

PRAKTIKUM MODUL 06
PEMROGRAMAN FRAMEWORK
LARAVEL DATABASE TAHAP LANJUT
Eloquent ORM & Blade Templates



DISUSUN OLEH:

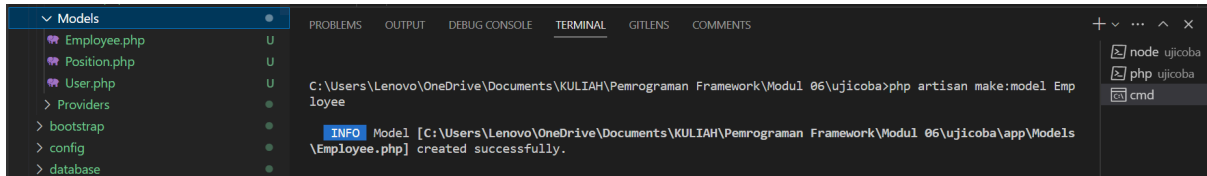
1204200081 RAFAEL FILLAH FIRMANSYAH

PROGRAM STUDI SISTEM INFORMASI
FAKULTAS REKAYASA INDUSTRI
TELKOM UNIVERSITY SURABAYA
2024

Membuat File Model

Link Github: <https://github.com/elfill03/Pemrograman-Framework>

1. Buat model melalui cmd dengan memasukkan “php artisan make:model (nama model)”, buat 2 buah model dengan nama employee dan position.

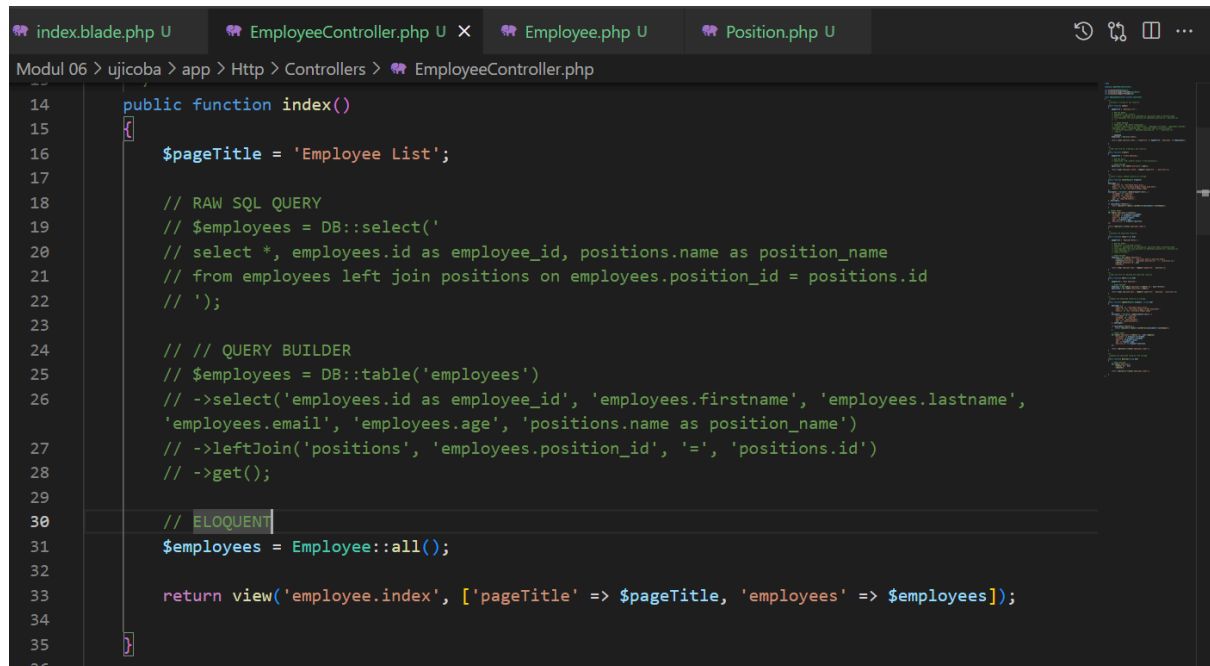


2. Kemudian buat sebuah function pada setiap model dengan tujuan untuk mendefinisikan keterhubungan antar model yang telah dibuat (model employee & model position)



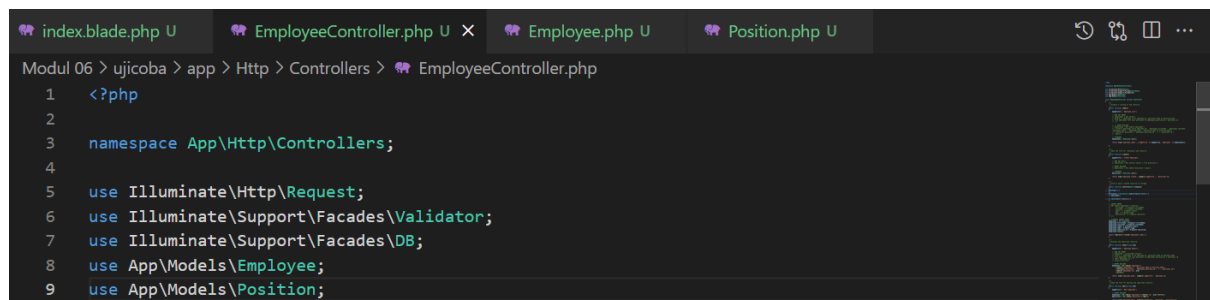
Implementasi Eloquent pada Fitur Menampilkan Data Pegawai

1. Untuk penerapan pendekatan Eloquent pada tampilan data pegawai pertama pada file `EmployeeController` bagian function `index` ubah yang sebelumnya masih menggunakan Query Builder menjadi Eloquent, contohnya seperti berikut (lane 30-31). Maksud dari `Employee::all()` yaitu melakukan request data dari database `Employee` dengan keseluruhan kolom



```
14 public function index()
15 {
16     $pageTitle = 'Employee List';
17
18     // RAW SQL QUERY
19     // $employees = DB::select('
20     // select *, employees.id as employee_id, positions.name as position_name
21     // from employees left join positions on employees.position_id = positions.id
22     // ');
23
24     // // QUERY BUILDER
25     // $employees = DB::table('employees')
26     //     ->select('employees.id as employee_id', 'employees.firstname', 'employees.lastname',
27     //         'employees.email', 'employees.age', 'positions.name as position_name')
28     //     ->leftJoin('positions', 'employees.position_id', '=', 'positions.id')
29     //     ->get();
30
31     // ELOQUENT
32     $employees = Employee::all();
33
34     return view('employee.index', ['pageTitle' => $pageTitle, 'employees' => $employees]);
35
36 }
```

2. Tambahkan juga import statement untuk model `employee` dan `position` (lane 8-9) pada file `EmployeeController`



```
1 <?php
2
3 namespace App\Http\Controllers;
4
5 use Illuminate\Http\Request;
6 use Illuminate\Support\Facades\Validator;
7 use Illuminate\Support\Facades\DB;
8 use App\Models\Employee;
9 use App\Models\Position;
```

3. Selanjutnya pada file index.blade.php tepatnya pada bagian tag tbody sesuaikan penggunaan query yang sebelumnya masih berjalan dengan query builder menjadi eloquent. Perubahan terjadi pada penggunaan nama variabel untuk route edit (lane 77-78) dan hapus (lane 80-88) dan pemanggilan data dari database pada tabel employee(lane 64-70)

```
index.blade.php U x EmployeeController.php U Employee.php U Position.php U
body > div.container.mt-4 > div.table-responsive.border.p-3.rounded-3 > table.table.table-bordered.table-hover.table-striped.mb-0.bg-white > tbody
63 <tbody>
64     @foreach ($employees as $employee)
65         <tr>
66             <td>{{ $employee->firstname }}</td>
67             <td>{{ $employee->lastname }}</td>
68             <td>{{ $employee->email }}</td>
69             <td>{{ $employee->age }}</td>
70             <td>{{ $employee->position->name }}</td>
71             <td>
72                 {{-- ACTIONS SECTION --}}
73                 <div class="d-flex">
74                     <a href="{{ route('employees.show', ['employee' =>
75                         $employee->id]) }}"
76                         class="btn btn-outline-dark btn-sm me-2"><i>
77                             class="bi-person-lines-fill"></i></a>
78                     <a href="{{ route('employees.edit', ['id' => $employee->id]) }}"
79                         class="btn btn-outline-dark btn-sm me-2"><i>
80                             class="bi-pencil-square"></i></a>
81                     <div>
82                         <form action="{{ route('employees.destroy', ['employee' =>
83                             $employee->id]) }}"
84                             method="POST">
85                             @csrf
86                             @method('delete')
87                             <button type="submit"
88                                 class="btn
89                                     btn-outline-dark btn-sm me-2"><i>
90                                     class="bi-trash"></i></button>
91                         </form>
92                     </div>
93                 </div>
94             </td>
95         </tr>
96     @endforeach
97 </tbody>
```

Implementasi Eloquent pada Fitur Form Input Pegawai

1. Untuk menerapkan eloquent pada fitur form input, pertama ubah query yang ada pada file EmployeeController bagian function dengan nama create menjadi jenis eloquent (lane 51-52)

```
41 public function create()  
42 {  
43     $pageTitle = 'Create Employee';  
44  
45     // RAW SQL Query  
46     // $positions = DB::select('select * from positions');  
47  
48     // QUERY BUILDER  
49     // $positions = DB::table('positions')->get();  
50  
51     // ELOQUENT  
52     $positions = Position::all();  
53  
54     return view('employee.create', compact('pageTitle', 'positions'));  
55 }
```

2. Kemudian bagian function store ubah yang sebelumnya masih menggunakan Query Builder menjadi Eloquent (lane 88-95)

```
Modul 06 > ujicoba > app > Http > Controllers > EmployeeController.php  
61 public function store(Request $request)  
62 {  
63     $messages = [...]  
67 ];  
68     $validator = Validator::make($request->all(), [...]  
73 ], $messages);  
74  
75     if ($validator->fails()) {  
77     }  
78  
79     // INSERT QUERY  
80     // DB::table('employees')->insert([  
81     //     'firstname' => $request->firstName,  
82     //     'lastname' => $request->lastName,  
83     //     'email' => $request->email,  
84     //     'age' => $request->age,  
85     //     'position_id' => $request->position,  
86     // ]);  
87  
88     // ELOQUENT INSERT QUERY  
89     $employee = new Employee;  
90     $employee->firstname = $request->firstName;  
91     $employee->lastname = $request->lastName;  
92     $employee->email = $request->email;  
93     $employee->age = $request->age;  
94     $employee->position_id = $request->position;  
95     $employee->save();  
96  
97     return redirect()->route('employees.index');
```

3. Lalu coba untuk membuat employee baru pada <http://localhost:8000/employees/create> apabila data berhasil tampil pada halaman index/employee list berarti fitur input/create berhasil dibuat tanpa perlu penyesuaian pada file create.blade.php, hal ini dikarenakan nama variabel sudah sesuai dengan yang didefinisikan pada EmployeeController

Implementasi Eloquent pada Fitur Detail Data Pegawai

1. Untuk menerapkan eloquent pada fitur detail data pegawai, pertama ubah query yang ada pada file EmployeeController bagian function dengan nama show menjadi jenis eloquent (lane 121-122). Untuk pencarian (where) berdasarkan id dapat menggunakan find yang merupakan method bawaan dari laravel untuk melakukan pencarian dari database berdasarkan kunci primernya

```
103 public function show(string $id)
104 {
105     $pageTitle = 'Employee Detail';
106
107     // RAW SQL QUERY
108     // $employee = collect(DB::select('
109     // select *, employees.id as employee_id, positions.name as position_name
110     // from employees left join positions on employees.position_id = positions.id
111     // where employees.id = ? ',
112     // [$id]))->first();
113
114     // QUERY BUILDER
115     // $employee = DB::table('employees')
116     //     ->select('employees.*', 'positions.name as position_name')
117     //     ->leftJoin('positions', 'employees.position_id', '=', 'positions.id')
118     //     ->where('employees.id', $id)
119     //     ->first();
120
121     // ELOQUENT
122     $employee = Employee::find($id);
123
124
125     return view('employee.show', compact('pageTitle', 'employee'));
126 }
```

2. Lalu coba untuk mengecek detail employee apabila data berhasil tampil pada halaman detail employee berarti fitur detail employee berhasil dibuat tanpa perlu penyesuaian pada file show.blade.php, hal ini dikarenakan nama variabel sudah sesuai dengan yang didefinisikan pada EmployeeController

Implementasi Eloquent pada Fitur Form Edit Pegawai

1. Untuk menerapkan eloquent pada fitur detail data pegawai, pertama ubah query yang ada pada file EmployeeController bagian function dengan nama edit menjadi jenis eloquent (lane 139-140).

```
130 public function edit(string $id)
131 {
132     $pageTitle = 'Edit Employee';
133
134     // QUERY BUILDER
135     // $employee = DB::table('employees')->where('id', $id)->first();
136     // $positions = DB::table('positions')->get();
137
138     // ELOQUENT
139     $positions = Position::all();
140     $employee = Employee::find($id);
141
142
143     return view('employee.edit', compact('pageTitle', 'employee', 'positions'));
144 }
```

2. Kemudian ubah juga query pada function update menjadi jenis eloquent (lane 176-183)

```
149 public function update(Request $request, string $id)
150 {
151     $messages = [ ...
152 ];
153
154     $validator = Validator::make($request->all(), [ ...
155 ], $messages);
156
157     if ($validator->fails()) { ...
158 }
159
160     // UPDATE QUERY
161     // DB::table('employees')->where('id', $id)->update([
162     //     'firstname' => $request->firstName,
163     //     'lastname' => $request->lastName,
164     //     'email' => $request->email,
165     //     'age' => $request->age,
166     //     'position_id' => $request->position,
167     // ]);
168
169     // ELOQUENT UPDATE QUERY
170     $employee = Employee::find($id);
171     $employee->firstname = $request->firstName;
172     $employee->lastname = $request->lastName;
173     $employee->email = $request->email;
174     $employee->age = $request->age;
175     $employee->position_id = $request->position;
176     $employee->save();
177
178     return redirect()->route('employees.index');
179 }
```

3. Lalu coba untuk mengubah salah satu employee apabila data berhasil berubah pada halaman employee list berarti fitur edit employee berhasil dibuat tanpa perlu penyesuaian pada file edit.blade.php, hal ini dikarenakan nama variabel sudah sesuai dengan yang didefinisikan pada EmployeeController

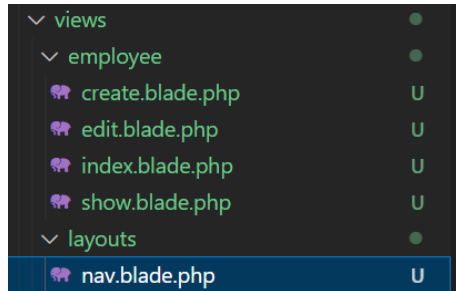
Implementasi Eloquent pada Fitur Hapus Data Pegawai

1. Untuk menerapkan eloquent pada fitur hapus data pegawai, pertama ubah query yang ada pada file EmployeeController bagian function dengan nama destroy menjadi jenis eloquent (lane 198-199).

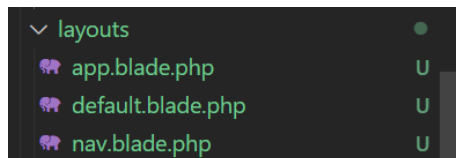
```
191 public function destroy(string $id)
192 {
193     // QUERY BUILDER
194     // DB::table('employees')
195     //     ->where('id', $id)
196     //     ->delete();
197
198     // ELOQUENT
199     Employee::find($id)->delete();
200
201     return redirect()->route('employees.index');
202
203 }
```


Memisahkan Nav Section

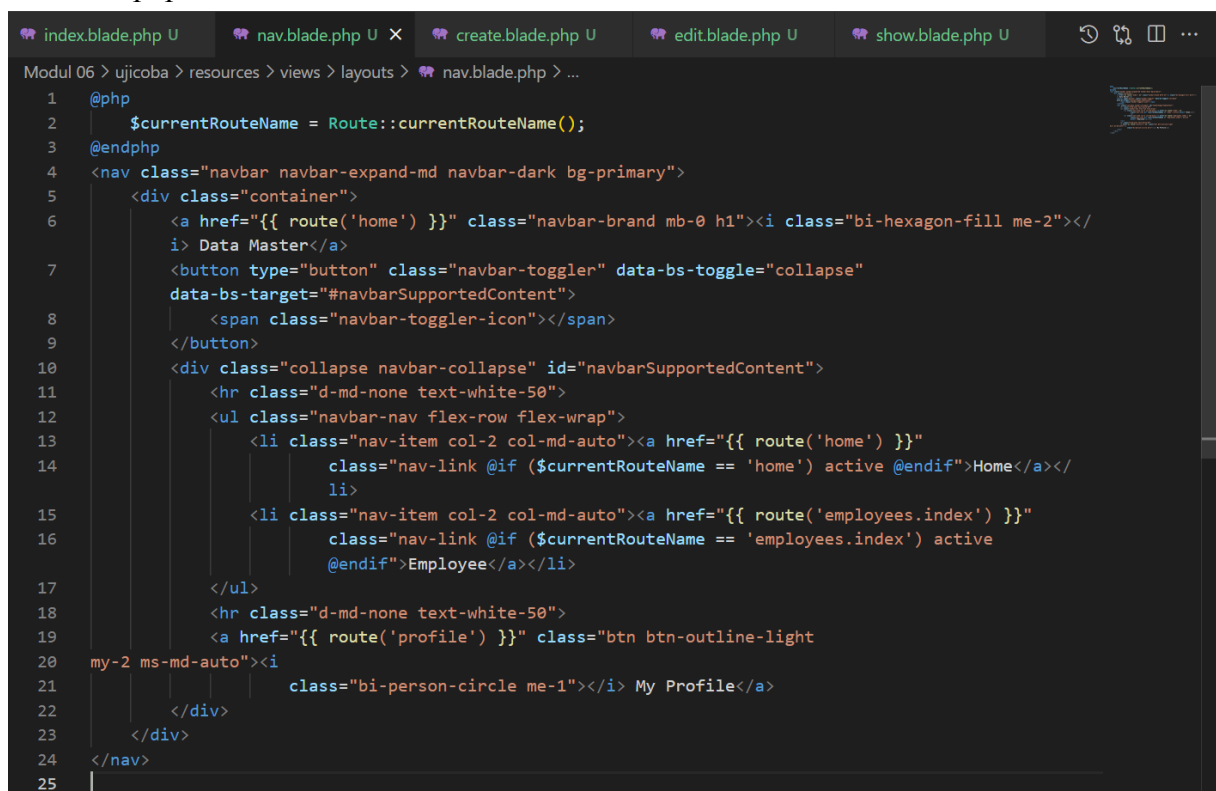
1. Pertama buat sebuah folder utama untuk layout pada views, nantinya folder tersebut akan berisi komponen-komponen view yang dapat dipanggil di beberapa halaman yang menggunakan komponen tersebut. Saya biasa memanggilnya reusable komponen seperti pada react



2. Lalu buat file nav.blade.php untuk navbar, app.blade.php untuk komponen utama, default.blade.php untuk komponen konten default atau dasar



3. Kemudian pada bagian navbar pindahkan komponennya dari setiap halaman ke file nav.blade.php



Mendefinisikan Layout Utama Aplikasi

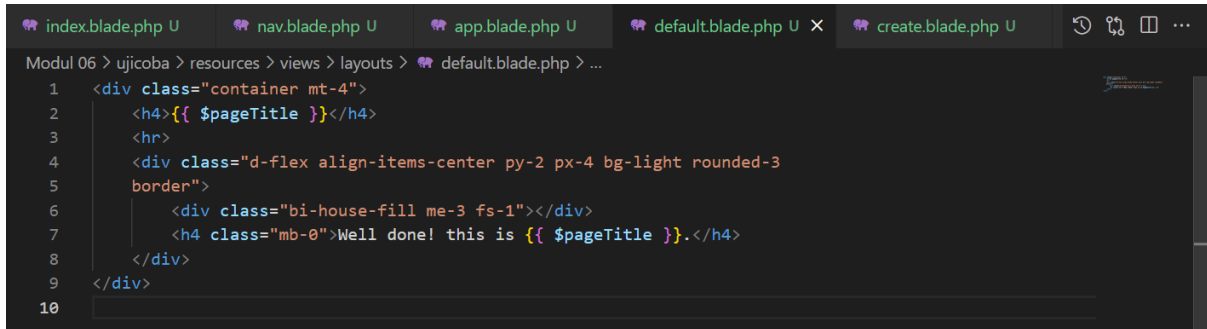
1. Pada `app.blade.php` untuk komponen utama berisi struktur utama html5 pada umumnya yang ditambahi pemanggilan komponen yang digunakan (lane 14)



```
index.blade.php U  nav.blade.php U  app.blade.php U x  create.blade.php U  edit.blade.php U  [refresh] [undo] [redo] [close] [menu]
Modul 06 > ujicoba > resources > views > layouts > app.blade.php > ...
1  <!DOCTYPE html>
2  <html lang="en">
3
4  <head>
5      <meta charset="UTF-8">
6      <meta http-equiv="X-UA-Compatible" content="IE=edge">
7      <meta name="viewport" content="width=device-width,
8  initial-scale=1.0">
9      <title>{{ $pageTitle }}</title>
10     @vite('resources/sass/app.scss')
11 </head>
12
13 <body>
14     @include('layouts.nav')
15     @yield('content')
16     @vite('resources/js/app.js')
17 </body>
18
19 </html>
20 |
```

Membuat Template untuk Default Content

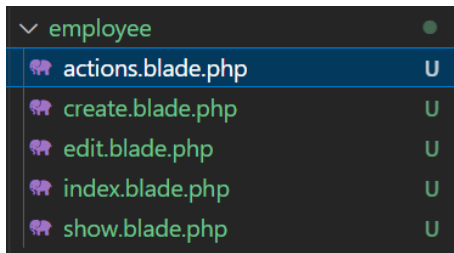
1. Buat sebuah halaman sederhana untuk dijadikan sebuah konten default



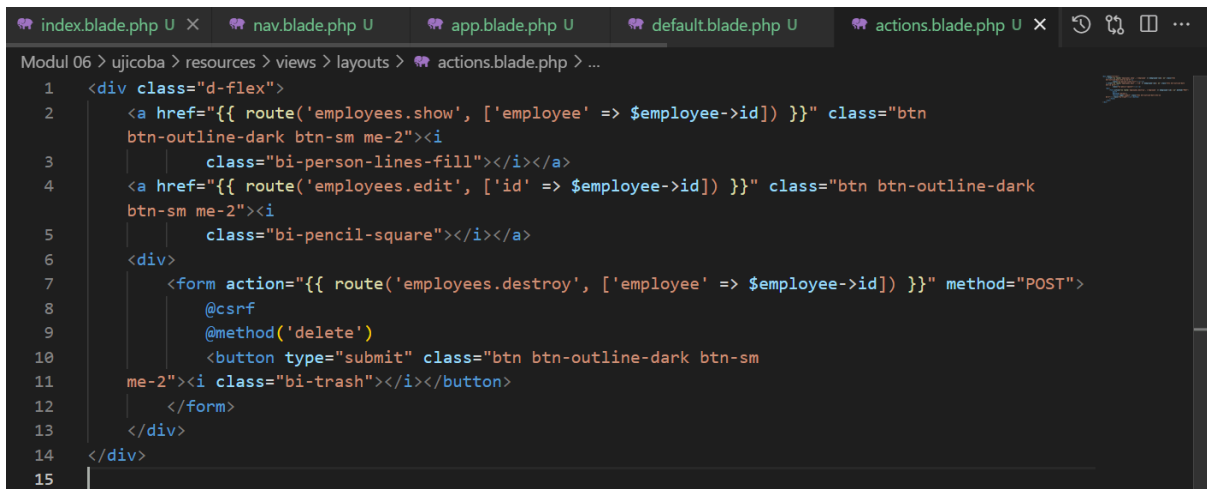
```
index.blade.php U  nav.blade.php U  app.blade.php U  default.blade.php U X  create.blade.php U
Modul 06 > ujicoba > resources > views > layouts > default.blade.php > ...
1  <div class="container mt-4">
2      <h4>{{ $pageTitle }}</h4>
3      <hr>
4      <div class="d-flex align-items-center py-2 px-4 bg-light rounded-3
5      border">
6          <div class="bi-house-fill me-3 fs-1"></div>
7          <h4 class="mb-0">Well done! this is {{ $pageTitle }}.</h4>
8      </div>
9  </div>
10
```

Memisahkan Actions Section

1. Buat file views baru pada folder employee dengan nama actions.blade.php



2. Pada file index.blade.php ambil dan pindahkan code action untuk logic detail employee, edit employee, dan hapus employee kedalam file actions



Penyesuaian File View untuk Fitur Menampilkan Data Pegawai

1. Pada file `index.blade.php` perlu dilakukannya pemanggilan komponen yang telah dibuat sebelumnya. Penerapan `@extends` merupakan pemanggilan komponen views yang digunakan yang kemudian dilakukannya pemanggilan untuk section yaitu isi konten. Kemudian pada line 36 terdapat komponen action tambahan di dalam section yang dipanggil menggunakan `include`

```
index.blade.php U X create.blade.php U edit.blade.php U show.blade.php U
Modul 06 > ujicoba > resources > views > employee > index.blade.php > div.container.mt-4 > div.table-responsive.border.p-3.rounded-3

1  @extends('layouts.app')
2  @section('content')
3      <div class="container mt-4">
4          <div class="row mb-0">
5              <div class="col-lg-9 col-xl-10">
6                  <h4 class="mb-3">{{ $pageTitle }}</h4>
7              </div>
8              <div class="col-lg-3 col-xl-2">
9                  <div class="d-grid gap-2">
10                     <a href="{{ route('employees.create') }}" class="btn
11 btn-primary">Create Employee</a>
12                 </div>
13             </div>
14         </div>
15         <hr>
16         <div class="table-responsive border p-3 rounded-3">
17             <table class="table table-bordered table-hover table-striped
18 mb-0 bg-white">
19                 <thead>
20                     <tr>
21                         <th>First Name</th>
22                         <th>Last Name</th>
23                         <th>Email</th>
24                         <th>Age</th>
25                         <th>Position</th>
26                     </tr>
27                 </thead>
28                 <tbody>
29                     @foreach ($employees as $employee)
30                         <tr>
31                             <td>{{ $employee->firstname }}</td>
32                             <td>{{ $employee->lastname }}</td>
33                             <td>{{ $employee->email }}</td>
34                             <td>{{ $employee->age }}</td>
35                             <td>{{ $employee->position->name }}</td>
36                             <td>@include('employee.actions')</td>
37                         </tr>
38                     @endforeach
39                 </tbody>
40             </table>
41         </div>
42     </div>
43 @endsection
```

Penyesuaian File View untuk Fitur Form Input Pegawai

1. Pada file create.blade.php lakukan hal serupa seperti sebelumnya dengan menggunakan pemanggilan komponen layout dan penerapan section untuk isi dari halaman create employee

```
index.blade.php U create.blade.php X edit.blade.php U show.blade.php U
Modul 06 > ujicoba > resources > views > employee > create.blade.php > div.container-sm.mt-5 > form > div.row.justify-content-center > div.p
1  @extends('layouts.app')
2  @section('content')
3      <div class="container-sm mt-5">
4          <form action="{{ route('employees.store') }}" method="POST">
5              @csrf
6              <div class="row justify-content-center">
7                  <div class="p-5 bg-light rounded-3 border col-xl-6">
8                      <div class="mb-3 text-center">
9                          <i class="bi-person-circle fs-1"></i>
10                         <h4>Create Employee</h4>
11                     </div>
12                     <hr>
13                     <div class="row">
14                         <div class="col-md-6 mb-3">
15                             <label for="firstName" class="form-label">First Name</label>
16                             <input class="form-control @error('firstName') is-invalid @enderror"
17                                 type="text"
18                                 name="firstName" id="firstName" value="{{ old('firstName') }}"
19                                 placeholder="Enter First Name">
20                             @error('firstName')
21                                 <div class="invalid-feedback">
22                                     {{ $message }}
23                                 </div>
24                             @enderror
25                         </div>
26                         <div class="col-md-6 mb-3">
27                             <label for="lastName" class="form-label">Last Name</label>
28                             <input class="form-control @error('lastName') is-invalid @enderror"
29                                 type="text"
30                                 name="lastName" id="lastName" value="{{ old('lastName') }}"
31                                 placeholder="Enter Last Name">
32                             @error('lastName')
33                                 <div class="invalid-feedback">
34                                     {{ $message }}
35                                 </div>
36                             @enderror
37                         </div>
38                         <div class="col-md-6 mb-3">
39                             <label for="email" class="form-label">Email</label>
40                             <input class="form-control @error('email') is-invalid @enderror"
41                                 type="text" name="email"
42                                 id="email" value="{{ old('email') }}" placeholder="Enter Email">
43                             @error('email')
44                                 <div class="invalid-feedback">
45                                     {{ $message }}
46                                 </div>
47                             @enderror
48                         </div>
49                         <div class="col-md-6 mb-3">
50                             <label for="age" class="form-label">Age</label>
51                             <input class="form-control @error('age') is-invalid @enderror"
52                                 type="text" name="age"
53                                 id="age" value="{{ old('age') }}" placeholder="Enter Age">
54                             @error('age')
55                                 <div class="invalid-feedback">
56                                     {{ $message }}
57                                 </div>
58                             @enderror
59                         </div>
60                     </div>
61                 </div>
62             </div>
63         </form>
64     </div>
65 @endsection
```

```

54     </div>
55     <div class="col-md-12 mb-3">
56         <label for="position" class="form-label">Position</label>
57         <select name="position" id="position" class="form-select">
58             @foreach ($positions as $position)
59                 <option value="{{ $position->id }}"
60                     {{ old('position') == $position->id ? 'selected' : '' }}>
61                     {{ $position->code . ' - ' . $position->name }}
62                 </option>
63             @endforeach
64         </select>
65     </div>
66
67 </div>
68 <hr>
69 <div class="row">
70     <div class="col-md-6 d-grid">
71         <a href="{{ route('employees.index') }}" class="btn btn-outline-dark
72             btn-lg mt-3"><i
73                 class="bi-arrow-left-circle me-2"></i> Cancel</a>
74     </div>
75     <div class="col-md-6 d-grid">
76         <button type="submit" class="btn btn-dark btn-lg mt-3"><i
77             class="bi-check-circle me-2"></i>
78             Save</button>
79     </div>
80 </div>

```

```

81 </form>
82 </div>
83 @endsection
84

```

Penyesuaian File View untuk Fitur Detail Data Pegawai

1. Pada file show.blade.php lakukan hal serupa seperti sebelumnya dengan menggunakan pemanggilan komponen layout dan penerapan section untuk isi dari halaman detail employee. Namun pada code yang aku buat perlu sedikit penyesuaian pada bagian position name karena nama variabel sedikit berbeda

```
index.blade.php U create.blade.php U show.blade.php U X EmployeeController.php U edit.blade.php I ...
hp > div.container-sm.my-5 > div.row.justify-content-center > div.p-5.bg-light.rounded-3.col-xl-4.border > div.row > div.col-md-12.mb-3 > h5
1  @extends('layouts.app')
2  @section('content')
3      <div class="container-sm my-5">
4          <div class="row justify-content-center">
5              <div class="p-5 bg-light rounded-3 col-xl-4 border">
6                  <div class="mb-3 text-center">
7                      <i class="bi-person-circle fs-1"></i>
8                      <h4>Detail Employee</h4>
9                  </div>
10                 <hr>
11                 <div class="row">
12                     <div class="col-md-12 mb-3">
13                         <label for="firstName" class="form-label">First
14                         Name</label>
15                         <h5>{{ $employee->firstname }}</h5>
16                     </div>
17                     <div class="col-md-12 mb-3">
18                         <label for="lastName" class="form-label">Last
19                         Name</label>
20                         <h5>{{ $employee->lastname }}</h5>
21                     </div>
22                     <div class="col-md-12 mb-3">
23                         <label for="email" class="form-label">Email</label>
24                         <h5>{{ $employee->email }}</h5>
25                     </div>
26                     <div class="col-md-12 mb-3">
27                         <label for="age" class="form-label">Age</label>
28                         <h5>{{ $employee->age }}</h5>
29                     </div>
30
31                     <div class="col-md-12 mb-3">
32                         <label for="age" class="form-label">Position</label>
33                         <h5>{{ $employee->position->name }}</h5>
34                     </div>
35                 </div>
36                 <hr>
37                 <div class="row">
38                     <div class="col-md-12 d-grid">
39                         <a href="{{ route('employees.index') }}" class="btn btn-outline-dark btn-lg
40                         mt-3"><i
41                         class="bi-arrow-left-circle me-2"></i> Back</a>
42                     </div>
43                 </div>
44             </div>
45         @endsection
46
```


Penyesuaian File View untuk Fitur Form Edit Pegawai

1. Pada file edit.blade.php lakukan hal serupa seperti sebelumnya dengan menggunakan pemanggilan komponen layout dan penerapan section untuk isi dari halaman edit employee.

```
index.blade.php U create.blade.php U show.blade.php U edit.blade.php X
Modul 06 > ujicoba > resources > views > employee > edit.blade.php > ...
1  @extends('layouts.app')
2  @section('content')
3      <div class="container-sm mt-5">
4          <form action="{{ route('employees.update', $employee->id) }}" method="POST">
5              @csrf
6              @method('PUT')
7              <div class="row justify-content-center">
8                  <div class="p-5 bg-light rounded-3 border col-xl-6">
9                      <div class="mb-3 text-center">
10                         <i class="bi-person-circle fs-1"></i>
11                         <h4>Edit Employee</h4>
12                     </div>
13                     <hr>
14                     <div class="row">
15                         <div class="col-md-6 mb-3">
16                             <label for="firstName" class="form-label">First Name</label>
17                             <input class="form-control @error('firstName') is-invalid @enderror"
18                                 type="text"
19                                 name="firstName" id="firstName" value="{{ $employee->firstname }}"
20                                 placeholder="Enter First Name">
21                             @error('firstName')
22                                 <div class="invalid-feedback">
23                                     {{ $message }}
24                                 </div>
25                             @enderror
26                         </div>
27                         <div class="col-md-6 mb-3">
28                             <label for="lastName" class="form-label">Last Name</label>
29                             <input class="form-control @error('lastName') is-invalid @enderror"
30                                 type="text"
31                                 name="lastName" id="lastName" value="{{ $employee->lastname }}"
32                                 placeholder="Enter Last Name">
33                             @error('lastName')
34                                 <div class="invalid-feedback">
35                                     {{ $message }}
36                                 </div>
37                             @enderror
38                         </div>
39                         <div class="col-md-6 mb-3">
40                             <label for="email" class="form-label">Email</label>
41                             <input class="form-control @error('email') is-invalid @enderror"
42                                 type="text" name="email"
43                                 id="email" value="{{ $employee->email }}" placeholder="Enter Email">
44                             @error('email')
45                                 <div class="invalid-feedback">
46                                     {{ $message }}
47                                 </div>
48                             @enderror
49                         </div>
50                         <div class="col-md-6 mb-3">
51                             <label for="age" class="form-label">Age</label>
52                             <input class="form-control @error('age') is-invalid @enderror"
53                                 type="text" name="age"
54                                 id="age" value="{{ $employee->age }}" placeholder="Enter Age">
55                             @error('age')
56                                 <div class="invalid-feedback">
57                                     {{ $message }}
58                                 </div>
59                             @enderror
60                         </div>
61                     </div>
62                 </div>
63             </div>
64         </form>
65     </div>
66 @endsection
```

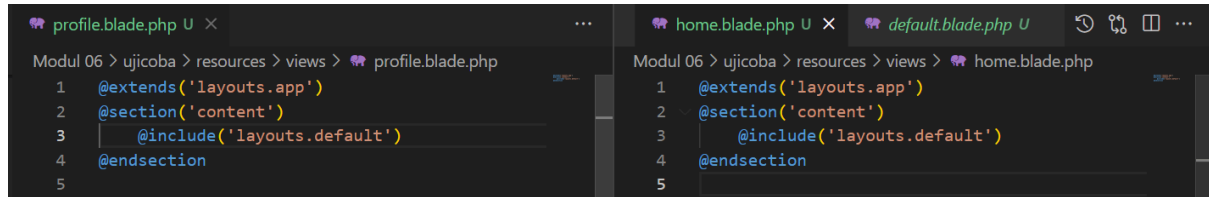
```

55         @enderror
56     </div>
57     <div class="col-md-12 mb-3">
58         <label for="position" class="form-label">Position</label>
59         <select name="position" id="position" class="form-select">
60             @foreach ($positions as $position)
61                 <option value="{{ $position->id }}"
62                     {{ old('position') == $position->id ? 'selected' : '' }}>
63                     {{ $position->code . ' - ' . $position->name }}
64                 </option>
65             @endforeach
66         </select>
67     </div>
68
69 </div>
70 <hr>
71 <div class="row">
72     <div class="col-md-6 d-grid">
73         <a href="{{ route('employees.index') }}" class="btn btn-outline-dark
74             btn-lg mt-3"><i
75                 class="bi-arrow-left-circle me-2"></i> Cancel</a>
76     </div>
77     <div class="col-md-6 d-grid">
78         <button type="submit" class="btn btn-dark btn-lg mt-3"><i
79             class="bi-check-circle me-2"></i>
80             Save</button>
81     </div>
82 </div>
83 </form>
84 </div>
85 @endsection

```

Penyesuaian File View untuk Halaman Home dan Profile

1. Pada file home.blade.php dan profile.blade.php lakukan hal serupa seperti sebelumnya dengan menggunakan pemanggilan komponen layout dan penerapan section untuk isi dari halaman home dan profil. Kemudian lakukan penyesuaian dengan file default dimana file default ini berada didalam folder layouts maka pada bagian include ditambahkan layouts.default



The screenshot displays a code editor with two tabs open: 'profile.blade.php' and 'home.blade.php'. The left pane shows the 'profile.blade.php' file with the following code:

```
Modul 06 > ujicoba > resources > views > profile.blade.php
1 @extends('layouts.app')
2 @section('content')
3     @include('layouts.default')
4 @endsection
5
```

The right pane shows the 'home.blade.php' file with the following code:

```
Modul 06 > ujicoba > resources > views > home.blade.php
1 @extends('layouts.app')
2 @section('content')
3     @include('layouts.default')
4 @endsection
5
```