



Mata Kuliah : Pemrograman Web Lanjut (PWL)
Program Studi : D4 – Teknik Informatika / D4 – Sistem Informasi Bisnis
Semester : 4 (empat) / 5 (lima)
Pertemuan ke- : 5 (lima)

JOBSHEET 05

Blade View, Web Templating(AdminLTE), Datatables

Sesuai dengan **Studi Kasus PWL.pdf**.

Jadi project Laravel 10 kita masih sama dengan menggunakan repositori **PWL_POS**.

Project PWL_POS akan kita gunakan sampai pertemuan 12 nanti, sebagai project yang akan kita pelajari

A. Blade View

Laravel Blade adalah template engine yang digunakan secara default oleh Laravel untuk mempermudah pembuatan tampilan (view) dalam aplikasi web. Blade menawarkan fitur-fitur yang powerful dan mudah digunakan untuk memisahkan logika bisnis dari tampilan antarmuka pengguna (UI). Berikut adalah penjelasan lengkap tentang Laravel Blade View:

1. Apa Itu Blade?

Blade adalah template engine yang disertakan dengan Laravel. Dengan Blade, Anda bisa menulis sintaks HTML yang terintegrasi dengan PHP dengan cara yang lebih bersih dan terstruktur. Meskipun menggunakan Blade, semua file template Blade di Laravel tetap memiliki ekstensi **.blade.php**.

2. Keuntungan Menggunakan Blade



- **Inheritance (Pewarisan Template):** Blade memungkinkan Anda untuk membuat layout dasar yang bisa di-extend oleh halaman lain. Ini mempermudah pembuatan struktur halaman yang konsisten.
- **Komponen:** Anda dapat membuat komponen yang reusable untuk elemen UI yang sering digunakan.
- **Control Structures:** Blade mendukung struktur kontrol seperti if, for, while, dan foreach yang mirip dengan sintaks PHP, namun dengan tampilan yang lebih bersih.
- **Tidak Ada Overhead:** Semua kode Blade dikompilasi menjadi PHP biasa, jadi tidak ada overhead tambahan.

3. Sintaks Dasar Blade

Blade memiliki sintaks yang sangat sederhana dan intuitif. Berikut adalah beberapa contoh penggunaan sintaks Blade:

- **Menampilkan Variabel:**

```
 {{ $name }}
```

Blade akan secara otomatis melindungi (escape) output untuk mencegah serangan XSS (Cross-Site Scripting).

- **Struktur Kontrol:**

```
@if($condition)
    <p>Condition is true</p>
@endif
```

```
@foreach($users as $user)
    <li>{{ $user->name }}</li>
@endforeach
```

- **Komentar**

```
 {{-- Ini adalah komentar yang tidak akan terlihat dalam output HTML --}}
```



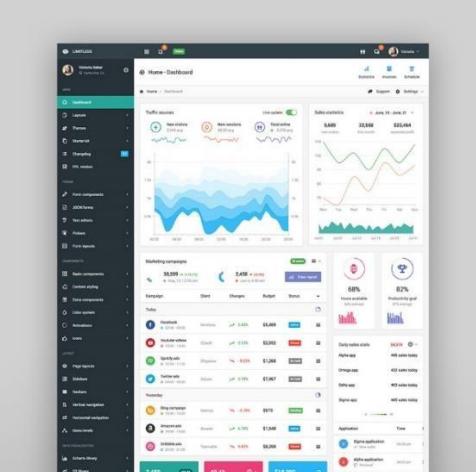
B. Web Template

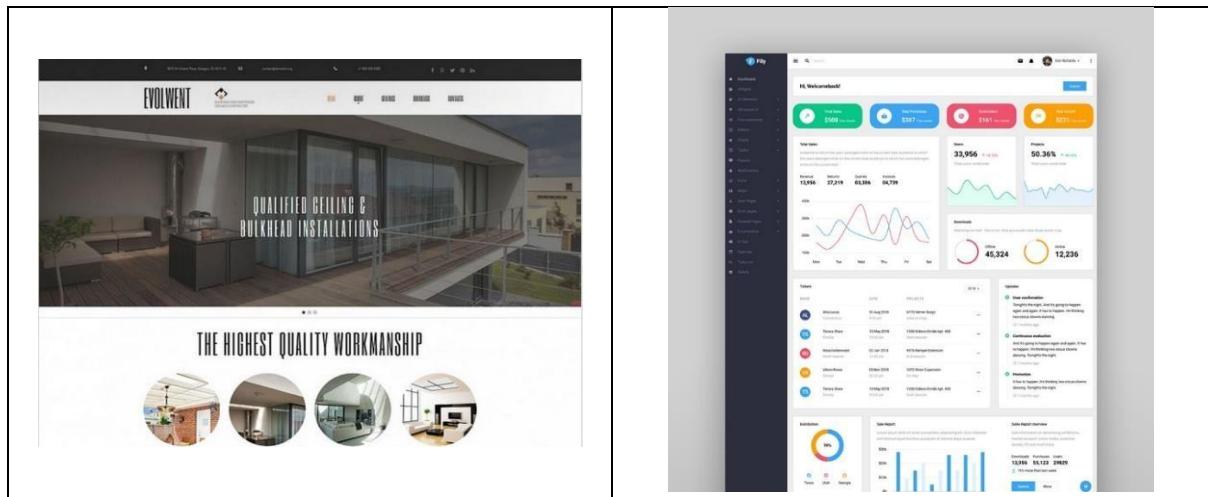
Merupakan suatu file yang sudah memiliki desain tertentu sehingga, kita tinggal memodifikasi bagian-bagian tertentu saja untuk mendapatkan hasil yang sudah bagus. Dengan menggunakan template yang sesuai dengan kebutuhan, kita bisa mengurangi waktu pembuatan suatu website secara drastis.

Secara umum kita bisa menggunakan template website yang sudah ada di internet untuk mempermudah kita dalam membuat website. Ada banyak sekali template yang ada di internet dan kita ambil satu contoh template untuk halaman administrator, seperti **AdminLTE** template. **AdminLTE** adalah salah satu template yang sering digunakan oleh web developer sebagai template backend/admin pada proyek yang sering dikerjakan. Template ini dibuat menggunakan framework bootstrap yang merupakan framework CSS yang paling banyak digunakan di kalangan web desainer

Template AdminLTE dapat di download di <https://adminlte.io>

Contoh Web template

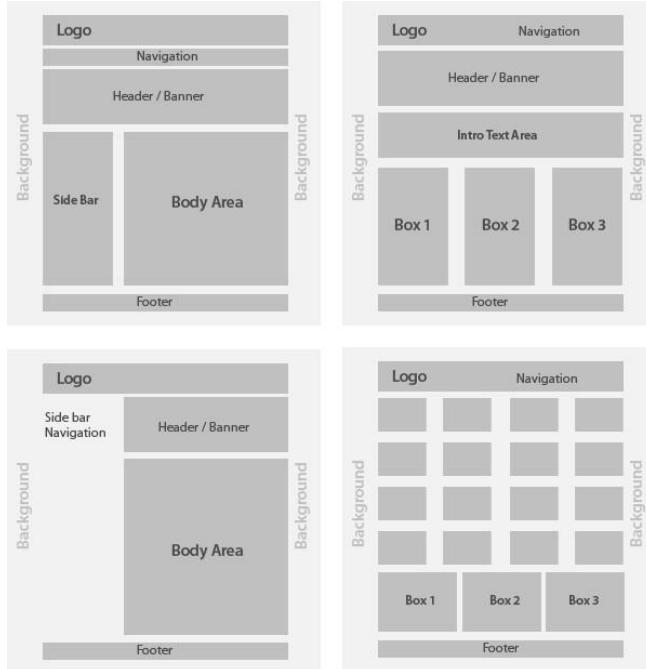
| Frontend Template | Backend Template |
|---|--|
|  |  |



A. Web Layout

Merupakan sekumpulan elemen atau variabel desain yang terkait dengan media tertentu untuk mendukung konsep pembuatan website. Pada dasarnya tujuan penggunaan layout website adalah untuk menggabungkan elemen yang digunakan untuk membangun website. Selain elemen di dalamnya, layout website juga harus memuat beberapa elemen, mulai dari header, navigasi, sidebar, hingga footer. Untuk memahami perbedaan elemen sebuah layout website, berikut penjelasan singkatnya

- **Header** → Pada bagian header, layout designer dapat mengisi bagian ini dengan logo website, navigasi situs, ikon media sosial, dan menu pencarian.
- **Navigasi** → Navigasi dapat dipahami sebagai panduan. Di website, navigasi dapat berkisar dari menu yang muncul di bagian atas halaman web hingga menu pendukung lainnya yang biasa ditemukan di bawah situs.
- **Body/Konten** → Selain itu, terdapat elemen body/konten yang biasanya diisi dengan informasi produk, fitur produk, dan deskripsi produk yang dijual.
- **Sidebar** → elemen ini merupakan elemen yang berada di samping kanan/kiri sebuah website. Element ini kadang sudah tidak dipakai, terutama pada Web Frontend. Namun, sidebar masih digunakan di banyak halaman artikel, yang kemudian akan diisi dengan informasi produk, produk terpopuler, dan navigasi tambahan.
- **Footer** → Elemen terakhir adalah footer. Footer merupakan elemen yang berada dibawah sendiri sebagai menu pendukung atau sebagai identitas dari sebuah website.



C. Layouting AdminLTE

Pada kali ini kita akan melakukan *layouting* template AdminLTE. Tujuan dari *layouting* ini adalah untuk memecah bagian-bagian dari template adminLTE menjadi berbagai elemen. Hal ini dapat mempermudah kita dalam membuat website karena kita bisa menggunakan berulang-ulang elemen yang sudah kita buat nantinya.

Pada layouting kali ini, kita membutuhkan **Blade Template Engine** dari Laravel untuk memecah web menjadi beberapa elemen. Komponen blade yang kita butuhkan untuk proses layouting pada pertemuan kali ini adalah

- `@include()`
- `@extend()`
- `@section()`
- `@push()`
- `@yield()`
- `@stack()`



Praktikum 1 - Layouting AdminLTE:

1. Kita download **AdminLTE v3.2.0** yang rilis pada 8 Feb 2022

Feb 8, 2022

REJack

v3.2.0

bd4d9c7

Compare

AdminLTE v3.2.0 Latest

Breaking Change!
We dropped official node v12, it might be still work but we won't give support for any node 12 bugs/errors.

Top-Features:

- Enhanced Dark Mode
- SVG Font Awesome Icon support
- Docker dev-preset
- Reworked Dark/Light colors
- Custom Scrollbar style (Non-Plugin-Scrollbar)

Contributors

forxer, liljack, and 16 other contributors

Assets 2

[Source code \(zip\)](#) ✓

[Source code \(tar.gz\)](#)

493 56 79 166 51 23 643 people reacted

2. Setelah kita berhasil download, kita ekstrak file yang sudah di download ke folder project **PWL_POS/public**, kemudian kita **rename** folder cukup menjadi **adminlte**

EXPLORER

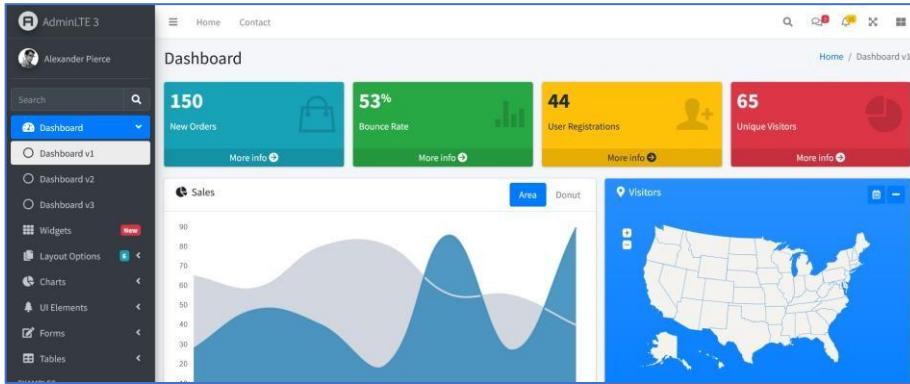
PWL_POS

- app
- bootstrap
- config
- database
- public
- adminlte

- .github
- build
- dist
- docs
- pages
- plugins
- .babelrc.js

3. Selanjutnya kita buka di browser dengan alamat

http://localhost/PWL_POS/public/adminlte maka akan muncul tampilan seperti berikut



4. Kita klik menu **Extras > Blank Page**, page inilah yang akan menjadi dasar web template

AdminLTE 3

- Mailbox
- Pages
- Extras** 1
- Login & Register v1
- Login & Register v2
- Lockscreen
- Legacy User Menu
- Language Menu
- Error 404
- Error 500
- Pace
- Blank Page** 2
- Starter Page

5. Dari sini kita bisa melakukan layouting halaman Blank Page ini menjadi 4 element seperti pada gambar berikut

AdminLTE 3

- Mailbox
- Pages
- Extras**
- Login & Register v1
- Login & Register v2
- 2 - Sidebar / Navigasi**
- Language Menu
- Error 404
- Error 500
- Pace
- Blank Page**
- Starter Page
- Search

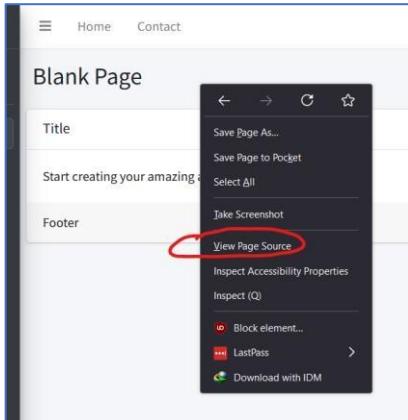
1 - Header / Navbar

3 - Breadcrumb

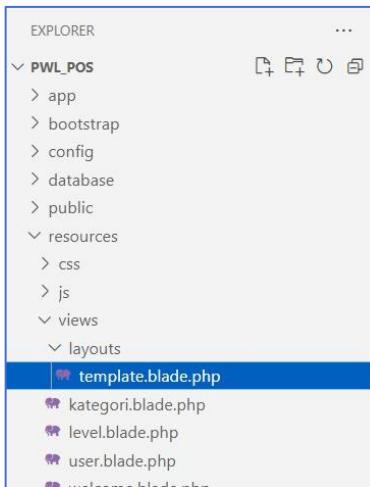
4 - Content

5 - Footer

6. Selanjutnya kita klik kanan halaman **Blank Page** dan klik **view page source**



7. Selanjutnya kita copy page source dari halaman **Blank Page**, kemudia kita *paste* pada **PWL_POS/resource/view/layouts/template.blade.php** (buat dulu folder **layouts** dan file **template.blade.php**)



```
template.blade.php
resources > views > layouts > template.blade.php > ...
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">
6      <title>AdminLTE 3 | Blank Page</title>
7
8      <!-- Google Font: Source Sans Pro -->
9      <link rel="stylesheet" href="https://fonts.googleapis.com/css?family=Source+Sans+Pro:300,400,400i,700&display=fallback">
10     <!-- Font Awesome -->
11     <link rel="stylesheet" href="https://use.fontawesome.com/releases/v5.11.2/assets/webfonts/all.css">
12     <!-- Theme style -->
13     <link rel="stylesheet" href="https://adminlte.io/themes/AdminLTE/dist/css/adminlte.min.css">
14 </head>
15 <body class="hold-transition sidebar-mini">
16     <!-- Site wrapper -->
17     <div class="wrapper">
```

8. File **layouts/template.blade.php** adalah file utama untuk templating website
9. Pada baris **1-14** file **template.blade.php**, kita modifikasi

```
template.blade.php
resources > views > layouts > template.blade.php > ...
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">
6      <title>AdminLTE 3 | Blank Page</title>
7
8      <!-- Google Font: Source Sans Pro -->
9      <link rel="stylesheet" href="https://fonts.googleapis.com/css?family=Source+Sans+Pro:300,400,400i,700&display=fallback">
10     <!-- Font Awesome -->
11     <link rel="stylesheet" href="https://use.fontawesome.com/releases/v5.11.2/assets/webfonts/all.css">
12     <!-- Theme style -->
13     <link rel="stylesheet" href="https://adminlte.io/themes/AdminLTE/dist/css/adminlte.min.css">
14 </head>
```

Menjadi



```
template.blade.php x
resources > views > layouts > template.blade.php > html > body.hold-transition.sidebar-mini > div.wrapper > nav.main-header.navbar.navbar-expand.navbar
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">
6      <title>{{ config('app.name', 'PWL Laravel Starter Code') }}</title>
7
8      <!-- Google Font: Source Sans Pro -->
9      <link rel="stylesheet" href="https://fonts.googleapis.com/css?family=Source+Sans+Pro:300,400,400i,700&display=fallback">
10     <!-- Font Awesome -->
11     <link rel="stylesheet" href="{{ asset('adminlte/plugins/fontawesome-free/css/all.min.css') }}>
12     <!-- Theme style -->
13     <link rel="stylesheet" href="{{ asset('adminlte/dist/css/adminlte.min.css') }}>
14 </head>
```

10. Kemudian kita blok baris **19-153** (baris untuk *element 1-header*), lalu kita **cut**, dan **paste**-kan di file **PWL_POS/resource/view/layouts/header.blade.php** (buat dulu file **header.blade.php** jika belum ada). Sehingga tampilan dari file **template.blade.php** menjadi seperti berikut

```
template.blade.php x  header.blade.php
template.blade.php > html > body.hold-transition.sidebar-mini > div.wrapper > aside.main
14  </head>
15  <body class="hold-transition sidebar-mini">
16  <!-- Site wrapper -->
17  <div class="wrapper">
18      <!-- Navbar -->
19      @include('layouts.header')
20      <!-- /.navbar -->
21
22      <!-- Main Sidebar Container -->
23      <aside class="main-sidebar sidebar-dark-primary elevation-4">
24          <!-- Brand Logo -->
25          <a href=".../index3.html" class="brand-link">
26              AdminLTE 3</span>
28          </a>
```

Baris **19** adalah komponen Blade untuk memanggil elemen **layouts/header.blade.php** agar menjadi satu dengan **template.blade.php** saat di-render nanti.

11. Kita modifikasi baris **25** dan **26** pada **template.blade.php**

```
17  <div class="wrapper">
18      <!-- Navbar -->
19      @include('layouts.header')
20      <!-- /.navbar -->
21
22      <!-- Main Sidebar Container -->
23      <aside class="main-sidebar sidebar-dark-primary elevation-4">
24          <!-- Brand Logo -->
25          <a href=".../index3.html" class="brand-link">
26              
27              <span class="brand-text font-weight-light">AdminLTE 3</span>
28          </a>
29
30      <!-- Sidebar -->
31      <div class="sidebar">
```

Menjadi



```
17 <div class="wrapper">
18   <!-- Navbar -->
19   @include('layouts.header')
20   <!-- /.navbar -->
21
22   <!-- Main Sidebar Container -->
23   <aside class="main-sidebar sidebar-dark-primary elevation-4">
24     <!-- Brand Logo -->
25     <a href="{{ url('/') }}" class="brand-link">
26       PWL - Starter Code</span>
28     </a>
29
30     <!-- Sidebar -->
31     <div class="sidebar">
```

12. Selanjutnya kita blok baris **31-693** (baris untuk **element 2-sidebar**), lalu kita **cut**, dan **paste**-kan di file **PWL_POS/resource/view/layouts/sidebar.blade.php** (buat dulu file **sidebar.blade.php** jika belum ada). Sehingga tampilan dari file **template.blade.php** menjadi seperti berikut

```
22   <!-- Main Sidebar Container -->
23   <aside class="main-sidebar sidebar-dark-primary elevation-4">
24     <!-- Brand Logo -->
25     <a href="{{ url('/') }}" class="brand-link">
26       PWL - Starter Code</span>
28     </a>
29
30     <!-- Sidebar -->
31     @include('layouts.sidebar')
32     <!-- /.sidebar -->
33   </aside>
```

13. Selanjutnya perhatikan baris **87-98** (baris untuk **element 5-footer**), lalu kita **cut**, dan **paste**-kan di file **PWL_POS/resource/view/layouts/footer.blade.php** (buat file **footer.blade.php** jika belum ada). Sehingga tampilan dari file **template.blade.php** menjadi seperti berikut

```
81
82   </section>
83   <!-- /.content -->
84 </div>
85   <!-- /.content-wrapper -->
86
87   @include('layouts.footer')
88 </div>
89   <!-- /.wrapper -->
90
91   <!-- jQuery -->
92   <script src="../../plugins/jquery/jquery.min.js"></script>
```

14. Kemudian kita modifikasi file **template.blade.php** baris **91-100**



```
91  <!-- jQuery -->
92  <script src="../../plugins/jquery/jquery.min.js"></script>
93  <!-- Bootstrap 4 -->
94  <script src="../../plugins/bootstrap/js/bootstrap.bundle.min.js"></script>
95  <!-- AdminLTE App -->
96  <script src="../../dist/js/adminlte.min.js"></script>
97  <!-- AdminLTE for demo purposes -->
98  <script src="../../dist/js/demo.js"></script>
99  </body>
100 </html>
```

Menjadi

```
91  <!-- jQuery -->
92  <script src="{{ asset('adminlte/plugins/jquery/jquery.min.js') }}"></script>
93  <!-- Bootstrap 4 -->
94  <script src="{{ asset('adminlte/plugins/bootstrap/js/bootstrap.bundle.min.js') }}"></script>
95  <!-- AdminLTE App -->
96  <script src="{{ asset('adminlte/dist/js/adminlte.min.js') }}"></script>
97  </body>
98 </html>
```

15. Sekarang masuk pada bagian konten. Konten kita bagi menjadi 2, yaitu elemen untuk **breadcrumb** dan elemen untuk **content**.
16. Perhatikan file `template.blade.php` pada baris **38-52** kita jadikan sebagai elemen **4- breadcrumb**. Kita blok baris **38-52** lalu kita **cut**, dan **paste**-kan di file `PWL_POS/resource/view/layouts/breadcrumb.blade.php` (buat file `breadcrumb.blade.php` jika belum ada). Sehingga tampilan dari file `template.blade.php` menjadi seperti berikut

```
30  <!-- Sidebar -->
31  @include('layouts.sidebar')
32  <!-- /.sidebar -->
33  </aside>
34
35  <!-- Content Wrapper. Contains page content -->
36  <div class="content-wrapper">
37  <!-- Content Header (Page header) -->
38  @include('layouts.breadcrumb')
39
40  <!-- Main content -->
41  <section class="content">
42  <!-- Default box -->
43  <div class="card">
44  <div class="card-header">
45  <h3 class="card-title">Title</h3>
46
```

17. Layout terakhir adalah pada bagian konten. Layout untuk konten bisa kita buat dinamis, sesuai dengan apa yang ingin kita sajikan pada web yang kita bangun.
18. Untuk **content**, kita akan menghapus baris **42-66** pada file `template.blade.php`. dan kita ganti dengan kode seperti ini `@yield('content')`
19. Hasil akhir pada file utama `layouts/template.blade.php` adalah seperti berikut



```
15 <body class="hold-transition sidebar-mini">
16   <!-- Site wrapper -->
17   <div class="wrapper">
18     <!-- Navbar -->
19     @include('layouts.header') 1
20   <!-- /.navbar -->
21
22   <!-- Main Sidebar Container -->
23   <aside class="main-sidebar sidebar-dark-primary elevation-4">
24     <!-- Brand Logo -->
25     <a href="{{ url('/') }}" class="brand-link">
26       PWL - Starter Code</span>
28     </a>
29
30     <!-- Sidebar -->
31     @include('layouts.sidebar') 2
32   <!-- /.sidebar -->
33 </aside>
34
35   <!-- Content Wrapper. Contains page content -->
36   <div class="content-wrapper">
37     <!-- Content Header (Page header) -->
38     @include('layouts.breadcrumb') 3
39
40     <!-- Main content -->
41     <section class="content">
42       @yield('content') 4
43     </section>
44     <!-- /.content -->
45   </div>
46   <!-- /.content-wrapper -->
47
48   @include('layouts.footer') 5
49 </div>
```

20. Selamat kalian sudah selesai dalam melakukan layouting website di laravel.

21. Jangan lupa commit dan push ke github untuk praktikum 1 ini

Praktikum 2 - Penerapan Layouting:

Sekarang kita akan mencoba melakukan penerapan terhadap layouting yang sudah kita lakukan.

1. Kita buat file controller dengan nama `WelcomeController.php`

```
1 <?php
2 namespace App\Http\Controllers;
3
4 class WelcomeController extends Controller
5 {
6   public function index()
7   {
8     $breadcrumb = (object) [
9       'title' => 'Selamat Datang',
10      'list'  => ['Home', 'Welcome']
11    ];
12
13    $activeMenu = 'dashboard';
14
15    return view('welcome', ['breadcrumb' => $breadcrumb, 'activeMenu' => $activeMenu]);
16  }
17}
```

2. Kita buat file pada `PWL_POS/resources/views/welcome.blade.php`



```
welcome.blade.php ×
1  @extends('layouts.template')
2
3  @section('content')
4
5  <div class="card">
6    <div class="card-header">
7      <h3 class="card-title">Halo, apakah??!!</h3>
8      <div class="card-tools"></div>
9    </div>
10   <div class="card-body">
11     Selamat datang semua, ini adalah halaman utama dari aplikasi ini.
12   </div>
13 </div>
14 @endsection
```

3. Kita modifikasi file [PWL_POS/resources/views/layouts/breadcrumb.blade.php](#)

```
welcome.blade.php breadcrumb.blade.php ×
1  <section class="content-header">
2    <div class="container-fluid">
3      <div class="row mb-2">
4        <div class="col-sm-6"><h1>{{ $breadcrumb->title }}</h1></div>
5        <div class="col-sm-6">
6          <ol class="breadcrumb float-sm-right">
7            @foreach($breadcrumb->list as $key => $value)
8              @if($key == count($breadcrumb->list) - 1)
9                <li class="breadcrumb-item active">{{ $value }}</li>
10              @else
11                <li class="breadcrumb-item">{{ $value }}</li>
12              @endif
13            @endforeach
14          </ol>
15        </div>
16      </div>
17    </div>
18  </section>
```

4. Kita modifikasi file [PWL_POS/resources/views/layouts/sidebar.blade.php](#)

```
<div class="sidebar">
  <!-- SidebarSearch Form -->
  <div class="form-inline mt-2">
    <div class="input-group" data-widget="sidebar-search">
      <input class="form-control form-control-sidebar" type="search"
placeholder="Search" aria-label="Search">
      <div class="input-group-append">
        <button class="btn btn-sidebar">
          <i class="fas fa-search fa-fw"></i>
        </button>
      </div>
    </div>
  </div>
  <!-- Sidebar Menu -->
  <nav class="mt-2">
    <ul class="nav nav-pills nav-sidebar flex-column" data-widget="treeview"
role="menu" data-accordion="false">
      <li class="nav-item">
        <a href="{{ url('/') }}" class="nav-link {{ ($activeMenu == 'dashboard')?
'active' : '' }} ">
          <i class="nav-icon fas fa-tachometer-alt"></i>
          <p>Dashboard</p>
        </a>
      </li>
```



```
<li class="nav-header">Data Pengguna</li>
<li class="nav-item">
    <a href="{{ url('/level') }}" class="nav-link {{ ($activeMenu == 'level')?
'active' : '' }} ">
        <i class="nav-icon fas fa-layer-group"></i>
        <p>Level User</p>
    </a>
</li>
<li class="nav-item">
    <a href="{{ url('/user') }}" class="nav-link {{ ($activeMenu == 'user')?
'active' : '' }} ">
        <i class="nav-icon far fa-user"></i>
        <p>Data User</p>
    </a>
</li>
<li class="nav-header">Data Barang</li>
<li class="nav-item">
    <a href="{{ url('/kategori') }}" class="nav-link {{ ($activeMenu ==
'kategori')? 'active' : '' }} ">
        <i class="nav-icon far fa-bookmark"></i>
        <p>Kategori Barang</p>
    </a>
</li>
<li class="nav-item">
    <a href="{{ url('/barang') }}" class="nav-link {{ ($activeMenu ==
'barang')? 'active' : '' }} ">
        <i class="nav-icon far fa-list-alt"></i>
        <p>Data Barang</p>
    </a>
</li>
<li class="nav-header">Data Transaksi</li>
<li class="nav-item">
    <a href="{{ url('/stok') }}" class="nav-link {{ ($activeMenu == 'stok')?
'active' : '' }} ">
        <i class="nav-icon fas fa-cubes"></i>
        <p>Stok Barang</p>
    </a>
</li>
<li class="nav-item">
    <a href="{{ url('/barang') }}" class="nav-link {{ ($activeMenu ==
'penjualan')? 'active' : '' }} ">
        <i class="nav-icon fas fa-cash-register"></i>
        <p>Transaksi Penjualan</p>
    </a>
</li>
</ul>
</nav>
</div>
```

5. Kita tambahkan kode berikut router web.php

```
Route::get('/', [WelcomeController::class, 'index']);
```

6. Sekarang kita coba jalankan di browser dengan mengetikkan url

http://localhost/PWL_POS/public



The screenshot shows a web application interface titled "PWL - Starter Code". On the left is a sidebar with a dark theme containing a navigation menu:

- Dashboard (selected)
- Data Pengguna
- Level User
- Data User
- Data Barang
- Kategori Barang
- Data Barang
- Data Transaksi
- Stok Barang
- Transaksi Penjualan

The main content area displays a "Selamat Datang" message with the text "Halo, apakah!!! Selamat datang semua, ini adalah halaman utama dari aplikasi ini." Below this is a copyright notice: "Copyright © 2014-2021 AdminLTE.io. All rights reserved." and "Version 3.2.0".

This screenshot is identical to the one above, showing the "PWL - Starter Code" application dashboard with the same sidebar and main content area.

7. Jangan lupa commit dan push ke github PWL_POS kalian

INFO

Terdapat 5 file utama pada folder *layouts* yang digunakan dalam membangun sebuah website, yaitu **template**, **header**, **sidebar**, **breadcrumb**, **footer**. Kalian bisa memodifikasi ke-5 file tersebut dalam membangun website.

The screenshot shows a file explorer interface with a sidebar containing icons for search, refresh, and other file operations. The main pane displays the directory structure of the "PWL_POS" project:

- > database
- > public
- < resources
 - > css
 - > js
 - < views
 - < layouts
 - breadcrumb.blade.php
 - footer.blade.php
 - header.blade.php
 - sidebar.blade.php
 - template.blade.php



D. jQuery Datatables di Laravel

jQuery DataTables adalah sebuah plugin jQuery yang sangat populer dan powerful untuk menampilkan dan mengelola data dalam bentuk tabel di halaman web. Untuk menampilkan banyak data, kita bisa menampilkan data tersebut dalam format tabel. DataTable ini merupakan plugin **jQuery** yang dibuat untuk mengelola data informasi dalam bentuk grid / table.

INFO

AdminLTE juga sudah menerapkan library Datatable dalam template nya. Datatable pada AdminLTE dapat kalian cek di
<http://localhost/PWL/public/adminlte/pages/tables/data.html>

Pada praktikum kali ini kita **tidak menggunakan Vite/NodeJS** dalam mengelola datatables, kita cukup menggunakan jQuery datatables bawaan AdminLTE dan library Yajra *laravel-datatables*.

E. Implementasi jQuery Datatables di Laravel

Kita akan menerapkan penggunaan jQuery datatable dari AdminLTE dengan Laravel. Dalam penerapan ini, kita menggunakan library Yajra laravel-datatables.

Praktikum 3 – Implementasi jQuery Datatable di AdminLTE :

1. Kita modifikasi proses CRUD pada tabel `m_user` pada praktikum ini
2. Kita gunakan library Yajra-datable dengan mengetikkan perintah pada CMD
`composer require yajra/laravel-datatables:^10.0` atau
`composer require yajra/laravel-datatables-oracle`
3. Kita modifikasi route `web.php` untuk proses CRUD user

```
6  use App\Http\Controllers\WelcomeController;
7  use Illuminate\Support\Facades\Route;
8  use App\Http\Controllers\UserController;
9
10
11 Route::get('/', [WelcomeController::class, 'index']);
12
13 Route::group(['prefix' => 'user'], function () {
14     Route::get('/', [UserController::class, 'index']);           // menampilkan halaman awal user
15     Route::post('/list', [UserController::class, 'list']);       // menampilkan data user dalam bentuk json untuk datatables
16     Route::get('/create', [UserController::class, 'create']);   // menampilkan halaman form tambah user
17     Route::post('/', [UserController::class, 'store']);         // menyimpan data user baru
18     Route::get('{id}', [UserController::class, 'show']);        // menampilkan detail user
19     Route::get('{id}/edit', [UserController::class, 'edit']);   // menampilkan halaman form edit user
20     Route::put('{id}', [UserController::class, 'update']);     // menyimpan perubahan data user
21     Route::delete('{id}', [UserController::class, 'destroy']); // menghapus data user
22 });
23 }
```

4. Kita buat atau modifikasi penuh untuk `UserController.php`. Kita buat fungsi `index()` untuk menampilkan halaman awal user



```
1 <?php
2 namespace App\Http\Controllers;
3
4 use App\Models\LevelModel;
5 use App\Models\UserModel;
6 use Illuminate\Http\Request;
7 use Yajra\DataTables\Facades\DataTables;
8
9 class UserController extends Controller
10 {
11     // Menampilkan halaman awal user
12     public function index()
13     {
14         $breadcrumb = (object) [
15             'title' => 'Daftar User',
16             'list' => ['Home', 'User']
17         ];
18
19         $page = (object) [
20             'title' => 'Daftar user yang terdaftar dalam sistem'
21         ];
22
23         $activeMenu = 'user'; // set menu yang sedang aktif
24
25         return view('user.index', ['breadcrumb' => $breadcrumb, 'page' => $page, 'activeMenu' => $activeMenu]);
26     }
}
```

5. Lalu kita buat view pada `PWL/resources/views/user/index.blade.php`

```
@extends('layouts.template')

@section('content')
    <div class="card card-outline card-primary">
        <div class="card-header">
            <h3 class="card-title">{{ $page->title }}</h3>
            <div class="card-tools">
                <a class="btn btn-sm btn-primary mt-1" href="{{ url('user/create') }}">Tambah</a>
            </div>
        </div>
        <div class="card-body">
            <table class="table table-bordered table-striped table-hover table-sm"
id="table_user">
                <thead>
                    <tr><th>ID</th><th>Username</th><th>>Nama</th><th>Level
Pengguna</th><th>Aksi</th></tr>
                </thead>
                <tbody>
                </tbody>
            </table>
        </div>
    @endsection

@push('css')
@endpush

@push('js')
<script>
$(document).ready(function() {
    var dataUser = $('#table_user').DataTable({
        // serverSide: true, jika ingin menggunakan server side processing
        serverSide: true,
        ajax: {
            "url": "{{ url('user/list') }}",
            "dataType": "json",
            "type": "POST"
        },
        columns: [
            {

```



```
// nomor urut dari laravel datatable addIndexColumn()
data: "DT_RowIndex",
className: "text-center",
orderable: false,
searchable: false
},{
data: "username",
className: "",
// orderable: true, jika ingin kolom ini bisa diurutkan
orderable: true,
// searchable: true, jika ingin kolom ini bisa dicari
searchable: true
},{
data: "nama",
className: "",
orderable: true,
searchable: true
},{
// mengambil data level hasil dari ORM berelasi
data: "level.level_nama",
className: "",
orderable: false,
searchable: false
},{
data: "aksi",
className: "",
orderable: false,
searchable: false
}
]);
});
</script>
@endpush
```

6. Kemudian kita modifikasi file `template.blade.php` untuk menambahkan library jquery datatables dari template AdminLTE yang kita download dan berada di folder `public`

```
template.blade.php x
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">
6      <title>{{ config('app.name', 'PWL Laravel Starter Code') }}</title>
7
8      <meta name="csrf-token" content="{{ csrf_token() }}> <!-- Untuk mengirimkan token Laravel CSRF pada setiap request ajax --&gt;
9
10     &lt!-- Google Font: Source Sans Pro --&gt;
11     &lt;link rel="stylesheet" href="https://fonts.googleapis.com/css?family=Source+Sans+Pro:300,400,400i,700&amp;display=fallback"&gt;
12     &lt!-- Font Awesome --&gt;
13     &lt;link rel="stylesheet" href="{{ asset('adminlte/plugins/fontawesome-free/css/all.min.css') }}&gt;
14     &lt!-- DataTables --&gt;
15     &lt;link rel="stylesheet" href="{{ asset('adminlte/plugins/datatables-bs4/css/dataTables.bootstrap4.min.css') }}&gt;
16     &lt;link rel="stylesheet" href="{{ asset('adminlte/plugins/datatables-responsive/css/responsive.bootstrap4.min.css') }}&gt;
17     &lt;link rel="stylesheet" href="{{ asset('adminlte/plugins/datatables-buttons/css/buttons.bootstrap4.min.css') }}&gt;
18     &lt!-- Theme style --&gt;
19     &lt;link rel="stylesheet" href="{{ asset('adminlte/dist/css/adminlte.min.css') }}&gt;
20
21     @stack('css') &lt!-- Digunakan untuk memanggil custom css dari perintah push('css') pada masing-masing view --&gt;
22 &lt;/head&gt;</pre>
```



```
60  <!-- jQuery -->
61  <script src="{{ asset('adminlte/plugins/jquery/jquery.min.js') }}"></script>
62  <!-- Bootstrap 4 -->
63  <script src="{{ asset('adminlte/plugins/bootstrap/js/bootstrap.bundle.min.js') }}"></script>
64  <!-- DataTables & Plugins -->
65  <script src="{{ asset('adminlte/plugins/datatables/jquery.dataTables.min.js') }}"></script>
66  <script src="{{ asset('adminlte/plugins/datatables-bs4/js/dataTables.bootstrap4.min.js') }}"></script>
67  <script src="{{ asset('adminlte/plugins/datatables-responsive/js/dataTables.responsive.min.js') }}"></script>
68  <script src="{{ asset('adminlte/plugins/datatables-responsive/js/responsive.bootstrap4.min.js') }}"></script>
69  <script src="{{ asset('adminlte/plugins/datatables-buttons/js/dataTables.buttons.min.js') }}"></script>
70  <script src="{{ asset('adminlte/plugins/datatables-buttons/js/buttons.bootstrap4.min.js') }}"></script>
71  <script src="{{ asset('adminlte/plugins/jscolor/jscolor.min.js') }}"></script>
72  <script src="{{ asset('adminlte/plugins/pdfmake/pdfmake.min.js') }}"></script>
73  <script src="{{ asset('adminlte/plugins/pdfmake/vfs_fonts.js') }}"></script>
74  <script src="{{ asset('adminlte/plugins/datatables-buttons/js/buttons.html5.min.js') }}"></script>
75  <script src="{{ asset('adminlte/plugins/datatables-buttons/js/buttons.print.min.js') }}"></script>
76  <script src="{{ asset('adminlte/plugins/datatables-buttons/js/buttons.colVis.min.js') }}"></script>
77  <!-- AdminLTE App -->
78  <script src="{{ asset('adminlte/dist/js/adminlte.min.js') }}"></script>
79  <script>
80  // Untuk mengirimkan token Laravel CSRF pada setiap request ajax
81  $.ajaxSetup({headers: {'X-CSRF-TOKEN': $('meta[name="csrf-token"]').attr('content')}});
82  </script>
83  @stack('js') <!-- Digunakan untuk memanggil custom js dari perintah push('js') pada masing-masing view -->
84  </body>
85  </html>
```

7. Untuk bisa menangkap request data untuk datatable, kita buat fungsi **list()** pada **UserController.php** seperti berikut

```
// Ambil data user dalam bentuk json untuk datatables
public function list(Request $request)
{
    $users = UserModel::select('user_id', 'username', 'nama', 'level_id')
                    ->with('level');

    return DataTables::of($users)
        // menambahkan kolom index / no urut (default nama kolom: DT_RowIndex)
        ->addIndexColumn()
        ->addColumn('aksi', function ($user) {
            // menambahkan kolom aksi
            $btn = '<a href="'.url('/user/' . $user->user_id).'" class="btn btn-info btn-sm">Detail</a> ';
            $btn .= '<a href="'.url('/user/' . $user->user_id . '/edit').'" class="btn btn-warning btn-sm">Edit</a> ';
            $btn .= '<form class="d-inline-block" method="POST" action="'.url('/user/' . $user->user_id).'">' .
                    '<input type="hidden" name="user_id" value="'.$user->user_id.'" />' .
                    '<input type="hidden" name="aksi" value="edit"' .
                    '<button type="submit" class="btn btn-danger btn-sm" onclick="return confirm(\'Apakah Anda yakin menghapus data ini?\');">Hapus</button></form>';
            return $btn;
        })
        ->rawColumns(['aksi']) // memberitahu bahwa kolom aksi adalah html
        ->make(true);
}
```



8. Sekarang coba jalankan browser, dan klik menu **Data User..!!!** perhatikan dan amati apa yang terjadi.

| ID | Username | Nama | Level Pengguna | Aksi |
|----|-------------|-------------------|----------------|-------------------------|
| 1 | admin | Administrator | Administrator | [Detail] [Edit] [Hapus] |
| 2 | manager | Manager | Manager | [Detail] [Edit] [Hapus] |
| 3 | staff | Staff/Kasir | Staff/Kasir | [Detail] [Edit] [Hapus] |
| 4 | customer_1 | Pelanggan | Staff/Kasir | [Detail] [Edit] [Hapus] |
| 5 | manager_dua | Manager 2 | Manager | [Detail] [Edit] [Hapus] |
| 6 | manager33 | Manager Tiga Tiga | Manager | [Detail] [Edit] [Hapus] |
| 7 | manager56 | Manager55 | Manager | [Detail] [Edit] [Hapus] |
| 8 | manager12 | Manager11 | Manager | [Detail] [Edit] [Hapus] |

9. Selanjutnya kita modifikasi [UserController.php](#) untuk form tambah data user

```
// Menampilkan halaman form tambah user
public function create()
{
    $breadcrumb = (object) [
        'title' => 'Tambah User',
        'list' => ['Home', 'User', 'Tambah']
    ];

    $page = (object) [
        'title' => 'Tambah user baru'
    ];

    $level = LevelModel::all(); // ambil data level untuk ditampilkan di form
    $activeMenu = 'user'; // set menu yang sedang aktif

    return view('user.create', ['breadcrumb' => $breadcrumb, 'page' => $page, 'level' => $level, 'activeMenu' => $activeMenu]);
}
```



10. Sekarang kita buat form untuk menambah data, kita buat file

[PWL/resources/views/user/create.blade.php](#)

```
@extends('layouts.template')

@section('content')
    <div class="card card-outline card-primary">
        <div class="card-header">
            <h3 class="card-title">{{ $page->title }}</h3>
            <div class="card-tools"></div>
        </div>
        <div class="card-body">
            <form method="POST" action="{{ url('user') }}" class="form-horizontal">
                @csrf
                <div class="form-group row">
                    <label class="col-1 control-label col-form-label">Level</label>
                    <div class="col-11">
                        <select class="form-control" id="level_id" name="level_id" required>
                            <option value="">- Pilih Level -</option>
                            @foreach($level as $item)
                                <option value="{{ $item->level_id }}>{{ $item->level_nama }}</option>
                            @endforeach
                        </select>
                        @error('level_id')
                            <small class="form-text text-danger">{{ $message }}</small>
                        @enderror
                    </div>
                </div>
                <div class="form-group row">
                    <label class="col-1 control-label col-form-label">Username</label>
                    <div class="col-11">
                        <input type="text" class="form-control" id="username" name="username" value="{{ old('username') }}" required>
                        @error('username')
                            <small class="form-text text-danger">{{ $message }}</small>
                        @enderror
                    </div>
                </div>
                <div class="form-group row">
                    <label class="col-1 control-label col-form-label">Nama</label>
                    <div class="col-11">
                        <input type="text" class="form-control" id="nama" name="nama" value="{{ old('nama') }}" required>
                        @error('nama')
                            <small class="form-text text-danger">{{ $message }}</small>
                        @enderror
                    </div>
                </div>
                <div class="form-group row">
                    <label class="col-1 control-label col-form-label">Password</label>
                </div>
            </form>
        </div>
    </div>

```



```
<div class="col-11">
    <input type="password" class="form-control" id="password" name="password"
required>
    @error('password')
        <small class="form-text text-danger">{{ $message }}</small>
    @enderror
</div>
</div>
<div class="form-group row">
    <label class="col-1 control-label col-form-label"></label>
    <div class="col-11">
        <button type="submit" class="btn btn-primary btn-sm">Simpan</button>
        <a class="btn btn-sm btn-default ml-1" href="{{ url('user') }}">Kembali</a>
    </div>
</div>
</form>
</div>
</div>
@endsection
@push('css')
@endpush
@push('js')
@endpush
```

11. Kemudian untuk bisa menng-*handle* data yang akan disimpan ke database, kita buat fungsi `store()` di `UserController.php`

```
// Menyimpan data user baru
public function store(Request $request)
{
    $request->validate([
        // username harus diisi, berupa string, minimal 3 karakter, dan bernilai unik di tabel m_user kolom username
        'username' => 'required|string|min:3|unique:m_user,username',
        'nama'      => 'required|string|max:100', // nama harus diisi, berupa string, dan maksimal 100 karakter
        'password'  => 'required|min:5',           // password harus diisi dan minimal 5 karakter
        'level_id'   => 'required|integer'         // level_id harus diisi dan berupa angka
    ]);

    UserModel::create([
        'username' => $request->username,
        'nama'      => $request->nama,
        'password'  => bcrypt($request->password), // password dienkripsi sebelum disimpan
        'level_id'   => $request->level_id
    ]);

    return redirect('/user')->with('success', 'Data user berhasil disimpan');
}
```

12. Sekarang coba kalian buka form tambah data user dengan klik tombol tambah. Amati dan pelajari..!!!



13. Selanjutnya, kita masuk pada bagian menampilkan detail data user (klik tombol **Detail**) pada halaman user. Route yang bertugas untuk menangkap request detail adalah

```
Route::group(['prefix' => 'user'], function () {
    Route::get('/', [UserController::class, 'index']); // menampilkan halaman awal user
    Route::post('/list', [UserController::class, 'list']); // menampilkan data user dalam bentuk json untuk datatables
    Route::get('/create', [UserController::class, 'create']); // menampilkan halaman form tambah user
    Route::post('/', [UserController::class, 'store']); // menyimpan data user baru
    Route::get('{id}', [UserController::class, 'show']); // menampilkan detail user
    Route::get('{id}edit', [UserController::class, 'edit']); // menampilkan halaman form edit user
    Route::put('{id}', [UserController::class, 'update']); // menyimpan perubahan data user
    Route::delete('{id}', [UserController::class, 'destroy']); // menghapus data user
});
```

| Detail user | |
|-------------|---------|
| ID | 2 |
| Level | Manager |
| Username | manager |
| Nama | Manager |
| Password | ***** |



14. Jadi kita buat/modifikasi fungsi **show()** pada **UserController.php** seperti berikut

```
// Menampilkan detail user
public function show(string $id)
{
    $user = UserModel::with('level')->find($id);

    $breadcrumb = (object) [
        'title' => 'Detail User',
        'list' => ['Home', 'User', 'Detail']
    ];

    $page = (object) [
        'title' => 'Detail user'
    ];

    $activeMenu = 'user'; // set menu yang sedang aktif

    return view('user.show', ['breadcrumb' => $breadcrumb, 'page' => $page, 'user' => $user, 'activeMenu' => $activeMenu]);
}
```

15. Kemudian kita buat *view* di **PWL/resources/views/user/show.blade.php**

```
@extends('layouts.template')

@section('content')


### {{ $page->title }}



@empty($user)
    <div class="alert alert-danger alert-dismissible">
        <i class="icon fas fa-ban"></i> Kesalahan!</h5>
        Data yang Anda cari tidak ditemukan.
    </div>
@else
    <table class="table table-bordered table-striped table-hover table-sm">
        <tr>
            <th>ID</th>
            <td>{{ $user->user_id }}</td>
        </tr>
        <tr>
            <th>Level</th>
            <td>{{ $user->level->level_nama }}</td>
        </tr>
        <tr>
            <th>Username</th>
            <td>{{ $user->username }}</td>
        </tr>
        <tr>
            <th>>Nama</th>
            <td>{{ $user->nama }}</td>
        </tr>
        <tr>
            <th>Password</th>
            <td>*****</td>
        </tr>
    </table>
@endempty
<a href="{{ url('user') }}" class="btn btn-sm btn-default mt-2">Kembali</a>
</div>
</div>
@endsection

@push('css')
@endpush


```



```
@push('js')
@endpush
```

16. Sekarang kalian coba untuk melihat detail data user di browser, dan coba untuk mengetikkan id yang salah contoh <http://localhost/PWL/public/user/100> amati apa yang terjadi, dan laporan!!!

The screenshot shows a web application interface. On the left is a sidebar with a dark background and white icons for various data management tasks like dashboard, users, categories, products, transactions, and stock. The 'Data User' option is highlighted with a blue background. The main content area has a light gray background. At the top, there's a navigation bar with links for Home and Contact, and a breadcrumb trail showing Home / User / Detail. Below that, the title 'Tambah User' is displayed. A prominent red error box contains the text 'Kesalahan! Data yang Anda cari tidak ditemukan.' (Error! The data you searched for was not found.). At the bottom of the page, there's a footer with copyright information: 'Copyright © 2014-2021 AdminLTE.io. All rights reserved.' and 'Version 3.2.0'.

Karena user_id 100 tidak ada



17. Selanjutnya, kita masuk pada bagian untuk memodifikasi data user. Route yang bertugas untuk menangkap request edit adalah

```
Route::group(['prefix' => 'user'], function () {
    Route::get('/', [UserController::class, 'index']); // menampilkan halaman awal user
    Route::post('/list', [UserController::class, 'list']); // menampilkan data user dalam bentuk json untuk datatables
    Route::get('/create', [UserController::class, 'create']); // menampilkan halaman form tambah user
    Route::post('/', [UserController::class, 'store']); // menyimpan data user baru
    Route::get('{id}', [UserController::class, 'show']); // menampilkan detail user
    Route::get('{id}/edit', [UserController::class, 'edit']); // menampilkan halaman form edit user
    Route::put('{id}', [UserController::class, 'update']); // menyimpan perubahan data user
    Route::delete('{id}', [UserController::class, 'destroy']); // menghapus data user
});
```

18. Jadi kita buat fungsi **edit()** dan **update()** pada UserController.php

```
// Menampilkan halaman form edit user
public function edit(string $id)
{
    $user = UserModel::find($id);
    $level = LevelModel::all();

    $breadcrumb = (object) [
        'title' => 'Edit User',
        'list' => ['Home', 'User', 'Edit']
    ];

    $page = (object) [
        'title' => 'Edit user'
    ];

    $activeMenu = 'user'; // set menu yang sedang aktif

    return view('user.edit', ['breadcrumb' => $breadcrumb, 'page' => $page, 'user' => $user, 'level' => $level, 'activeMenu' => $activeMenu]);
}

// Menyimpan perubahan data user
public function update(Request $request, string $id)
{
    $request->validate([
        // username harus diisi, berupa string, minimal 3 karakter,
        // dan bernilai unik di tabel m_user kolom username kecuali untuk user dengan id yang sedang diedit
        'username' => 'required|string|min:3|unique:m_user,username,'.$id.',user_id',
        'nama' => 'required|string|max:100', // nama harus diisi, berupa string, dan maksimal 100 karakter
        'password' => 'nullable|min:5', // password bisa diisi (minimal 5 karakter) dan bisa tidak diisi
        'level_id' => 'required|integer' // level_id harus diisi dan berupa angka
    ]);

    UserModel::find($id)->update([
        'username' => $request->username,
        'nama' => $request->nama,
        'password' => $request->password ? bcrypt($request->password) : UserModel::find($id)->password,
        'level_id' => $request->level_id
    ]);

    return redirect('/user')->with('success', 'Data user berhasil diubah');
}
```

19. Selanjutnya, kita buat view untuk melakukan proses edit data user di PWL/resources/views/user/edit.blade.php

```
@extends('layouts.template')

@section('content')
```



```
<div class="card card-outline card-primary">
    <div class="card-header">
        <h3 class="card-title">{{ $page->title }}</h3>
        <div class="card-tools"></div>
    </div>
    <div class="card-body">
        @empty($user)
            <div class="alert alert-danger alert-dismissible">
                <i class="icon fas fa-ban"></i> Kesalahan!
                Data yang Anda cari tidak ditemukan.
            </div>
            <a href="{{ url('user') }}" class="btn btn-sm btn-default mt-2">Kembali</a>
        @else
            <form method="POST" action="{{ url('/user/'.$user->user_id) }}" class="form-horizontal">
                @csrf
                {!! method_field('PUT') !!}<!-- tambahkan baris ini untuk proses edit yang butuh
method PUT -->
                <div class="form-group row">
                    <label class="col-1 control-label col-form-label">Level</label>
                    <div class="col-11">
                        <select class="form-control" id="level_id" name="level_id" required>
                            <option value="">- Pilih Level -</option>
                            @foreach($level as $item)
                                <option value="{{ $item->level_id }}" @if($item->level_id == $user-
level_id) selected @endif>{{ $item->level_nama }}</option>
                            @endforeach
                        </select>
                        @error('level_id')
                            <small class="form-text text-danger">{{ $message }}</small>
                        @enderror
                    </div>
                </div>
                <div class="form-group row">
                    <label class="col-1 control-label col-form-label">Username</label>
                    <div class="col-11">
                        <input type="text" class="form-control" id="username" name="username"
value="{{ old('username', $user->username) }}" required>
                        @error('username')
                            <small class="form-text text-danger">{{ $message }}</small>
                        @enderror
                    </div>
                </div>
                <div class="form-group row">
                    <label class="col-1 control-label col-form-label">Nama</label>
                    <div class="col-11">
                        <input type="text" class="form-control" id="nama" name="nama" value="{{
old('nama', $user->nama) }}" required>
                        @error('nama')
                            <small class="form-text text-danger">{{ $message }}</small>
                        @enderror
                    </div>
                </div>
                <div class="form-group row">
                    <label class="col-1 control-label col-form-label">Password</label>
                    <div class="col-11">
                        <input type="password" class="form-control" id="password" name="password">
                        @error('password')
                            <small class="form-text text-danger">{{ $message }}</small>
                        @else
                            <small class="form-text text-muted">Abaikan (jangan diisi) jika tidak ingin
mengganti password user.</small>
                        @enderror
                    </div>
                </div>
                <div class="form-group row">
                    <label class="col-1 control-label col-form-label"></label>
```



```
<div class="col-11">
    <button type="submit" class="btn btn-primary btn-sm">Simpan</button>
    <a class="btn btn-sm btn-default ml-1" href="{{ url('user') }}>Kembali</a>
</div>
</div>
</form>
@endempty
</div>
</div>
@endsection

@push('css')
@endpush
@push('js')
@endpush
```

20. Sekarang kalian coba untuk mengedit data user di browser, amati, pahami, dan laporan!

Sebelum di edit

The screenshot shows a web-based administration interface titled "PWL - Starter Code". The left sidebar has a dark theme with white icons and text. The "Data User" option is highlighted in blue. The main content area is titled "Edit user". It contains four input fields: "Level" set to "Manager", "Username" set to "Xonexorcist", "Nama" set to "Audric Andhika", and a password field which is empty. Below the form is a note: "Abaikan (jangan diisi) jika tidak ingin mengganti password user." At the bottom are two buttons: "Simpan" and "Kembali". The browser's address bar shows the URL: "localhost/PWL_2025/Minggu05.2/Jobsheet5.2/public/user/19/edit".

Sesudah di edit

| | |
|-----------------|----------------|
| ID | 19 |
| Level | Manager |
| Username | Manager969 |
| Nama | Audric Andhika |
| Password | ***** |



21. Selanjutnya kita akan membuat penanganan untuk tombol hapus. Router `web.php` yang berfungsi untuk menangkap request hapus dengan method DELETE adalah

```
Route::delete('/{id}', [UserController::class, 'destroy']);
```

22. Jadi kita buat fungsi `destroy()` pada `UserController.php`

```
// Menghapus data user
public function destroy(string $id)
{
    $check = UserModel::find($id);
    if (!$check) { // untuk mengecek apakah data user dengan id yang dimaksud ada atau tidak
        return redirect('/user')->with('error', 'Data user tidak ditemukan');
    }

    try{
        UserModel::destroy($id); // Hapus data level

        return redirect('/user')->with('success', 'Data user berhasil dihapus');
    }catch (\Illuminate\Database\QueryException $e){

        // Jika terjadi error ketika menghapus data, redirect kembali ke halaman dengan membawa pesan error
        return redirect('/user')->with('error', 'Data user gagal dihapus karena masih terdapat tabel lain yang terkait dengan data ini');
    }
}
```

23. Selanjutnya kita modifikasi file `PWL/resources/views/user/index.blade.php` untuk menambahkan tampilan jika ada pesan error

```
11 | <div class="card-body">
12 |     @if (session('success'))
13 |         <div class="alert alert-success">{{ session('success') }}</div>
14 |     @endif
15 |     <table class="table table-bordered table-striped table-hover table-sm" id="table_user">
16 |         <thead>
17 |             <tr><th>ID</th><th>Username</th><th>Nama</th><th>Level Pengguna</th><th>Aksi</th></tr>
18 |         </thead>
19 |         </table>
20 |     </div>
```

Menjadi

```
11 | <div class="card-body">
12 |     @if (session('success'))
13 |         <div class="alert alert-success">{{ session('success') }}</div>
14 |     @endif
15 |     @if (session('error'))
16 |         <div class="alert alert-danger">{{ session('error') }}</div>
17 |     @endif
18 |     <table class="table table-bordered table-striped table-hover table-sm" id="table_user">
19 |         <thead>
20 |             <tr><th>ID</th><th>Username</th><th>Nama</th><th>Level Pengguna</th><th>Aksi</th></tr>
21 |         </thead>
22 |         </table>
23 |     </div>
```



24. Kemudian jalankan browser untuk menghapus salah satu data user. Amati dan laporan!

The screenshots show a Laravel application interface for managing users. The top screenshot displays a modal dialog from 'localhost' asking 'Apakah Anda yakin menghapus data ini?' (Are you sure you want to delete this data?). The bottom screenshot shows the user list after one row has been deleted, with a green header bar indicating 'Data user berhasil dihapus' (User data deleted successfully).

| ID | Username | Nama | Level Pengguna | Aksi |
|----|-------------|-------------------|----------------|-------------------------|
| 1 | admin | Administrator | Administrator | [Detail] [Edit] [Hapus] |
| 2 | manager | Manager | Manager | [Detail] [Edit] [Hapus] |
| 3 | staff | Staff/Kasir | Staff/Kasir | [Detail] [Edit] [Hapus] |
| 4 | customer-1 | Pelanggan | Staff/Kasir | [Detail] [Edit] [Hapus] |
| 5 | manager_dua | Manager 2 | Manager | [Detail] [Edit] [Hapus] |
| 6 | manager33 | Manager Tiga Tiga | Manager | [Detail] [Edit] [Hapus] |
| 7 | manager56 | Manager55 | Manager | [Detail] [Edit] [Hapus] |
| 8 | manager12 | Manager11 | Manager | [Detail] [Edit] [Hapus] |
| 9 | Manager969 | Audric Andhika | Manager | [Detail] [Edit] [Hapus] |

25. Selamat, kalian sudah membuat Laravel Starter Code untuk proses CRUD dengan menggunakan template AdminLTE dan plugin jQuery Datatables.

26. Jangan lupa commit dan push ke github PWL_POS kalian

F. Data Searching dan Filtering

Dalam menampilkan sebuah data, kadang kita perlu yang namanya melakukan pencarian (*searching*) berdasarkan kata-kunci (*keyword*) tertentu ataupun penyaringan data berdasarkan suatu kategori (*filtering*).

1. *Searching* (Pencarian):

- Searching adalah proses mencari informasi atau data tertentu yang sesuai dengan kriteria yang diberikan.
- Biasanya, pencarian dilakukan dengan menggunakan kata kunci atau frasa tertentu untuk mencocokkan dengan data yang ada.



- Tujuan dari pencarian adalah untuk menemukan entitas yang cocok dengan kriteria pencarian, baik itu dalam basis data, dalam dokumen teks, atau dalam konteks lainnya.
- Pencarian sering kali melibatkan pencocokan pola atau kata kunci dalam teks atau data yang ada, dan hasilnya mungkin mencakup entitas yang sebagian cocok atau mirip dengan kriteria pencarian.

2. ***Filtering*** (Penyaringan):

- Filtering adalah proses memilih atau membatasi sekumpulan data atau entitas berdasarkan kriteria tertentu.
- Biasanya, filtering dilakukan untuk menyaring data yang sudah ada berdasarkan atribut atau karakteristik tertentu, seperti tanggal, kategori, atau atribut lainnya.



- Tujuan dari penyaringan adalah untuk menyajikan data yang relevan atau relevan dengan kebutuhan pengguna atau kriteria tertentu.
- Filtering dapat dilakukan dengan memilih data berdasarkan nilai yang sesuai dengan kriteria tertentu atau dengan mengecualikan data yang tidak memenuhi kriteria tersebut.

Perbedaan utama antara **searching** dan **filtering** adalah bahwa **searching** berkaitan dengan pencarian data yang sesuai dengan kriteria tertentu, sementara **filtering** berkaitan dengan penyaringan data yang sudah ada berdasarkan kriteria tertentu. Meskipun keduanya sering digunakan bersama-sama dalam aplikasi atau sistem informasi, mereka melayani tujuan yang berbeda dalam proses pengelolaan dan analisis data.

Secara default, jQuery datatables sudah memiliki fitur searching yang sudah terintegrasi dengan laravel yajra-datatables. Akan tetapi, belum ada untuk proses **filtering** datanya. Untuk itu kita akan membuat filtering data pada Laravel Starter Code kita.

The screenshot shows a Laravel application interface. On the left is a sidebar with navigation links: PWL - Starter Code, Home, Contact, Dashboard, Data Pengguna, Level User, Data User (which is selected and highlighted in blue), Data Barang, Kategori Barang, Data Barang, Data Transaksi, Stok Barang, and Transaksi Penjualan. The main content area has a header with Home, Contact, and a search bar. Below it is a title 'Daftar User' and a sub-header 'Daftar user yang terdaftar dalam sistem'. There is a 'Tambah' button. The central part is a table with columns: ID, Username, Nama, Level Pengguna, and Aksi. The 'Aksi' column contains three buttons: Detail, Edit, and Hapus. A red box highlights the search input field at the top right of the table. The table shows 6 entries. At the bottom, there are buttons for Previous, Next, and a page number indicator (1).

| ID | Username | Nama | Level Pengguna | Aksi |
|----|---------------|---------------|----------------|-------------------------|
| 1 | admin | Administrator | Administrator | [Detail] [Edit] [Hapus] |
| 2 | manager | Manager | Manager | [Detail] [Edit] [Hapus] |
| 3 | supervisor | Supervisor | Supervisor | [Detail] [Edit] [Hapus] |
| 4 | kasir1 | Kasir 1 | Kasir/Staff | [Detail] [Edit] [Hapus] |
| 5 | kasir2 | Kasir 2 | Kasir/Staff | [Detail] [Edit] [Hapus] |
| 6 | administrator | Administrator | Administrator | [Detail] [Edit] [Hapus] |

Praktikum 4 – Implementasi *Filtering* Datatables:

Kita akan menerapkan filtering pada datatable yang sudah kita buat. Hal ini akan mempermudah kita dalam mengelompokkan suatu data sesuai kategori tertentu. Langkah-langkah yang kita kerjakan sebagai berikut

1. Kita modifikasi fungsi `index()` di `UserController.php` untuk menambahkan data yang ingin dijadikan kategori untuk data **filtering**



```
// Menampilkan halaman awal user
public function index()
{
    $breadcrumb = (object) [
        'title' => 'Daftar User',
        'list' => ['Home', 'User']
    ];

    $page = (object) [
        'title' => 'Daftar user yang terdaftar dalam sistem'
    ];

    $activeMenu = 'user'; // set menu yang sedang aktif

    $level = LevelModel::all(); // ambil data level untuk filter level

    return view('user.index', ['breadcrumb' => $breadcrumb, 'page' => $page, 'level' => $level, 'activeMenu' => $activeMenu]);
}
```

2. Kemudian kita modifikasi view untuk menampilkan data filtering pada `PWL/resources/views/user/index.blade.php`

```
@if (session('error'))
    <div class="alert alert-danger">{{ session('error') }}</div>
@endif
<div class="row">
    <div class="col-md-12">
        <div class="form-group row">
            <label class="col-1 control-label col-form-label">Filter:</label>
            <div class="col-3">
                <select class="form-control" id="level_id" name="level_id" required>
                    <option value="">- Semua -</option>
                    @foreach($level as $item)
                        <option value="{{ $item->level_id }}>{{ $item->level_nama }}</option>
                    @endforeach
                </select>
                <small class="form-text text-muted">Level Pengguna</small>
            </div>
        </div>
    </div>
</div>
<table class="table table-bordered table-striped table-hover table-sm" id="table_user">
    <thead>
        <tr><th>ID</th><th>Username</th><th>Nama</th><th>Level Pengguna</th><th>Aksi</th></tr>
    </thead>
    <tbody>
    </tbody>
</table>
</div>
@endif
```

3. Selanjutnya, tetap pada view `index.blade.php`, kita tambahkan kode berikut pada deklarasi ajax di datatable. Kode ini digunakan untuk mengirimkan data untuk filtering



```
46 @push('js')
47 <script>
48 $(document).ready(function() {
49     var dataUser = $('#table_user').DataTable({
50         serverSide: true, // serverSide: true, jika ingin menggunakan server side processing
51         ajax: {
52             "url": "{{ url('user/list') }}",
53             "dataType": "json",
54             "type": "POST",
55             "data": function (d) {
56                 d.level_id = $('#level_id').val();
57             }
58         },
59         columns: [
60             {
61                 data: "DT_RowIndex", // nomor urut dari laravel datatable addIndexColumn()
```

4. Kemudian kita edit pada bagian akhir script @push('js') untuk menambahkan listener jika data filtering dipilih

```
81         data: "aksi",
82         className: "",
83         orderable: false, // orderable: true, jika ingin kolom ini bisa diurutkan
84         searchable: false // searchable: true, jika ingin kolom ini bisa dicari
85     }
86
87 ];
88 });
89
90 $('#level_id').on('change', function() {
91     dataUser.ajax.reload();
92 });
93
94 });
95 </script>
96 @endpush
```

5. Tahapan akhir adalah memodifikasi fungsi **list()** pada **UserController.php** yang digunakan untuk menampilkan data pada datatable

```
// Ambil data user dalam bentuk json untuk datatables
public function list(Request $request)
{
    $users = UserModel::select('user_id', 'username', 'nama', 'level_id')
        ->with('level');

    // Filter data user berdasarkan level_id
    if ($request->level_id) {
        $users->where('level_id', $request->level_id);
    }

    return DataTables::of($users)
        ->addIndexColumn() // menambahkan kolom index / no urut (default nama kolom: DT_RowIndex)
        ->addColumn('aksi', function ($user) { // menambahkan kolom aksi
            $btn = '<a href="'.url('/user/' . $user->user_id).'" class="btn btn-info btn-sm">Detail</a> ';
            $btn .= '<a href="'.url('/user/' . $user->user_id . '/edit).'" class="btn btn-warning btn-sm">Edit</a> ';
            $btn .= '<form class="d-inline-block" method="POST" action="'.url('/user/' . $user->user_id).'">';
            $btn .= csrf_field() . method_field('DELETE') .
            '<button type="submit" class="btn btn-danger btn-sm" onclick="return confirm(\'Apakah Anda yakin menghapus data ini?\')>Delete</button>';
            return $btn;
        })
        ->rawColumns(['aksi']) // memberitahu bahwa kolom aksi adalah html
        ->make(true);
```



6. Bagian akhir adalah kita coba jalankan di browser dengan akses menu user, maka akan tampil seperti berikut

The screenshot shows a 'Daftar User' (User List) page from a Laravel application. The sidebar on the left includes links for Dashboard, Data Pengguna, Level User, and Data User (which is highlighted in blue). The main content area has a search bar at the top right and a 'Filtering' section with a dropdown menu set to '- Semua -'. A 'Searching' section with a 'Search:' input field is also present. Below these are buttons for 'Show 10 entries' and 'Show All'. The main table lists six users with columns for ID, Username, Name, Level Pengguna (User Type), and Aksi (Actions). Each row contains three buttons: 'Detail' (blue), 'Edit' (yellow), and 'Hapus' (red). At the bottom, a message says 'Showing 1 to 6 of 6 entries'.

| ID | Username | Nama | Level Pengguna | Aksi |
|----|---------------|---------------|----------------|-------------------------|
| 1 | admin | Administrator | Administrator | [Detail] [Edit] [Hapus] |
| 2 | manager | Manager | Manager | [Detail] [Edit] [Hapus] |
| 3 | supervisor | Supervisor | Supervisor | [Detail] [Edit] [Hapus] |
| 4 | kasir1 | Kasir 1 | Kasir/Staff | [Detail] [Edit] [Hapus] |
| 5 | kasir2 | Kasir 2 | Kasir/Staff | [Detail] [Edit] [Hapus] |
| 6 | administrator | Administrator | Administrator | [Detail] [Edit] [Hapus] |

This screenshot shows the same 'Daftar User' page but with a different set of users listed in the table. The table now contains eight users, each with a different combination of Username, Name, and User Type. The actions column remains the same with 'Detail', 'Edit', and 'Hapus' buttons.

| ID | Username | Nama | Level Pengguna | Aksi |
|----|-------------|-------------------|----------------|-------------------------|
| 1 | admin | Administrator | Administrator | [Detail] [Edit] [Hapus] |
| 2 | manager | Manager | Manager | [Detail] [Edit] [Hapus] |
| 3 | staff | Staff/Kasir | Staff/Kasir | [Detail] [Edit] [Hapus] |
| 4 | customer-1 | Pelanggan | Staff/Kasir | [Detail] [Edit] [Hapus] |
| 5 | manager_dua | Manager 2 | Manager | [Detail] [Edit] [Hapus] |
| 6 | manager33 | Manager Tiga Tiga | Manager | [Detail] [Edit] [Hapus] |
| 7 | manager56 | Manager55 | Manager | [Detail] [Edit] [Hapus] |
| 8 | manager12 | Manager11 | Manager | [Detail] [Edit] [Hapus] |

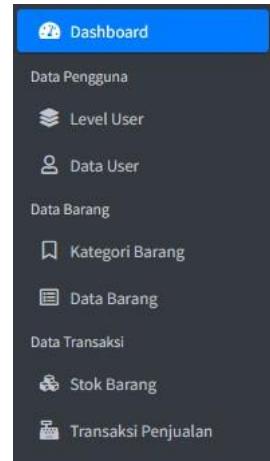
7. Selamat, sekarang Laravel Starter Code sudah ada filtering dan searching data. Starter Code sudah bisa digunakan dalam membangun sebuah sistem berbasis website.
8. Jangan lupa commit dan push ke github PWL_POS kalian



G. Tugas

Implementasikan web layout dan datatables, pada menu berikut ini

- ✓ Tabel m_level
- ✓ Tabel m_kategori
- ✓ Tabel m_supplier
- ✓ Tabel m_barang





1. Tabel m_level

- Views

Index.blade.php

```
1 @extends('layouts.template')
2
3 @section('content')
4     <div class="card card-outline card-primary">
5         <div class="card-header">
6             <h3 class="card-title">{{ $page->title }}</h3>
7             <div class="card-tools">
8                 <a class="btn btn-sm btn-primary mt-1" href="{{ url('level/create') }}">Tambah</a>
9             </div>
10        </div>
11        <div class="card-body">
12            @if (session('success'))
13                <div class="alert alert-success">{{ session('success') }}</div>
14            @endif
15            @if (session('error'))
16                <div class="alert alert-danger">{{ session('error') }}</div>
17            @endif
18            <table class="table table-bordered table-striped table-hover table-sm" id="table_level">
19                <thead>
20                    <tr>
21                        <th>ID</th>
22                        <th>Kode Level</th>
23                        <th>Nama Level</th>
24                        <th>Aksi</th>
25                    </tr>
26                </thead>
27            </table>
28        </div>
29    </div>
30 @endsection
31
32 @push('css')
33 @endpush
34
35 @push('js')
36     <script>
37         $(document).ready(function() {
38             var dataUser = $('#table_level').DataTable
39             ({
40                 // serverside: true, jika ingin menggunakan server side processing
41                 serverSide: true,
42                 ajax: {
43                     "url": "{{ url('level/list') }}",
44                     "dataType": "json",
45                     "type": "POST",
46                 },
47                 columns: [
48                     {
49                         data: "DT_RowIndex",
50                         className: "text-center",
51                         orderable: false,
52                         searchable: false
53                     },{
54                         data: "level_kode",
55                         className: "",
56                         orderable: true,
57                         searchable: true
58                     },{
59                         data: "level_nama",
60                         className: "",
61                         orderable: true,
62                         searchable: true
63                     },{
64                         data: "aksi",
65                         className: ""
66                     }
67                 ]
68             })
69         });
70     </script>
71 
```



Create.blade.php

```
resources > views > level > create.blade.php > ...
1   @extends('layouts.template')
2
3   @section('content')
4     <div class="card card-outline card-primary">
5       <div class="card-header">
6         <h3 class="card-title">{!! $page->title !!}
```

Edit.blade.php

```
resources > views > level > edit.blade.php > ...
4     <div class="card card-outline card-primary">
5       <div class="card-header">
6         <h3 class="card-title">{!! $page->title !!}
```



Show.blade.php

```
resources > views > level > show.blade.php > ...
1  @extends('layouts.template')
2
3  @section('content')
4      <div class="card card-outline card-primary">
5          <div class="card-header">
6              <h3 class="card-title">{( $page->title )}</h3>
7              <div class="card-tools"></div>
8          </div>
9          <div class="card-body">
10             @empty($level)
11                 <div class="alert alert-danger alert-dismissible">
12                     <h5><i class="icon fas fa-ban"></i> Kesalahan!</h5>
13                     Data yang Anda cari tidak ditemukan.
14                 </div>
15             @else
16                 <table class="table table-bordered table-striped table-hover table-sm">
17                     <thead>
18                         <tr>
19                             <th>ID</th>
20                             <td>{($level->level_id )}</td>
21                         </tr>
22                         <th>Kode Level</th>
23                         <td>{($level->level_kode )}</td>
24                     </tr>
25                     <th>Nama Level</th>
26                     <td>{($level->level_nama )}</td>
27                 </tr>
28             </table>
29         @endifempty
30         <a href="{{ url('level') }}" class="btn btn-sm btn-default mt-2">Kembali</a>
31     </div>
32  </div>
33  @endsection
34  @push('css')
35
36  @endpush
37
```

• LevelController

```
public function index()
{
    $breadcrumb = (object)[
        'title' => 'Daftar Level',
        'list' => ['Home', 'Level']
    ];

    $page = (object)[
        'title' => 'Daftar Level yang terdaftar dalam sistem'
    ];

    $activeMenu = 'level';
    return view('level.index', ['breadcrumb' => $breadcrumb, 'page' => $page, 'activeMenu' => $activeMenu]);
}

public function list()
{
    $levels = LevelModel::select('level_id', 'level_kode', 'level_nama');

    return Datatables::of($levels)
        ->addIndexColumn()
        ->addColumn('aksi', function ($level) {
            $btn = '<a href="'.url('/level')."/$level->level_id.'" class="btn btn-info btn-sm">Detail</a> ';
            $btn .= '<a href="'.url('/level')."/$level->level_id . '/edit'" class="btn btn-warning btn-sm">Edit</a> ';
            $btn .= '<form class="d-inline-block" method="POST" action="'.url('/level/'.$level->level_id).'">';
            $btn .= csrf_field();
            $btn .= method_field('DELETE');
            $btn .= '<button type="submit" class="btn btn-danger btn-sm" onclick="return confirm(`Apakah Anda yakin menghapus data ini?`);">Hapus</button></form>';
            return $btn;
        })
        ->rawColumns(['aksi']);
    $make(true);
}

public function create()
{
    $breadcrumb = (object)[
        'title' => 'Tambah Level',
        'list' => ['Home', 'Level', 'Tambah']
    ];

    $page = (object)[
        'title' => 'Tambah level baru'
    ];

    $activeMenu = 'level';
    return view('level.create', ['breadcrumb' => $breadcrumb, 'page' => $page, 'activeMenu' => $activeMenu]);
}

public function store(Request $request)
{
    $request->validate([
        'level_kode' => 'required|string|unique:m_level,level_kode',
        'level_nama' => 'required|string|max:100',
    ]);

    LevelModel::create([
        'level_kode' => $request->level_kode,
        'level_nama' => $request->level_nama,
    ]);

    return redirect('/level')->with('success', 'Data level berhasil disimpan');
}
```



```
public function show(string $id)
{
    $level = LevelModel::find($id);

    $breadcrumb = (object)[
        'title' => 'Detail Level',
        'list' => ['Home', 'Level', 'Detail']
    ];

    $page = (object)[
        'title' => 'Detail level'
    ];

    $activeMenu = 'level';

    return view('level.show', ['breadcrumb' => $breadcrumb, 'page' => $page, 'level' => $level, 'activeMenu' => $activeMenu]);
}

public function edit(string $id)
{
    $level = LevelModel::find($id);

    $breadcrumb = (object) [
        'title' => 'Edit Level',
        'list' => ['Home', 'Level', 'Edit']
    ];

    $page = (object) [
        'title' => 'Edit level'
    ];

    $activeMenu = 'level';
    return view('level.edit', ['breadcrumb' => $breadcrumb, 'page' => $page, 'level' => $level, 'activeMenu' => $activeMenu]);
}

public function update(Request $request, string $id)
{
    $request->validate([
        'level_kode' => 'required|string|unique:m_level,level_kode,'.$id.',level_id',
        'level_nama' => 'required|string|max:100',
    ]);

    LevelModel::find($id)->update([
        'level_kode' => $request->level_kode,
        'level_nama' => $request->level_nama,
    ]);

    return redirect('/level')->with('success', 'Data level berhasil diubah');
}

public function destroy(string $id)
{
    $check = LevelModel::find($id);
    if (!$check) {
        return redirect('/level')->with('error', 'Data level tidak ditemukan');
    }

    try {
        LevelModel::destroy($id);

        return redirect('/level')->with('success', 'Data level berhasil dihapus');
    } catch (\Illuminate\Database\QueryException $e) {

        return redirect('/level')->with('error', 'Data level gagal dihapus karena masih terdapat tabel lain yang terkait dengan data ini');
    }
}
```

- Route

```
Route::group(['prefix' => 'level'], function () {
    Route::get('/', [LevelController::class, 'index']);
    Route::post('/list', [LevelController::class, 'list']);
    Route::get('/create', [LevelController::class, 'create']);
    Route::post('/', [LevelController::class, 'store']);
    Route::get('/{id}', [LevelController::class, 'show']);
    Route::get('/{id}/edit', [LevelController::class, 'edit']);
    Route::put('/{id}', [LevelController::class, 'update']);
    Route::delete('/{id}', [LevelController::class, 'destroy']);
});
```



Screenshot of the 'Daftar Level' (List of Levels) page. The sidebar shows 'Level User' is selected. The main table lists three levels:

| ID | Kode Level | Nama Level | Aksi |
|----|------------|---------------|-------------------------|
| 1 | ADM | Administrator | [Detail] [Edit] [Hapus] |
| 2 | MNG | Manager | [Detail] [Edit] [Hapus] |
| 3 | STF | Staff/Kasir | [Detail] [Edit] [Hapus] |

Form tambah level

Tambah Level

Tambah level baru

| | |
|------------|-------------------------|
| Kode Level | CEO |
| Nama Level | Chief Executive Officer |

[Simpan] [Kembali]

Daftar Level

Daftar Level yang terdaftar dalam sistem

Data level berhasil disimpan

| ID | Kode Level | Nama Level | Aksi |
|----|------------|-------------------------|-------------------------|
| 1 | ADM | Administrator | [Detail] [Edit] [Hapus] |
| 2 | MNG | Manager | [Detail] [Edit] [Hapus] |
| 3 | STF | Staff/Kasir | [Detail] [Edit] [Hapus] |
| 4 | CEO | Chief Executive Officer | [Detail] [Edit] [Hapus] |

Showing 1 to 4 of 4 entries

Detail level

Detail Level

Detail level

| | |
|------------|-------------------------|
| ID | 7 |
| Kode Level | CEO |
| Nama Level | Chief Executive Officer |

[Kembali]

Delete

Daftar Level

Daftar Level yang terdaftar dalam sistem

Data level berhasil dihapus

| ID | Kode Level | Nama Level | Aksi |
|----|------------|---------------|-------------------------|
| 1 | ADM | Administrator | [Detail] [Edit] [Hapus] |
| 2 | MNG | Manager | [Detail] [Edit] [Hapus] |
| 3 | STF | Staff/Kasir | [Detail] [Edit] [Hapus] |

Showing 1 to 3 of 3 entries



2. Tabel m_kategori

- Views

Index.blade.php

```
resources > views > kategori > index.blade.php > script > <function> > @dataUser
1   @extends('layouts.template')
2
3   @section('content')
4       <div class="card card-outline card-primary">
5           <div class="card-header">
6               <h3 class="card-title">{{ $page->title }}
```



Create.blade.php

```
resources > views > kategori >  create.blade.php > ...
1 @extends('layouts.template')
2
3 @section('content')
4     <div class="card card-outline card-primary">
5         <div class="card-header">
6             <h3 class="card-title">{{ $page->title }}</h3>
7             <div class="card-tools"></div>
8         </div>
9         <div class="card-body">
10            <form method="POST" action="{{ url('kategori') }}" class="form-horizontal">
11                @csrf
12                <div class="form-group row">
13                    <label class="col-1 control-label col-form-label">Kode Kategori</label>
14                    <div class="col-11">
15                        <input type="text" class="form-control" id="kategori_kode" name="kategori.kode" value="{{ old('kategori.kode') }}" required>
16                        @error('kategori.kode')
17                            <small class="form-text text-danger">{{ $message }}</small>
18                        @enderror
19                    </div>
20                </div>
21                <div class="form-group row">
22                    <label class="col-1 control-label col-form-label">Nama Kategori</label>
23                    <div class="col-11">
24                        <input type="text" class="form-control" id="kategori_nama" name="kategori.nama" value="{{ old('kategori.nama') }}" required>
25                        @error('kategori.nama')
26                            <small class="form-text text-danger">{{ $message }}</small>
27                        @enderror
28                    </div>
29                </div>
30                <div class="form-group row">
31                    <label class="col-1 control-label col-form-label"></label>
32                    <div class="col-11">
33                        <button type="submit" class="btn btn-primary btn-sm">Simpan</button>
34                        <a class="btn btn-sm btn-default ml-1" href="{{ url('kategori') }}>Kembali</a>
35                    </div>
36                </div>
37            </form>
38        </div>
39    @endsection
40
41 @push('css')
42 @endpush
43
44 @push('js')
45 @endpush
```

Show.blade.php

```
resources > views > kategori >  show.blade.php > ...
1 @extends('layouts.template')
2
3 @section('content')
4     <div class="card card-outline card-primary">
5         <div class="card-header">
6             <h3 class="card-title">{{ $page->title }}</h3>
7             <div class="card-tools"></div>
8         </div>
9         <div class="card-body">
10            @empty($kategori)
11                <div class="alert alert-danger alert-dismissible">
12                    <h5><i class="icon fas fa-ban"></i> Kesalahan!</h5>
13                    Data yang Anda cari tidak ditemukan.
14                </div>
15            @else
16                <table class="table table-bordered table-striped table-hover table-sm">
17                    <tr>
18                        <th>ID</th>
19                        <td>{{ $kategori->kategori_id }}</td>
20                    </tr>
21                    <tr>
22                        <th>Kode Kategori</th>
23                        <td>{{ $kategori->kategori.kode }}</td>
24                    </tr>
25                    <tr>
26                        <th>Nama Kategori</th>
27                        <td>{{ $kategori->kategori.nama }}</td>
28                    </tr>
29                </table>
30            @endempty
31            <a href="{{ url('kategori') }}" class="btn btn-sm btn-default mt-2">Kembali</a>
32        </div>
33    </div>
34 @endsection
35
36 @push('css')
37 @endpush
38
39 @push('js')
40 @endpush
```



Create.blade.php

```
resources > views > kategori >  create.blade.php > ...
1  @extends('layouts.template')
2
3  @section('content')
4      <div class="card card-outline card-primary">
5          <div class="card-header">
6              <h3 class="card-title">{{ $page->title }}</h3>
7          <div class="card-tools"></div>
8      </div>
9      <div class="card-body">
10         <form method="POST" action="{{ url('kategori') }}" class="form-horizontal">
11             @csrf
12             <div class="form-group row">
13                 <label class="col-1 control-label col-form-label">Kode Kategori</label>
14                 <div class="col-11">
15                     <input type="text" class="form-control" id="kategori_kode" name="kategori_kode" value="{{ old('kategori_kode') }}" required>
16                     @error('kategori_kode')
17                         <small class="form-text text-danger">{{ $message }}</small>
18                     @enderror
19                 </div>
20             </div>
21             <div class="form-group row">
22                 <label class="col-1 control-label col-form-label">Nama Kategori</label>
23                 <div class="col-11">
24                     <input type="text" class="form-control" id="kategori_nama" name="kategori_nama" value="{{ old('kategori_nama') }}" required>
25                     @error('kategori_nama')
26                         <small class="form-text text-danger">{{ $message }}</small>
27                     @enderror
28                 </div>
29             <div class="form-group row">
30                 <label class="col-1 control-label col-form-label"></label>
31                 <div class="col-11">
32                     <button type="submit" class="btn btn-primary btn-sm">Simpan</button>
33                     <a class="btn btn-sm btn-default ml-1" href="{{ url('kategori') }}>Kembali</a>
34                 </div>
35             </div>
36         </form>
37     </div>
38     </div>
39     </div>
40     @endsection
41
42     @push('css')
43     @endpush
44
45     @push('js')
```

• KategoriController

```
public function index(){
    $breadcrumb = (object)[
        'title' => 'Daftar Kategori',
        'list' => ['Home', 'Kategori']
    ];

    $page = (object)[
        'title' => 'Daftar Kategori yang terdaftar dalam sistem'
    ];

    $activeMenu = 'kategori';
    return view('kategori.index', ['breadcrumb' => $breadcrumb, 'page' => $page, 'activeMenu' => $activeMenu]);
}

public function list()
{
    $kategori = KategoriModel::select('kategori_id', 'kategori.kode', 'kategori.nama');

    return Datatables::of($kategori)
        ->addIndexColumn()
        ->addColumn('aksi', function ($kategori) {
            $btn = '<a href="'.url('kategori').'/'. $kategori->kategori_id.'>'. class="btn btn-info btn-sm">Detail</a> ';
            $btn .= '<a href="'.url('kategori').'/'. $kategori->kategori_id . '/edit'.'>'. class="btn btn-warning btn-sm">Edit</a> ';
            $btn .= '<form class="d-inline-block" method="POST" action="'. url('kategori').'/'. $kategori->kategori_id .'>' .
                csrf_field() . method_field('DELETE') .
                '<button type="submit" class="btn btn-danger btn-sm" onclick="return confirm(\''Apakah Anda yakin menghapus data ini?\')">>Hapus</button>' ;
            return $btn;
        })
        ->rawColumns(['aksi'])
        ->make(true);
}

public function create()
{
    $breadcrumb = (object)[
        'title' => 'Tambah Kategori',
        'list' => ['Home', 'Kategori', 'Tambah']
    ];

    $page = (object)[
        'title' => 'Tambah kategori baru'
    ];

    $activeMenu = 'kategori';
    return view('kategori.create', ['breadcrumb' => $breadcrumb, 'page' => $page, 'activeMenu' => $activeMenu]);
}

public function store(Request $request)
{
    $request->validate([
        'kategori.kode' => 'required|string|unique:m_kategori,kategori.kode',
        'kategori.nama' => 'required|string|max:100',
    ]);

    KategoriModel::create([
        'kategori.kode' => $request->kategori.kode,
        'kategori.nama' => $request->kategori.nama,
    ]);

    return redirect('/kategori')->with('success', 'Data kategori berhasil disimpan');
}
```



```
public function show(string $id)
{
    $kategori = KategoriModel::find($id);

    $breadcrumb = (object)[
        'title' => 'Detail Kategori',
        'list' => ['Home', 'Kategori', 'Detail']
    ];

    $page = (object)[
        'title' => 'Detail kategori'
    ];

    $activeMenu = 'kategori';

    return view('kategori.show', ['breadcrumb' => $breadcrumb, 'page' => $page, 'kategori' => $kategori, 'activeMenu' => $activeMenu]);
}

public function edit(string $id)
{
    $kategori = KategoriModel::find($id);

    $breadcrumb = (object)[
        'title' => 'Edit Kategori',
        'list' => ['Home', 'Kategori', 'Edit']
    ];

    $page = (object)[
        'title' => 'Edit kategori'
    ];

    $activeMenu = 'kategori';

    return view('kategori.edit', ['breadcrumb' => $breadcrumb, 'page' => $page, 'kategori' => $kategori, 'activeMenu' => $activeMenu]);
}

public function update(Request $request, string $id)
{
    $request->validate([
        'kategori.kode' => 'required|string|unique:m_kategori,kategori.kode,'.$id.',kategori_id',
        'kategori.nama' => 'required|string|max:100',
    ]);

    KategoriModel::find($id)->update([
        'kategori.kode' => $request->kategori.kode,
        'kategori.nama' => $request->kategori.nama,
    ]);

    return redirect('/kategori')->with('success', 'Data kategori berhasil diubah');
}

public function destroy(string $id)
{
    $check = KategoriModel::find($id);
    if (!$check) {
        return redirect('/kategori')->with('error', 'Data kategori tidak ditemukan');
    }

    try {
        KategoriModel::destroy($id);

        return redirect('/kategori')->with('success', 'Data kategori berhasil dihapus');
    } catch (\Illuminate\Database\QueryException $e) []
    return redirect('/kategori')->with('error', 'Data kategori gagal dihapus karena masih terdapat tabel lain yang terkait dengan data ini');
}
```

- Route

```
Route::group(['prefix' => 'kategori'], function () {
    Route::get('/', [KategoriController::class, 'index']);
    Route::post('/list', [KategoriController::class, 'list']);
    Route::get('/create', [KategoriController::class, 'create']);
    Route::post('/', [KategoriController::class, 'store']);
    Route::get('/{id}', [KategoriController::class, 'show']);
    Route::get('/{id}/edit', [KategoriController::class, 'edit']);
    Route::put('/{id}', [KategoriController::class, 'update']);
    Route::delete('/{id}', [KategoriController::class, 'destroy']);
});
```



localhost/PWL_2025/Mirggid05.2/jobsheet5.2/public/kategori

PWL - Starter Code

Daftar Kategori

| ID | Kode Kategori | Nama Kategori | Aksi |
|----|---------------|----------------|-------------------------|
| 1 | MKN | Makanan | [Detail] [Edit] [Hapus] |
| 2 | PKN | Pakaian | [Detail] [Edit] [Hapus] |
| 3 | SNCK | Snack | [Detail] [Edit] [Hapus] |
| 4 | BHBH | Buah-buahan | [Detail] [Edit] [Hapus] |
| 5 | SYR | Sayuran | [Detail] [Edit] [Hapus] |
| 6 | MNR | Minuman Ringan | [Detail] [Edit] [Hapus] |

Showing 1 to 6 of 6 entries

Copyright © 2014-2021 AdminLTE.io. All rights reserved.

Version 3.2.0

Tambah Data

Tambah Kategori

Tambah kategori baru

| | |
|---------------|--------|
| Kode Kategori | SPT |
| Nama Kategori | Sepatu |

Simpan **Kembali**

Daftar Kategori

| ID | Kode Kategori | Nama Kategori | Aksi |
|----|---------------|----------------|-------------------------|
| 1 | MKN | Makanan | [Detail] [Edit] [Hapus] |
| 2 | PKN | Pakaian | [Detail] [Edit] [Hapus] |
| 3 | SNCK | Snack | [Detail] [Edit] [Hapus] |
| 4 | BHBH | Buah-buahan | [Detail] [Edit] [Hapus] |
| 5 | SYR | Sayuran | [Detail] [Edit] [Hapus] |
| 6 | MNR | Minuman Ringan | [Detail] [Edit] [Hapus] |
| 7 | SPT | Sepatu | [Detail] [Edit] [Hapus] |

Showing 1 to 7 of 7 entries

Data kategori berhasil disimpan

Detail Data

Detail Kategori

Detail kategori

| | |
|---------------|--------|
| ID | 15 |
| Kode Kategori | SPT |
| Nama Kategori | Sepatu |

Kembali

Edit

Edit Kategori

Edit kategori

| | |
|---------------|--------|
| Kode Kategori | SPT |
| Nama Kategori | Sepatu |

Simpan **Kembali**



Hapus

| Daftar Kategori | | | |
|---|---------------|----------------|---|
| Daftar Kategori yang terdaftar dalam sistem | | | |
| Data kategori berhasil dihapus | | | |
| Show 10 entries | Search: | | |
| ID | Kode Kategori | Nama Kategori | Aksi |
| 1 | MKN | Makanan | Detail Edit Hapus |
| 2 | PKN | Pakaian | Detail Edit Hapus |
| 3 | SNCK | Snack | Detail Edit Hapus |
| 4 | BHBH | Buah-buahan | Detail Edit Hapus |
| 5 | SYR | Sayuran | Detail Edit Hapus |
| 6 | MNR | Minuman Ringan | Detail Edit Hapus |

Showing 1 to 6 of 6 entries

Previous **1** Next



3. Tabel t_stok

- Views

Index.blade.php

```
resources > views > stok > index.blade.php > ...
1  @extends('layouts.template')
2  |
3  @section('content')
4      <div class="card card-outline card-primary">
5          <div class="card-header">
6              <h3 class="card-title">{{ $page->title }}</h3>
7              <div class="card-tools">
8                  <a class="btn btn-sm btn-primary mt-1" href="{{ url('stok/create') }}">Tambah</a>
9              </div>
10         </div>
11         <div class="card-body">
12             @if (session('success'))
13                 <div class="alert alert-success">{{ session('success') }}</div>
14             @endif
15             @if (session('error'))
16                 <div class="alert alert-danger">{{ session('error') }}</div>
17             @endif
18             <div class="row">
19                 <div class="col-md-12">
20                     <div class="form-group row">
21                         <label class="col-1 control-label col-form-label">Filter:</label>
22                         <div class="col-3">
23                             <select class="form-control" id="user_id" name="user_id">
24                                 <option value=""> Semua</option>
25                                 @foreach ($user as $item)
26                                     <option value="{{ $item->user_id }}>{{ $item->nama }}</option>
27                                 @endforeach
28                             </select>
29                             <small class="form-text text-muted">Nama User</small>
30                         </div>
31                     </div>
32                 </div>
33             </div>
34             <table class="table table-bordered table-striped table-hover table-sm" id="table_stok">
35                 <thead>
36                     <tr>
37                         <th>ID</th>
38                         <th>Nama Barang</th>
39                         <th>Nama User</th>
40                         <th>Tanggal Stok</th>
41                         <th>Jumlah Stok</th>
42                         <th>Aksi</th>
43                     </tr>
44                 </thead>
45             </table>
46         </div>
47     @endsection
48
49     @push('css')
50     @endpush
51
52     @push('js')
53         <script>
54             $(document).ready(function() {
55                 var dataStok = $('#table_stok').DataTable({
56                     serverSide: true,
57                     ajax: {
58                         "url": "{{ url('stok/list') }}",
59                         "dataType": "json",
60                         "type": "POST",
61                         "data": function (d) {
62                             d.user_id = $('#user_id').val(); // Filter berdasarkan user_id
63                             d._token = "{{ csrf_token() }}"; // Tambahkan CSRF token
64                         }
65                     },
66                     columns: [
67                         {
68                             data: "DT_RowIndex",
69                             className: "text-center",
70                             orderable: false,
71                             searchable: false
72                         },
73                         {
74                             data: "barang.barang_nama", // Mengambil nama barang dari relasi
75                             className: "",
76                             orderable: false,
77                             searchable: false
78                         },
79                         {
80                             data: "user.nama", // Mengambil nama user dari relasi
81                             className: "",
82                             orderable: false,
83                             searchable: false
84                         },
85                         {
86                         }
87                     ]
88                 });
89             });
90         </script>
91     @endpush
92 
```



```
87      },
88      {
89        data: "stok_tanggal",
90        className: "",
91        orderable: true,
92        searchable: true
93      },
94      {
95        data: "stok_jumlah",
96        className: "",
97        orderable: true,
98        searchable: false
99      },
100     {
101       data: "aksi",
102       className: "",
103       orderable: false,
104       searchable: false
105     }
106   );
107   $('#user_id').on('change', function() {
108     dataStok.ajax.reload();
109   });
110 });
111 </script>
112 @endpush
```

Create.blade.php

```
resources > views > stok >  create.blade.php > ...
1  @extends('layouts.template')
2
3  @section('content')
4    <div class="card card-outline card-primary">
5      <div class="card-header">
6        <h3 class="card-title">{{ $page->title }}</h3>
7        <div class="card-tools"></div>
8      </div>
9      <div class="card-body">
10        <form method="POST" action="{{ url('stok') }}" class="form-horizontal">
11          @csrf
12          <div class="form-group row">
13            <label class="col-1 control-label col-form-label">Barang</label>
14            <div class="col-11">
15              <select class="form-control" id="barang_id" name="barang_id" required>
16                <option value=""> - Pilih Barang </option>
17                @foreach($barang as $item)
18                  <option value="{{ $item->barang_id }}>{{ $item->barang_nama }}</option>
19                @endforeach
20              </select>
21              @error('barang_id')
22                <small class="form-text text-danger">{{ $message }}</small>
23              @enderror
24            </div>
25          </div>
26          <div class="form-group row">
27            <label class="col-1 control-label col-form-label">User</label>
28            <div class="col-11">
29              <select class="form-control" id="user_id" name="user_id" required>
30                <option value=""> - Pilih User </option>
31                @foreach($user as $item)
32                  <option value="{{ $item->user_id }}>{{ $item->nama }}</option>
33                @endforeach
34              </select>
35              @error('user_id')
36                <small class="form-text text-danger">{{ $message }}</small>
37              @enderror
38            </div>
39          </div>
40          <div class="form-group row">
41            <label class="col-1 control-label col-form-label">Tanggal Stok</label>
42            <div class="col-11">
43              <input type="date" class="form-control" id="stok_tanggal" name="stok_tanggal" value="{{ old('stok_tanggal') }}" required>
44              @error('stok_tanggal')
45                <small class="form-text text-danger">{{ $message }}</small>
46              @enderror
47            </div>
48          </div>
49          <div class="form-group row">
50            <label class="col-1 control-label col-form-label">Jumlah Stok</label>
51            <div class="col-11">
52              <input type="number" class="form-control" id="stok_jumlah" name="stok_jumlah" value="{{ old('stok_jumlah') }}" required>
53              @error('stok_jumlah')
54                <small class="form-text text-danger">{{ $message }}</small>
55              @enderror
56            </div>
57          </div>
58          <div class="form-group row">
59            <label class="col-1 control-label col-form-label"></label>
60            <div class="col-11">
61              <button type="submit" class="btn btn-primary btn-sm">Simpan</button>
62              <a class="btn btn-sm btn-default ml-1" href="{{ url('stok') }}">Kembali</a>
63            </div>
64          </div>
65        </form>
66      </div>
67    </div>
68  @endsection
69
70  @push('css')
71  @endpush
72
73  @push('js')
74  @endpush
```



Edit.blade.php

```
resources > views > stok > edit.blade.php > ...
1  @extends('layouts.template')
2
3  @section('content')
4      <div class="card card-outline card-primary">
5          <div class="card-header">
6              <h3 class="card-title">{!! $page->title !!}
7              <div class="card-tools"></div>
8          </div>
9          <div class="card-body">
10             @empty($stok)
11                 <div class="alert alert-danger alert-dismissible">
12                     <i class="icon fas fa-ban"></i> Kesalahan!
13                     Data yang Anda cari tidak ditemukan.
14                 </div>
15                 <a href="{!! url('stok') !!}" class="btn btn-sm btn-default mt-2">Kembali</a>
16             @else
17                 <form method="POST" action="{!! url('/stok/'.$stok->stok_id) !!}" class="form-horizontal">
18                     @csrf
19                     {!! method_field('PUT') !!}<!-- Method PUT untuk proses edit -->
20                     <div class="form-group row">
21                         <label class="col-1 control-label col-form-label">Barang</label>
22                         <div class="col-11">
23                             <select class="form-control" id="barang_id" name="barang_id" required>
24                                 <option value="">- Pilih Barang -</option>
25                                 @foreach($barang as $item)
26                                     <option value="{!! $item->barang_id !!}">
27                                         @if($item->barang_id == $stok->barang_id) selected @endif
28                                         {!! $item->barang_nama !!}
29                                     </option>
30                                 @endforeach
31                             </select>
32                             @error('barang_id')
33                                 <small class="form-text text-danger">{!! $message !!}Simpan</button>
75                             <a class="btn btn-sm btn-default ml-1" href="{!! url('stok') !!}">Kembali</a>
76                         </div>
77                     </div>
78                 </div>
79             </form>
80             @endempty
81         </div>
82     </div>
83     @endsection
```



Show.blade.php

```
resources > views > stok > show.blade.php > ...
1  @extends('layouts.template')
2
3  @section('content')
4      <div class="card card-outline card-primary">
5          <div class="card-header">
6              <h3 class="card-title">{{ $page->title }}</h3>
7              <div class="card-tools"></div>
8          </div>
9          <div class="card-body">
10             @empty($stok)
11                 <div class="alert alert-danger alert-dismissible">
12                     <i class="icon fas fa-ban"></i> Kesalahan!
13                     Data yang Anda cari tidak ditemukan.
14                 </div>
15             @else
16                 <table class="table table-bordered table-striped table-hover table-sm">
17                     <tr>
18                         <th>ID</th>
19                         <td>{{ $stok->stok_id }}</td>
20                     </tr>
21                     <tr>
22                         <th>Nama Barang</th>
23                         <td>{{ $stok->barang->barang_nama }}</td>
24                     </tr>
25                     <tr>
26                         <th>Nama User</th>
27                         <td>{{ $stok->user->nama }}</td>
28                     </tr>
29                     <tr>
30                         <th>Tanggal Stok</th>
31                         <td>{{ $stok->stok_tanggal }}</td>
32                     </tr>
33                     <tr>
34                         <th>Jumlah Stok</th>
35                         <td>{{ $stok->stok_jumlah }}</td>
36                     </tr>
37                 </table>
38             @endempty
39             <a href="{{ url('stok') }}" class="btn btn-sm btn-default mt-2">Kembali</a>
40         </div>
41     </div>
42 @endsection
43
44 @push('css')
45 @endpush
```

- Route

```
Route::group(['prefix' => 'stok'], function () {
    Route::get('/', [StokController::class, 'index']);
    Route::post('/list', [StokController::class, 'list']);
    Route::get('/create', [StokController::class, 'create']);
    Route::post('/', [StokController::class, 'store']);
    Route::get('/{id}', [StokController::class, 'show']);
    Route::get('/{id}/edit', [StokController::class, 'edit']);
    Route::put('/{id}', [StokController::class, 'update']);
    Route::delete('/{id}', [StokController::class, 'destroy']);
});
```



• StokController

```
public function index()
{
    $breadcrumb = (object)[
        'title' => 'Daftar Stok',
        'list' => ['Home', 'Stok']
    ];

    $page = (object)[
        'title' => 'Daftar stok yang terdaftar dalam sistem'
    ];

    $activeMenu = 'stok';
    $user = UserModel::all();

    return view('stok.index', ['breadcrumb' => $breadcrumb, 'page' => $page, 'user' => $user, 'activeMenu' => $activeMenu]);
}

public function list(Request $request)
{
    $stok = StokModel::select('stok_id', 'barang_id', 'user_id', 'stok_tanggal', 'stok_jumlah')
        ->with(['barang', 'user']);

    if ($request->user_id) {
        $stok->where('user_id', $request->user_id);
    }

    return DataTables::of($stok)
        ->addIndexColumn()
        ->addColumn('aksi', function ($stok) {
            $btn = '<a href="'.url('/stok/'.$stok->stok_id).'" class="btn btn-info btn-sm">Detail</a> ';
            $btn .= '<a href="'.url('/stok/'.$stok->stok_id).'/edit'" class="btn btn-warning btn-sm">Edit</a> ';
            $btn .= '<form class="d-inline-block" method="POST" action="'.url('/stok/'.$stok->stok_id).'">';
            . csrf_field(). method_field('DELETE');
            . '<button type="submit" class="btn btn-danger btn-sm" onclick="return confirm(\'Apakah Anda yakin menghapus data ini?\')">>Hapus</button></form>';
            return $btn;
        })
        ->rawColumns(['aksi'])
        ->make(true);
}

public function create()
{
    $breadcrumb = (object)[
        'title' => 'Tambah Stok',
        'list' => ['Home', 'Stok', 'Tambah']
    ];

    $page = (object)[
        'title' => 'Tambah stok baru'
    ];

    $barang = BarangModel::all();
    $user = UserModel::all();
    $activeMenu = 'stok';

    return view('stok.create', ['breadcrumb' => $breadcrumb, 'page' => $page, 'barang' => $barang, 'user' => $user, 'activeMenu' => $activeMenu]);
}

public function store(Request $request)
{
    $request->validate([
        'barang_id' => 'required|integer',
        'user_id' => 'required|integer',
        'stok_tanggal' => 'required|date',
        'stok_jumlah' => 'required|integer|min:1',
    ]);

    StokModel::create([
        'barang_id' => $request->barang_id,
        'user_id' => $request->user_id,
        'stok_tanggal' => $request->stok_tanggal,
        'stok_jumlah' => $request->stok_jumlah,
    ]);
}

public function show(string $id)
{
    $stok = StokModel::with(['barang', 'user'])->find($id);

    $breadcrumb = (object)[
        'title' => 'Detail Stok',
        'list' => ['Home', 'Stok', 'Detail']
    ];

    $page = (object)[
        'title' => 'Detail stok'
    ];

    $activeMenu = 'stok';

    return view('stok.show', ['breadcrumb' => $breadcrumb, 'page' => $page, 'stok' => $stok, 'activeMenu' => $activeMenu]);
}

public function edit(string $id)
{
    $stok = StokModel::find($id);

    $breadcrumb = (object)[
        'title' => 'Edit Stok',
        'list' => ['Home', 'Stok', 'Edit']
    ];

    $page = (object)[
        'title' => 'Edit stok'
    ];

    $barang = BarangModel::all();
    $user = UserModel::all();
    $activeMenu = 'stok';

    return view('stok.edit', ['breadcrumb' => $breadcrumb, 'page' => $page, 'stok' => $stok, 'barang' => $barang, 'user' => $user, 'activeMenu' => $activeMenu]);
}
```



```
public function update(Request $request, string $id)
{
    $request->validate([
        'barang_id' => 'required|integer',
        'user_id' => 'required|integer',
        'stok_tanggal' => 'required|date',
        'stok_jumlah' => 'required|integer|min:1',
    ]);

    StokModel::find($id)->update([
        'barang_id' => $request->barang_id,
        'user_id' => $request->user_id,
        'stok_tanggal' => $request->stok_tanggal,
        'stok_jumlah' => $request->stok_jumlah,
    ]);

    return redirect('/stok')->with('success', 'Data stok berhasil diubah');
}

public function destroy(string $id)
{
    $check = StokModel::find($id);
    if (!$check) {
        return redirect('/stok')->with('error', 'Data stok tidak ditemukan');
    }

    try {
        StokModel::destroy($id);

        return redirect('/stok')->with('success', 'Data stok berhasil dihapus');
    } catch (\Illuminate\Database\QueryException $e) {

        return redirect('/stok')->with('error', 'Data stok gagal dihapus karena masih terdapat tabel lain yang terkait dengan data ini');
    }
}
```

| ID | Nama Barang | Nama User | Tanggal Stok | Jumlah Stok | Aksi |
|----|-----------------|---------------|---------------------|-------------|-----------------------|
| 1 | Mie Goreng | Administrator | 2025-03-02 17:48:40 | 30 | [Detail, Edit, Hapus] |
| 2 | Ayam Bakar | Administrator | 2025-03-02 17:48:40 | 25 | [Detail, Edit, Hapus] |
| 3 | Kaos Polos | Manager | 2025-03-02 17:48:40 | 20 | [Detail, Edit, Hapus] |
| 4 | Celana Jeans | Manager | 2025-03-02 17:48:40 | 15 | [Detail, Edit, Hapus] |
| 5 | Keripik Kentang | Manager | 2025-03-02 17:48:40 | 10 | [Detail, Edit, Hapus] |
| 6 | Biskuit Coklat | Staff/Kasir | 2025-03-02 17:48:40 | 50 | [Detail, Edit, Hapus] |
| 7 | Apel Fuji | Staff/Kasir | 2025-03-02 17:48:40 | 40 | [Detail, Edit, Hapus] |
| 8 | Jeruk Manis | Staff/Kasir | 2025-03-02 17:48:40 | 35 | [Detail, Edit, Hapus] |
| 9 | Wortel Segar | Administrator | 2025-03-02 17:48:40 | 45 | [Detail, Edit, Hapus] |
| 10 | Bayam Hijau | Administrator | 2025-03-02 17:48:40 | 60 | [Detail, Edit, Hapus] |

Tambah Data

Tambah Stok

Tambah stok baru

Barang: - Pilih Barang -

User: - Pilih User -

Tanggal Stok: dd/mm/yyyy

Jumlah Stok:



Detail Data

Detail Stok

Home / Stok / Detail

| Detail stok | |
|--------------|---------------------|
| ID | 1 |
| Nama Barang | Mie Goreng |
| Nama User | Administrator |
| Tanggal Stok | 2025-03-02 17:48:40 |
| Jumlah Stok | 30 |

[Kembali](#)

Edit Data

Edit Stok

Home / Stok / Edit

| Edit stok | |
|--------------|---------------|
| Barang | Mie Goreng |
| User | Administrator |
| Tanggal Stok | dd/mm/yyyy |
| Jumlah Stok | 30 |

[Simpan](#) [Kembali](#)

Delete Data

Daftar Stok

Home / Stok

Daftar stok yang terdaftar dalam sistem

[Tambah](#)

Data stok berhasil dihapus

| ID | Nama Barang | Nama User | Tanggal Stok | Jumlah Stok | Aksi |
|----|-----------------|---------------|---------------------|-------------|---|
| 1 | Mie Goreng | Administrator | 2025-03-02 17:48:40 | 30 | Detail Edit Hapus |
| 2 | Ayam Bakar | Administrator | 2025-03-02 17:48:40 | 25 | Detail Edit Hapus |
| 3 | Kaos Polos | Manager | 2025-03-02 17:48:40 | 20 | Detail Edit Hapus |
| 4 | Celana Jeans | Manager | 2025-03-02 17:48:40 | 15 | Detail Edit Hapus |
| 5 | Keripik Kentang | Manager | 2025-03-02 17:48:40 | 10 | Detail Edit Hapus |
| 6 | Biskuit Coklat | Staff/Kasir | 2025-03-02 17:48:40 | 50 | Detail Edit Hapus |
| 7 | Apel Fuji | Staff/Kasir | 2025-03-02 17:48:40 | 40 | Detail Edit Hapus |
| 8 | Jeruk Manis | Staff/Kasir | 2025-03-02 17:48:40 | 35 | Detail Edit Hapus |
| 9 | Wortel Segar | Administrator | 2025-03-02 17:48:40 | 45 | Detail Edit Hapus |

Show 10 entries Search:

Previous [1](#) Next



4. Tabel m_barang

- Views

Index.blade.php

```
1  @extends('layouts.template')
2
3  @section('content')
4      <div class="card card-outline card-primary">
5          <div class="card-header">
6              <h3 class="card-title">{{ $page->title }}</h3>
7              <div class="card-tools">
8                  | <a class="btn btn-sm btn-primary mt-1" href="{{ url('barang/create') }}">Tambah</a>
9              </div>
10         </div>
11         <div class="card-body">
12             @if (session('success'))
13                 <div class="alert alert-success">{{ session('success') }}</div>
14             @endif
15             @if (session('error'))
16                 <div class="alert alert-danger">{{ session('error') }}</div>
17             @endif
18             <div class="row">
19                 <div class="col-md-12">
20                     <div class="form-group row">
21                         <label class="col-1 control-label col-form-label">Filter:</label>
22                         <div class="col-3">
23                             <select class="form-control" id="kategori_id" name="kategori_id">
24                                 <option value=""> Semua</option>
25                                 @foreach ($kategori as $item)
26                                     <option value="{{ $item->kategori_id }}>{{ $item->kategori_nama }}</option>
27                                 @endforeach
28                             </select>
29                             <small class="form-text text-muted">Kategori Barang</small>
30                         </div>
31                     </div>
32                 </div>
33             <table class="table table-bordered table-striped table-hover table-sm" id="table_barang">
34                 <thead>
35                     <tr>
36                         <th>ID</th>
37                         <th>Kode Barang</th>
38                         <th>Nama Barang</th>
39                         <th>Kategori</th>
40                         <th>Harga Beli</th>
41                         <th>Harga Jual</th>
42                         <th>Aksi</th>
43                     </tr>
44                 </thead>
45             </table>
46         </div>
47     </div>
48     @endsection
49
50     @push('css')
51     @endpush
52
53
54     @push('js')
55         <script>
56             $(document).ready(function() {
57                 var dataBarang = $('#table_barang').DataTable({
58                     serverSide: true,
59                     ajax: {
60                         "url": "{{ url('barang/list') }}",
61                         "dataType": "json",
62                         "type": "POST",
63                         "data": function (d) {
64                             d.kategori_id = $('#kategori_id').val();
65                             d._token = "{{ csrf_token() }}"; // Tambahkan CSRF token
66                         }
67                     },
68                     columns: [
69                         {
70                             data: "DT_RowIndex",
71                             className: "text-center",
72                             orderable: false,
73                             searchable: false
74                         },
75                         {
76                             data: "barang_kode",
77                             className: "",
78                             orderable: true,
79                             searchable: true
80                         },
81                         {
82                             data: "barang_nama",
83                             className: ""
84                         }
85                     ]
86                 });
87             });
88         </script>
89     @endpush
90 
```



```
82         data: "barang_nama",
83         className: "",
84         orderable: true,
85         searchable: true
86     },
87     ],
88     data: "kategori.kategori_nama", // Mengambil nama kategori dari relasi
89     className: "",
90     orderable: false,
91     searchable: false
92 },
93 [
94     {
95         data: "harga_beli",
96         className: "",
97         orderable: true,
98         searchable: false
99     },
100    {
101        data: "harga_jual",
102        className: "",
103        orderable: true,
104        searchable: false
105    },
106    {
107        data: "aksi",
108        className: "",
109        orderable: false,
110        searchable: false
111    }
112]);
113
114 $("#kategori_id").on('change', function() {
115     dataBarang.ajax.reload();
116 });
117 });
118 </script>
119 @endpush
```

Create.blade.php

```
1  @extends('layouts.template')
2
3  @section('content')
4      <div class="card card-outline card-primary">
5          <div class="card-header">
6              <h3 class="card-title">{{ $page->title }}</h3>
7              <div class="card-tools"></div>
8          </div>
9          <div class="card-body">
10             <form method="POST" action="{{ url('barang') }}" class="form-horizontal">
11                 @csrf
12                 <div class="form-group row">
13                     <label class="col-1 control-label col-form-label">Kategori</label>
14                     <div class="col-11">
15                         <select class="form-control" id="kategori_id" name="kategori_id" required>
16                             <option value=""> Pilih Kategori </option>
17                             @foreach($kategori as $item)
18                                 <option value="{{ $item->kategori_id }}>{{ $item->kategori_nama }}</option>
19                             @endforeach
20                         </select>
21                         @error('kategori_id')
22                             <small class="form-text text-danger">{{ $message }}</small>
23                         @enderror
24                     </div>
25                 </div>
26                 <div class="form-group row">
27                     <label class="col-1 control-label col-form-label">Kode Barang</label>
28                     <div class="col-11">
29                         <input type="text" class="form-control" id="barang_kode" name="barang_kode" value="{{ old('barang_kode') }}" required>
30                         @error('barang_kode')
31                             <small class="form-text text-danger">{{ $message }}</small>
32                         @enderror
33                     </div>
34                 </div>
35                 <div class="form-group row">
36                     <label class="col-1 control-label col-form-label">Nama Barang</label>
37                     <div class="col-11">
38                         <input type="text" class="form-control" id="barang_nama" name="barang_nama" value="{{ old('barang_nama') }}" required>
39                         @error('barang_nama')
40                             <small class="form-text text-danger">{{ $message }}</small>
41                         @enderror
42                     </div>
43                 </div>
44                 <div class="form-group row">
45                     <label class="col-1 control-label col-form-label">Harga Beli</label>
46                     <div class="col-11">
47                         <input type="number" class="form-control" id="harga_beli" name="harga_beli" value="{{ old('harga_beli') }}" required>
48                         @error('harga_beli')
49                             <small class="form-text text-danger">{{ $message }}</small>
50                         @enderror
51                     </div>
52                 </div>
53                 <div class="form-group row">
54                     <label class="col-1 control-label col-form-label">Harga Jual</label>
55                     <div class="col-11">
56                         <input type="number" class="form-control" id="harga_jual" name="harga_jual" value="{{ old('harga_jual') }}" required>
57                         @error('harga_jual')
58                             <small class="form-text text-danger">{{ $message }}</small>
59                         @enderror
60                     </div>
61                 </div>
62                 <div class="form-group row">
63                     <label class="col-1 control-label col-form-label"></label>
64                     <div class="col-11">
65                         <button type="submit" class="btn btn-primary btn-sm">Simpan</button>
66                         <a class="btn btn-sm btn-default ml-1" href="{{ url('barang') }}>Kembali</a>
67                     </div>
68                 </div>
69             </form>
70         </div>
71     @endsection
72
73     @push('css')
74     @endpush
75
76     @push('js')
77     @endpush
78 @endpush
```



Edit.blade.php

```
1 @extends('layouts.template')
2
3 @section('content')
4 <div class="card card-outline card-primary">
5   <div class="card-header">
6     <h3 class="card-title">{{ $page->title }}
```

Show.blade.php

```
1 @extends('layouts.template')
2
3 @section('content')
4 <div class="card card-outline card-primary">
5   <div class="card-header">
6     <h3 class="card-title">{{ $page->title }}
```



- BarangController

```
public function index()
{
    $breadcrumb = (object)[
        'title' => 'Daftar Barang',
        'list' => ['Home', 'Barang']
    ];

    $page = (object)[
        'title' => 'Daftar barang yang terdaftar dalam sistem'
    ];

    $activeMenu = 'barang';
    $kategori = KategoriModel::all(); // Ambil data kategori untuk filter

    return view('barang.index', ['breadcrumb' => $breadcrumb, 'page' => $page, 'kategori' => $kategori, 'activeMenu' => $activeMenu]);
}

public function list(Request $request)
{
    $barang = BarangModel::select('barang_id', 'kategori_id', 'barang_kode', 'barang_nama', 'harga_beli', 'harga_jual')
        ->with(['kategori']);

    if ($request->kategori_id) {
        $barang->where('kategori_id', $request->kategori_id);
    }

    return DataTables::of($barang)
        ->addIndexColumn()
        ->addColumn('aksi', function ($barang) {
            $tn = '<a href="'.url('/barang/'.$barang->barang_id).'" class="btn btn-info btn-sm">Detail</a> ';
            $tn .= '<a href="'.url('/barang/'.$barang->barang_id . '/edit').'" class="btn btn-warning btn-sm">Edit</a> ';
            $tn .= '<form class="d-inline-block" method="POST" action="'.url('/barang/'.$barang->barang_id).'">';
            $tn .= '<button type="submit" class="btn btn-danger btn-sm" onclick="return confirm(\\'Apakah Anda yakin menghapus data ini?\')";>Hapus</button></form>';
            return $tn;
        })
        ->rawColumns(['aksi'])
        ->make(true);
}

public function create()
{
    $breadcrumb = (object)[
        'title' => 'Tambah Barang',
        'list' => ['Home', 'Barang', 'Tambah']
    ];

    $page = (object)[
        'title' => 'Tambah barang baru'
    ];

    $kategori = KategoriModel::all(); // Ambil data kategori untuk dropdown
    $activeMenu = 'barang';

    return view('barang.create', ['breadcrumb' => $breadcrumb, 'page' => $page, 'kategori' => $kategori, 'activeMenu' => $activeMenu]);
}

public function store(Request $request)
{
    $request->validate([
        'kategori_id' => 'required|integer',
        'barang_kode' => 'required|string|unique:m_barang,barang_kode',
        'barang_nama' => 'required|string|max:100',
        'harga_beli' => 'required|numeric|min:0',
        'harga_jual' => 'required|numeric|min:0',
    ]);

    BarangModel::create([
        'kategori_id' => $request->kategori_id,
        'barang_kode' => $request->barang_kode,
        'barang_nama' => $request->barang_nama,
        'harga_beli' => $request->harga_beli,
        'harga_jual' => $request->harga_jual,
    ]);

    return redirect('/barang')->with('success', 'Data barang berhasil disimpan');
}

public function show(string $id)
{
    $barang = BarangModel::with(['kategori'])->find($id);

    $breadcrumb = (object)[
        'title' => 'Detail Barang',
        'list' => ['Home', 'Barang', 'Detail']
    ];

    $page = (object)[
        'title' => 'Detail barang'
    ];

    $activeMenu = 'barang';

    return view('barang.show', ['breadcrumb' => $breadcrumb, 'page' => $page, 'barang' => $barang, 'activeMenu' => $activeMenu]);
}

public function edit(string $id)
{
    $barang = BarangModel::find($id);

    $breadcrumb = (object)[
        'title' => 'Edit Barang',
        'list' => ['Home', 'Barang', 'Edit']
    ];

    $page = (object)[
        'title' => 'Edit barang'
    ];

    $kategori = KategoriModel::all(); // Ambil data kategori untuk dropdown
    $activeMenu = 'barang';

    return view('barang.edit', ['breadcrumb' => $breadcrumb, 'page' => $page, 'barang' => $barang, 'kategori' => $kategori, 'activeMenu' => $activeMenu]);
}
```



```
public function update(Request $request, string $id)
{
    $request->validate([
        'kategori_id' => 'required|integer',
        'barang_kode' => 'required|string|unique:m_barang,barang.kode','.$id.',barang_id',
        'barang_nama' => 'required|string|max:100',
        'harga_beli' => 'required|numeric|min:0',
        'harga_jual' => 'required|numeric|min:0',
    ]);

    BarangModel::find($id)->update([
        'kategori_id' => $request->kategori_id,
        'barang_kode' => $request->barang_kode,
        'barang_nama' => $request->barang_nama,
        'harga_beli' => $request->harga_beli,
        'harga_jual' => $request->harga_jual,
    ]);

    return redirect('/barang')->with('success', 'Data barang berhasil diubah');
}

public function destroy(string $id)
{
    $check = BarangModel::find($id);
    if (!$check) {
        return redirect('/barang')->with('error', 'Data barang tidak ditemukan');
    }

    try {
        BarangModel::destroy($id);

        return redirect('/barang')->with('success', 'Data barang berhasil dihapus');
    } catch (\Illuminate\Database\QueryException $e) {
        return redirect('/barang')->with('error', 'Data barang gagal dihapus karena masih terdapat tabel lain yang terkait dengan data ini');
    }
}
```

- Route

```
Route::group(['prefix' => 'barang'], function () {
    Route::get('/', [BarangController::class, 'index']);
    Route::post('/list', [BarangController::class, 'list']);
    Route::get('/create', [BarangController::class, 'create']);
    Route::post('/', [BarangController::class, 'store']);
    Route::get('/{id}', [BarangController::class, 'show']);
    Route::get('/{id}/edit', [BarangController::class, 'edit']);
    Route::put('/{id}', [BarangController::class, 'update']);
    Route::delete('/{id}', [BarangController::class, 'destroy']);
});|
```



localhost/PWL_2023/Jobsheet05.2/jobsheet5.2/public/barang

PWL - Starter Code

Daftar Barang

Daftar barang yang terdaftar dalam sistem.

Filter: - Semua

Kategori Barang

Show 10 entries Search:

| ID | Kode Barang | Nama Barang | Kategori | Harga Beli | Harga Jual | Aksi |
|----|-------------|-----------------|-------------|------------|------------|---|
| 1 | BRG002 | Mie Goreng | Makanan | 9000 | 14000 | Detail Edit Hapus |
| 2 | BRG003 | Ayam Bakar | Makanan | 12000 | 18000 | Detail Edit Hapus |
| 3 | BRG004 | Kaos Polos | Pakaian | 30000 | 45000 | Detail Edit Hapus |
| 4 | BRG005 | Celana Jeans | Pakaian | 70000 | 100000 | Detail Edit Hapus |
| 5 | BRG006 | Keripik Kentang | Snack | 8000 | 12000 | Detail Edit Hapus |
| 6 | BRG007 | Biskuit Coklat | Snack | 5000 | 9000 | Detail Edit Hapus |
| 7 | BRG008 | Apel Fuji | Buah-buahan | 15000 | 20000 | Detail Edit Hapus |
| 8 | BRG009 | Jeruk Manis | Buah-buahan | 10000 | 15000 | Detail Edit Hapus |
| 9 | BRG010 | Wortel Segar | Sayuran | 7000 | 10000 | Detail Edit Hapus |
| 10 | BRG011 | Bayam Hijau | Sayuran | 5000 | 8000 | Detail Edit Hapus |

Tambah Data

Tambah Barang

Tambah barang baru

| | |
|-------------|------------|
| Kategori | Makanan |
| Kode Barang | BRG0012 |
| Nama Barang | Bubur Ayam |
| Harga Beli | 10000 |
| Harga Jual | 12000 |

[Simpan](#) [Kembali](#)

Detail Data

Detail Barang

Detail barang

| | |
|-------------|------------|
| ID | 12 |
| Kategori | Makanan |
| Kode Barang | BRG012 |
| Nama Barang | Bubur Ayam |
| Harga Beli | 10000 |
| Harga Jual | 12000 |

[Kembali](#)

Edit Data

Edit Barang

Edit barang

| | |
|-------------|------------|
| Kategori | Makanan |
| Kode Barang | BRG012 |
| Nama Barang | Bubur Ayam |
| Harga Beli | 10000 |
| Harga Jual | 15000 |

[Simpan](#) [Kembali](#)



Delete Data

Daftar Barang

Home / Barang

Tambah

Data barang berhasil dihapus

Filter: - Semua

Kategori Barang

Show 10 entries

Search:

| ID | Kode Barang | Nama Barang | Kategori | Harga Beli | Harga Jual | Aksi |
|----|-------------|-----------------|-------------|------------|------------|-------------------------|
| 1 | BRG002 | Mie Goreng | Makanan | 9000 | 14000 | [Detail] [Edit] [Hapus] |
| 2 | BRG003 | Ayam Bakar | Makanan | 12000 | 18000 | [Detail] [Edit] [Hapus] |
| 3 | BRG004 | Kaos Polos | Pakaian | 30000 | 45000 | [Detail] [Edit] [Hapus] |
| 4 | BRG005 | Celana Jeans | Pakaian | 70000 | 100000 | [Detail] [Edit] [Hapus] |
| 5 | BRG006 | Keripik Kentang | Snack | 8000 | 12000 | [Detail] [Edit] [Hapus] |
| 6 | BRG007 | Biskuit Coklat | Snack | 5000 | 9000 | [Detail] [Edit] [Hapus] |
| 7 | BRG008 | Apel Fuji | Buah-buahan | 15000 | 20000 | [Detail] [Edit] [Hapus] |
| 8 | BRG009 | Jeruk Manis | Buah-buahan | 10000 | 15000 | [Detail] [Edit] [Hapus] |
| 9 | BRG010 | Wortel Segar | Sayuran | 7000 | 10000 | [Detail] [Edit] [Hapus] |

*** Sekian, dan selamat belajar ***