



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 --}}
```



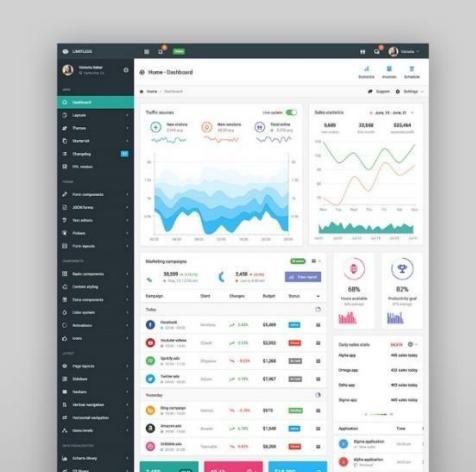
B. Web Template

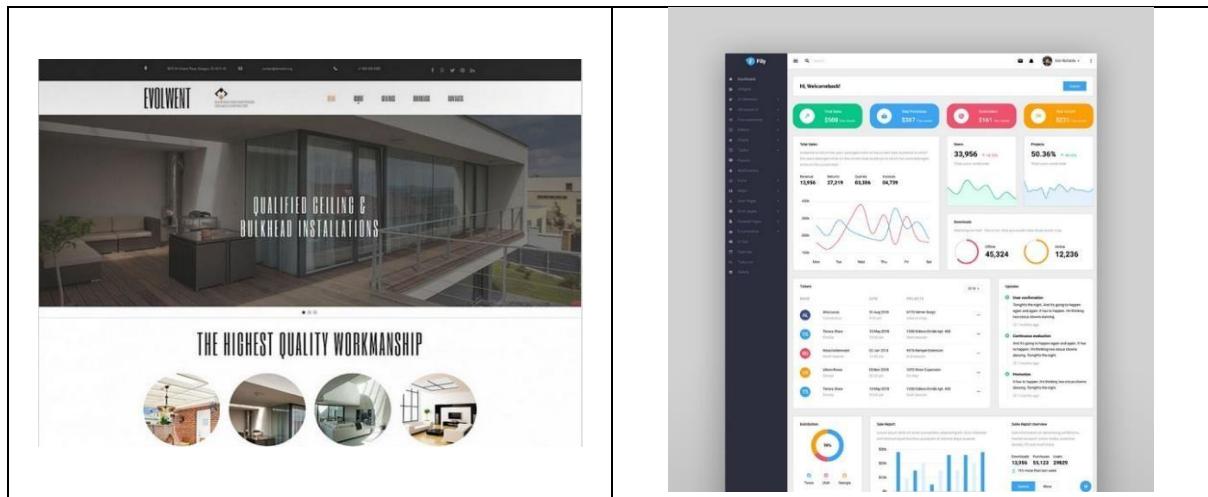
Merupakan suatu file yang sudah memiliki desain tertentu sehingga, kita tinggal memodifikasi bagian-bagian tertentu saja untuk mendapatkan hasil yang sudah bagus. Dengan menggunakan template yang sesuai dengan kebutuhan, kita bisa mengurangi waktu pembuatan suatu website secara drastis.

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

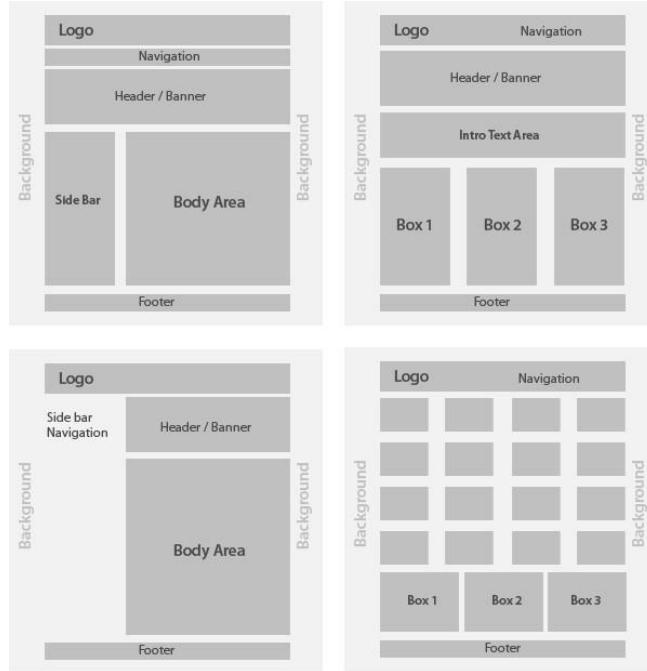
Frontend Template	Backend Template
	



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

The screenshot shows the GitHub release page for AdminLTE v3.2.0. At the top, there's a navigation bar with 'Releases' and 'Tags'. Below it, the release date is Feb 8, 2022, and the author is REJack. There are two download links: 'Source code (zip)' and 'Source code (tar.gz)'. A red checkmark is placed over the 'Source code (zip)' link. Below the links, there are social sharing icons and a reaction count of 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**

The screenshot shows the VS Code Explorer sidebar. The 'PWL_POS' folder is expanded, showing its contents: app, bootstrap, config, database, public, and adminlte. The 'adminlte' folder is selected, indicated by a blue selection bar and a red checkmark placed over it.

3. Selanjutnya kita buka di browser dengan alamat http://localhost/PWL_POS/public/adminlte maka akan muncul tampilan seperti berikut



The screenshot shows two side-by-side instances of the AdminLTE 3 dashboard. Both instances have a similar layout with a sidebar on the left containing navigation links like 'Dashboard', 'Widgets', 'Layout Options', 'Charts', 'UI Elements', 'Forms', and 'Tables'. The main area displays four cards: 'New Orders' (150), 'Bounce Rate' (53%), 'User Registrations' (44), and 'Unique Visitors' (65). Below these cards are two charts: a line chart for 'Sales' showing monthly trends from January to July, and a map of the United States labeled 'Visitors'. The bottom section of the dashboard includes a 'Direct Chat' window showing messages between Alexander Pierce and Sarah Bullock, and a 'Sales Graph' window.

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

The screenshot shows the 'Extras' section of the AdminLTE 3 sidebar, specifically the 'Blank Page' configuration. A red number '1' is placed over the 'Extras' menu item, and a red number '2' is placed over the 'Blank Page' link. The main panel displays a 'Blank Page' template with sections for 'Title' and 'Footer', both of which contain placeholder text: 'Start creating your amazing application!' and 'Footer' respectively.



localhost/PWL_2025/Minggu5.2/public/adminlte/pages/examples/blank.html

AdminLTE 3

Alexander Pierce

Search

Dashboard Widgets New

Layout Options

Charts

UI Elements

Forms

Tables

EXAMPLES

Calendar 2

Gallery

Kanban Board

Mailbox

Pages

Extras

Login & Register v1

Blank Page

Title

Start creating your amazing application!

Footer

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

1 - Header / Navbar

2 - Sidebar / Navigasi

Blank Page

3 - Breadcrumb

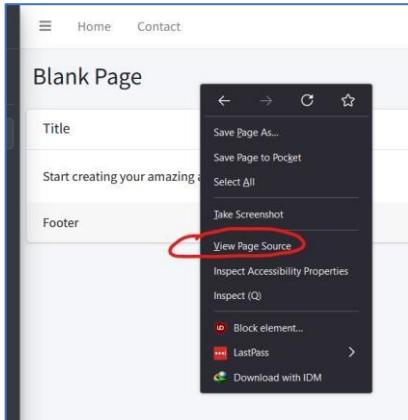
4 - Content

5 - Footer

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

Version 3.2.0

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**)

```
EXPLORER ... template.blade.php ...
PWL_POS ...
  app
  bootstrap
  config
  database
  public
  resources ...
    css
    js
  views ...
    layouts ...
      template.blade.php
      kategori.blade.php
      level.blade.php
      user.blade.php
      welcome.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=".../plugins/fontawesome-free/css/all.min.css">
12   <!-- Theme style -->
13   <link rel="stylesheet" href=".../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=".../plugins/fontawesome-free/css/all.min.css">
12   <!-- Theme style -->
13   <link rel="stylesheet" href=".../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>
  <!-- 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

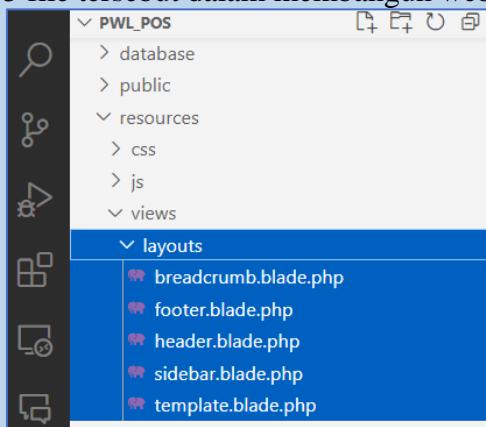


Berikut tampilan pada dasboard

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.





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).'">' .
                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']) // memberitahu bahwa kolom aksi adalah html
        ->make(true);
}
```

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

Berikut tampilan pada data user



ID	Username	Nama	Level Pengguna	Aksi
1	admin	Administrator	Administrasi	[Detail] [Edit] [Hapus]
2	manager	Manager	Manager	[Detail] [Edit] [Hapus]
3	staff	Staff/Kasir	Staff/Kasir	[Detail] [Edit] [Hapus]
4	customer-1	Pelanggan Pertama	Staff/Kasir	[Detail] [Edit] [Hapus]
5	manager33	Manager Tiga Tiga	Manager	[Detail] [Edit] [Hapus]
6	manager56	Manager55	Manager	[Detail] [Edit] [Hapus]
7	manager12	Manager11	Manager	[Detail] [Edit] [Hapus]
8	manager11	Manager111	Manager	[Detail] [Edit] [Hapus]
9	manager_dua	Manager 2	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')


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



<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>


```



```
<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..!!!

Berikut tampilan jika menekan tombol tambah

```
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
});
```



KEMENTERIAN PENDIDIKAN, KEBUDAYAAN, RISET, DAN TEKNOLOGI
POLITEKNIK NEGERI MALANG
JURUSAN TEKNOLOGI INFORMASI
Jl. Soekarno Hatta No. 9, Jatimulyo, Lowokwaru, Malang 65141
Telp. (0341) 404424 – 404425, Fax (0341) 404420
<http://www.polinema.ac.id>

The screenshot shows a web-based application for managing users. On the left, there's a sidebar with various menu items: Dashboard, Data Pengguna, Level User, **Data User** (which is currently selected), Data Barang, Kategori Barang, Data Barang, Data Transaksi, Stok Barang, and Transaksi Penjualan. The main content area is titled 'Tambah User' and contains a form for adding a new user. The form includes fields for 'Level' (with a dropdown placeholder '- Pilih Level -'), 'Username', 'Nama', and 'Password'. At the bottom of the form are two buttons: 'Simpan' (Save) and 'Kembali' (Back). The URL in the browser bar is 'localhost/PWL_2025/Minggu5.2/jobsheet5.2/public/user/create'.

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



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]);
}
```

ID	Level
1	Administrasi

Detail user

Username: admin

Nama: Administrator

Password: *****

Kembali

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">
        <h5><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>


@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!!!

Error karena user_id 100 tidak ada pada database

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('/store', [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>
```



KEMENTERIAN PENDIDIKAN, KEBUDAYAAN, RISET, DAN TEKNOLOGI
POLITEKNIK NEGERI MALANG
JURUSAN TEKNOLOGI INFORMASI

```
// 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');
    }
}

<a class="btn btn-sm btn-default m1-1" 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!

Mengedit data

Edit user

Level Manager

Username manager_duaa

Nama Manager 2

Password

Abakan (pengan dilisi jika tidak ingin mengganti password user.)

Setelah data diedit

Detail User

Detail user	
ID	24
Level	Manager
Username	manager_duaa
Nama	Manager 2
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`



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>
```

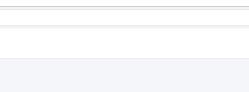
24. Kemudian jalankan browser untuk menghapus salah satu data user. Amati dan laporan!
Data dengan nama Bayu Triwibowo akan dihapus



The screenshot shows a Laravel application's user management interface. On the left is a sidebar with navigation links: Dashboard, Data Pengguna, Level User, and Data User (which is currently selected). The main content area is titled "Daftar User" and displays a table of users. A modal window is overlaid on the page, asking "Apakah Anda yakin menghapus data ini?" (Are you sure you want to delete this data?). The modal has two buttons: "OK" and "Cancel".

ID	Username	Nama	Level Pengguna	Aksi
1	admin	Administrator	Administrasi	[Detail] [Edit] [Hapus]
2	manager	Manager	Manager	[Detail] [Edit] [Hapus]
3	staff	Staff/Kasir	Staff/Kasir	[Detail] [Edit] [Hapus]
4	customer-1	Pelanggan Pertama	Staff/Kasir	[Detail] [Edit] [Hapus]
5	manager33	Manager Tiga Tiga	Manager	[Detail] [Edit] [Hapus]
6	manager56	Manager55	Manager	[Detail] [Edit] [Hapus]
7	manager12	Manager11	Manager	[Detail] [Edit] [Hapus]
8	manager11	Manager111	Manager	[Detail] [Edit] [Hapus]
9	Manager99	Bayu Triwibowo	Manager	[Detail] [Edit] [Hapus]

Data terhapus



This screenshot shows the same Laravel application interface after the deletion. The modal confirmation window is no longer present. The user "Bayu Triwibowo" is missing from the list of users in the table.

ID	Username	Nama	Level Pengguna	Aksi
1	admin	Administrator	Administrasi	[Detail] [Edit] [Hapus]
2	manager	Manager	Manager	[Detail] [Edit] [Hapus]
3	staff	Staff/Kasir	Staff/Kasir	[Detail] [Edit] [Hapus]
4	customer-1	Pelanggan Pertama	Staff/Kasir	[Detail] [Edit] [Hapus]
5	manager33	Manager Tiga Tiga	Manager	[Detail] [Edit] [Hapus]
6	manager56	Manager55	Manager	[Detail] [Edit] [Hapus]
7	manager12	Manager11	Manager	[Detail] [Edit] [Hapus]
8	manager11	Manager111	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 and 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 a search bar and a 'Daftar User' title. Below it is a table titled 'Daftar user yang terdaftar dalam sistem'. The table has 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 in the top right corner of the table's header. The table displays 6 entries. At the bottom, there are navigation 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="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>





@endsection


```

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

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]

Berikut tampilan dengan menambahkan filtering dan search

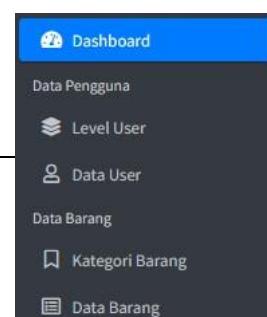
ID	Username	Nama	Level Pengguna	Aksi
1	admin	Administrator	Administrator	[Detail] [Edit] [Hapus]
2	manager	Manager	Manager	[Detail] [Edit] [Hapus]
3	staff	Staff/Kasir	Staff/Kasir	[Detail] [Edit] [Hapus]
4	customer-1	Pelanggan Pertama	Staff/Kasir	[Detail] [Edit] [Hapus]
5	manager33	Manager Tiga Tiga	Manager	[Detail] [Edit] [Hapus]
6	manager56	Manager55	Manager	[Detail] [Edit] [Hapus]
7	manager12	Manager11	Manager	[Detail] [Edit] [Hapus]
8	manager11	Manager111	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
- View





Index.blade

```
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

```
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

```
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                 <a href="{{ url('level') }}" class="btn btn-sm btn-default mt-2">Kembali</a>
16             @else
17                 <form method="POST" action="{{ url('/level/'.$level->level_id) }}" class="form-horizontal">
18                     @csrf
19                     @if (method()=='PUT') <!-- Method PUT untuk proses edit -->
20                         <div class="form-group row">
21                             <label class="col-1 control-label col-form-label">Kode Level</label>
22                             <div class="col-11">
23                                 <input type="text" class="form-control" id="level_kode" name="level_kode" value="{{ old('level_kode', $level->level_kode) }}" required>
24                                 @error('level_kode')
25                                     <small class="form-text text-danger">{{ $message }}</small>
26                                 @enderror
27                             </div>
28                         </div>
29                         <div class="form-group row">
30                             <label class="col-1 control-label col-form-label">Nama Level</label>
31                             <div class="col-11">
32                                 <input type="text" class="form-control" id="level_nama" name="level_nama" value="{{ old('level_nama', $level->level_nama) }}" required>
33                                 @error('level_nama')
34                                     <small class="form-text text-danger">{{ $message }}</small>
35                                 @enderror
36                             </div>
37                         </div>
38                         <div class="form-group row">
39                             <label class="col-1 control-label col-form-label"></label>
40                             <div class="col-11">
41                                 <button type="submit" class="btn btn-primary btn-sm">Simpan</button>
42                                 <a class="btn btn-sm btn-default ml-1" href="{{ url('level') }}">Kembali</a>
43                             </div>
44                         </div>
45                     </form>
46                 @endempty
47             </div>
48         @endsection
49         @push('css')
50             @endpush
51         @push('js')
52             @endpush
53         @endpush
54     
```

Show.blade

```
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                     <thead>
23                         <tr>
24                             <th>Kode Level</th>
25                             <td>{{ $level->level_kode }}</td>
26                         </tr>
27                     <tbody>
28                         <tr>
29                             <th>Nama Level</th>
30                             <td>{{ $level->level_nama }}</td>
31                         </tr>
32                     </tbody>
33                 </table>
34             @endempty
35             <a href="{{ url('level') }}" class="btn btn-sm btn-default mt-2">Kembali</a>
36         </div>
37     @endsection
38     @push('css')
39     @endpush
40     @push('js')
41     @endpush
42 
```



➤ Controller

```
26
27     public function index(){
28         $breadcrumb = (object)[
29             'title' => 'Daftar Level',
30             'list' => ['Home', 'Level']
31         ];
32
33         $page = (object)[
34             'title' => 'Daftar Level yang terdaftar dalam sistem'
35         ];
36
37         $activeMenu = 'level';
38         return view('level.index', ['breadcrumb' => $breadcrumb, 'page' => $page, 'activeMenu' => $activeMenu]);
39     }
40
41     public function list()
42     {
43         $levels = LevelModel::select('level_id', 'level_kode', 'level_nama');
44
45         return Datatables::of($levels)
46             ->addIndexColumn()
47             ->addColumn('aksi', function ($level) {
48                 $url = 'level/detail/' . $level->level_id;
49                 $edit = 'level/edit/' . $level->level_id;
50                 $del = 'form class="d-inline-block" method="POST" actions="" url="/level/' . $level->level_id . '"';
51                 $del .= '<button type="submit" class="btn btn-danger btn-sm" onclick="return confirm(`Apakah Anda yakin menghapus data ini?`);">' . Hapus . '</button></form>';
52                 $del .= '<button type="button" class="btn btn-warning btn-sm" onclick="location.href=' . $url . '"></button>';
53             })
54             ->rawColumns(['aksi'])
55             ->make(true);
56     }
57
58     public function create()
59     {
60         $breadcrumb = (object)[
61             'title' => 'Tambah Level',
62             'list' => ['Home', 'Level', 'Tambah']
63         ];
64
65         $page = (object)[
66             'title' => 'Tambah level baru'
67         ];
68
69         $activeMenu = 'level';
70         return view('level.create', ['breadcrumb' => $breadcrumb, 'page' => $page, 'activeMenu' => $activeMenu]);
71     }
72
73     public function store(Request $request)
74     {
75         $request->validate([
76             'level_kode' => 'required|string|unique:m_level,level_kode',
77             'level_nama' => 'required|string|max:100',
78         ]);
79
80         LevelModel::create([
81             'level_kode' => $request->level_kode,
82             'level_nama' => $request->level_nama,
83         ]);
84
85         return redirect('/level')->with('success', 'Data level berhasil disimpan');
86     }
87
88     public function show(string $id)
89     {
90         $level = LevelModel::find($id);
91
92         $breadcrumb = (object)[
93             'title' => 'Detail Level',
94             'list' => ['Home', 'Level', 'Detail']
95         ];
96
97         $page = (object)[
98             'title' => 'Detail Level'
99         ];
100
101         $activeMenu = 'level';
102
103         return view('level.show', ['breadcrumb' => $breadcrumb, 'page' => $page, 'level' => $level, 'activeMenu' => $activeMenu]);
104     }
105
106     public function edit(string $id)
107     {
108         $level = LevelModel::find($id);
109
110         $breadcrumb = (object) [
111             'title' => 'Edit Level',
112             'list' => ['Home', 'Level', 'Edit']
113         ];
114
115         $page = (object) [
116             'title' => 'Edit Level'
117         ];
118
119         $activeMenu = 'level';
120         return view('level.edit', ['breadcrumb' => $breadcrumb, 'page' => $page, 'level' => $level, 'activeMenu' => $activeMenu]);
121     }
122
123     public function update(Request $request, string $id)
124     {
125         $request->validate([
126             'level_kode' => 'required|string|unique:m_level,level_kode,' . $id . ',level_id',
127             'level_nama' => 'required|string|max:100',
128         ]);
129
130         LevelModel::find($id)->update([
131             'level_kode' => $request->level_kode,
132             'level_nama' => $request->level_nama,
133         ]);
134
135         return redirect('/level')->with('success', 'Data level berhasil diubah');
136     }
137
138     public function destroy(string $id)
139     {
140         $check = LevelModel::find($id);
141         if (!$check) {
142             return redirect('/level')->with('error', 'Data level tidak ditemukan');
143         }
144
145         try {
146             LevelModel::destroy($id);
147
148             return redirect('/level')->with('success', 'Data level berhasil dihapus');
149         } catch (\Laravel\midtone\database\QueryException $e) {
150
151             return redirect('/level')->with('error', 'Data level gagal dihapus karena masih terdapat tabel lain yang terkait dengan data ini');
152         }
153     }
154 }
155 }
```

➤ Routes (web.php)

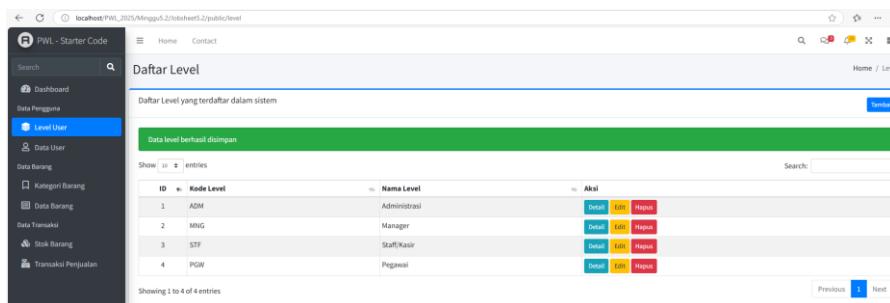
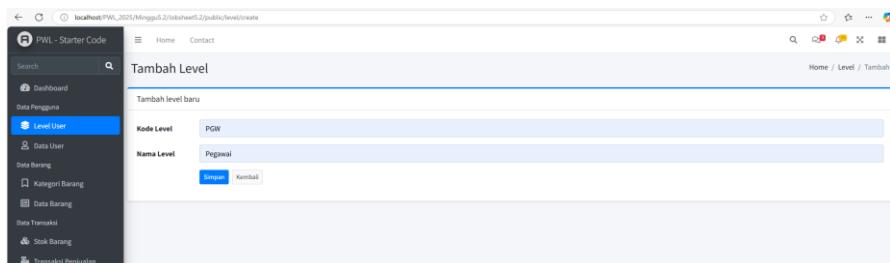
a



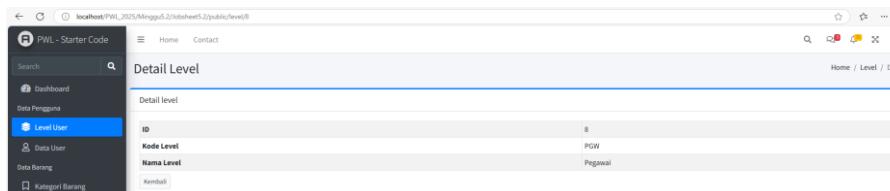
```
8 Route::group(['prefix' => 'level'], function () {
9     Route::get('/', [LevelController::class, 'index']);
10    Route::post('/list', [LevelController::class, 'list']);
11    Route::get('/create', [LevelController::class, 'create']);
12    Route::post('/', [LevelController::class, 'store']);
13    Route::get('/{id}', [LevelController::class, 'show']);
14    Route::get('/{id}/edit', [LevelController::class, 'edit']);
15    Route::put('/{id}', [LevelController::class, 'update']);
16    Route::delete('/{id}', [LevelController::class, 'destroy']);
17});
```

Berikut tampilan pada web

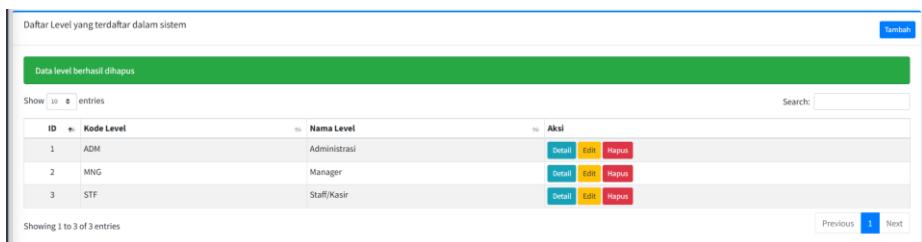
○ Tambah data



○ Form detail



○ Delete



✓ Tabel m_kategori

➤ View

Index.blade



```
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('kategori/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_kategori">
19       <thead>
20         <tr>
21           <th>ID</th>
22           <th>Kode Kategori</th>
23           <th>Nama Kategori</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_kategori").DataTable
39       ({
40         // serverSide: true, jika ingin menggunakan server side processing
41         serverSide: true,
42         ajax: {
43           url: "{{ url('kategori/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           {
55             data: "kategori_kode",
56             className: "",
57             orderable: true,
58             searchable: true
59           },
60           {
61             data: "kategori_nama",
62             className: "",
63             orderable: true,
64             searchable: true
65           },
66           {
67             data: "aksi",
68             className: "",
69             orderable: false,
70             searchable: false
71           }
72         });
73     </script>
74 @endpush
```

Create.blade



```
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   </div>
40   @endsection
41
42 @push('css')
43 @endpush
44
45 @push('js')
46 @endpush
```

Edit.blade

```
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           <i class="fas fa-ban"></i> Kesalahan!
13           Data yang anda cari tidak ditemukan.
14         </div>
15         <a href="{{ url('kategori') }}" class="btn btn-sm btn-default mt-2">Kembali</a>
16       @else
17         <form method="POST" action="{{ url('/kategori/'.$kategori->kategori_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">Kode Kategori</label>
22             <div class="col-11">
23               <input type="text" class="form-control" id="kategori_kode" name="kategori.kode" value="{{ old('kategori.kode', $kategori->kategori.kode) }}" required>
24               @error('kategori.kode')
25                 <small class="form-text text-danger">{{ $message }}</small>
26               @enderror
27             </div>
28           </div>
29           <div class="form-group row">
30             <label class="col-1 control-label col-form-label">Nama Kategori</label>
31             <div class="col-11">
32               <input type="text" class="form-control" id="kategori_nama" name="kategori.nama" value="{{ old('kategori.nama', $kategori->kategori.nama) }}" required>
33               @error('kategori.nama')
34                 <small class="form-text text-danger">{{ $message }}</small>
35               @enderror
36             </div>
37           </div>
38           <div class="form-group row">
39             <label class="col-1 control-label col-form-label"></label>
40             <div class="col-11">
41               <button type="submit" class="btn btn-primary btn-sm">Simpan</button>
42               <a class="btn btn-sm btn-default ml-1" href="{{ url('kategori') }}">Kembali</a>
43             </div>
44           </div>
45         </form>
46       @endempty
47     </div>
48   </div>
49   @endsection
50
51 @push('css')
52 @endpush
53
54 @push('js')
55 @endpush
```

Show.blade



```
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
```

➤ Controller



KEMENTERIAN PENDIDIKAN, KEBUDAYAAN, RISET, DAN TEKNOLOGI
POLITEKNIK NEGERI MALANG
JURUSAN TEKNOLOGI INFORMASI

Jl. Soekarno Hatta No. 9, Jatimulyo, Lowokwaru, Malang 65141
Telp. (0341) 404424 – 404425, Fax (0341) 404420
<http://www.polinema.ac.id>

```

1  public function index()
2  {
3      $breadcrumbs = ($obj)[1];
4      $title = "Definisi Kategori yang tersedia dalam sistem";
5      $list = [
6          [
7              "name" => "index",
8              "title" => "Definisi Kategori yang tersedia dalam sistem"
9          ],
10         [
11             "setelahmenu" => "kategori"
12         ]
13     ];
14
15     return view('kategori/index', [
16         'breadcrumbs' => $breadcrumbs,
17         'msg' => $msg,
18         'aktivitas' => 'definisi',
19         'list' => $list
20     ]);
21 }
22
23 public function list()
24 {
25     $categoryModel = $this->select('kategori_id', 'kategori_kode', 'kategori_nama');
26
27     return $categoryModel;
28 }
29
30 public function detail($id)
31 {
32     $categoryModel = $this->select('kategori_id', 'kategori_kode', 'kategori_nama');
33
34     $category = $categoryModel->where('kategori_id', $id);
35
36     $msg = [
37         [
38             "name" => "index",
39             "title" => "Definisi Kategori yang tersedia dalam sistem"
40         ],
41         [
42             "setelahmenu" => "kategori"
43         ]
44     ];
45
46     return view('kategori/detail', [
47         'category' => $category,
48         'msg' => $msg,
49         'aktivitas' => 'detail'
50     ]);
51 }
52
53 public function create()
54 {
55     $breadcrumbs = ($obj)[1];
56     $title = "Tambah Kategori";
57     $list = [
58         [
59             "name" => "index",
60             "title" => "Tambah Kategori"
61         ],
62         [
63             "setelahmenu" => "kategori"
64         ]
65     ];
66
67     $msg = [
68         [
69             "name" => "index",
70             "title" => "Tambah Kategori Baru"
71         ],
72         [
73             "setelahmenu" => "kategori"
74         ]
75     ];
76
77     $categoryModel = $this->insert([
78         'kategori_kode' => $request->get('kategori_kode'),
79         'kategori_nama' => $request->get('kategori_nama'),
80         'aktivitas' => 'create'
81     ]);
82
83     return redirect('/kategori')->with('success', "Data kategori berhasil disimpan");
84 }
85
86 public function store(Request $request)
87 {
88     $request->validate([
89         'kategori_kode' => 'required|string|unique:tbl_kategori,kategori_kode',
90         'kategori_nama' => 'required|string|max:100',
91         'aktivitas' => 'required'
92     ]);
93
94     $categoryModel = $this->create([
95         'kategori_kode' => $request->get('kategori_kode'),
96         'kategori_nama' => $request->get('kategori_nama'),
97         'aktivitas' => 'store'
98     ]);
99
100    return redirect('/kategori')->with('success', "Data kategori berhasil disimpan");
101 }
102
103 public function show(string $id)
104 {
105     $categoryModel = $this->select($id);
106
107     $breadcrumbs = ($obj)[1];
108
109     $category = $categoryModel->where('kategori_id', $id);
110
111     $msg = [
112         [
113             "name" => "index",
114             "title" => "Detail Kategori"
115         ],
116         [
117             "setelahmenu" => "kategori"
118         ]
119     ];
120
121     return view('kategori/show', [
122         'breadcrumbs' => $breadcrumbs,
123         'msg' => $msg,
124         'category' => $category,
125         'aktivitas' => 'show'
126     ]);
127 }
128
129 public function edit(string $id)
130 {
131     $categoryModel = $this->select($id);
132
133     $breadcrumbs = ($obj)[1];
134
135     $category = $categoryModel->where('kategori_id', $id);
136
137     $msg = [
138         [
139             "name" => "index",
140             "title" => "Edit Kategori"
141         ],
142         [
143             "setelahmenu" => "kategori"
144         ]
145     ];
146
147     return view('kategori/edit', [
148         'breadcrumbs' => $breadcrumbs,
149         'msg' => $msg,
150         'category' => $category,
151         'aktivitas' => 'edit'
152     ]);
153 }
154
155 public function update(Request $request, string $id)
156 {
157     $request->validate([
158         'kategori_kode' => 'required|string|unique:tbl_kategori,kategori_kode,'.$request->get('kategori_id'),
159         'kategori_nama' => 'required|string|max:100',
160         'aktivitas' => 'update'
161     ]);
162
163     $categoryModel = $this->update($request->get('kategori_kode'), [
164         'kategori_kode' => $request->get('kategori_kode'),
165         'kategori_nama' => $request->get('kategori_nama'),
166         'aktivitas' => 'update'
167     ]);
168
169     return redirect('/kategori')->with('success', "Data kategori berhasil diubah");
170 }
171
172 public function destroy(string $id)
173 {
174     $categoryModel = $this->select($id);
175
176     if ($categoryModel->count() == 0) {
177         return redirect('/kategori')->with('error', "Data kategori tidak ditemukan");
178     }
179
180     $categoryModel->destroy($id);
181
182     return redirect('/kategori')->with('success', "Data kategori berhasil dihapus");
183 }
184
185 public function destroyAll()
186 {
187     $categoryModel = $this->all();
188
189     if ($categoryModel->count() > 0) {
190         return redirect('/kategori')->with('error', "Data kategori gagal dihapus karena masih terdapat relasi yang terkait dengan data ini");
191     }
192 }

```

➤ Routes

```
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']);
});
```

Berikut tampilan pada web



localhost/PWL_2025/Minggu5.2/jobsheet5.2/public/kategori

PWL - Starter Code

Daftar Kategori

Daftar Kategori yang terdaftar dalam sistem

ID	Kode Kategori	Nama Kategori	Aksi
1	ELK	Elektronik	[Detail] [Edit] [Hapus]
2	PAK	Pakaian	[Detail] [Edit] [Hapus]
3	MAK	Makanan	[Detail] [Edit] [Hapus]
4	MIN	Minuman	[Detail] [Edit] [Hapus]
5	PRR	Peralatan Rumahh	[Detail] [Edit] [Hapus]
6	CML	Cemilan	[Detail] [Edit] [Hapus]
7	MNR	Minuman Ringan	[Detail] [Edit] [Hapus]

Showing 1 to 7 of 7 entries

Search:

Tambah

Tambah data

Tambah Kategori

Tambah kategori baru

Kode Kategori	PFM
Nama Kategori	Parfum

Simpan **Kembali**

Daftar Kategori yang terdaftar dalam sistem

Data kategori berhasil disimpan

ID	Kode Kategori	Nama Kategori	Aksi
1	ELK	Elektronik	[Detail] [Edit] [Hapus]
2	PAK	Pakaian	[Detail] [Edit] [Hapus]
3	MAK	Makanan	[Detail] [Edit] [Hapus]
4	MIN	Minuman	[Detail] [Edit] [Hapus]
5	PRR	Peralatan Rumahh	[Detail] [Edit] [Hapus]
6	CML	Cemilan	[Detail] [Edit] [Hapus]
7	MNR	Minuman Ringan	[Detail] [Edit] [Hapus]
8	PFM	Parfum	[Detail] [Edit] [Hapus]

Showing 1 to 8 of 8 entries

Search:

Tambah

Form detail

Detail kategori

ID	9
Kode Kategori	PFM
Nama Kategori	Parfum

Kembali

Hapus



localhost says

Apakah Anda yakin menghapus data ini?

OK

Cancel

Data kategori berhasil dihapus			
Show	10	entries	Search:
ID	Kode Kategori	Nama Kategori	Aksi
1	ELK	Elektronik	Detail Edit Hapus
2	PAK	Pakaian	Detail Edit Hapus
3	MAK	Makanan	Detail Edit Hapus
4	MIN	Minuman	Detail Edit Hapus
5	PRR	Peralatan Rumahh	Detail Edit Hapus
6	CML	Cemilan	Detail Edit Hapus
7	MNR	Minuman Ringan	Detail Edit Hapus

Showing 1 to 7 of 7 entries

Previous **1** Next

✓ Tabel m_supplier

➤ View

Index.blade



KEMENTERIAN PENDIDIKAN, KEBUDAYAAN, RISET, DAN TEKNOLOGI
POLITEKNIK NEGERI MALANG
JURUSAN TEKNOLOGI INFORMASI
Jl. Soekarno Hatta No. 9, Jatimulyo, Lowokwaru, Malang 65141
Telp. (0341) 404424 – 404425, Fax (0341) 404420
<http://www.polinema.ac.id>

```
1 #extends('layouts.template')
2
3 #section('content')
4     <div class="card card-outline card-primary">
5         <div class="card-header">
6             <h3>{{ $page->title }}</h3>
7             <div class="card-tools">
8                 <a class="btn btn-sm btn-primary" href="{{ url('stok/create') }}>Tambah</a>
9             </div>
10        </div>
11        <div class="card-body">
12            @if ($session['success'])
13                <div class="alert alert-success">{{ session('success') }}</div>
14            @endif
15            @if ($session['error'])
16                <div class="alert alert-danger">{{ session('error') }}</div>
17            @endif
18        </div>
19        <div class="row">
20            <div class="col-md-12">
21                <div class="form-group row">
22                    <label class="control-label col-form-label">Filter:</label>
23                    <div class="col-3">
24                        <select class="form-control" id="user_id" name="user_id">
25                            <option value="">> Semua</option>
26                            @foreach ($user as $item)
27                                <option value="{{ $item->user_id }}">{!! $item->nama !!}
28                            @endforeach
29                        </select>
30                        <small class="form-text text-muted">Nama User</small>
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                <tbody>
46                </tbody>
47            </table>
48        </div>
49    @endsection
50 @push('css')
51 @endpush
52
53 @push('js')
54     <script>
55         $(document).ready(function() {
56             var dataStok = $('#table_stok').DataTable({
57                 serverSide: true,
58                 ajax: {
59                     url: "{{ url('stok/list') }}",
60                     "datatype": "json",
61                     "type": "POST",
62                     "data": function (d) {
63                         d.user_id = $('#user_id').val(); // Filter berdasarkan user_id
64                         d_token = '{{ csrf_token() }}'; // Tambahkan CSRF token
65                     }
66                 },
67                 columns: [
68                     {
69                         data: "DT_RowIndex",
70                         className: "text-center",
71                         orderable: false,
72                         searchable: false
73                     },
74                     {
75                         data: "barang_barang_nama", // Mengambil nama barang dari relasi
76                         className: "",
77                         orderable: false,
78                         searchable: false
79                     },
80                     {
81                         data: "user_nama", // Mengambil nama user dari relasi
82                         className: "",
83                         orderable: false,
84                         searchable: false
85                     },
86                     {
87                         data: "stok_tanggal",
88                         className: "",
89                         orderable: true,
90                         searchable: true
91                     },
92                     {
93                         data: "stok_jumlah",
94                         className: "",
95                         orderable: true,
96                         searchable: false
97                     },
98                     {
99                         data: "aksi",
100                         className: "",
101                         orderable: false,
102                         searchable: false
103                     }
104                 ],
105             });
106             $('#user_id').on('change', function() {
107                 dataStok.ajax.reload();
108             });
109         });
110     </script>
111 @endpush
112
```

Create.blade



```
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             </div>
66         </div>
67     </div>
68     @endsection
69
70     @push('css')
71     @endpush
72
73     @push('js')
74     @endpush
```

Edit.blade



```
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">{{ $page->title }}</h3>
7          </div>
8          <div class="card-body">
9              @empty($stok)
10                  <div class="alert alert-danger alert-dismissible">
11                      <h5><i class="icon fas fa-ban"></i> Kesalahan!</h5>
12                      Data yang Anda cari tidak ditemukan.
13                  </div>
14                  <a href="{{ url('stok') }}" class="btn btn-sm btn-default mt-2">Kembali</a>
15
16  @else
17      <form method="POST" actions="{{ url('/stok/'.$stok->stok_id) }}" class="form-horizontal">
18          @csrf
19          (@if method_field('PUT') !) <!-- Method PUT untuk proses edit -->
20          <div class="form-group row">
21              <label class="col-1 control-label col-form-label">Barang:</label>
22              <div class="col-11">
23                  <select class="form-control" id="barang_id" name="barang_id" required>
24                      <option value="">> Pilih Barang </option>
25                      @foreach($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">{{ $message }}</small>
34                  @enderror
35              </div>
36          </div>
37          <div class="form-group row">
38              <label class="col-1 control-label col-form-label">User:</label>
39              <div class="col-11">
40                  <select class="form-control" id="user_id" name="user_id" required>
41                      <option value=""> Pilih User </option>
42                      @foreach($user as $item)
43                          <option value="{{ $item->user_id }}">
44                              @if($item->user_id == $stok->user_id) selected @endif
45                              {{ $item->nama }}
46                          </option>
47                      @endforeach
48                  </select>
49                  @error('user_id')
50                      <small class="form-text text-danger">{{ $message }}</small>
51                  @enderror
52              </div>
53          </div>
54          <div class="form-group row">
55              <label class="col-1 control-label col-form-label">Tanggal Stok:</label>
56              <div class="col-11">
57                  <input type="date" class="form-control" id="stok_tanggal" name="stok_tanggal" value="{{ old('stok_tanggal', $stok->stok_tanggal) }}" required>
58                  @error('stok_tanggal')
59                      <small class="form-text text-danger">{{ $message }}</small>
60                  @enderror
61              </div>
62          </div>
63          <div class="form-group row">
64              <label class="col-1 control-label col-form-label">Jumlah Stok:</label>
65              <div class="col-11">
66                  <input type="number" class="form-control" id="stok_jumlah" name="stok_jumlah" value="{{ old('stok_jumlah', $stok->stok_jumlah) }}" required>
67                  @error('stok_jumlah')
68                      <small class="form-text text-danger">{{ $message }}</small>
69                  @enderror
70              </div>
71          </div>
72          <div class="form-group row">
73              <label class="col-1 control-label col-form-label">Simpan</button>
76                  <a class="btn btn-sm btn-default ml-1" href="{{ url('stok') }}">Kembali</a>
77              </div>
78          </div>
79      </form>
80  @endifempty
81  </div>
82  </div>
83  @endsection
84
85  @push('css')
86  @endpush
87
88  @push('js')
89  @endpush
```

Show.blade



```
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">{{ $page->title }}</h3>
7              <div class="card-tools"></div>
8      </div>
9      <div class="$stok">
10         @empty($stok)
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>{{ $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
46
47  @push('js')
48  @endpush
```

➤ Controller





```
1  Route::group(['prefix' => 'stok'], function () {
2      Route::get('/', [StokController::class, 'index']);
3      Route::post('/list', [StokController::class, 'list']);
4      Route::get('/create', [StokController::class, 'create']);
5      Route::post('/', [StokController::class, 'store']);
6      Route::get('/{id}', [StokController::class, 'show']);
7      Route::get('/{id}/edit', [StokController::class, 'edit']);
8      Route::put('/{id}', [StokController::class, 'update']);
9      Route::delete('/{id}', [StokController::class, 'destroy']);
10 });
});
```

Berikut output yang dihasilkan

ID	Nama Barang	Nama User	Tanggal Stok	Jumlah Stok	Aksi
1	Laptop	Administrator	2025-03-04 18:26:17	30	Detail Edit Hapus
2	Smartphone	Administrator	2025-03-04 18:26:17	25	Detail Edit Hapus
3	Kaos	Manager	2025-03-04 18:26:17	20	Detail Edit Hapus
4	Jaket	Manager	2025-03-04 18:26:17	15	Detail Edit Hapus
5	Roti	Manager	2025-03-04 18:26:17	10	Detail Edit Hapus
6	Kue	Staff/Kasir	2025-03-04 18:26:17	50	Detail Edit Hapus
7	Teh Botol	Staff/Kasir	2025-03-04 18:26:17	40	Detail Edit Hapus
8	Kopi	Staff/Kasir	2025-03-04 18:26:17	35	Detail Edit Hapus
9	Panci	Administrator	2025-03-04 18:26:17	45	Detail Edit Hapus
10	Sapu	Administrator	2025-03-04 18:26:17	60	Detail Edit Hapus

Tambah data

Tambah stok baru	
Barang	Laptop
User	Manager55
Tanggal Stok	10/03/2025
Jumlah Stok	50

Detail data



Detail Stok

Home / Stok / Detail

Detail stok	
ID	1
Nama Barang	Laptop
Nama User	Administrator
Tanggal Stok	2025-03-04 18:26:17
Jumlah Stok	30

[Kembali](#)

Edit data

Edit stok

Barang	Laptop
User	Administrator
Tanggal Stok	dd/mm/yyyy
Jumlah Stok	30

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

Hapus

localhost says

Apakah Anda yakin menghapus data ini?

OK Cancel

Home / Stok

Daftar Stok

Daftar stok yang terdaftar dalam sistem

Data stok berhasil diubah

ID	Nama Barang	Nama User	Tanggal Stok	Jumlah Stok	Aksi
1	Laptop	Administrator	2025-03-24 00:00:00	50	Detail Edit Hapus
2	Smartphone	Administrator	2025-03-04 18:26:17	25	Detail Edit Hapus
3	Kaos	Manager	2025-03-04 18:26:17	20	Detail Edit Hapus
4	Jaket	Manager	2025-03-04 18:26:17	15	Detail Edit Hapus
5	Roti	Manager	2025-03-04 18:26:17	10	Detail Edit Hapus
6	Kue	Staff/Kasir	2025-03-04 18:26:17	50	Detail Edit Hapus
7	Teh Botol	Staff/Kasir	2025-03-04 18:26:17	40	Detail Edit Hapus
8	Kopi	Staff/Kasir	2025-03-04 18:26:17	35	Detail Edit Hapus
9	Panci	Administrator	2025-03-04 18:26:17	45	Detail Edit Hapus
10	Sapu	Administrator	2025-03-04 18:26:17	60	Detail Edit Hapus

Showing 1 to 10 of 11 entries

Previous [1](#) [2](#) Next



Daftar Stok						Home	/ Stok
Daftar stok yang terdaftar dalam sistem						Tambah	
Data stok berhasil dihapus							
Filter: <input type="text" value="- Semua"/> <input type="button" value="Nama User"/>							
Show <input type="button" value="10"/> <input type="button" value="entries"/>						Search: <input type="text"/>	
ID	Nama Barang	Nama User	Tanggal Stok	Jumlah Stok	Aksi		
1	Laptop	Administrator	2025-03-24 00:00:00	50	Detail Edit Hapus		
2	Smartphone	Administrator	2025-03-04 18:26:17	25	Detail Edit Hapus		
3	Kaos	Manager	2025-03-04 18:26:17	20	Detail Edit Hapus		
4	Jaket	Manager	2025-03-04 18:26:17	15	Detail Edit Hapus		
5	Roti	Manager	2025-03-04 18:26:17	10	Detail Edit Hapus		
6	Kue	Staff/Kasir	2025-03-04 18:26:17	50	Detail Edit Hapus		
7	Teh Botol	Staff/Kasir	2025-03-04 18:26:17	40	Detail Edit Hapus		
8	Kopi	Staff/Kasir	2025-03-04 18:26:17	35	Detail Edit Hapus		
9	Panci	Administrator	2025-03-04 18:26:17	45	Detail Edit Hapus		
10	Laptop	Manager55	2025-03-10 00:00:00	50	Detail Edit Hapus		

✓ Tabel m_barang

➤ View

Index.blade

Create.blade



KEMENTERIAN PENDIDIKAN, KEBUDAYAAN, RISET, DAN TEKNOLOGI
POLITEKNIK NEGERI MALANG
JURUSAN TEKNOLOGI INFORMASI

Jl. Soekarno Hatta No. 9, Jatimulyo, Lowokwaru, Malang 65141
Telp. (0341) 404424 – 404425, Fax (0341) 404420
<http://www.polinema.ac.id>

Edit.blade



```
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">{{ $page->title }}</h3>
7          </div>
8          <div class="card-body">
9              @empty($barang)
10                  <div class="alert alert-danger alert-dismissible">
11                      <h5><i class="icon fas fa-ban"></i> Kesalahan!</h5>
12                      Data yang Anda cari tidak ditemukan.
13                  </div>
14                  <a href="{{ url('barang') }}" class="btn btn-sm btn-default mt-2">Kembali</a>
15              @else
16                  <form method="POST" action="{{ url('/barang/'.$barang->barang_id) }}" class="form-horizontal">
17                      @csrf
18                      {{ ($method_field=='PUT') ? '-- Method PUT untuk proses edit -->' }}
19                      <div class="form-group row">
20                          <label class="col-1 control-label col-form-label">Kategori</label>
21                          <div class="col-11">
22                              <select class="form-control" id="kategori_id" name="kategori_id" required>
23                                  <option value=""> Pilih Kategori </option>
24                                  @foreach($kategori as $item)
25                                      <option value="{{$item->kategori_id }}" @if($item->kategori_id == $barang->kategori_id) selected @endif>
26                                      {{ $item->kategori_nama }} </option>
27                                  @endforeach
28                              </select>
29                          <small class="form-text text-danger">{{ $message }}Kode Barang</label>
33                      <div class="col-11">
34                          <input type="text" class="form-control" id="barang_kode" name="barang_kode" value="{{ old('barang_kode', $barang->barang_kode) }}" required>
35                          @error('barang_kode')
36                              <small class="form-text text-danger">{{ $message }}Nama Barang</label>
42                      <div class="col-11">
43                          <input type="text" class="form-control" id="barang_nama" name="barang_nama" value="{{ old('barang_nama', $barang->barang_nama) }}" required>
44                          @error('barang_nama')
45                              <small class="form-text text-danger">{{ $message }}Harga Beli</label>
51                      <div class="col-11">
52                          <input type="number" class="form-control" id="harga_beli" name="harga_beli" value="{{ old('harga_beli', $barang->harga_beli) }}" required>
53                          @error('harga_beli')
54                              <small class="form-text text-danger">{{ $message }}Harga Jual</label>
60                      <div class="col-11">
61                          <input type="number" class="form-control" id="harga_jual" name="harga_jual" value="{{ old('harga_jual', $barang->harga_jual) }}" required>
62                          @error('harga_jual')
63                              <small class="form-text text-danger">{{ $message }}Simpan</button>
71                              <a class="btn btn-sm btn-default ml-1" href="{{ url('barang') }}">Kembali</a>
72                          </div>
73                      </div>
74                  </div>
75              @endsection
76          </div>
77          @push('css')
78          @endpush
79          @push('js')
80          @endpush
81      </div>
82  </div>
83  </div>
84  @endsection
85  @push('css')
86  @endpush
87  @push('js')
88  @endpush
89  @push('js')
90  @endpush
91  </div>
```

Show.blade



```
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">{{ $page->title }}</h3>
7              <div class="card-tools"></div>
8      </div>
9      <div class="card-body">
10         @empty($barang)
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>{{ $barang->barang_id }}</td>
20                 </tr>
21                 <tr>
22                     <th>Kategori</th>
23                     <td>{{ $barang->kategori->kategori_nama }}</td>
24                 </tr>
25                 <tr>
26                     <th>Kode Barang</th>
27                     <td>{{ $barang->barang_kode }}</td>
28                 </tr>
29                 <tr>
30                     <th>Nama Barang</th>
31                     <td>{{ $barang->barang_nama }}</td>
32                 </tr>
33                 <tr>
34                     <th>Harga Beli</th>
35                     <td>{{ $barang->harga_beli }}</td>
36                 </tr>
37                 <tr>
38                     <th>Harga Jual</th>
39                     <td>{{ $barang->harga_jual }}</td>
40                 </tr>
41             </table>
42         @endempty
43         <a href="{{ url('barang') }}" class="btn btn-sm btn-default mt-2">Kembali</a>
44     </div>
45  </div>
46  @endsection
47
48  @push('css')
49  @endpush
50
51  @push('js')
52  @endpush
```

➤ Controller



```
1 <?php
2
3 namespace App\http\Controllers;
4
5 use Illuminate\Http\Request;
6 use App\Models\BarangModel;
7 use App\Models\KategoriModel;
8 use Illuminate\Database\Eloquent\Model;
9
10 class BarangController extends Controller
11 {
12     public function index()
13     {
14         $breadcrumb = (object)[
15             'title' => 'Daftar Barang',
16             'list' => ['home', 'barang']
17         ];
18
19         $page = (object)[
20             'title' => 'Daftar barang yang terdaftar dalam sistem'
21         ];
22
23         $activeMenu = 'barang';
24         $kategori = KategoriModel::all(); // Ambil data kategori untuk filter
25
26         return view('barang.index', ['breadcrumb' => $breadcrumb, 'page' => $page, 'kategori' => $kategori, 'activeMenu' => $activeMenu]);
27     }
28
29     public function list(Request $request)
30     {
31         $barang = BarangModel::select(['barang_id', 'kategori_id', 'barang_kode', 'barang_nama', 'harga_beli', 'harga_jual'])
32             ->with('kategori');
33
34         if ($request->kategori_id != 0)
35         {
36             $barang->where('kategori_id', $request->kategori_id);
37
38             return $barang->get();
39         }
40
41         $barang->addIndexColumn()
42             ->addColumn('aksi', function($barang) {
43                 $breadcrumb = (object)[
44                     'title' => 'Detail Barang',
45                     'list' => ['home', 'barang', 'barang']
46                 ];
47
48                 $page = (object)[
49                     'title' => 'Detail barang'
50                 ];
51
52                 $kategori = KategoriModel::all(); // Ambil data kategori untuk dropdown
53                 $activeMenu = 'barang';
54
55                 return view('barang.detail', ['breadcrumb' => $breadcrumb, 'page' => $page, 'kategori' => $kategori, 'activeMenu' => $activeMenu]);
56             });
57
58         $barang->with('kategori');
59
60         return $barang;
61     }
62
63     public function create()
64     {
65         $breadcrumb = (object)[
66             'title' => 'Tambah Barang',
67             'list' => ['home', 'barang', 'tambah']
68         ];
69
70         $page = (object)[
71             'title' => 'Tambah barang baru'
72         ];
73
74         $kategori = KategoriModel::all(); // Ambil data kategori untuk dropdown
75         $activeMenu = 'barang';
76
77         return view('barang.create', ['breadcrumb' => $breadcrumb, 'page' => $page, 'kategori' => $kategori, 'activeMenu' => $activeMenu]);
78     }
79
80     public function store(Request $request)
81     {
82         $request->validate([
83             'kategori_id' => 'required|integer',
84             'barang_kode' => 'required|string|max:100',
85             'barang_nama' => 'required|string|max:100',
86             'harga_beli' => 'required|numeric|min:0',
87             'harga_jual' => 'required|numeric|min:0',
88         ]);
89
90         BarangModel::create([
91             'kategori_id' => $request->kategori_id,
92             'barang_kode' => $request->barang_kode,
93             'barang_nama' => $request->barang_nama,
94             'harga_beli' => $request->harga_beli,
95             'harga_jual' => $request->harga_jual,
96         ]);
97
98         return redirect('/barang')->with('success', 'Data barang berhasil disimpan');
99     }
100
101     public function show(string $id)
102     {
103         $barang = BarangModel::with('kategori')->find($id);
104
105         $breadcrumb = (object)[
106             'title' => 'Detail Barang',
107             'list' => ['home', 'barang', 'detail']
108         ];
109
110         $page = (object)[
111             'title' => 'Detail barang'
112         ];
113
114         $activeMenu = 'barang';
115
116         return view('barang.show', ['breadcrumb' => $breadcrumb, 'page' => $page, 'barang' => $barang, 'activeMenu' => $activeMenu]);
117     }
118
119     public function edit(string $id)
120     {
121         $barang = BarangModel::find($id);
122
123         $breadcrumb = (object)[
124             'title' => 'Edit Barang',
125             'list' => ['home', 'barang', 'edit']
126         ];
127
128         $page = (object)[
129             'title' => 'Edit barang'
130         ];
131
132         $kategori = KategoriModel::all(); // Ambil data kategori untuk dropdown
133         $activeMenu = 'barang';
134
135         return view('barang.edit', ['breadcrumb' => $breadcrumb, 'page' => $page, 'barang' => $barang, 'kategori' => $kategori, 'activeMenu' => $activeMenu]);
136     }
137
138     public function update(Request $request, string $id)
139     {
140         $request->validate([
141             'kategori_id' => 'required|integer',
142             'barang_kode' => 'required|string|max:100',
143             'barang_nama' => 'required|string|max:100',
144             'harga_beli' => 'required|numeric|min:0',
145             'harga_jual' => 'required|numeric|min:0',
146         ]);
147
148         BarangModel::find($id)->update([
149             'kategori_id' => $request->kategori_id,
150             'barang_kode' => $request->barang_kode,
151             'barang_nama' => $request->barang_nama,
152             'harga_beli' => $request->harga_beli,
153             'harga_jual' => $request->harga_jual,
154         ]);
155
156         return redirect('/barang')->with('success', 'Data barang berhasil diubah');
157     }
158
159     public function destroy(string $id)
160     {
161         $check = BarangModel::find($id);
162
163         if (!$check) {
164             return redirect('/barang')->with('error', 'Data barang tidak ditemukan');
165         }
166
167         try {
168             $barangModel->destroy($id);
169
170             return redirect('/barang')->with('success', 'Data barang berhasil dihapus');
171         } catch (\Illuminate\Database\QueryException $e)
172         {
173             return redirect('/barang')->with('error', 'Data barang gagal dihapus karena masih terdapat tabel lain yang terkait dengan data ini');
174         }
175     }
176 }
```

➤ Routes



```
● ● ●
```

```
1 Route::group(['prefix' => 'barang'], function () {
2     Route::get('/', [BarangController::class, 'index']);
3     Route::post('/list', [BarangController::class, 'list']);
4     Route::get('/create', [BarangController::class, 'create']);
5     Route::post('/', [BarangController::class, 'store']);
6     Route::get('/{id}', [BarangController::class, 'show']);
7     Route::get('/{id}/edit', [BarangController::class, 'edit']);
8     Route::put('/{id}', [BarangController::class, 'update']);
9     Route::delete('/{id}', [BarangController::class, 'destroy']);
10 });
});
```

Berikut tampilan pada web

ID	Kode Barang	Nama Barang	Kategori	Harga Beli	Harga Jual	Aksi
1	ELK001	Laptop	Elektronik	5000000	5500000	[Detail] [Edit] [Hapus]
2	ELK002	Smartphone	Elektronik	3000000	3500000	[Detail] [Edit] [Hapus]
3	PAK001	Kaos	Pakaian	50000	75000	[Detail] [Edit] [Hapus]
4	PAK002	Jaket	Pakaian	150000	200000	[Detail] [Edit] [Hapus]
5	MAK001	Roti	Makanan	10000	15000	[Detail] [Edit] [Hapus]
6	MAK002	Kue	Makanan	20000	25000	[Detail] [Edit] [Hapus]
7	MIN001	Teh Botol	Minuman	5000	7000	[Detail] [Edit] [Hapus]
8	MIN002	Kopi	Minuman	10000	12000	[Detail] [Edit] [Hapus]
9	PRR001	Panci	Peralatan Rumahh	50000	75000	[Detail] [Edit] [Hapus]
10	PRR002	Sapu	Peralatan Rumahh	25000	30000	[Detail] [Edit] [Hapus]

Tambah data

Tambah barang baru	
Kategori	Pakaian
Kode Barang	PAK003
Nama Barang	Flanel
Harga Beli	150000
Harga Jual	200000



Daftar barang yang terdaftar dalam sistem						
Data barang berhasil disimpan						
Filter: - Semua						
Kategori Barang						
Show:	10	entries	Search:			
ID	Kode Barang	Nama Barang	Kategori	Harga Beli	Harga Jual	Aksi
11	PAK003	Flanel	Pakaian	150000	200000	<button>Detail</button> <button>Edit</button> <button>Hapus</button>

Detail data

Detail Barang	
Detail barang	
ID	11
Kategori	Pakaian
Kode Barang	PAK003
Nama Barang	Flanel
Harga Beli	150000
Harga Jual	200000
Kembali	

Edit data

Edit Barang

Edit barang

Kategori	Pakaian
Kode Barang	PAK003
Nama Barang	Flanel1
Harga Beli	150000
Harga Jual	200000

Simpan **Kembali**

Hapus

localhost says
Apakah Anda yakin menghapus data ini?

OK Cancel

Daftar Barang						
Daftar barang yang terdaftar dalam sistem						
Data barang berhasil diubah						
Filter: - Semua						
Show:	10	entries	Search:			
ID	Kode Barang	Nama Barang	Kategori	Harga Beli	Harga Jual	Aksi
11	PAK003	Flanel1	Pakaian	150000	200000	<button>Detail</button> <button>Edit</button> <button>Hapus</button>



Daftar Barang

Home / Barang

Daftar barang yang terdaftar dalam sistem

Tambah

Data barang berhasil dihapus

Filter: - Semua

Kategori Barang

Show 10 5 entries

Search:

ID	Kode Barang	Nama Barang	Kategori	Harga Beli	Harga Jual	Aksi
1	ELK001	Laptop	Elektronik	5000000	5500000	Detail Edit Hapus
2	ELK002	Smartphone	Elektronik	3000000	3500000	Detail Edit Hapus
3	PAK001	Kaos	Pakaian	50000	75000	Detail Edit Hapus
4	PAK002	Jaket	Pakaian	150000	200000	Detail Edit Hapus
5	MAK001	Roti	Makanan	10000	15000	Detail Edit Hapus
6	MAK002	Kue	Makanan	20000	25000	Detail Edit Hapus
7	MIN001	Teh Botol	Minuman	5000	7000	Detail Edit Hapus
8	MIN002	Kopi	Minuman	10000	12000	Detail Edit Hapus
9	PRR001	Panci	Peralatan Rumahh	50000	75000	Detail Edit Hapus
10	PRR002	Sapu	Peralatan Rumahh	25000	30000	Detail Edit Hapus

Showing 1 to 10 of 10 entries

Previous [1](#) Next

*** Sekian, dan selamat belajar ***