



PRAKTIKUM

Pemrograman Web Lanjut - Minggu 8

DISUSUN OLEH:

NAMA : Ferdi Riansyah Ramadhani Kusuma
NIM : 2241720264
JURUSAN : TEKNOLOGI INFORMASI
KELAS/ ABSEN : INFORMATIKA - 2H

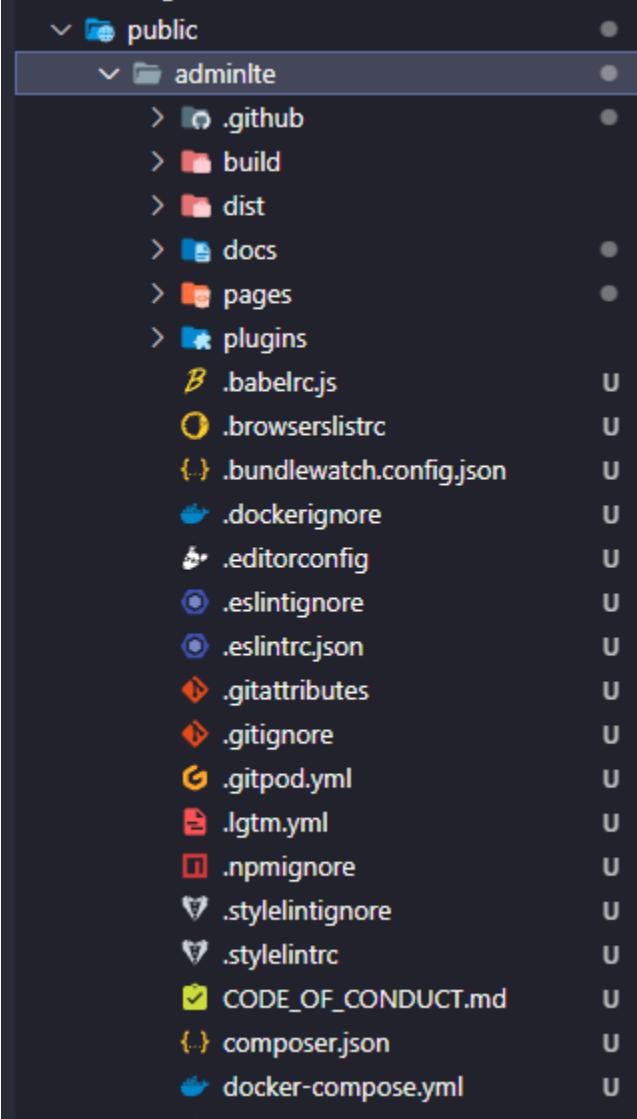
PROGAM STUDI D-IV TEKNIK INFORMATIKA

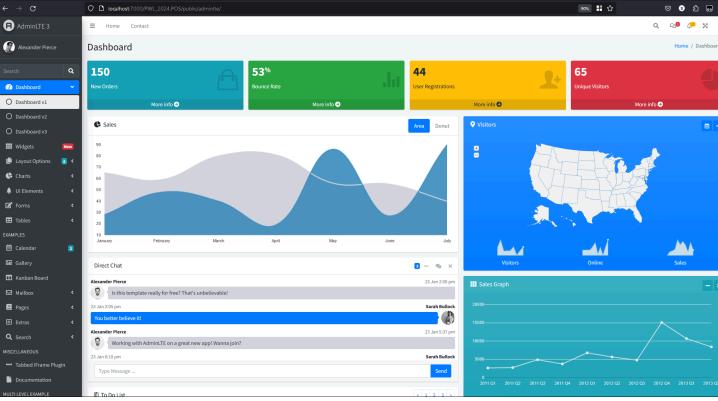
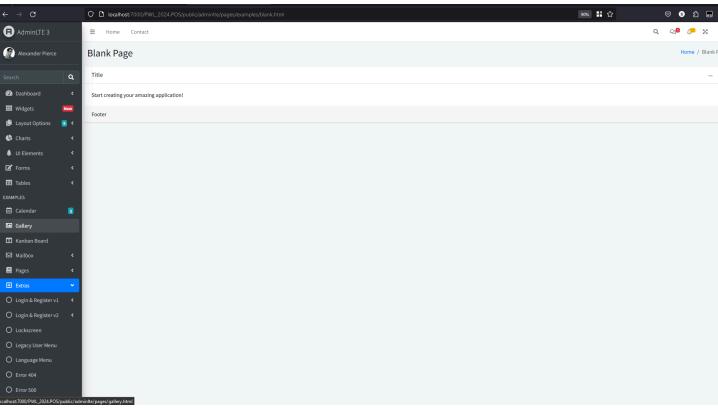
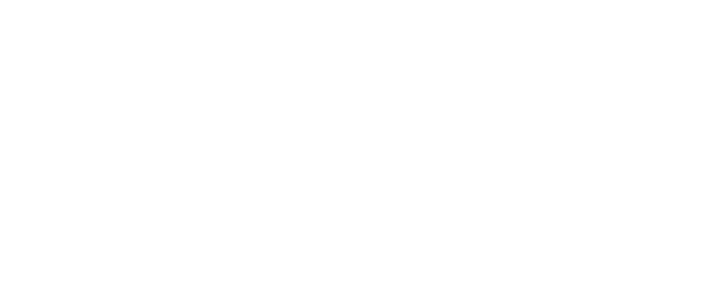
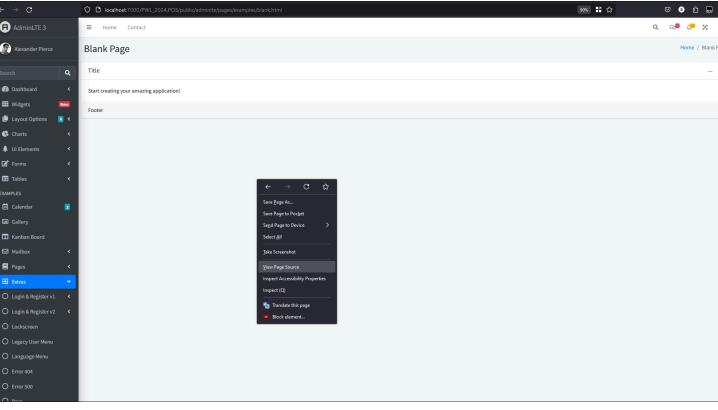
JURUSAN TEKNOLOGI INFORMASI

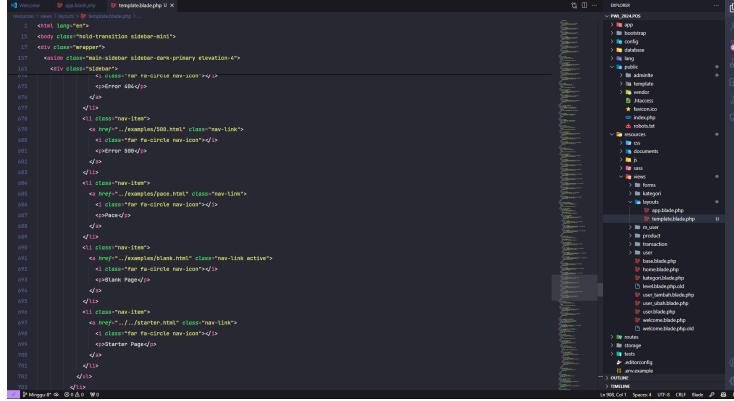
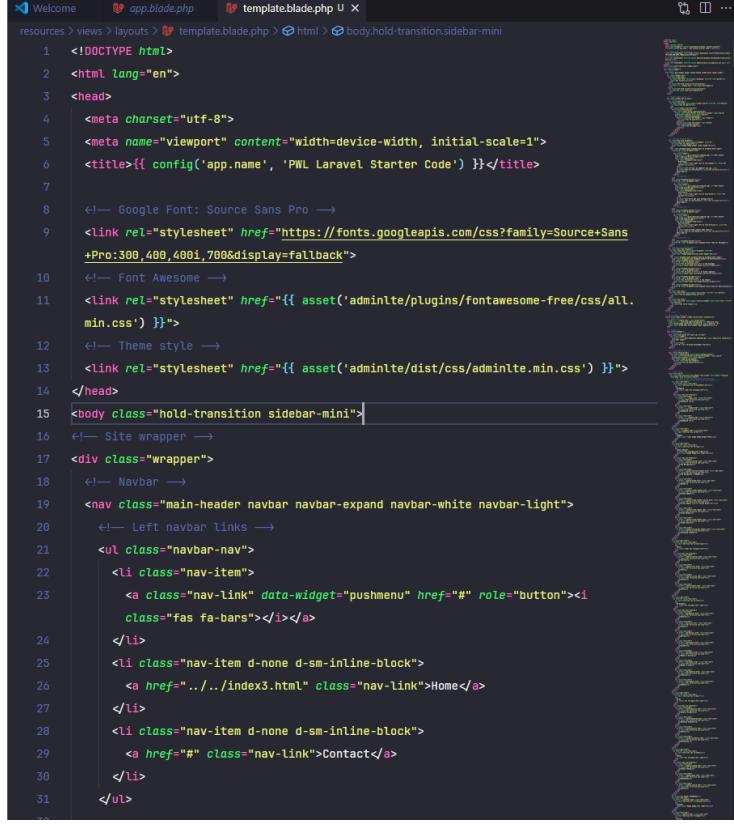
POLITEKNIK NEGERI MALANG

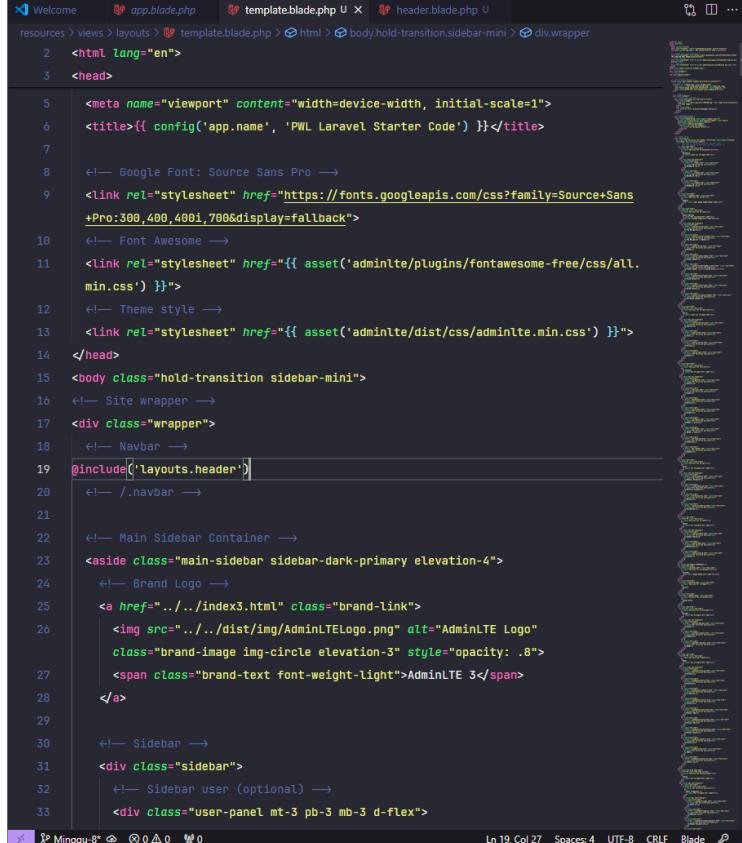
2024

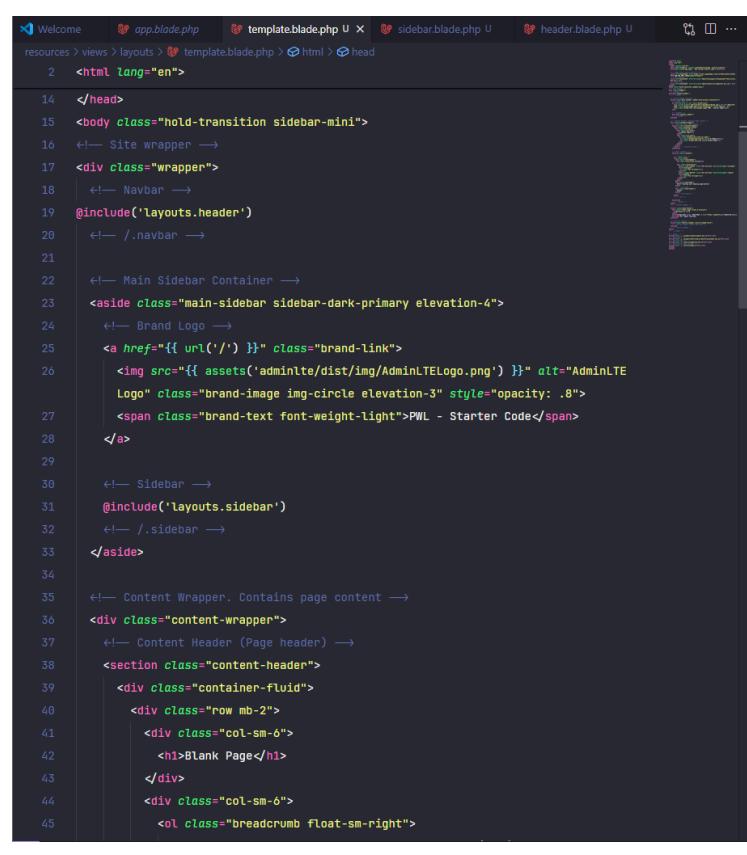
Praktikum 1 – Layouting AdminLTE

No	Steps	Attachments
1.	Kita download AdminLTE v3.2.0 yang rilis pada 8 Feb 2022	
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	

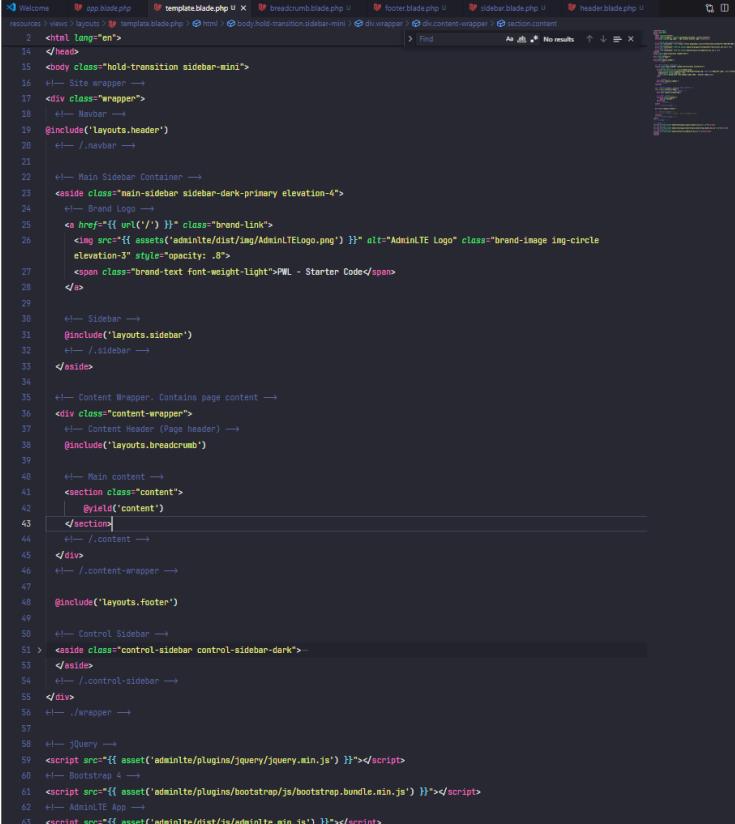
3.	<p>Selanjutnya kita buka di browser dengan alamat http://localhost/PWL_POS/public/adminlte maka akan muncul tampilan seperti berikut</p>	
4.	<p>Kita klik menu Extras > Blank Page, page inilah yang akan menjadi dasar web template</p>	
5.	<p>Dari sini kita bisa melakukan layouting halaman Blank Page ini menjadi 4 element seperti pada gambar berikut</p>	
6.	<p>Selanjutnya kita klik kanan halaman Blank Page dan klik view page source</p>	

7.	<p>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)</p>	 <pre> <?php // app.blade.php <!DOCTYPE html> <html lang="en"> <head> <meta charset="utf-8"> <meta name="viewport" content="width=device-width, initial-scale=1"> <title>{{ config('app.name', 'PWL Laravel Starter Code') }}</title> <!-- Google Font: Source Sans Pro --> <link rel="stylesheet" href="https://fonts.googleapis.com/css?family=Source+Sans+Pro:300,400,400i,700&display=fallback"> <!-- Font Awesome --> <link rel="stylesheet" href="{{ asset('adminlte/plugins/fontawesome-free/css/all.min.css') }}"> <!-- Theme style --> <link rel="stylesheet" href="{{ asset('adminlte/dist/css/adminlte.min.css') }}"> </head> <body class="hold-transition sidebar-mini"> <!-- Site wrapper --> <div class="wrapper"> <!-- Navbar --> <nav class="main-header navbar navbar-expand navbar-white navbar-light"> <!-- Left navbar links --> <ul class="navbar-nav"> <li class="nav-item"> <i class="fas fa-bars"></i> <li class="nav-item d-none d-sm-inline-block"> Home <li class="nav-item d-none d-sm-inline-block"> Contact </nav> </div> <!-- Content Wrapper --> <div class="content-wrapper"> <!-- Content Header (Optional) --> <div class="content-header"> <h1>Blank Page</h1> <ol class="breadcrumb"> <li class="breadcrumb-item">Home <li class="breadcrumb-item active">Blank Page </div> <!-- Main content --> <div class="content"> <p>This is a blank page.</p> </div> </div> <!-- Main Footer --> <div class="main-footer"> <small>Copyright © 2018. All Rights Reserved. Template by PWL</small> </div> </body> </pre>
8.	<p>File layouts/template.blade.php adalah file utama untuk templating website</p>	
9.	<p>Pada baris 1-14 file template.blade.php, kita modifikasi</p>	 <pre> <?php // app.blade.php <!DOCTYPE html> <html lang="en"> <head> <meta charset="utf-8"> <meta name="viewport" content="width=device-width, initial-scale=1"> <title>{{ config('app.name', 'PWL Laravel Starter Code') }}</title> <!-- Google Font: Source Sans Pro --> <link rel="stylesheet" href="https://fonts.googleapis.com/css?family=Source+Sans+Pro:300,400,400i,700&display=fallback"> <!-- Font Awesome --> <link rel="stylesheet" href="{{ asset('adminlte/plugins/fontawesome-free/css/all.min.css') }}"> <!-- Theme style --> <link rel="stylesheet" href="{{ asset('adminlte/dist/css/adminlte.min.css') }}"> </head> <body class="hold-transition sidebar-mini"> <!-- Site wrapper --> <div class="wrapper"> <!-- Navbar --> <nav class="main-header navbar navbar-expand navbar-white navbar-light"> <!-- Left navbar links --> <ul class="navbar-nav"> <li class="nav-item"> <i class="fas fa-bars"></i> <li class="nav-item d-none d-sm-inline-block"> Home <li class="nav-item d-none d-sm-inline-block"> Contact </nav> </div> <!-- Content Wrapper --> <div class="content-wrapper"> <!-- Content Header (Optional) --> <div class="content-header"> <h1>Blank Page</h1> <ol class="breadcrumb"> <li class="breadcrumb-item">Home <li class="breadcrumb-item active">Blank Page </div> <!-- Main content --> <div class="content"> <p>This is a blank page.</p> </div> </div> <!-- Main Footer --> <div class="main-footer"> <small>Copyright © 2018. All Rights Reserved. Template by PWL</small> </div> </body> </pre>

10.	<p>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</p> <p>Baris 19 adalah komponen Blade untuk memanggil elemen layouts/header.blade.php agar menjadi satu dengan template.blade.php saat di-render nanti.</p>	 <pre> 1 Welcome app.blade.php template.blade.php U X header.blade.php U resources > views > layouts > template.blade.php > html > body.hold-transition.sidebar-mini > div.wrapper 2 <html lang="en"> 3 <head> 4 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> 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 26 28 AdminLTE 3 29 30 31 <!-- Sidebar --> 32 <div class="sidebar"> 33 <!-- Sidebar user (optional) --> 34 <div class="user-panel mt-3 pb-3 mb-3 d-flex"> </pre> <p>Minggu-8* ⌂ 0 △ 0 ⌂ 0 Ln 19, Col 27 Spaces: 4 UTF-8 CRLF Blade ⌂</p>
11.	Kita modifikasi baris 25 dan 26 pada template.blade.php	 <pre> <!-- Main Sidebar Container --> <aside class="main-sidebar sidebar-dark-primary elevation-4"> <!-- Brand Logo --> PWL - Starter Code </pre>

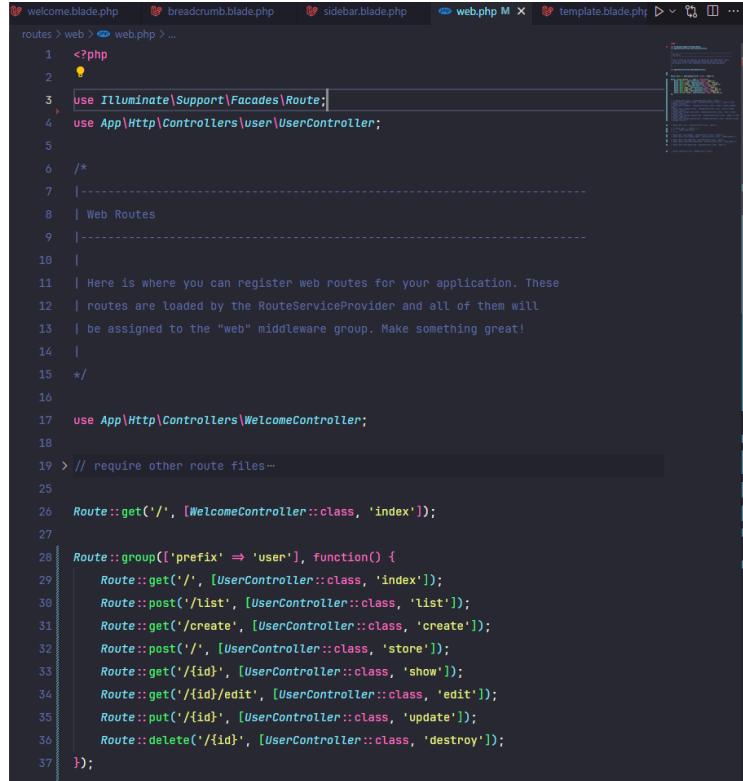
12.	<p>Selanjutnya kita blok baris 31-693 (baris untuk element 2-sidebar), lalu kita cut, dan paste-kan di file PWL_POS/resource/view/layouts/sidebar.blade.php (buat dulu file sidebar.blade.php jika belum ada). Sehingga tampilan dari file template.blade.php menjadi seperti berikut</p>	 <pre> 2 <html lang="en"> 3 </head> 4 <body class="hold-transition sidebar-mini"> 5 <!-- Site wrapper --> 6 <div class="wrapper"> 7 <!-- Navbar --> 8 @include('layouts.header') 9 <!-- /.navban --> 10 11 <!-- Main Sidebar Container --> 12 <aside class="main-sidebar sidebar-dark-primary elevation-4"> 13 <!-- Brand Logo --> 14 15 16 PWL - Starter Code 17 18 19 <!-- Sidebar --> 20 @include('layouts.sidebar') 21 <!-- /.sidebar --> 22 </aside> 23 24 <!-- Content Wrapper. Contains page content --> 25 <div class="content-wrapper"> 26 <!-- Content Header (Page header) --> 27 <section class="content-header"> 28 <div class="container-fluid"> 29 <div class="row mb-2"> 30 <div class="col-sm-6"> 31 <h1>Blank Page</h1> 32 </div> 33 <div class="col-sm-6"> 34 <ol class="breadcrumb float-sm-right"> 35 36 37 38 39 30 31 32 33 34 35 36 37 38 39 40 41 42 43 44 45 </pre>
13.	<p>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</p>	<pre> 84 </div> 85 <!-- /.content-wrapper --> 86 87 @include('layouts.footer') 88 89 <!-- Control Sidebar --> 90 <aside class="control-sidebar control-sidebar-dark"> </pre>

14.	<p>Kemudian kita modifikasi file template.blade.php baris 91-100</p>	<pre> <!-- jquery --> <script src="{{ asset('adminlte/plugins/jquery/jquery.min.js') }}"></script> <!-- Bootstrap 4 --> <script src="{{ asset('adminlte/plugins/bootstrap/js/bootstrap.bundle.min.js') }}"></script> <!-- AdminLTE App --> <script src="{{ asset('adminlte/dist/js/adminlte.min.js') }}"></script> </body> </html> </pre>
15.	<p>Sekarang masuk pada bagian konten. Konten kita bagi menjadi 2, yaitu elemen untuk breadcrumb dan elemen untuk content.</p>	
16.	<p>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</p>	<pre> 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 </pre>
17.	<p>Layout terakhir adalah pada bagian konten. Layout untuk</p>	

	<p>konten bisa kita buat dinamis,</p> <p>sesuai dengan apa yang ingin kita sajikan pada web yang kita bangun.</p>	
18.	<p>Untuk content, kita akan menghapus baris 42-66 pada file template.blade.php. dan kita</p> <p>ganti dengan kode seperti ini @yield('content')</p>	
19.	<p>Hasil akhir pada file utama layouts/template.blade.php adalah seperti berikut</p>  <pre> 1 Welcome app.blade.php template.blade.php x breadcrumb.blade.php x footer.blade.php x sidebar.blade.php x header.blade.php resources \ views \ layouts > template.blade.php 2 <html lang="en"> 3 <head> 4 <body class="hold-transition sidebar-mini"> 5 <!-- Site wrapper --> 6 <div class="wrapper"> 7 <!-- Navbar --> 8 @include('layouts.header') 9 <!-- /.navbar --> 10 11 <!-- Main Sidebar Container --> 12 <aside class="main-sidebar sidebar-dark-primary elevation-4"> 13 <!-- Brand Logo --> 14 15 16 PWL - Starter Page 17 18 19 <!-- Sidebar --> 20 @include('layouts.sidebar') 21 <!-- /.sidebar --> 22 </aside> 23 24 <!-- Content Wrapper. Contains page content --> 25 <div class="content-wrapper"> 26 <!-- Content Header (Page header) --> 27 @include('layouts.breadcrumb') 28 29 <!-- Main content --> 30 <section class="content"> 31 @yield('content') 32 </section> 33 </div> 34 35 <!-- Control Sidebar --> 36 <aside class="control-sidebar control-sidebar-dark"> 37 <!-- /.control-sidebar --> 38 </aside> 39 40 <!-- jQuery --> 41 <script src="{{ asset('adminlte/plugins/jquery/jquery.min.js') }}></script> 42 43 <!-- Bootstrap 4 --> 44 <script src="{{ asset('adminlte/plugins/bootstrap/js/bootstrap.bundle.min.js') }}></script> 45 46 <!-- AdminLTE App --> 47 <script src="{{ asset('adminlte/dist/js/adminlte.min.js') }}></script> 48 49 </body> 50 </html> </pre>	
20.	<p>Selamat kalian sudah selesai dalam melakukan layouting</p>	

	website di laravel.	
--	---------------------	--

Praktikum 3 – Implementasi jQuery Datatable di AdminLTE

No	Steps	Attachments
1.	Kita modifikasi proses CRUD pada tabel m_user pada praktikum ini	
2.	Kita gunakan library Yajra-datatable 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	 <pre> 1 <?php 2 3 use Illuminate\Support\Facades\Route; 4 use App\Http\Controllers\User\ UserController; 5 6 /* 7 ----- 8 Web Routes 9 ----- 10 11 Here is where you can register web routes for your application. These 12 routes are loaded by the RouteServiceProvider and all of them will 13 be assigned to the "web" middleware group. Make something great! 14 15 */ 16 17 use App\Http\Controllers\WelcomeController; 18 19 > // require other route files... 20 21 22 Route::get('/', [WelcomeController::class, 'index']); 23 24 25 Route::group(['prefix' => 'user'], function() { 26 Route::get('/', [UserController::class, 'index']); 27 Route::post('/list', [UserController::class, 'list']); 28 Route::get('/create', [UserController::class, 'create']); 29 Route::post('/', [UserController::class, 'store']); 30 Route::get('{id}', [UserController::class, 'show']); 31 Route::get('{id}/edit', [UserController::class, 'edit']); 32 Route::put('{id}', [UserController::class, 'update']); 33 Route::delete('{id}', [UserController::class, 'destroy']); 34 }); 35 36 37 }); 38 </pre>

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



```
<?php  
namespace App\Http\Controllers;  
  
use App\Models\LevelModel;  
  
class UserController extends Controller  
{  
    public function index()  
    {  
        $breadcrumb = (object) [  
            'title' => 'Daftar User',  
            'list' => ['home', 'User']  
        ];  
  
        $page = (object) [  
            'title' => 'Daftar user yang terdaftar dalam sistem'  
        ];  
  
        $activeMenu = 'user';  
  
        $level = LevelModel::all();  
  
        return view('user.index', compact('breadcrumb', 'page', 'activeMenu', 'level'));  
    }  
}
```

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

```

web.php M UserController.php.old U UserController.php M index.blade.php X template.blade.php
resources > views > user > index.blade.php > div.card.card-outline.card-primary > div.card-header > div.card-tools > a.btn.btn-sm.btn-primary
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">{{ $page->title }}</h3>
7             <div class="card-tools">
8                 <a class="btn btn-sm btn-primary mt-1" href="{{ url('user/create') }}">Tambah</a>
9             </div>
10        </div>
11        <div class="card-body">
12            @if (session('success'))
13                <div class="alert alert-success">{{ session('success') }}</div>
14            @endif
15            @if (session('error'))
16                <div class="alert alert-danger">{{ session('error') }}</div>
17            @endif
18            <div class="row">
19                <div class="col-md-12">
20                    <div class="form-group row">
21                        <label class="col-1 control-label col-form-label">Filter : </label>
22                        <div class="col-3">
23                            <select name="level_id" id="level_id" class="form-control" required>
24                                <option value=""> Semua </option>
25                                @foreach ($level as $item)
26                                    <option value="{{ $item->level_id }}>{{ $item->level_nama }}</option>
27                                @endforeach
28                            </select>
29                            <small class="form-text text-muted">Level Pengguna</small>
30                        </div>
31                    </div>
32                </div>
33            </div>
34        </div>
35    </div>
36
```

6.	<p>Kemudian kita modifikasi file template.blade.php untuk menambahkan library jquery datatables dari template AdminLTE yang kita download dan berada di folder public</p> <pre data-bbox="693 221 1428 1047"><!DOCTYPE html> <html lang="en"> <head> <meta charset="utf-8"> <meta name="viewport" content="width=device-width, initial-scale=1"> <title>{{ config('app.name', 'PWL Laravel Starter Code') }}</title> <meta name="csrf-token" content="{{ csrf_token() }}> <!-- Google Font: Source Sans Pro --> <link rel="stylesheet" href="https://fonts.googleapis.com/css?family=Source+Sans *Pro:300,400,400i,700&display=fallback"> <!-- Font Awesome --> <link rel="stylesheet" href="{{ asset('adminlte/plugins/fontawesome-free/css/all. min.css') }}> <!-- Theme style --> <link rel="stylesheet" href="{{ asset('adminlte/dist/css/adminlte.min.css') }}> <!-- DataTables --> <link rel="stylesheet" href="{{ asset('adminlte/plugins/datatables-bs4/css/ dataTables.bootstrap4.min.css') }}> <link rel="stylesheet" href="{{ asset('adminlte/plugins/datatables-responsive/css/ responsive.bootstrap4.min.css') }}> <link rel="stylesheet" href="{{ asset('adminlte/plugins/datatables-buttons/css/ buttons.bootstrap4.min.css') }}> @stack('css') </head> <body class="hold-transition sidebar-mini"> <!-- Site wrapper --> <div class="wrapper"> <!-- Navbar --> @include('layouts.header')</pre>
7.	<p>Untuk bisa menangkap request data untuk datatable, kita buat fungsi list() pada UserController.php seperti berikut</p> <pre data-bbox="693 1066 1428 1809">public function list(Request \$request) { \$users = UserModel::select(['user_id', 'username', 'nama', 'level_id'])->with ('level'); if (\$request->level_id) { \$users->where('level_id', \$request->level_id); } return DataTables::of(\$users) ->addIndexColumn() ->addColumn('aksi', function (\$user) { \$btn = 'user_id).'" class="btn btn-info btn-sm">Detail'; \$btn .= 'user_id . '/edit').'"' class="btn btn-warning btn-sm">Edit'; \$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']) ->make(true); }</pre>

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

ID	Username	Nama	Level Pengguna	Aksi
1	admin	Administrator	Administrator	[Detail] [Edit] [Hapus]
2	manager	Manager	Manager	[Detail] [Edit] [Hapus]
3	staff	Staff/Kasir	Staff/Kasir	[Detail] [Edit] [Hapus]
4	customer-1	Pelanggan Pertama	Customer	[Detail] [Edit] [Hapus]
5	manager_dua	Manager 2	Manager	[Detail] [Edit] [Hapus]
6	manager22	Manager Dua Dua	Manager	[Detail] [Edit] [Hapus]
7	manager33	Manager Tiga Tiga	Manager	[Detail] [Edit] [Hapus]
8	manager56	Manager55	Manager	[Detail] [Edit] [Hapus]
9	manager12	Manager11	Manager	[Detail] [Edit] [Hapus]
10	test	test	Administrator	[Detail] [Edit] [Hapus]

Showing 1 to 10 of 10 entries

Copyright © 2014-2021 AdminLTE.io. All rights reserved. Version 3.2.0

Method list dalam class UserController digunakan untuk menghandle list tampilan data user menggunakan data table.

9. Selanjutnya kita modifikasi UserController.php untuk form tambah data user

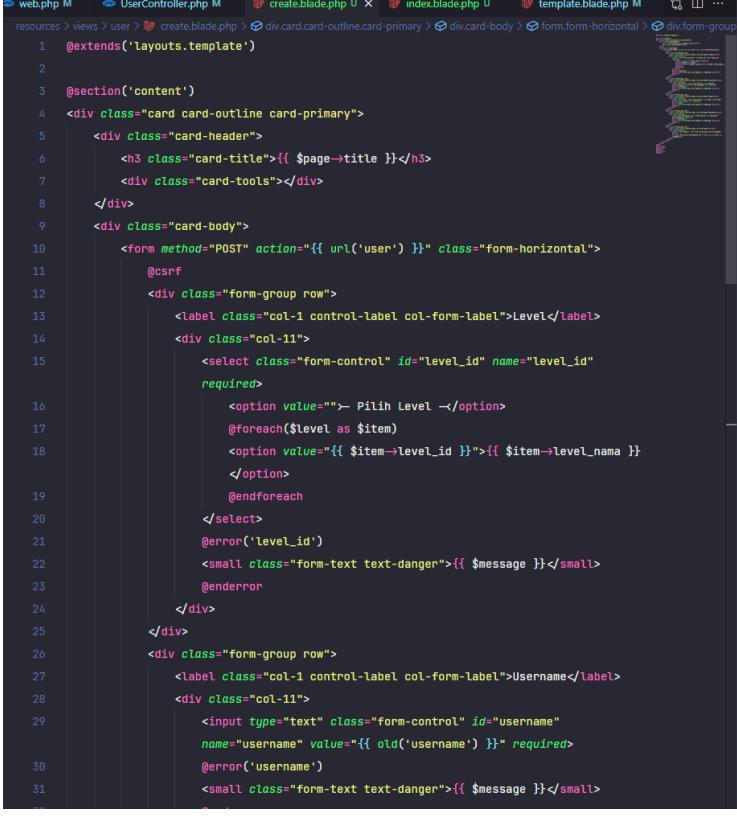
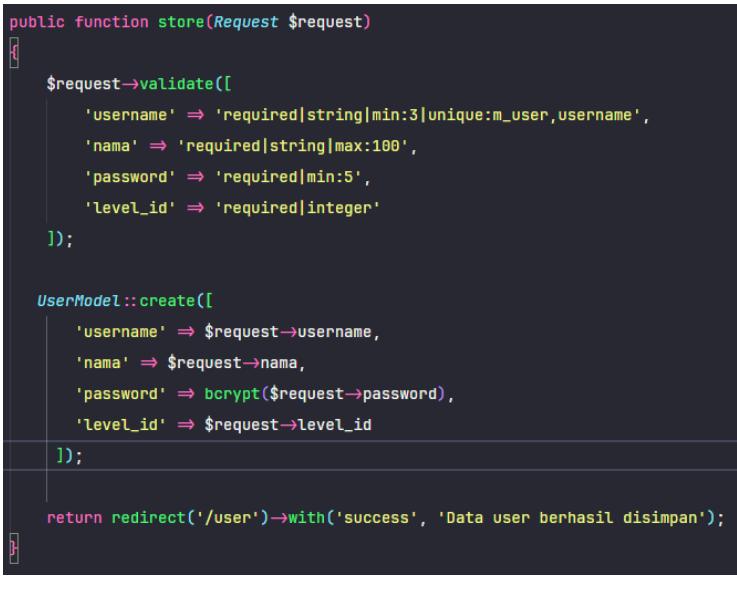
```
1 reference | 0 overrides

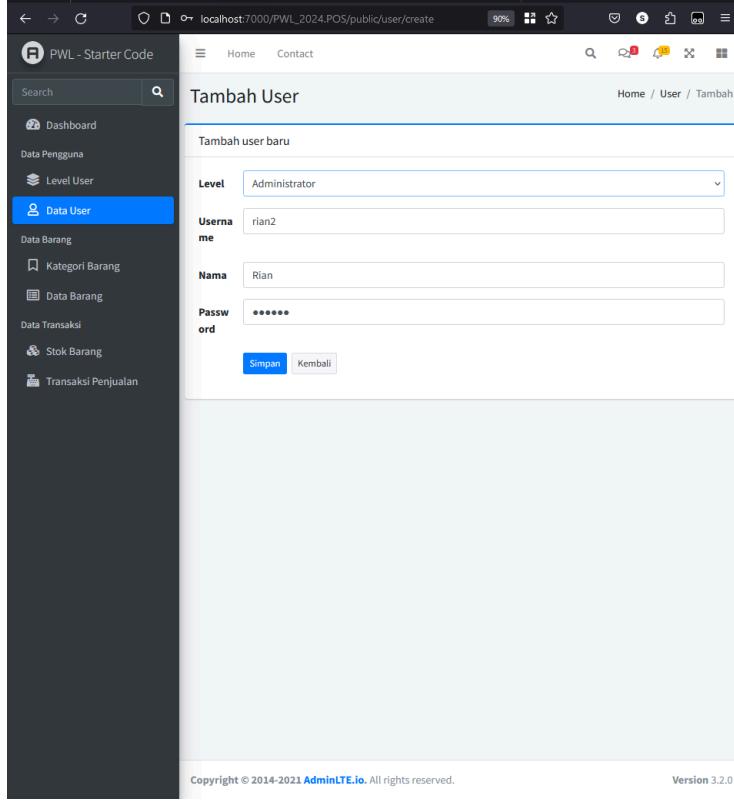
public function create()
{
    $breadcrumb = (object) [
        'title' => 'Tambah User',
        'list' => ['Home', 'User', 'Tambah']
    ];

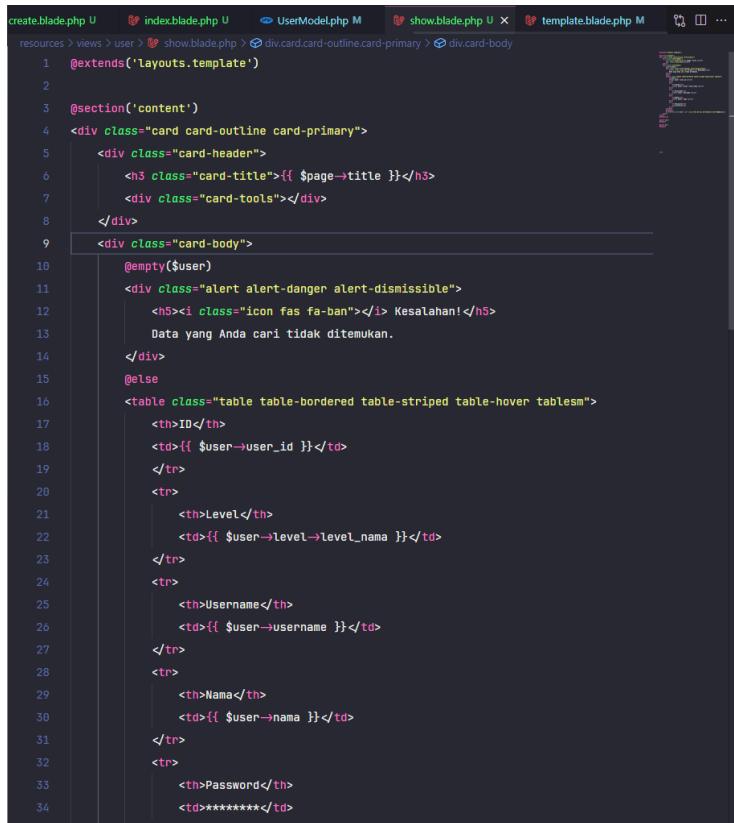
    $page = (object) [
        'title' => 'Tambah user baru'
    ];

    $level = LevelModel::all();
    $activeMenu = 'user';

    return view('user.create', compact
        ('breadcrumb', 'page', 'level',
        'activeMenu'));
}
```

10.	<p>Sekarang kita buat form untuk menambah data, kita buat file PWL_POS/resources/views/user/create.blade.php</p>	 <pre> 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('user') }}" class="form-horizontal"> 11 @csrf 12 <div class="form-group row"> 13 <label class="col-1 control-label col-form-label">Level</label> 14 <div class="col-11"> 15 <select class="form-control" id="level_id" name="level_id" required> 16 <option value=""> Pilih Level </option> 17 @foreach(\$level as \$item) 18 <option value="{{ \$item->level_id }}>{{ \$item->level_nama }}</option> 19 @endforeach 20 </select> 21 @error('level_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">Username</label> 28 <div class="col-11"> 29 <input type="text" class="form-control" id="username" name="username" value="{{ old('username') }}" required> 30 @error('username') 31 <small class="form-text text-danger">{{ \$message }}</small> 32 @enderror 33 </div> 34 </div> 35 </form> 36 </div> 37 </pre>
11.	<p>Kemudian untuk bisa menng-handle data yang akan disimpan ke database, kita buat fungsi store() di UserController.php</p>	 <pre> public function store(Request \$request) { \$request->validate(['username' => 'required string min:3 unique:m_user,username', 'nama' => 'required string max:100', 'password' => 'required min:5', 'level_id' => 'required integer']); UserModel::create(['username' => \$request->username, 'nama' => \$request->nama, 'password' => bcrypt(\$request->password), 'level_id' => \$request->level_id]); return redirect('/user')->with('success', 'Data user berhasil disimpan'); } </pre>

12.	<p>Sekarang coba kalian buka form tambah data user dengan klik tombol tambah. Amati dan pelajari..!!!</p>	 <p>Setelah menekan tombol tambah maka akan diarahkan menuju halaman tambah user, user dapat ditambahkan menggunakan tombol Simpan.</p>
13.	<p>Selanjutnya, kita masuk pada bagian menampilkan detail data user (klik tombol) pada halaman user. Route yang bertugas untuk menangkap request detail adalah</p>	

14.	<p>Jadi kita buat/modifikasi fungsi show() pada UserController.php seperti berikut</p>	<pre>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'; return view('user.show', compact('breadcrumb', 'page', 'user', 'activeMenu')); }</pre>
15.	<p>Kemudian kita buat view di PWL_POS/resources/views/user/show.blade.php</p>	 <pre>create.blade.php U index.blade.php U UserModel.php M show.blade.php X template.blade.php M resources > views > user > show.blade.php > div.card.card-outline.card-primary > div.card-body 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(\$user) 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 tablesm"> 17 <thead> 18 <tr> 19 <th>ID</th> 20 <td>{{ \$user->user_id }}</td> 21 </tr> 22 <tbody> 23 <tr> 24 <th>Level</th> 25 <td>{{ \$user->level->level_nama }}</td> 26 </tr> 27 <tr> 28 <th>Username</th> 29 <td>{{ \$user->username }}</td> 30 </tr> 31 <tr> 32 <th>>Nama</th> 33 <td>{{ \$user->nama }}</td> 34 </tr> 35 <tr> 36 <th>>Password</th> 37 <td>*****</td> 38 </tr> 39 </tbody> 40 </table> 41 </div> 42 </div> 43</div></pre>
16.	<p>Sekarang kalian coba untuk melihat detail data user di browser, dan coba untuk mengetikkan id yang salah contoh http://localhost/PWL_POS/public</p>	Id Benar

lic/user/100 amati

apa yang terjadi, dan
laporkan!!!

The screenshot shows a web application interface titled "PWL - Starter Code". The left sidebar contains navigation links: Dashboard, Data Pengguna, Level User, Data User (which is highlighted in blue), Data Barang, Kategori Barang, Data Transaksi, Stok Barang, and Transaksi Penjualan. The main content area is titled "Detail User" and displays the following information:

Detail user	
ID	1
Level	Administrator
Username	admin
Nama	Administrator
Password	*****

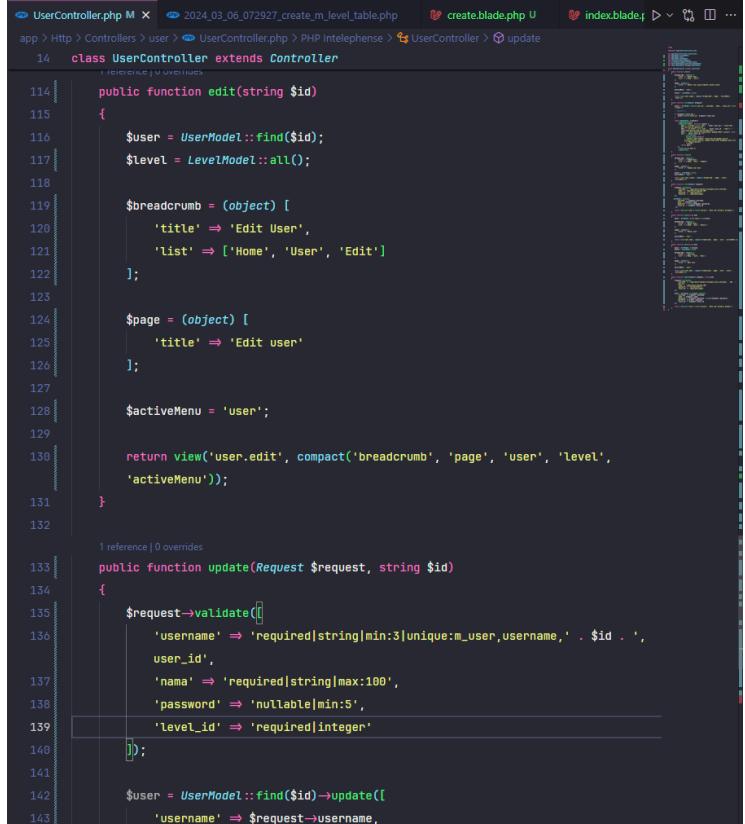
A "Kembali" button is located at the bottom left of the content area.

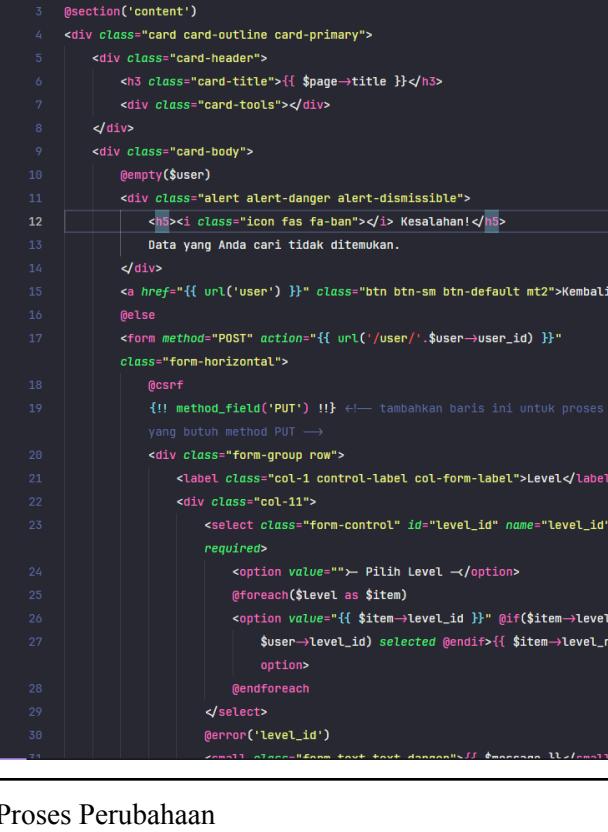
Id Salah

The screenshot shows a web application interface titled "PWL - Starter Code". The left sidebar contains navigation links: Dashboard, Data Pengguna, Level User, Data User (highlighted in blue), Data Barang, Kategori Barang, Data Transaksi, Stok Barang, and Transaksi Penjualan. The main content area is titled "Detail User" and displays an error message in a red box:

Kesalahan!
Data yang Anda cari tidak ditemukan.

A "Kembali" button is located at the bottom left of the content area.

17.	<p>Selanjutnya, kita masuk pada bagian untuk memodifikasi data user. Route yang bertugas untuk menangkap request edit adalah</p>	
18.	<p>Jadi kita buat fungsi edit() dan update() pada UserController.php</p>	 <pre> 14 class UserController extends Controller 15 { 16 public function edit(string \$id) 17 { 18 \$user = UserModel::find(\$id); 19 \$level = LevelModel::all(); 20 21 \$breadcrumb = (object) [22 'title' => 'Edit User', 23 'list' => ['Home', 'User', 'Edit'] 24]; 25 26 \$page = (object) [27 'title' => 'Edit user' 28]; 29 30 \$activeMenu = 'user'; 31 32 return view('user.edit', compact('breadcrumb', 'page', 'user', 'level', 33 'activeMenu')); 34 } 35 36 public function update(Request \$request, string \$id) 37 { 38 \$request->validate([39 'username' => 'required string min:3 unique:m_user,username,' . \$id . ', 40 'user_id', 41 'nama' => 'required string max:100', 42 'password' => 'nullable min:5', 43 'level_id' => 'required integer' 44]); 45 46 \$user = UserModel::find(\$id)->update([47 'username' => \$request->username, 48]); 49 } 50 } </pre>

19.	<p>Selanjutnya, kita buat view untuk melakukan proses edit data user di</p> <p>PWL_POS/resources/views/user/edit.blade.php</p>	 <pre> UserController.php M edit.blade.php U 2024-03-06_072927_create_m_level_table.php create.blade.php U resources > views > user > edit.blade.php > edit.blade.php > div.card.card-outline.card-primary > div.card-body > div.alert.alert-dismissible 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(\$user) 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 Kembali 16 @else 17 <form method="POST" action="{{ url('/user/'. \$user->user_id) }}" 18 class="form-horizontal"> 19 @csrf 20 {!! method_field('PUT') !!}<!-- tambahkan baris ini untuk proses edit 21 yang butuh method PUT --> 22 <div class="form-group row"> 23 <label class="col-1 control-label col-form-label">Level</label> 24 <div class="col-11"> 25 <select class="form-control" id="level_id" name="level_id" 26 required> 27 <option value="">- Pilih Level -</option> 28 @foreach(\$level as \$item) 29 <option value="{{ \$item->level_id }}> @if(\$item->level_id == 30 \$user->level_id) selected @endif{{ \$item->level_nama }}</ 31 option> 32 @endforeach 33 </select> 34 @error('level_id') 35 <div class="invalid-feedback">{{ \$message }}</div> 36 @enderror 37 </div> 38 </form> 39 @endsection 40 </div> 41 <div class="card-footer"></div> 42</div> </pre>
20.	<p>Sekarang kalian coba untuk mengedit data user di browser, amati, pahami, dan laporkan!</p>	<p>Proses Perubahan</p>

localhost:7000/PWL_2024.POS/public/user/13/edit

Edit User

Level: Administrator

Username: rian2

Nama: Rian [Updated]

Password: *****

Abaikan (jangan diisi) jika tidak ingin mengganti password user.

Simpan Kembali

Copyright © 2014-2021 AdminLTE.io. All rights reserved. Version 3.2.0

Setelah Perubahan

localhost:7000/PWL_2024.POS/public/user

Daftar User

Data user berhasil diubah

Tambah

Filter: - Semua - Level Pengguna

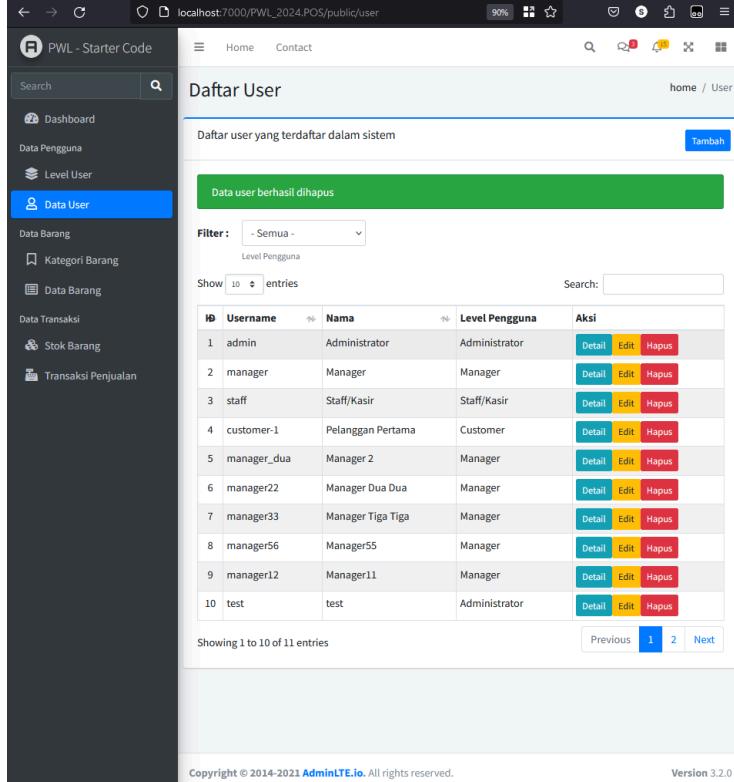
Show 10 entries Search:

ID	Username	Nama	Level Pengguna	Aksi
11	rian	Rian	Customer	[Detail] [Edit] [Hapus]
12	rian2	Rian [Updated]	Administrator	[Detail] [Edit] [Hapus]

Showing 11 to 12 of 12 entries Previous 1 2 Next

Copyright © 2014-2021 AdminLTE.io. All rights reserved. Version 3.2.0

21.	<p>Selanjutnya kita akan membuat penanganan untuk tombol hapus. Router web.php yang berfungsi untuk menangkap request hapus dengan method DELETE adalah</p> <pre>Route::delete('/{id}', [UserController::class, 'destroy']);</pre>	
22.	<p>Jadi kita buat fungsi destroy() pada UserController.php</p>	<pre>1 reference 0 overrides public function destroy(string \$id) { \$check = UserModel::find(\$id); if (!\$check) { return redirect('/user')->with('error', 'Data user tidak ditemukan'); } try { UserModel::destroy(\$id); return redirect('/user')->with ('success', 'Data user berhasil dihapus'); } catch (\Illuminate\Database\QueryException \$e) { return redirect('/user')->with('error', 'Data user gagal dihapus karena masih terdapat tabel lain yang terikat dengan data ini'); } }</pre>

23.	<p>Selanjutnya kita modifikasi file PWL_POS/resources/views/user/index.blade.php</p> <p>untuk menambahkan tampilan jika ada pesan error</p>	<pre><code>@if (session('success')) <div class="alert alert-success">{{ session('success') }}</div> @endif @if (session('error')) <div class="alert alert-danger">{{ session('error') }}</div> @endif</code></pre>																																																							
24.	<p>Kemudian jalankan browser untuk menghapus salah satu data user. Amati dan laporan!</p>	 <table border="1" data-bbox="780 798 1405 1136"> <thead> <tr> <th>ID</th> <th>Username</th> <th>Nama</th> <th>Level Pengguna</th> <th>Aksi</th> </tr> </thead> <tbody> <tr><td>1</td><td>admin</td><td>Administrator</td><td>Administrator</td><td>[Detail] [Edit] [Hapus]</td></tr> <tr><td>2</td><td>manager</td><td>Manager</td><td>Manager</td><td>[Detail] [Edit] [Hapus]</td></tr> <tr><td>3</td><td>staff</td><td>Staff/Kasir</td><td>Staff/Kasir</td><td>[Detail] [Edit] [Hapus]</td></tr> <tr><td>4</td><td>customer-1</td><td>Pelanggan Pertama</td><td>Customer</td><td>[Detail] [Edit] [Hapus]</td></tr> <tr><td>5</td><td>manager_dua</td><td>Manager 2</td><td>Manager</td><td>[Detail] [Edit] [Hapus]</td></tr> <tr><td>6</td><td>manager22</td><td>Manager Dua Dua</td><td>Manager</td><td>[Detail] [Edit] [Hapus]</td></tr> <tr><td>7</td><td>manager33</td><td>Manager Tiga Tiga</td><td>Manager</td><td>[Detail] [Edit] [Hapus]</td></tr> <tr><td>8</td><td>manager56</td><td>Manager55</td><td>Manager</td><td>[Detail] [Edit] [Hapus]</td></tr> <tr><td>9</td><td>manager12</td><td>Manager11</td><td>Manager</td><td>[Detail] [Edit] [Hapus]</td></tr> <tr><td>10</td><td>test</td><td>test</td><td>Administrator</td><td>[Detail] [Edit] [Hapus]</td></tr> </tbody> </table>	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	Customer	[Detail] [Edit] [Hapus]	5	manager_dua	Manager 2	Manager	[Detail] [Edit] [Hapus]	6	manager22	Manager Dua Dua	Manager	[Detail] [Edit] [Hapus]	7	manager33	Manager Tiga Tiga	Manager	[Detail] [Edit] [Hapus]	8	manager56	Manager55	Manager	[Detail] [Edit] [Hapus]	9	manager12	Manager11	Manager	[Detail] [Edit] [Hapus]	10	test	test	Administrator	[Detail] [Edit] [Hapus]
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	Customer	[Detail] [Edit] [Hapus]																																																					
5	manager_dua	Manager 2	Manager	[Detail] [Edit] [Hapus]																																																					
6	manager22	Manager Dua Dua	Manager	[Detail] [Edit] [Hapus]																																																					
7	manager33	Manager Tiga Tiga	Manager	[Detail] [Edit] [Hapus]																																																					
8	manager56	Manager55	Manager	[Detail] [Edit] [Hapus]																																																					
9	manager12	Manager11	Manager	[Detail] [Edit] [Hapus]																																																					
10	test	test	Administrator	[Detail] [Edit] [Hapus]																																																					
25.	<p>Selamat, kalian sudah membuat Laravel Starter Code untuk proses CRUD dengan menggunakan template AdminLTE dan plugin jQuery Datatables.</p>																																																								

Praktikum 4 – Implementasi Filtering Datatables

No	Steps	Attachments
----	-------	-------------

1.	<p>Kita modifikasi fungsi index() di UserController.php untuk menambahkan data yang ingin dijadikan kategori untuk data filtering</p>	<pre><code>public function index() { \$breadcrumb = (object) ['title' => 'Daftar User', 'list' => ['home', 'User']]; \$page = (object) ['title' => 'Daftar user yang terdaftar dalam sistem']; \$activeMenu = 'user'; \$level = LevelModel::all(); return view('user.index', compact('breadcrumb', 'page', 'activeMenu', 'level')); }</code></pre>
2.	<p>Kemudian kita modifikasi view untuk menampilkan data filtering pada PWL_POS/resources/views/user/index.blade.php</p>	<pre><code><div class="row"> <div class="col-md-12"> <div class="form-group row"> <label class="col-1 control-label col-form-label">Filter : </label> <div class="col-3"> <select name="level_id" id="level_id" class="form-control" 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></code></pre>
3.	<p>Selanjutnya, tetap pada view index.blade.php, kita tambahkan kode berikut pada deklarasi ajax di datatable. Kode ini digunakan untuk mengirimkan data untuk filtering</p>	<pre><code>@push('js') <script> \$(document).ready(function() { var dataUser = \$('#table_user').DataTable({ serverSide: true, // serverSide: true, jika ingin menggunakan server side processing ajax: { "url": "{{ url('user/list') }}", "dataType": "json", "type": "POST", "data": function(d) { d.level_id = \$('#level_id').val(); } }, columns: [data: "DT_RowIndex", // nomor urut dari laravel datatable] }); });</code></pre> 

4.	Kemudian kita edit pada bagian akhir script @push('js') untuk menambahkan listener jika data filtering dipilih	<pre>\$('#level_id').on('change', function() { dataUser.ajax.reload(); });</pre>
5.	Tahapan akhir adalah memodifikasi fungsi list() pada UserController.php yang digunakan untuk menampilkan data pada datatable	<pre>public function list(Request \$request) { \$users = UserModel::select('user_id', 'username', 'nama', 'level_id') ->with(['level']); // dd(\$users); if (\$request->level_id) { \$users->where('level_id', \$request->level_id); } return DataTables::of(\$users) ->addIndexColumn() ->addColumn('aksi', function (\$user) { \$btn = 'user_id) . '" class="btn btn-info btn-sm">Detail'; \$btn .= 'user_id . '/edit') . '" class="btn btn-warning btn-sm">Edit'; \$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>'; }); ->rawColumns(['aksi']); ->make(true); }</pre>

6.

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

The screenshot shows a web application interface titled 'PWL - Starter Code'. On the left, there is a dark sidebar with various menu items: Dashboard, Data Pengguna, Level User, and Data User (which is highlighted in blue). The main content area is titled 'Daftar User' and displays a table of users. The table has columns for ID, Username, Name, Level Pengguna (User Level), and Aksi (Actions). There is one entry shown:

ID	Username	Nama	Level Pengguna	Aksi
1	admin	Administrator	Administrator	Detail Edit Hapus

Below the table, it says 'Showing 1 to 1 of 1 entries (filtered from 11 total entries)'. At the bottom right, there are links for 'Previous' and 'Next'.

This screenshot shows the same application interface as the previous one, but with a different filter applied. The 'Filter' dropdown now shows 'Manager'. The main content area is titled 'Daftar User' and displays a table of users. The table has columns for ID, Username, Name, Level Pengguna (User Level), and Aksi (Actions). There are six entries shown:

ID	Username	Nama	Level Pengguna	Aksi
1	manager	Manager	Manager	Detail Edit Hapus
2	manager_dua	Manager 2	Manager	Detail Edit Hapus
3	manager22	Manager Dua Dua	Manager	Detail Edit Hapus
4	manager33	Manager Tiga Tiga	Manager	Detail Edit Hapus
5	manager56	Manager55	Manager	Detail Edit Hapus
6	manager12	Manager11	Manager	Detail Edit Hapus

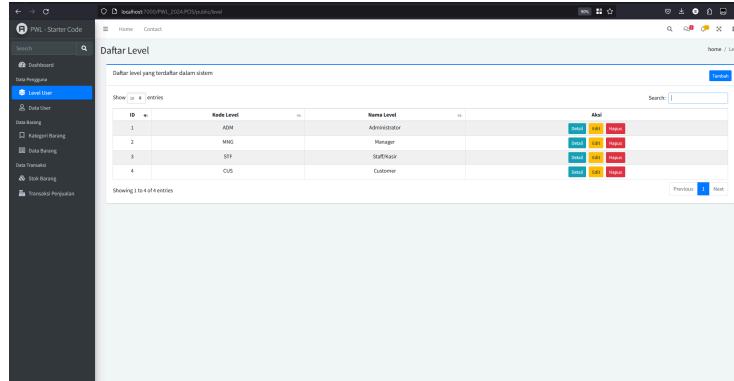
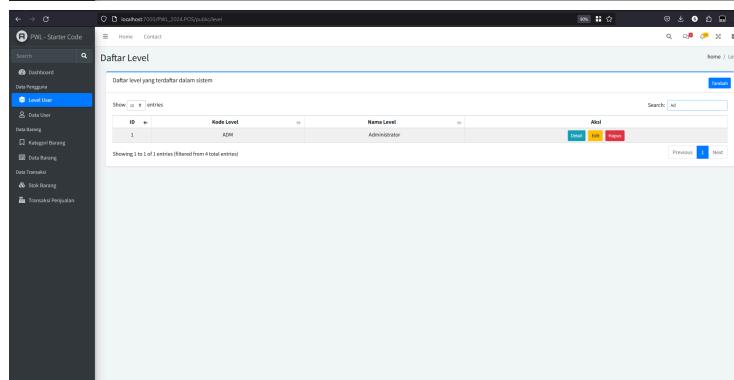
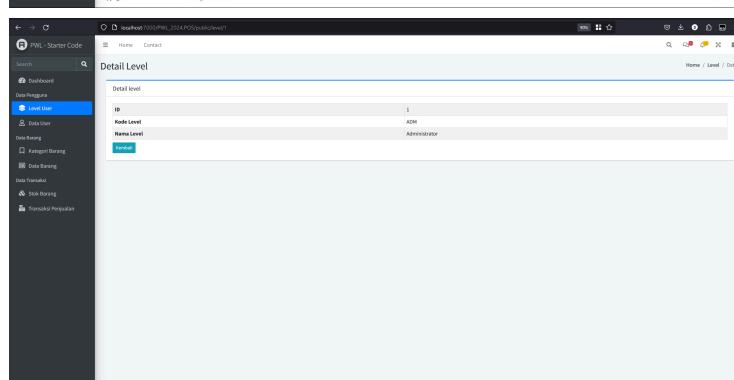
Below the table, it says 'Showing 1 to 6 of 6 entries'. At the bottom right, there are links for 'Previous' and 'Next'.

7.	Selamat, sekarang Laravel Starter Code sudah ada filtering dan searching data. Starter Code sudah bisa digunakan dalam membangun sebuah sistem berbasis website.	
----	--	--

Pertanyaan		
No	Steps	Attachments
1.	Apa perbedaan frontend template dengan backend template?	Perbedaan frontend template dan backend template yang cukup jelas terletak pada fungsionalitas pada template, frontend template menawarkan tampilan yang menarik tanpa adanya fungsionalitas, sedangkan backend template menawarkan keduanya, atau setidaknya menawarkan template fungsionalitas umum pada website.
2.	Apakah layouting itu penting dalam membangun sebuah website?	Layouting tentu penting dalam membangun sebuah website, apalagi dalam skala menengah - besar. Layouting membantu developer dalam menerapkan prinsip DRY (Don't Repeat Yourself) dimana melakukan repetisi pada sebuah blok kode dianggap tidak efisien.
3.	Jelaskan fungsi dari komponen laravel blade berikut : @include(), @extend(), @section(), @push(), @yield(), dan @stack()	<ul style="list-style-type: none"> • @include() : digunakan untuk melakukan include view di dalam view lain. • @extends() : digunakan untuk melakukan spesifikasi layout mana child harus mewarisi struktur. • @section() : digunakan untuk melakukan definisi seksi pada view • @push() : digunakan untuk menambahkan konten pada parent view @stack() directive. • @stack() : mirip dengan @section(), hanya saja dapat ditambah isinya sebanyak yang diperlukan.
4.	Apa fungsi dan tujuan dari variable \$activeMenu ?	Variable \$activeMenu digunakan untuk menyimpan nilai sidebar yang sedang aktif.

Tugas
Implementasikan menu yang belum ada di laravel starter code ini sesuai dengan Studi Kasus Point of Sales Sederhana. Silahkan terapkan kode di laravel starter code untuk menu-menu yang sesuai dengan menu di samping ini.

Jawaban

No	Steps	Attachments
1.	Level User	  

Edit Level

Kode Level: ADM
Nama Level: Administrator (Updated)

Daftar Level

ID	Kode Level	Nama Level	Aksi
1	ADM	Administrator (Updated)	
2	MNG	Manager	
3	STK	Staff/Kasir	
4	CUS	Customer	

Tambah Level

Kode Level: NEW
Nama Level: New Level

Daftar Level

ID	Kode Level	Nama Level	Aksi
1	ADM	Administrator (Updated)	
2	MNG	Manager	
3	STK	Staff/Kasir	
4	CUS	Customer	
5	NEW	New Level	

--	--	--

2. Kategori Barang

Daftar Kategori

Daftar kategori yang terdapat dalam sistem.

ID	Kode Kategori	Nama Kategori	Aksi
1	MNN	Makanan (Updated)	
2	MINN	Makanan	
3	JKT	Jaket	
4	CLN	Celana	
5	TSH	Topi	
6	NAN	Man Stuff	
7	NEW	New Kategori	

Showing 1 to 7 of 7 entries

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

Version 2.2.0

Detail Kategori

Detail Kategori

ID	Kode Kategori	Nama Kategori
1	MNN	Makanan (Updated)

[Edit](#)

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

Version 2.2.0

Tambah Kategori

Tambah kategori baru.

Kode Kategori	NEW
Nama Kategori	New Kategori

[Simpan](#) [Batal](#)

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

Version 2.2.0

Daftar Kategori

Daftar kategori yang terdapat dalam sistem.

ID	Kode Kategori	Nama Kategori	Aksi
1	MNN	Makanan (Updated)	
2	MINN	Makanan	
3	JKT	Jaket	
4	CLN	Celana	
5	TSH	Topi	
6	NAN	Man Stuff	
7	NEW	New Kategori	

Showing 1 to 7 of 7 entries

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

Version 2.2.0

--	--	--

3. Data Barang

--	--	--

--	--	--

--	--	--

Edit Barang

Kategori: Makanan (Updated)

Kode Barang: MMN-002

Nama Barang: Nasi Goreng Jawa

Harga Beli: 10000

Harga Jual: 15000

[Simpan](#) [Batal](#)

Daftar Barang

Daftar barang yang terdapat dalam sistem.

ID	Kategori	Kode Barang	Nama Barang	Harga Beli	Harga Jual	Aksi
1	Makanan (Updated)	MMN-001	Nasi Goreng Jawa	10000	15000	Edit Hapus Rincian
2	Makanan (Updated)	MMN-002	Nasi Goreng	8000	12000	Edit Hapus Rincian
3	Makanan	MMN-003	Eg Tahu	3000	5000	Edit Hapus Rincian
4	Makanan	MMN-004	Eg Jeruk	4000	6000	Edit Hapus Rincian
5	Jaket	JFT-001	Jaket Parasut	60000	100000	Edit Hapus Rincian
6	Jaket	JFT-002	Jaket Kulit	100000	200000	Edit Hapus Rincian
7	Celana	CLN-001	Celana Jeans	100000	150000	Edit Hapus Rincian
8	Celana	CLN-002	Celana Joger	60000	120000	Edit Hapus Rincian
9	Topi	TPH-001	Topi Tracker	30000	50000	Edit Hapus Rincian
10	Topi	TPH-002	Topi Snapback	25000	40000	Edit Hapus Rincian

Showing 1 to 10 of 20 entries

Tambah Barang

Tambah barang baru

Kategori: Makanan (Updated)

Kode Barang: MMN-003

Nama Barang: Pecel

Harga Beli: 4000

Harga Jual: 7000

User: Rian

Stok Jumlah: 1000

[Simpan](#) [Batal](#)

Daftar Barang

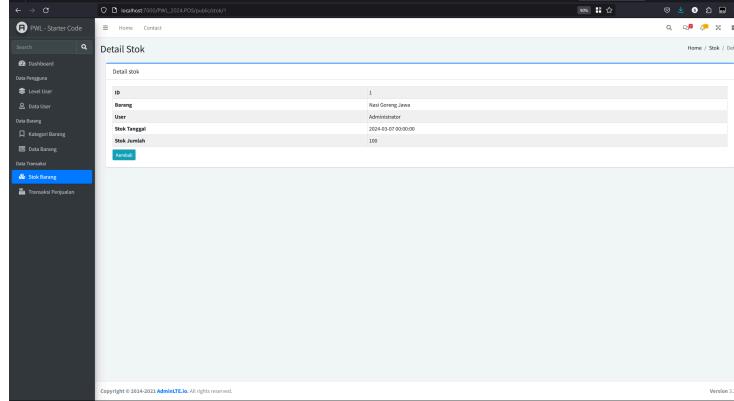
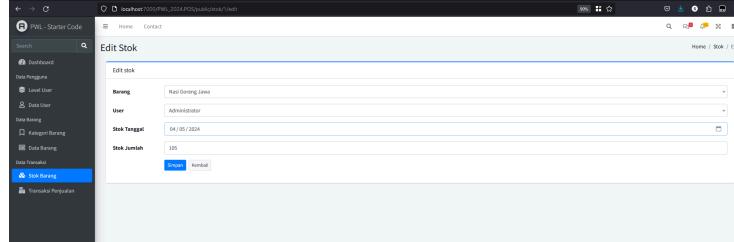
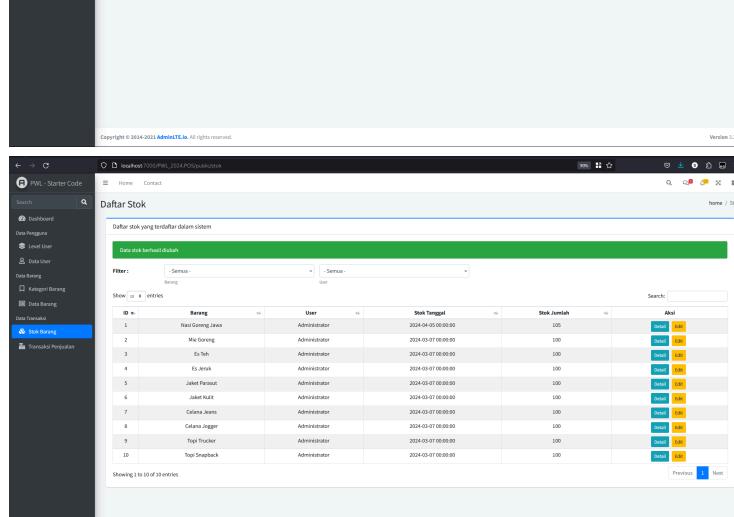
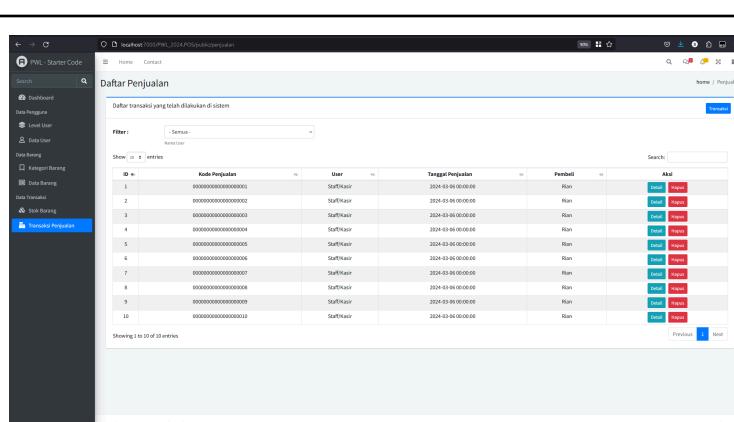
Daftar barang yang terdapat dalam sistem.

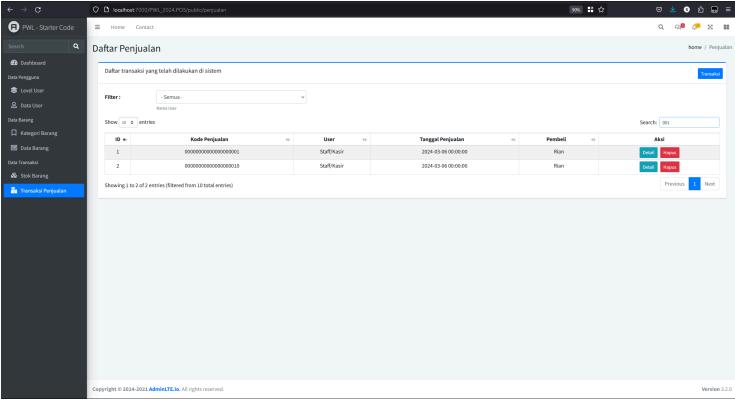
ID	Kategori	Kode Barang	Nama Barang	Harga Beli	Harga Jual	Aksi
11	Makanan (Updated)	MMN-003	Pecel	4000	7000	Edit Hapus Rincian

Showing 11 to 11 of 11 entries

--	--

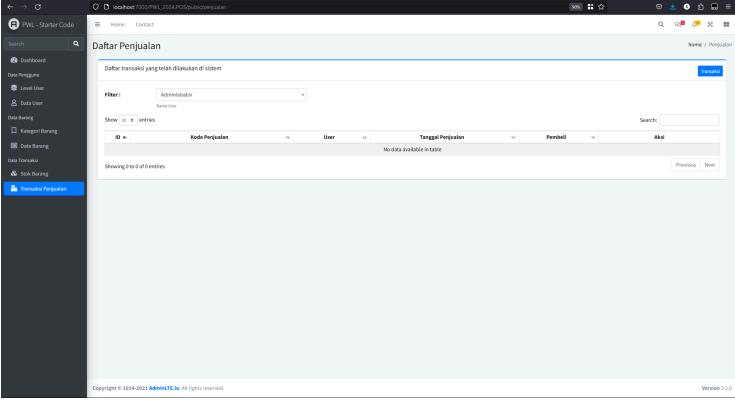
4. Stok Barang

	  
<p>5.</p> <h2>Transaksi Penjualan</h2>	

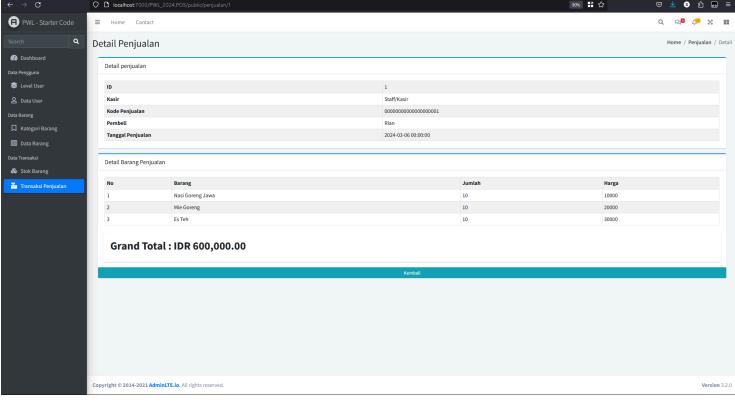


This screenshot shows the 'Daftar Penjualan' (Sales Report) page. It displays a table of sales transactions with columns for ID, Kode Penjualan, User, Tanggal Penjualan, Pembeli, and Aktif. Two entries are listed:

ID	Kode Penjualan	User	Tanggal Penjualan	Pembeli	Aktif
1	000000000000000001	StaffMaster	2024-03-01 00:00:00	Ran	Detail Print
2	000000000000000012	StaffMaster	2024-03-01 00:00:00	Ran	Detail Print



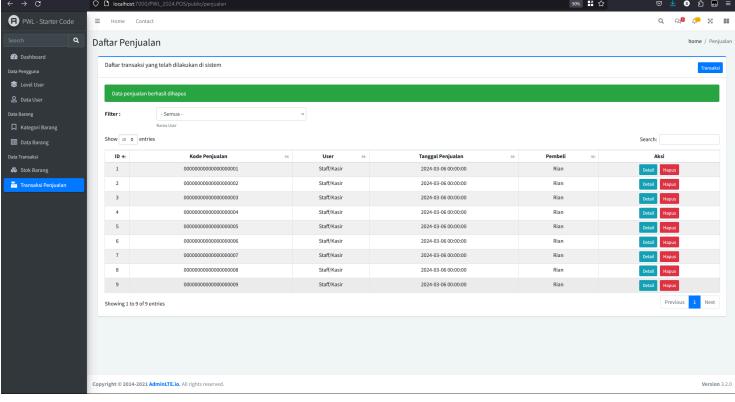
This screenshot shows the same 'Daftar Penjualan' page, but it indicates 'No data available in table'. A message at the bottom says 'Showing 0 to 0 of 0 entries'.



This screenshot shows the 'Detail Penjualan' (Sale Details) page for a single transaction. It includes a summary table and a detailed table of items sold.

No	Barang	Jumlah	Harga
1	Nasi Goreng Jumbo	10	10000
2	Mie Goreng	10	20000
3	Es Teh	10	30000

Grand Total : IDR 600,000.00



This screenshot shows the 'Daftar Penjualan' page again, displaying a list of nine sales entries from March 1st, 2024. The table structure is identical to the first screenshot.

