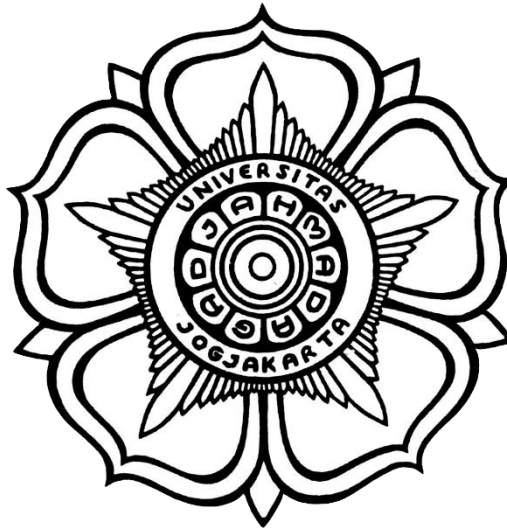


LAPORAN PRAKTIKUM
PEMROGRAMAN BERBASIS WEB 2
PERTEMUAN 3
ROUTING, VIEW DAN BLADE LARAVEL



Disusun Oleh:

Ghita Najmi Naqasy

22/496466/SV/20961

Dosen Pengampu:

Dinar Nugroho Pratomo, S.Kom., M.IM., M.Cs.

Faza Maula Azif, S.Kom., M.Eng.

PROGRAM STUDI TEKNOLOGI REKAYASA PERANGKAT LUNAK
DEPARTEMEN TEKNIK ELEKTRO DAN INFORMATIKA
SEKOLAH VOKASI
UNIVERSITAS GADJAH MADA
YOGYAKARTA

2023

DAFTAR ISI

DAFTAR ISI	i
DAFTAR GAMBAR	ii
BAB I PENDAHULUAN	1
1.1. Latar Belakang	1
1.1. Rumusan Masalah	1
1.2. Tujuan	2
BAB II PEMBAHASAN	3
2.1. Routing dan view	3
2.2. Blade Laravel	3
BAB III LATIHAN PRAKTIKUM	5
3.1. Routing dan View	5
3.2. Blade Laravel	10
BAB IV KESIMPULAN	11
4.1. Kesimpulan	11
DAFTAR PUSTAKA	12

DAFTAR GAMBAR

Gambar 3.1. 1. Folder routes	5
Gambar 3.1. 2. Folder resources.....	6
Gambar 3.1. 3. Mengaktifkan Laravel.....	6
Gambar 3.1. 4 Membuat route	7
Gambar 3.1. 5. Tampilan Laravel	8
Gambar 3.1. 6. Kode HTML.....	8
Gambar 3.1. 7. Kode CSS.....	9
Gambar 3.1. 8. Kode Javascript.....	9
Gambar 3.1. 9. Tampilan web	10

BAB I

PENDAHULUAN

1.1.Latar Belakang

Jaringan terdiri dari sejumlah perangkat, seperti komputer, server, dan perangkat lainnya, yang terhubung satu sama lain tujuan utama dari jaringan adalah memungkinkan perangkat untuk berkomunikasi dan berbagi data. Router adalah perangkat khusus dalam jaringan yang berfungsi sebagai pengatur lalu lintas data, router mengetahui cara mengirim paket data ke tujuan yang benar. Routing melibatkan perangkat dalam jaringan yang bekerja sama untuk memastikan data dikirimkan dengan efisien dan akurat dari satu tempat ke tempat lain. Router berperan penting dalam proses ini dengan menggunakan protokol routing dan tabel routing untuk memilih jalur terbaik.

Pada project yang ingin dibuat terdapat beberapa folder, folder tersebut memiliki fungsi yang berbeda-beda yakni `api.php` bagian dari aplikasi web yang mengizinkan komunikasi antara perangkat lunak yang berbeda, `channels.php` berfungsi sebagai tempat pengaturan atau konfigurasi untuk saluran dalam aplikasi yang menggunakan teknologi komunikasi real-time, `console.php` berisi pengaturan dan konfigurasi yang diperlukan untuk menjalankan perintah tertentu misalnya perintah `artisan`, `web.php` adalah tempat di mana mendefinisikan rute atau URL yang dapat diakses oleh pengguna melalui browser. Dalam kerangka kerja Laravel, `views` merujuk pada bagian dari aplikasi yang bertanggung jawab untuk menghasilkan tampilan atau halaman yang dilihat oleh pengguna. `Views` mendefinisikan cara tampilan akhir akan diatur mencakup penggunaan templating engine (seperti `Blade` dalam Laravel) untuk menyisipkan variabel, loop, kondisi, dan elemen logika lainnya ke dalam tampilan, folder `views` terletak pada `resources`.

1.1.Rumusan Masalah

Berikut rumusan masalah dalam praktikum ini:

1. Bagaimana cara mengaplikasikan Routing, View dan Blade Laravel.
2. Apa saja manfaat yang dapat diambil dari menggunakan Routing, View dan Blade Laravel.
3. Bagaimana proses Routing, View dan Blade Laravel bekerja.

1.2.Tujuan

Berikut adalah tujuan yang ingin dicapai pada praktikum kali ini:

4. Mampu memahami Routing, View dan Blade Laravel.
5. Mampu menerapkan Routing, View dan Blade Laravel pada project.
6. Mengenali Routing, View dan Blade Laravel.

BAB II

PEMBAHASAN

2.1. Routing dan view

Dalam pengembangan web, routing mengacu pada proses mengarahkan permintaan dari URL yang dimasukkan pengguna ke tindakan yang sesuai dalam kode aplikasi berguna untuk mengaitkan URL dengan fungsi atau tindakan yang dilakukan oleh server. Routing membantu membuat struktur URL yang jelas dan deskriptif hal ini membuat navigasi pengguna lebih mudah dan membantu mesin pencari memahami hierarki halaman pada situs website. Tampilan dalam pengembangan web mengacu pada antarmuka pengguna yang dilihat oleh pengguna akhir melalui browser. Tampilan terdiri dari elemen HTML dan CSS dan biasanya menggunakan templat untuk menyisipkan data dinamis.

Pengaturan routing mengidentifikasi langkah-langkah yang diperlukan oleh aplikasi ketika suatu URL tertentu diakses saat langkah-langkah ini diterapkan, data umumnya diambil dari model, diolah, dan disiapkan agar dapat ditampilkan di dalam view. View selanjutnya mengambil data ini dan menghasilkan antarmuka yang akan diperlihatkan kepada pengguna dalam pengembangan web, istilah view merujuk pada antarmuka atau halaman yang terlihat oleh pengguna akhir melalui peramban web. Tampilan ini terdiri dari komponen-komponen HTML dan CSS serta seringkali memanfaatkan template untuk menyisipkan data yang dapat berubah-ubah. Secara keseluruhan, kerja sama antara pengaturan routing dan view memungkinkan penciptaan pengalaman yang interaktif dan penuh informasi bagi pengguna aplikasi web. Pengaturan routing mengarahkan pengguna ke tujuan yang sesuai, sementara tampilan menyajikan informasi dengan cara yang menarik dan efisien.

2.2. Blade Laravel

Blade Laravel adalah komponen yang sangat berharga dalam ekosistem pengembangan web menggunakan kerangka kerja Laravel. Blade memudahkan pengembang untuk memisahkan logic dari tampilan yang memungkinkan pembuatan kode yang lebih terstruktur, mudah dikelola, dan lebih efisien. Fitur penting dalam Laravel yang menyediakan sistem templating yang kuat dan efisien dengan Blade dapat menghasilkan tampilan dengan lebih terstruktur, lebih mudah dipelihara, dan lebih aman serta memberikan pengalaman pengguna yang lebih baik dalam sebuah web.

Keuntungan menggunakan Blade:

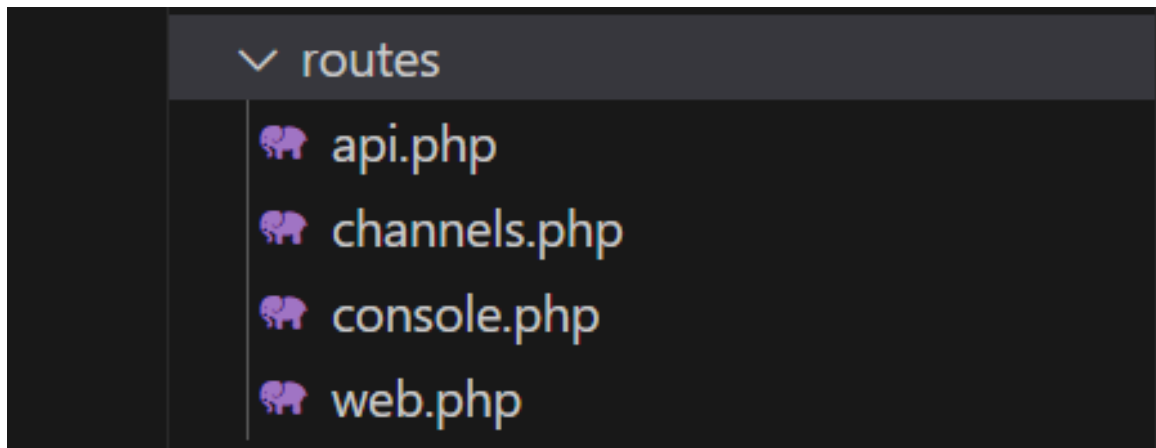
- Struktur yang Jelas: Blade memungkinkan untuk mengorganisir tampilan dalam struktur yang lebih teratur dengan menggunakan komponen dan templat.
- Pewarisan dan Komponen: Dapat mewarisi tampilan dari tampilan lain dan menggunakan komponen untuk memudahkan penggunaan kembali kode.
- Escaping Otomatis: Blade secara otomatis melakukan proses escaping pada data yang dimasukkan ke dalam tampilan, mencegah potensi serangan injeksi skrip silang.

BAB III

LATIHAN PRAKTIKUM

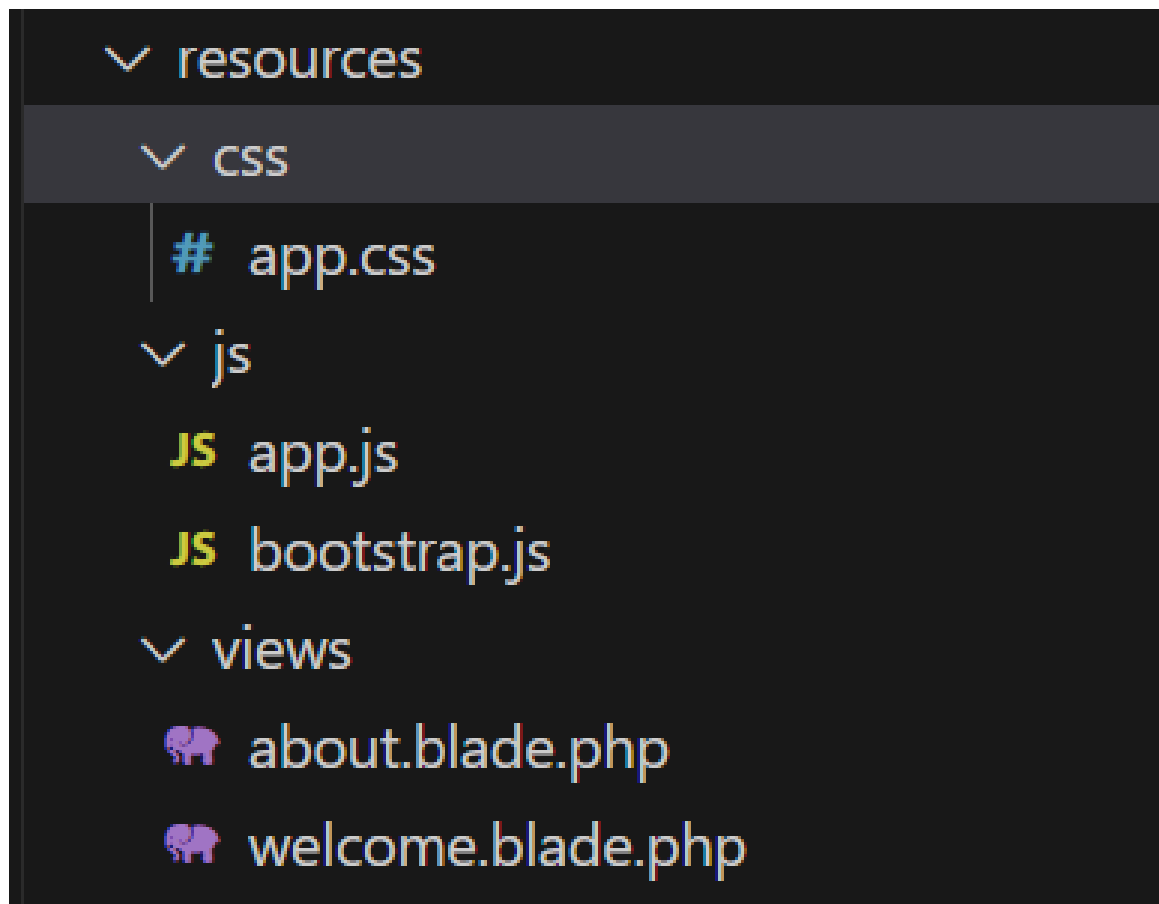
3.1. Routing dan View

3.1.1. Ketika ingin mengakses routing maka langkah pertama carilah folder dengan nama routes jika kita lihat pada folder terdapat beberapa file yang berbeda , api.php dirancang untuk menyajikan data dalam format yang mudah diakses oleh perangkat lunak lain melalui permintaan HTTP. Channels.php membantu dalam konfigurasi dan manajemen saluran komunikasi real-time dalam aplikasi web. Console.php mendefinisikan perintah yang dapat dijalankan dari baris perintah untuk menjalankan tugas-tugas seperti migrasi basis data, pengelolaan cache yang memerlukan akses ke bagian-bagian dalam aplikasi dan web.php berfokus pada pengaturan rute dan logika aplikasi yang terkait dengan antarmuka pengguna melalui peramban web. Masing-masing memiliki tujuan dan peran khusus dalam menjalankan fungsi-fungsi tertentu dalam aplikasi.



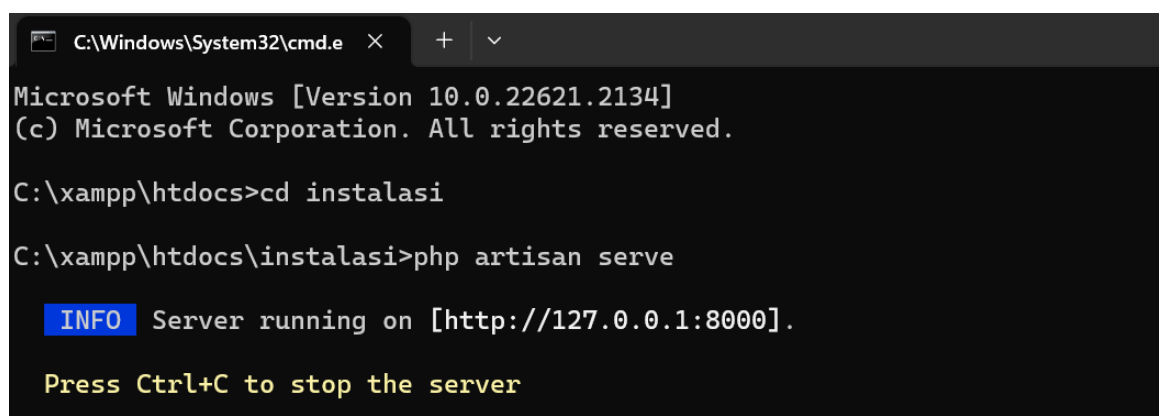
Gambar 3.1. 1. Folder routes

3.1.2. Direktori "resources" beserta struktur dan berkas-berkas yang telah dihasilkan oleh kerangka kerja Laravel telah dibentuk. Di dalam "resources," terdapat "views," yang mengandung seluruh tampilan yang dibangun untuk aplikasi. Salah satu berkas yang sudah tersedia adalah "welcome.blade.php," yang akan ditampilkan saat Laravel dijalankan untuk pertama kalinya. Oleh karena itu, ketika Anda mengakses halaman melalui peramban web, perhatian akan tertuju pada berkas "web.blade.php," atau juga mungkin Anda ingin membuat berkas baru dengan format "nama_file.blade.php."



Gambar 3.1. 2. Folder resources

3.1.3. Ikuti perintah berikut untuk mengakses Laravel “cd nama folder” pada contoh kali ini nama foldernya “instalasi” klik enter, ketikkan “php artisan serve” klik enter maka akan muncul ip address [http://127.0.0.1:8000].



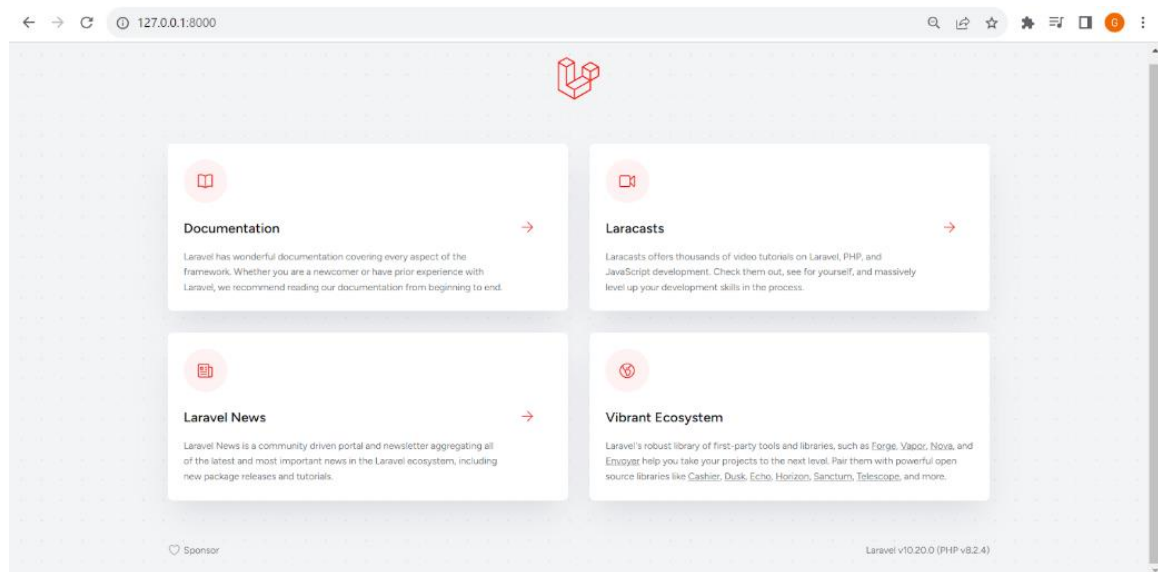
Gambar 3.1. 3. Mengaktifkan Laravel

3.1.4. Ilustrasi yang tertera menggambarkan struktur folder yang terletak dalam bagian "packages" di dalam kerangka kerja Laravel. Pada berkas "web.php," ditemukan potongan kode "Route::get('/')", yang serupa dengan representasi visual di atas. Salin "route::get('/')" dari berkas "web.php" dan berlanjut dengan penambahan identitas halaman setelah garis miring ("/") sesuai dengan nama berkas yang sudah dihasilkan dalam direktori "views" sebelumnya. Juga dimungkinkan untuk memasukkan data di dalamnya atau sejumlah data daftar menggunakan tanda kurung siku kosong "[]" dengan menuliskan pasangan kunci dan nilai yang dihubungkan oleh simbol panah "=>" potongan kode ini merujuk pada halaman awal dari aplikasi Laravel setelah proses instalasi.

```
routes > web.php
10 | Here is where you can register web routes for your application. These
11 | routes are loaded by the RouteServiceProvider and all of them will
12 | be assigned to the "web" middleware group. Make something great!
13 |
14 */
15
16 Route::get('/', function () {
17     return view('welcome');
18 });
19
20 Route::get('/about', function () {
21     return view('about', [
22         "name" => "lala",
23         "email" => "lala@gmail.com"
24     ]);
25 });
```

Gambar 3.1. 4 Membuat route

3.1.5. Potongan kode ini merujuk pada halaman awal dari aplikasi Laravel setelah proses instalasi. Di bawah ini disajikan tampilan halaman yang dimaksud.





Gambar 3.1. 5. Tampilan Laravel

3.1.6. Selanjutnya, sertakan kode HTML dalam berkas `about.blade.php` dan sisipkan data yang telah disebutkan ke dalam bagian `body`. Saat menuliskan data, cukup gunakan kode dan kunci seperti contoh di bawah ini. Tanda `<? =` adalah pengganti singkat dari `<?php echo`.

```
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     <link rel="stylesheet" href="css/style.css">
8     <title>About</title>
9 </head>
10 <body>
11     <h1>Halaman About</h1>
12     <h1> <? = $name;?> </h1>
13     <h1> <? = $email;?> </h1>
14 <script src="js/script.js"></script>
15 </body>
16 </html>
```

Gambar 3.1. 6. Kode HTML

3.1.5. Sertakan gaya di dalam berkas CSS. Berikut ini adalah ilustrasinya:

```
public > css > # style.css >  body
1   body{
2       background-color:  beige;
3   }
```

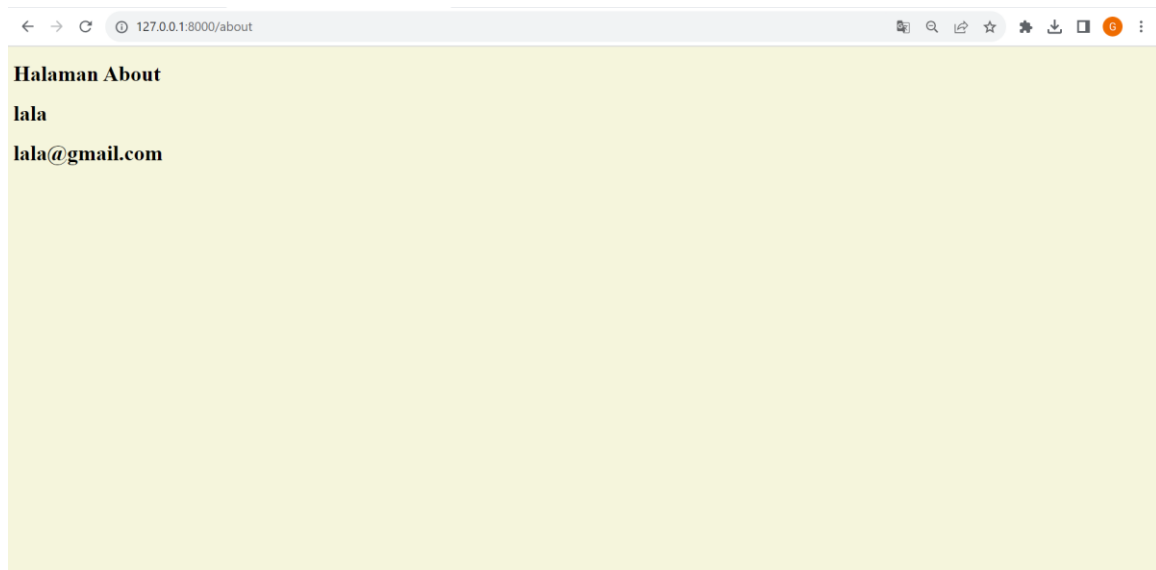
Gambar 3.1. 7. Kode CSS

3.1.6. Tambahkan alert pada javascript dengan string “haihai”.

```
public > js > JS script.js
1   alert("haihai")
```

Gambar 3.1. 8. Kode Javascript

3.1.5. Terakhir web akan mengalihkan ke halaman yang diinginkan, yaitu halaman "about" dan akan memuat program yang sudah disiapkan sebelumnya.



Gambar 3.1. 9. Tampilan web

3.2. Blade Laravel

Blade menggunakan format file blade.php yang ditempatkan dalam direktori resources/views, dan menggunakan konsep warisan dan bagian-bagian (section) dalam pembuatan tampilan. Blade memiliki berbagai manfaat, di antaranya memungkinkan penggunaan tata letak (layout) untuk menghindari duplikasi tampilan seperti header, footer, dan sidebar yang mungkin timbul akibat pengulangan statis, yang dapat menimbulkan ketidaksesuaian. Dengan cara ini, sebuah file induk yang disebut template utama dapat dibuat, di mana kode dan struktur tampilan yang ditampilkan umumnya tetap sama untuk beberapa halaman, hanya terdapat perbedaan konten di setiap halaman, untuk halaman lain yang menggunakan tata letak yang sama, fungsi-fungsi tertentu digunakan, seperti @extends(), @section(), dan @yield(). @yield berfungsi untuk menunjukkan bagian-bagian tertentu dalam tata letak situs, dan nama setiap bagian dapat ditentukan dalam parameter @yield(). Sementara @include, yang serupa dengan PHP dasar, menggabungkan tampilan "partial" ke dalam tampilan view yang diinginkan. Kemudian, fungsi @extends() memungkinkan perluasan template dengan mendefinisikan bagian-bagian yang unik. Template yang dapat "di-extends" akan menentukan bagian-bagian (section) mereka sendiri menggunakan fungsi yield, yang nantinya dapat dimasukkan ke dalam file tampilan.

BAB IV

KESIMPULAN

4.1. Kesimpulan

Dalam pengembangan web, routing, view, dan Blade adalah tiga konsep yang saling melengkapi dan penting untuk menciptakan pengalaman pengguna yang efektif dan konsisten. Routing adalah proses mengarahkan permintaan HTTP dari URL pengguna ke tindakan atau kode yang sesuai di aplikasi. Ini membantu menciptakan struktur URL yang intuitif, memisahkan logika aplikasi, dan menangani berbagai jenis permintaan. View mengacu pada antarmuka pengguna yang dilihat oleh pengguna akhir melalui peramban web. Views terdiri dari komponen HTML, CSS, dan kadang-kadang menggunakan templating engine untuk menyisipkan data dinamis. Pemisahan tampilan dari logika memungkinkan pengembang dan desainer bekerja secara terpisah. Blade, fitur templating di Laravel, menyediakan sintaks yang efisien dan kuat untuk membangun tampilan.

Dengan menyimpan file `.blade.php` dalam direktori `resources/views`, Blade memungkinkan pemisahan logika tampilan dari bisnis, menyederhanakan pembuatan tampilan yang konsisten, dan mengintegrasikan fitur seperti layout, section, dan komponen. Ketiganya bersama-sama membentuk dasar untuk mengembangkan tampilan yang responsif, konsisten, dan menarik dalam aplikasi web. Routing memandu pengguna ke tempat yang sesuai, view mengatur cara data disajikan, sementara Blade membantu dalam merancang tampilan yang mudah dipelihara dan dioptimalkan. Dengan pemahaman yang kuat tentang ketiga konsep ini, pengembang mampu menciptakan pengalaman web yang menarik dan berfungsi dengan baik bagi pengguna akhir.

DAFTAR PUSTAKA

- Pengertian Routing*. (2023, May 13). FIKTI. <https://fikti.umsu.ac.id/pengertian-routing/>
- SEW. (2019, September 27). *Laravel View dan Blade (Belajar Laravel #5) - Informatika Universitas Ciputra*. Informatika Universitas Ciputra.
<https://informatika.uc.ac.id/2019/09/laravel-view-blade/>
- K, A. (2023, May 4). *Routing Adalah: Pengertian, Sejarah, Jenis dan Algoritma Yang Digunakan*. Gramedia Literasi. <https://www.gramedia.com/literasi/routing/>
- Ahmad, T. (2020, April 12). *Tutorial Views dan Passing data Controller ke Views Blade pada Laravel | dotlocal*. Medium; dotlocal. <https://medium.com/dotlocal/belajar-laravel-chapter-10-tutorial-views-82a67aadee74>
- shintack. (2014). *Mari Mengenal Blade - Template Engine Untuk Laravel | Kursus Website & Jasa Pembuatan Website*. Babastudio.com; babastudio.com.
<https://www.babastudio.com/blog/mari-mengenal-blade-template-engine-untuk-laravel>