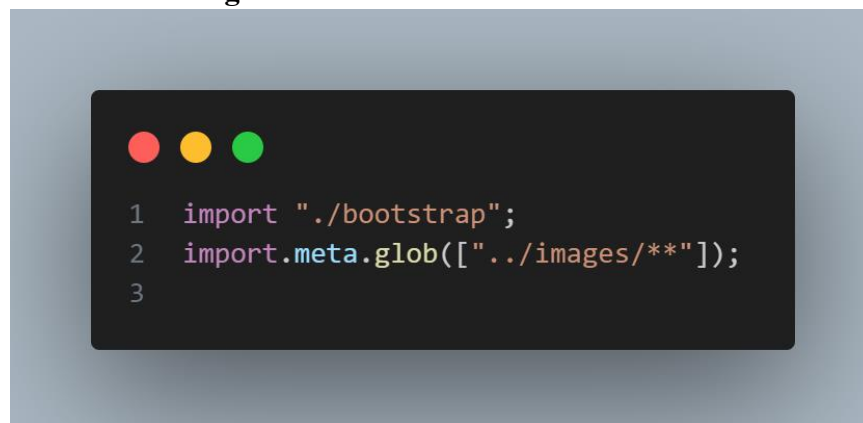
2. Terapkan fitur “Refreshing on Save”



```
1 import { defineConfig } from 'vite';
2 import laravel from 'laravel-vite-plugin';
3
4 export default defineConfig({
5   plugins: [
6     laravel({
7       input: [
8         'resources/sass/app.scss',
9         'resources/js/app.js',
10      ],
11      refresh: true,
12    }),
13  ],
14 });
15
```

Pada vite.config.js memungkinkan aplikasi web yang sedang kita kembangkan agar otomatis me refresh browser setiap kali kita menyimpan suatu perubahan dalam sebuah file . dalam hal ini vite.config sangat berguna untuk membantuk pengembangan agar dapat melihat perubahan secara langsung tanpa perlu merifresh browser secara manual

3. Terapkan fitur “Processing Static Assets With Vite”



```
1 import './bootstrap';
2 import.meta.glob(["../images/**"]);
3
```

Pada kode ini kita dapat mengelola file statis dengan mudah dan efisien emnggunakan struktur direktori yang sudah ada. Danvite akan secara otomatis merujuk ke asset asset saat hendak membuat aplikasi untuk memastikan bahwa file dapat diatasi dengan optimal. Dalam hal ini juga import.meta.glob(["../images/**"]); dapat dgunakan untuk mengidentifikasi dan mengimpor semua file yang ada dalam folder gambar dan juga subfolddernya

4. Buka file View bernama **welcome.blade.php**

```
1 <!DOCTYPE html>
2 <html lang="en">
3 <head>
4   <meta charset="UTF-8">
5   <meta http-equiv="X-UA-Compatible" content="IE=edge">
6   <meta name="viewport" content="width=device-width, initial-scale=1.0">
7   <title>Welcome</title>
8   @vite('resources/sass/app.scss')
9 </head>
10 <body>
11   <div class="container m-5">
12     {{-- Contoh cara mereferensikan gambar di dalam file blade dengan menggunakan pendekatan Vite --}}
13     
14   </div>
15   @vite('resources/js/app.js')
16 </body>
17 </html>
18
```

Laravel menyediakan vite() untuk membantu mengintegrasikan asset yang diproses oleh vite ke dalam basis file blade. @vite('resources/sass/app.scss'): dapat mengarahkan Laravel untuk memuat file CSS yang akan diproses oleh vite sebelum dimuat

5. Jalankan Vite

```
found 0 vulnerabilities
PS C:\Users\ASUS\OneDrive\Documents\project laravel\Modul11-app> npm run dev

> dev
> vite
```

```
→ APP_URL: http://localhost
PS C:\Users\ASUS\OneDrive\Documents\project laravel\Modul11-app> npm run build

> build
> vite build
```

6. Instal bootstrap dan bootstrap icon

```
PS C:\Users\ASUS\OneDrive\Documents\project laravel\Modul11-app> npm install bootstrap@5.3.0-alpha3 bootstrap-icons @popperjs/core
added 1 package, changed 1 package, and audited 39 packages in 6s

12 packages are looking for funding
  run `npm fund` for details

found 0 vulnerabilities
PS C:\Users\ASUS\OneDrive\Documents\project laravel\Modul11-app> php artisan make:controller RouteController
>>
```

```
1 // Fonts
2 // @import url("https://fonts.bunny.net/css?family=Nunito");
3
4 // Variables
5 // @import "variables";
6
7 // Bootstrap
8 @import "bootstrap/scss/bootstrap";
9 @import "bootstrap-icons/font/bootstrap-icons.css";
10
```

@import "bootstrap/scss/bootstrap"; digunakan untuk mengimpor file SCSS utama Bootstrap. Bagian aktif pada kode diatas berisi gaya dan komponen bootstrap biasa. Komentar pada font digunakan untuk mengomentari import untuk font karena tidak diperlukan pada implementasi saat ini. Sedangkan variable akan mengomentari import variables jika kita tidak menggunakan variable SCSS kustom

7. Praktik Laravel Routing

```
1 Route::get('/routing', function() {  
2     return view('routing');  
3 });
```

8. Routing.blade.php

```
1 <!DOCTYPE html>  
2 <html lang="en">  
3 <head>  
4     <meta charset="UTF-8">  
5     <meta http-equiv="X-UA-Compatible" content="IE=edge">  
6     <meta name="viewport" content="width=device-width, initial-scale=1.0">  
7     <title>Welcome</title>  
8     @vite('resources/sass/app.scss')  
9 </head>  
10 <body>  
11     <div class="container m-5">  
12         {{-- Contoh cara mereferensikan gambar di dalam file blade dengan menggunakan pendekatan Vite --}}  
13           
14     </div>  
15     @vite('resources/js/app.js')  
16 </body>  
17 </html>  
18
```

Pemisahan antara kode view dan logika rute dapat membuat proyek lebih terstruktur , dan helper route() memastikan URL berubah secara otomatis jika nama rute diubah

9. Basic Routing

```
1 Route::get('/basic_routing', function() {  
2     return 'Hello World';  
3 });
```

Basic Routing di Laravel digunakan untuk mendefinisikan rute yang sederhana tanpa memerlukan controller atau view. Kita cukup mendefinisikan rute menggunakan `Route::get()` dan memberikan respons langsung. Kita dapat mendefinisikan rute yang menampilkan sebuah pesan teks Ketika diakses dan menggunakan sebuah tautan untuk mengarahkannya. Basic routing ini dapat memudahkan kita untuk emnagani jenis tampilan yang terbilang rumit

10. View Route

```
1 <!DOCTYPE html>  
2 <html lang="en">  
3 <head>  
4     <meta charset="UTF-8">  
5     <meta name="viewport" content="width=device-width, initial-scale=1.0">  
6     <meta http-equiv="X-UA-Compatible" content="ie=edge">  
7     <title>View Route</title>  
8     @vite('resources/sass/app.scss')  
9 </head>  
10 <body>  
11     <div class="container m-5">  
12         <h1>This is from View Route</h1>  
13         <p>Hello, My name is {{ $name }}</p>  
14     </div>  
15     @vite('resources/js/app.js')  
16 </body>  
17 </html>  
18
```

Tujuan file `view.route.blade.php` adalah view Laravel yang menampilkan data dinamis seperti nama pengguna melalui sebuah variable yang dikirim dari controller atau rute

11. Routing blade

```
1 <!DOCTYPE html>
2 <html lang="en">
3 <head>
4     <meta charset="UTF-8">
5     <meta name="viewport" content="width=device-width, initial-scale=1.0">
6     <meta http-equiv="X-UA-Compatible" content="ie=edge">
7     <title>Belajar Laravel Routing</title>
8     @vite('resources/sass/app.scss')
9 </head>
10 <body>
11     <div class="container m-5">
12         <h1>Belajar Laravel Routing</h1>
13
14         <div class="list-group list-group-numbered mt-4">
15             <h6 class="mt-4">Route Groups (Route Prefixes & Route Name Prefixes)</h6>
16             <div class="list-group list-group-numbered mt-4">
17                 <a href="{{ route('admin.dashboard') }}" class="list-group-item list-group-item-action">
18                     Admin Dashboard
19                 </a>
20                 <a href="{{ route('admin.users') }}" class="list-group-item list-group-item-action">
21                     Admin Users
22                 </a>
23                 <a href="{{ route('admin.items') }}" class="list-group-item list-group-item-action">
24                     Admin Items
25                 </a>
26             </div>
27         </div>
28
29         <a href="{{ url('/basic_routing') }}" class="list-group-item list-group-item-action">
30             Basic Routing (No View, No Controller)
31         </a>
32
33         <a href="{{ url('/view_route') }}" class="list-group-item list-group-item-action">
34             View Route
35         </a>
36
37         <a href="{{ url('/controller_route') }}" class="list-group-item list-group-item-action">
38             Controller Route
39         </a>
40
41         <a href="{{ url('/') }}" class="list-group-item list-group-item-action">
42             Redirect Route
43         </a>
44
45         <a href="{{ url('/user/12345') }}" class="list-group-item list-group-item-action">
46             Route Parameter (Required Parameter) - 1
47         </a>
48         <a href="{{ url('/posts/01/comments/20') }}" class="list-group-item list-group-item-action">
49             Route Parameter (Required Parameter) - 2
50         </a>
51
52         <a href="{{ url('/username') }}" class="list-group-item list-group-item-action">
53             Route Parameter (Optional Parameter)
54         </a>
55
56         <a href="{{ url('/title/this-is-my-title') }}" class="list-group-item list-group-item-action">
57             Route With Regular Expression Constraints
58         </a>
59
60         <a href="{{ route('profileRouteName', ['profileId' => '123']) }}" class="list-group-item list-group-item-action">
61             Named Route
62         </a>
63
64         <a href="{{ url('/route_priority/user') }}" class="list-group-item list-group-item-action">
65             Route Priority
66         </a>
67
68         <a href="{{ url('/asdqwezxc') }}" class="list-group-item list-group-item-action">
69             Fallback Routes
70         </a>
71
72     </div>
73     @vite('resources/js/app.js')
74 </body>
75 </html>
76
77
78
79
80
81
82
83
84
85
86
87
```

12. Controller Route

```
found 0 vulnerabilities
PS C:\Users\ASUS\OneDrive\Documents\project laravel\Modul11-app> php artisan make:controller RouteController
>>
```

```

1 <?php
2
3 namespace App\Http\Controllers;
4
5 use App\Http\Controllers\RouteController;
6 use Illuminate\Support\Facades\Route;

```

Mengimpor RouteController adalah Langkah penting untuk memastikan bahwa controller dapat digunakan dalam mendefinisikan rute di Laravel. Hal ini mendukung struktur model view controller agar rapi dan terorganisir

```

1 <a href="{{ url('/controller_route') }}" class="list-group-item list-group-item-action">
2     Controller Route
3 </a>

```

Kode ini menunjukkan cara untuk membuat tautan yang akan mengarahkan ke rute yang dikelola oleh controller. Ini adalah suatu bentuk implementasi Laravel untuk memisahkan controller dan view

13. Redirect Route

```

1 Route::get('/routing', function() {
2     return view('routing');
3 });

```

```

1 <a href="{{ url('/') }}" class="list-group-item list-group-item-action">
2     Redirect Route
3 </a>

```

Redirect route dapat mempermudah pengalihan tanpa perlu membuat logika tambahan dalam controller. Hal ini sangat berguna untuk kasus seperti memindahkan halaman untuk membuat rute sementara atau meningkatkan navigasi pada sebuah aplikasi

14. Route Parameter (Required Parameter)

```
1 Route::get('/user/{id}', function($id) {  
2     return "User Id: ".$id;  
3 });  
4 Route::get('/posts/{post}/comments/{comment}', function($postId, $commentId) {  
5     return "Post Id: ".$postId.", Comment Id: ".$commentId;  
6 });
```

Route parameter ini berguna untuk mennagani URL dengan mudah dan dapat mengambil nilai dari URL lalu menggunakannya dalam aplikasi. Parameter required ini harus ada dalam URL saat mengakses rute yang bersangkutan

15. Buka file view **routing.blade.php**

```
1 <a href="{{ url('/user/12345') }}" class="list-group-item list-group-item-action">  
2     Route Parameter (Required Parameter) - 1  
3 </a>  
4 <a href="{{ url('/posts/01/comments/20') }}" class="list-group-item list-group-item-action">  
5     Route Parameter (Required Parameter) - 2  
6 </a>  
7
```

16. Praktik router parameter

```
1 <a href="{{ url('/title/this-is-my-title') }}" class="list-group-item list-group-item-action">  
2     Route With Regular Expression Constraints  
3 </a>
```

Optional parameter berguna Ketika kita ingin membuat sebuah URL yang dapat menerima parameter tetapi parameter tersebut tidak diwajibkan ada. Dengan menggunakan parameter juga kita dapat memberi fleksibilitas kepada user untuk memilih apakah mereka ingin menyertakan sebuah nilai dalam URL atau tidak

17. Named route

```
1 Route::get('/profile/{profileId}', [RouteController::class, 'profile'])->name('profileRouteName');
```

```

1 public function profile($profileId) {
2     return "This is Profile from Controller, profile id: ".$profileId;
3 }

```

```

1 <a href="{{ route('profileRouteName', ['profileId' => '123']) }}" class="list-group-item list-group-item-action">
2     Named Route
3 </a>

```

18. Route priority

```

1 // Route::get('/route_priority/{rpId}', function($rpId) {
2 //     return "This is Route One";
3 // });
4 Route::get('/route_priority/user', function() {
5     return "This is Route 1";
6 });
7 Route::get('/route_priority/user', function() {
8     return "This is Route 2";
9 });
10

```

Route priority di Laravel sangat bergantung pada urutan deklarasi pada rute dalam file web.php. Rute yang lebih spesifik seharusnya dapat mendeklarasi sebelum rute yang lebih umum. Jika ada dua rute dengan URL yang sama hanya rute pertama kali dideklarasikan akan dieksekusi.

19. Fallback routes

```

1 Route::fallback(function() {
2     return 'This is Fallback Route';
3 });
4

```

```

1 <a href="{{ url('/asdqwezxc') }}" class="list-group-item list-group-item-action">
2     Fallback Routes
3 </a>

```

Fallback routes di Laravel memungkinkan kita untuk menangani sebuah permintaan yang tidak mempunyai sebuah rute yang sesuai dengan cara yang fleksible.

. fallback routes juga berguna untuk menangani URL sata terjadi salah ketik atau jika aplikasi yang ingin memberikan respons khusus Ketika tidak ada rute yang cocok.. kita hanya perlu mendefinisikan satu fallback route di aplikasi laravel dan laravel akan secara otomatis menangani semua permintaan yang dirasa tidak cocok dengan yang lain

20. Route groups

```
1 Route::prefix('admin')->name('admin.')->group(function() {
2     Route::get('/dashboard', function() {
3         return "This is admin dashboard";
4     })->name('dashboard');
5     Route::get('/users', function() {
6         return "This is user data on admin page";
7     })->name('users');
8     Route::get('/items', function() {
9         return "This is item data on admin page";
10    })->name('items');
11 });
```

```
1 <h6 class="mt-4">Route Groups (Route Prefixes & Route Name Prefixes)</h6>
2 <div class="list-group list-group-numbered mt-4">
3     <a href="{{ route('admin.dashboard') }}" class="list-group-item list-group-item-action">
4         Admin Dashboard
5     </a>
6     <a href="{{ route('admin.users') }}" class="list-group-item list-group-item-action">
7         Admin Users
8     </a>
```

21. php artisan route:list

```
c:\Users\ASUS\OneDrive\Documents\project laravel\Modul11-app>php artisan route:list

GET|HEAD / ..... Illuminate\Routing > RedirectController
ANY / .....
POST /_ignition/execute-solution ..... Ignition\ExecuteSolutionController
GET|HEAD /_ignition/health-check ..... Ignition\HealthCheckController
POST /_ignition/update-config ..... Ignition\UpdateConfigController
GET|HEAD admin/dashboard ..... Admin\DashboardController
GET|HEAD admin/items ..... Admin\ItemsController
GET|HEAD admin/users ..... Admin\UsersController
GET|HEAD api/user .....
GET|HEAD basic routing .....
GET|HEAD controller route ..... RouteController@index
GET|HEAD posts/{post}/comments/{comment} ..... ProfileRouteName > RouteController@profile
GET|HEAD profile/{profileId} .....
GET|HEAD route_priority/user .....
GET|HEAD routing .....
GET|HEAD sanctum/csrf-cookie ..... Sanctum\CsrfCookieController@show
GET|HEAD title/{title} .....
GET|HEAD user/{id} .....
GET|HEAD username/{name?} .....
GET|HEAD view route .....
GET|HEAD {fallbackPlaceholder} .....
```

22. php artisan route:cache

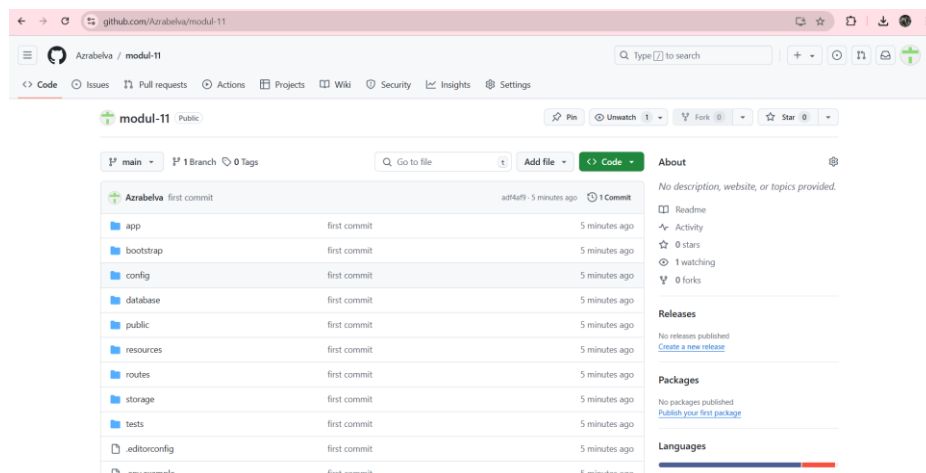
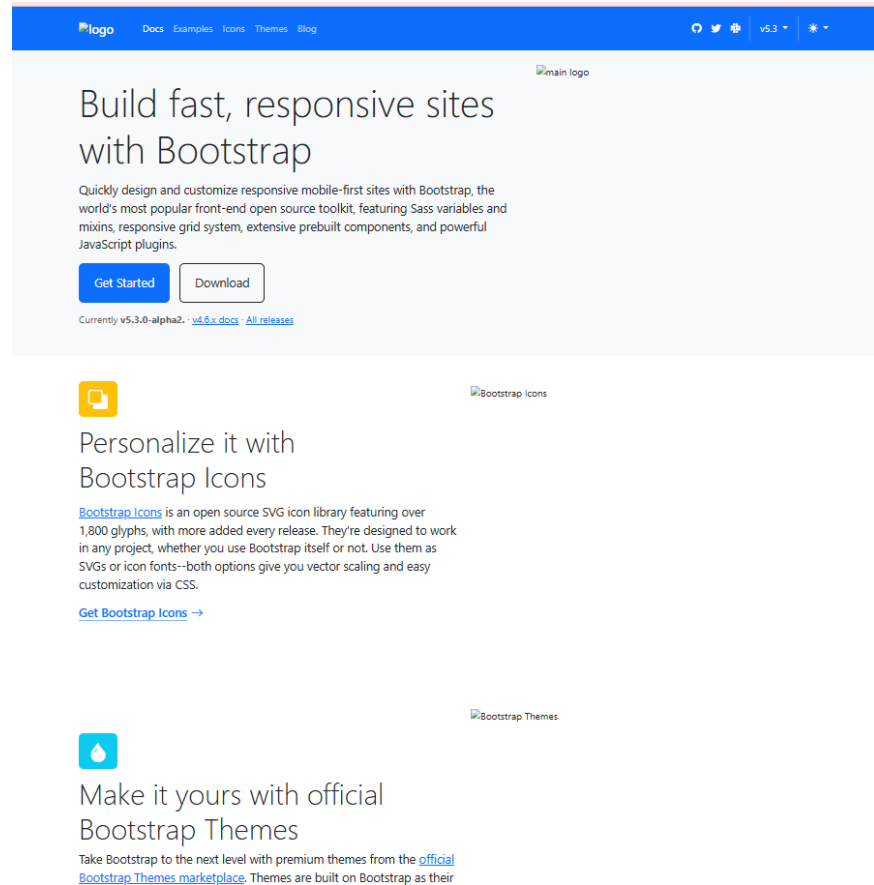
```
C:\Users\ASUS\OneDrive\Documents\project laravel\Modul11-app>php artisan route:cache  
  
INFO Routes cached successfully.  
  
C:\Users\ASUS\OneDrive\Documents\project laravel\Modul11-app>
```

23. php artisan route:clear

```
C:\Users\ASUS\OneDrive\Documents\project laravel\Modul11-app>php artisan route:clear  
  
INFO Route cache cleared successfully.  
  
C:\Users\ASUS\OneDrive\Documents\project laravel\Modul11-app>
```

TUGAS

➤ Output Project



Link : <https://github.com/Azrabelva/modul-11.git>