



### **Jobsheet 9**

Statistika Komputasi - Minggu 11

#### **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

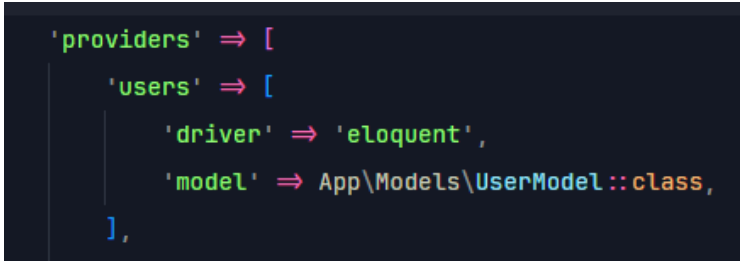
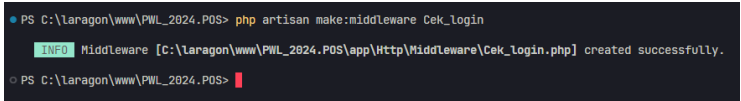
POLITEKNIK NEGERI MALANG

**2024**

## Praktikum – Auth dan Middleware

Praktikum ini lanjutan dari pertemuan ke-4. Tabel yang digunakan untuk praktikum ini menggunakan m\_user. Sebelum melakukan praktikum jalankan perintah di bawah ini

- Start Laragon atau Xampp
- Jalankan “php artisan serve”
- Jalankan “npm run dev”

No	Steps	Attachment
1.	Sebelum melangkah lebih lanjut setting file pada config/auth.php ubah pada bagian model menjadi model yang telah kita buat sebelumnya untuk menyimpan username dan password	 <pre>'providers' =&gt; [     'users' =&gt; [         'driver' =&gt; 'eloquent',         'model' =&gt; App\Models\UserModel::class,     ], ],</pre>
2.	Membuat middleware dengan perintah pada cmd atau terminal  <pre>php artisan make:middleware Cek_login</pre> Setelah itu cek didalam folder app/Http/Middleware apakah sudah ada file bernama Cek_login.php	 <pre>PS C:\laragon\www\PWL_2024.POS&gt; php artisan make:middleware Cek_login INFO   Middleware [C:\laragon\www\PWL_2024.POS\app\Http\Middleware\Cek_login.php] created successfully. PS C:\laragon\www\PWL_2024.POS&gt;</pre>

3.	<p>Kemudian tambahkan kode dibawah ini, letakkan di function handle()</p>	<pre> 0 references   0 implementations class Cek_login {     /**      * Handle an incoming request.      *      * @param \Closure(\Illuminate\Http\Request): (\Symfony\Component\HttpFoundation\Response) \$next      */     0 references   0 overrides     public function handle(Request \$request, Closure \$next, \$roles): Response     {         // cek sudah login atau belum. Jika belum kembali ke halaman login         if ('Auth::check()') {             return redirect('login');         }          // simpan data user pada variabel \$user         \$user = Auth::user();          // jika user memiliki level sesuai pada kolom, lanjutkan request         if (\$user-&gt;level_id == \$roles) {             return \$next(\$request);         }          // jika tidak memiliki akses maka kembalikan ke halaman login         return redirect('login')-&gt;with('error', 'maaf anda tidak memiliki akses');     } } </pre>
4.	<p>Ketika sudah membuat middleware, langkah berikutnya harus registrasikan pada kernel.php. Agar middleware dapat dibaca oleh sistem. Buka folder App/Http/Kernel.php dan tambahkan code pada variabel \$middlewareAliases seperti dibawah</p>	<pre> protected \$middlewareAliases = [     'auth' =&gt; \App\Http\Middleware\Authenticate::class,     'auth.basic' =&gt; \Illuminate\Auth\Middleware\AuthenticateWithBasicAuth::class,     'auth.session' =&gt; \Illuminate\Session\Middleware\AuthenticateSession::class,     'cache.headers' =&gt; \Illuminate\Http\Middleware\SetCacheHeaders::class,     'can' =&gt; \Illuminate\Auth\Middleware\Authorize::class,     'guest' =&gt; \App\Http\Middleware\RedirectIfAuthenticated::class,     'password.confirm' =&gt; \Illuminate\Auth\Middleware\RequirePassword::class,     'precognitive' =&gt; \Illuminate\Foundation\Http\Middleware\HandlePrecognitiveRequests::class,     'signed' =&gt; \App\Http\Middleware\ValidateSignature::class,     'throttle' =&gt; \Illuminate\Routing\Middleware\ThrottleRequests::class,     'verified' =&gt; \Illuminate\Auth\Middleware\EnsureEmailIsVerified::class,     // tambahkan middleware alias dengan nama cek_login     'cek_login' =&gt; \App\Http\Middleware\Cek_login::class, ]; </pre>
5.	<p>Membuat Controller Auth. Pada langkah ini kita akan membuat controller auth, yang dimana di dalamnya terdapat proses autentikasi dari request login. Jalankan perintah Make:controller</p> <pre>php artisan make:controller AuthController</pre> <p>Buka file App/Http/Controllers/AuthController.php yang sudah kita buat. Lalu kita akan isi perintah code untuk membuat proses autentikasi, verifikasi, dan logout</p>	<pre> PS C:\laragon\www\PWL_2024.POS&gt; php artisan make:controller AuthController INFO Controller [C:\laragon\www\PWL_2024.POS\app\Http\Controllers\AuthController.php] created successfully. PS C:\laragon\www\PWL_2024.POS&gt; </pre>


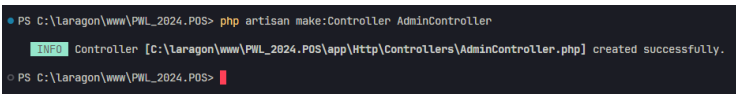

6.


Kemudian buat kode AuthController.php seperti dibawah ini

```

1 <?php
2
3 namespace App\Http\Controllers;
4
5 use App\Models\UserModel;
6 use Dotenv\Validator;
7 use Illuminate\Contracts\Validation\Validator as ValidationValidator;
8 use Illuminate\Http\Request;
9 use Illuminate\Support\Facades\Auth;
10 use Illuminate\Support\Facades\Hash;
11 use Illuminate\Support\Facades\Validator as FacadesValidator;
12 use Illuminate\Validation\Validator as IlluminateValidationValidator;
13
14 class AuthController extends Controller
15 {
16     public function index()
17     {
18         // jika masih data user yang ada variabel $user
19         $user = Auth::user();
20
21         // kondisi jika pengguna ada
22         if ($user) {
23             // jika pengguna memiliki level admin
24             if ($user->level_id == '1') {
25                 return redirect()->intended('admin');
26             }
27             // jika pengguna memiliki level manager
28             atau if ($user->level_id == '2') {
29                 return redirect()->intended('manager');
30             }
31         }
32         return view('login');
33     }
34
35     public function proses_login(Request $request)
36     {
37         // kita buat validasi pada saat login dihalaman
38         // validasi username & password wajib diisi
39         $request->validate([
40             'username' => 'required',
41             'password' => 'required',
42         ]);
43
44         // ambil data request username & password saja
45         $credential = $request->only('username', 'password');
46         // cek jika data username dan password valid (sesuai) dengan data
47         if (Auth::attempt($credential)) {
48             // kalau berhasil akan ada user yang di variabel $user
49             $user = Auth::user();
50
51             // cek lagi jika level user admin maka arahkan ke halaman admin
52             if ($user->level_id == '1') {
53                 // dd($user->level_id);
54                 return redirect()->intended('admin');
55             }
56             // tapi jika level user manager atau biasa maka arahkan ke halaman user
57             atau if ($user->level_id == '2') {
58                 return redirect()->intended('manager');
59             }
60             return redirect()->intended('/');
61         }
62         // jika ya ada data user yang valid maka kembalikan lagi ke halaman login
63         // tambahkan pesan error jika login gagal ya
64         return redirect('login')
65             ->withInput()
66             ->withErrors(['login_gagal' => 'Pastikan kembali username dan password yang dimasukkan sudah benar!']);
67     }
68
69     public function register()
70     {
71         // tampilkan view register
72         return view('register');
73     }
74
75     // proses form register
76     public function proses_register(Request $request)
77     {
78         // kita buat validasi saat proses register
79         // validasi username & password wajib diisi
80         // validasi username itu harus unik atau tidak boleh duplicate username ya
81         $validator = FacadesValidator::make($request->all(), [
82             'name' => 'required',
83             'username' => 'required|unique:users',
84             'password' => 'required',
85         ]);
86
87         // kalau gagal kembali ke halaman register dengan pesan error
88         if ($validator->fails()) {
89             return redirect('/register')
90                 ->withErrors($validator)
91                 ->withInput();
92         }
93
94         // kalau berhasil isi level & hash passwordnya ya biar secure
95         $request['level_id'] = '2';
96         $request['password'] = Hash::make($request->password);
97
98         // masukkan semua data pada request ke table user
99         UserModel::create($request->all());
100
101         // kalau berhasil arahkan ke halaman login
102         return redirect()->route('login');
103     }
104
105     public function logout(Request $request)
106     {
107         // logout itu harus menghapus sessionnya
108         $request->session()->flush();
109
110         // lanjutkan form logout pada auth
111         Auth::logout();
112
113         // kembalikan ke halaman login
114         return Redirect('login');
115     }
116 }

```

7.	Kemudian tambahkan kode pada route pada file routes/web.php seperti kode program dibawah	 <pre> 1 Route::get('login', [AuthController::class, 'index'])-&gt;name('login'); 2 Route::get('register', [AuthController::class, 'register'])-&gt;name('register'); 3 Route::post('proses_login', [AuthController::class, 'proses_login'])-&gt;name('proses_login'); 4 Route::get('logout', [AuthController::class, 'logout'])-&gt;name('logout'); 5 Route::post('proses_register', [AuthController::class, 'proses_register'])-&gt;name('proses_register'); 6 7 // kita atur juga untuk middleware menggunakan group pada routing 8 // di dalamnya terdapat group untuk mengecek kondisi login 9 // jika user yang login merupakan admin maka akan diarahkan ke AdminController 10 // jika user yang login merupakan manager maka akan diarahkan ke UserController 11 12 Route::group(['middleware' =&gt; ['auth']], function () { 13     Route::group(['middleware' =&gt; ['cek_login:1']], function () { 14         Route::resource('admin', AdminController::class); 15     }); 16     Route::group(['middleware' =&gt; ['cek_login:2']], function () { 17         Route::resource('manager', ManagerController::class); 18     }); 19 }); </pre>
8.	Kemudian buat file controller dengan nama AdminController sebagai halaman yang boleh diakses oleh admin saja. Jalankan perintah dibawah php artisan make:controller AdminController	 <pre> PS C:\laragon\www\PWL_2024.POS&gt; php artisan make:controller AdminController INFO Controller [C:\laragon\www\PWL_2024.POS\app\Http\Controllers\AdminController.php] created successfully. PS C:\laragon\www\PWL_2024.POS&gt; </pre>
9.	Buka folder App/Http/Controllers dan isi file AdminController seperti dibawah	 <pre> 1 &lt;?php 2 3 namespace App\Http\Controllers; 4 5 use Illuminate\Http\Request; 6 7 class AdminController extends Controller 8 { 9     public function index() 10     { 11         return view('admin'); 12     } 13 } </pre>
10.	Kemudian buat file controller dengan nama ManagerController sebagai	 <pre> PS C:\laragon\www\PWL_2024.POS&gt; php artisan make:controller ManagerController INFO Controller [C:\laragon\www\PWL_2024.POS\app\Http\Controllers\ManagerController.php] created successfully. PS C:\laragon\www\PWL_2024.POS&gt; </pre>

	halaman yang boleh diakses oleh admin saja. Jalankan perintah dibawah php artisan make:controller ManagerController	
11.	Buka folder App/Http/Controllers dan isi file ManagerController seperti dibawah	
12.	Pada langkah terakhir yaitu membuat layout sederhana	

13.

Buatlah halaman login dengan membuka folder resources/views dengan nama file login.blade.php. Lalu kita isi filenya seperti contoh dibawah

```

1 @extends('adminlte::auth.auth-page', ['auth_type' => 'login'])
2
3 @section('adminlte_css_pre')
4 <link rel="stylesheet" href="{{ asset('vendor/checkbox-bootstrap/checkbox-bootstrap.min.css') }}">
5 @stop
6
7 @php( $login_url = View::getSection('login_url') ?? config('adminlte.login_url', 'login') )
8 @php( $register_url = View::getSection('register_url') ?? config('adminlte.register_url', 'register') )
9 @php( $password_reset_url = View::getSection('password_reset_url') ?? config('adminlte.password_reset_url', 'password/reset') )
10
11 @if (config('adminlte.use_route_url', false))
12 @php( $login_url = $login_url ? route($login_url) : '' )
13 @php( $register_url = $register_url ? route($register_url) : '' )
14 @php( $password_reset_url = $password_reset_url ? route($password_reset_url) : '' )
15 @else
16 @php( $login_url = $login_url ? url($login_url) : '' )
17 @php( $register_url = $register_url ? url($register_url) : '' )
18 @php( $password_reset_url = $password_reset_url ? url($password_reset_url) : '' )
19 @endif
20
21 @section('auth_header', __('adminlte::adminlte.login_message'))
22
23 @section('auth_body')
24 @error('login_gagal')
25 <div class="alert alert-warning alert-dismissible fade show" role="alert">
26 <span class="alert-inner--text"><strong>Warning!</strong> {{ $message }} </span>
27 <button type="button" class="close" data-dismiss="alert" aria-label="Close">
28 <span aria-hidden="true"><times></span>
29 </button>
30 </div>
31 @enderror
32 <form action="{{ url('process_login') }}" method="post">
33 @csrf
34
35 {{-- Email field --}}
36 <div class="input-group mb-3">
37 <input type="text" name="username" class="form-control" @error('username') is-invalid @enderror
38 value="{{ old('username') }}" placeholder="Username" autofocus>
39
40 <div class="input-group-append">
41 <div class="input-group-text">
42 <span class="fas fa-envelope" {{ config('adminlte.classes_auth_icon', '') }}></span>
43 </div>
44 </div>
45
46 @error('username')
47 <span class="invalid-feedback" role="alert">
48 <strong>{{ $message }}</strong>
49 </span>
50 @enderror
51 </div>
52
53 {{-- Password field --}}
54 <div class="input-group mb-3">
55 <input type="password" name="password" class="form-control" @error('password') is-invalid @enderror
56 placeholder="{{ __('adminlte::adminlte.password') }}">
57
58 <div class="input-group-append">
59 <div class="input-group-text">
60 <span class="fas fa-lock" {{ config('adminlte.classes_auth_icon', '') }}></span>
61 </div>
62 </div>
63
64 @error('password')
65 <span class="invalid-feedback" role="alert">
66 <strong>{{ $message }}</strong>
67 </span>
68 @enderror
69 </div>
70
71 {{-- Login field --}}
72 <div class="row">
73 <div class="col-7">
74 <div class="checkbox-primary" title="{{ __('adminlte::adminlte.remember_me_hint') }}">
75 <input type="checkbox" name="remember" id="remember" {{ old('remember') ? 'checked' : '' }}>
76
77 <label for="remember">
78 {{ __('adminlte::adminlte.remember_me') }}
79 </label>
80 </div>
81 </div>
82
83 <div class="col-5">
84 <button type="submit" class="btn btn-block" {{ config('adminlte.classes_auth_btn', 'btn-flat btn-primary') }}">
85 <span class="fas fa-sign-in-alt"></span>
86 {{ __('adminlte::adminlte.sign_in') }}
87 </button>
88 </div>
89 </div>
90 </form>
91 @stop
92
93 @section('auth_footer')
94 @if ($register_url)
95 <p class="my-0">
96 <a href="{{ route('register') }}">
97 {{ __('adminlte::adminlte.register_a_new_membership') }}
98 </a>
99 </p>
100 @endif
101 @stop

```

14.



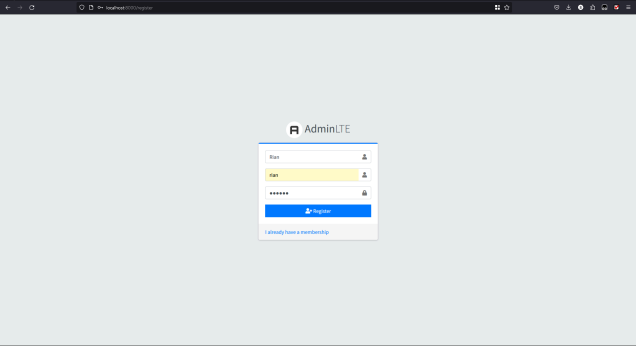
Buatlah halaman register dengan membuka folder resources/views dengan nama file register.blade.php. Lalu kita isi filenya seperti contoh dibawah

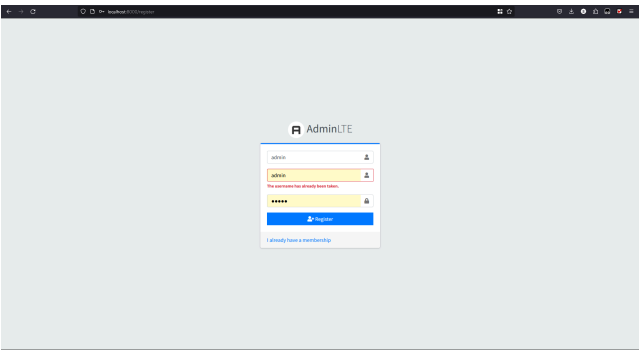
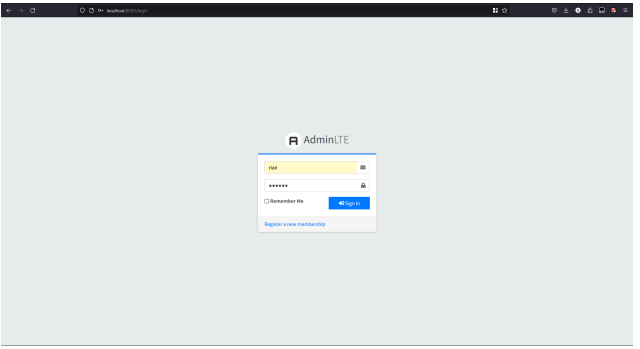
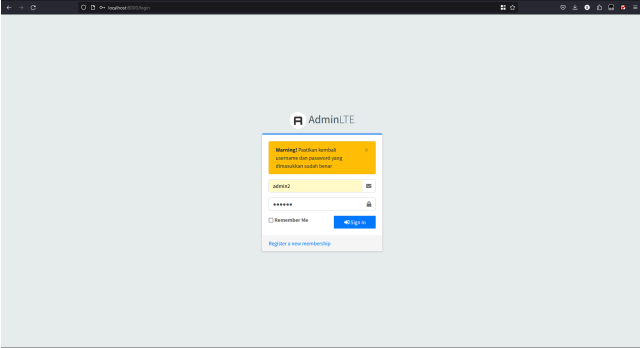
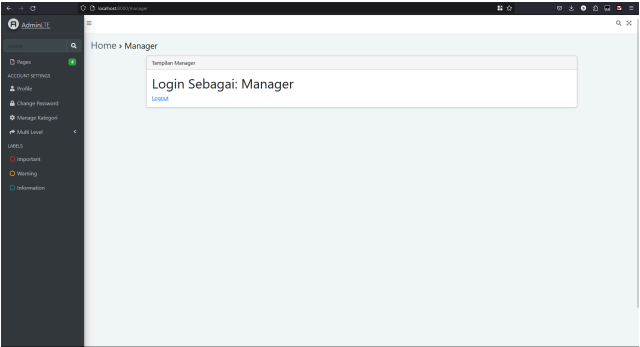
```

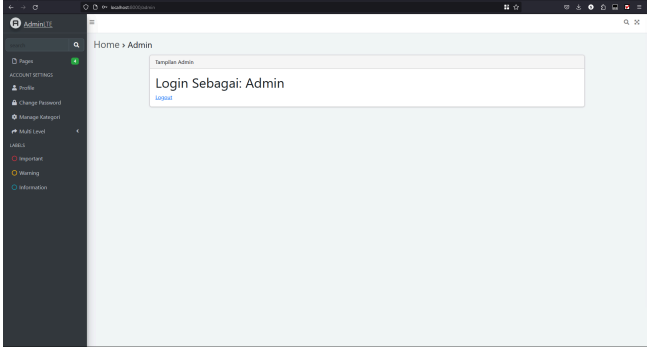
1 @extends('adminlte::auth.auth-page', ['auth type' => 'register'])
2
3 @section('adminlte_css_pre')
4 <link rel="stylesheet" href="{{ asset('vendor/checkbox-bootstrap/checkbox-bootstrap.min.css') }}">
5 @stop
6
7 @php( $login_url = View::getSection('login_url') ?? config('adminlte.login_url', 'login') )
8 @php( $register_url = View::getSection('register_url') ?? config('adminlte.register_url', 'register') )
9
10 @if (config('adminlte.use_route_url', false))
11 @php( $login_url = $login_url ? route($login_url) : '' )
12 @php( $register_url = $register_url ? route($register_url) : '' )
13 @else
14 @php( $login_url = $login_url ? url($login_url) : '' )
15 @php( $register_url = $register_url ? url($register_url) : '' )
16 @endif
17
18 @section('auth-header', __('adminlte::adminlte.register_message'))
19
20 @section('auth-body')
21 <form action="{{ url('process_register') }}" method="post">
22 @csrf
23
24 {{-- Name field --}}
25 <div class="input-group mb-3">
26 <input type="text" name="name" class="form-control @error('name') is-invalid @enderror"
27 value="{{ old('name') }}" placeholder="Name" autofocus>
28
29 <div class="input-group-append">
30 <div class="input-group-text">
31 <span class="fas fa-user {{ config('adminlte.classes_auth_icon', '') }}"></span>
32 </div>
33 </div>
34
35 @error('name')
36 <span class="invalid-feedback" role="alert">
37 <strong>{{ $message }}</strong>
38 </span>
39 @enderror
40 </div>
41
42 {{-- Username field --}}
43 <div class="input-group mb-3">
44 <input type="text" name="username" class="form-control @error('username') is-invalid @enderror"
45 value="{{ old('username') }}" placeholder="Username">
46
47 <div class="input-group-append">
48 <div class="input-group-text">
49 <span class="fas fa-user {{ config('adminlte.classes_auth_icon', '') }}"></span>
50 </div>
51 </div>
52
53 @error('username')
54 <span class="invalid-feedback" role="alert">
55 <strong>{{ $message }}</strong>
56 </span>
57 @enderror
58 </div>
59
60 {{-- Password field --}}
61 <div class="input-group mb-3">
62 <input type="password" name="password" class="form-control @error('password') is-invalid @enderror"
63 placeholder="{{ __('adminlte::adminlte.password') }}">
64
65 <div class="input-group-append">
66 <div class="input-group-text">
67 <span class="fas fa-lock {{ config('adminlte.classes_auth_icon', '') }}"></span>
68 </div>
69 </div>
70
71 @error('password')
72 <span class="invalid-feedback" role="alert">
73 <strong>{{ $message }}</strong>
74 </span>
75 @enderror
76 </div>
77
78 {{-- Register button --}}
79 <button type="submit" class="btn btn-block {{ config('adminlte.classes_auth_btn', 'btn-flat btn-primary') }}">
80 <span class="fas fa-user-plus"></span>
81 {{ __('adminlte::adminlte.register') }}
82 </button>
83 </form>
84 @stop
85
86 @section('auth-footer')
87 <p class="my-0">
88 <a href="{{ route('login') }}">
89 {{ __('adminlte::adminlte.i_already_have_a_membership') }}
90 </a>
91 </p>
92 @stop

```



15.	<p>Buatlah halaman register dengan membuka folder resources/views dengan nama file admin.blade.php. Lalu kita isi filenya seperti contoh dibawah</p>	 <pre> 1 @extends('layouts.app') 2 3 {{-- Customize layout sections --}} 4 5 @section('subtitle', 'Admin') 6 @section('content_header_title', 'Home') 7 @section('content_header_subtitle', 'Admin') 8 9 @section('content') 10 &lt;div class="container"&gt; 11   &lt;div class="card"&gt; 12     &lt;div class="card-header"&gt;Tampilkan &lt;?php echo (Auth::user()-&gt;level_id == 1) ? 'Admin' : 'Manager'; ?&gt;&lt;/div&gt; 13     &lt;div class="card-body"&gt; 14       &lt;h1&gt;Login Sebagai: 15       &lt;?php echo (Auth::user()-&gt;level_id == 1) ? 'Admin' : 'Manager'; ?&gt; 16       &lt;/h1&gt; 17       &lt;a href="{{ route('logout') }}"&gt;Logout&lt;/a&gt; 18     &lt;/div&gt; 19   &lt;/div&gt; 20 &lt;/div&gt; 21 @endsection 22 23 @push('js') 24 @endpush </pre>
16.	<p>Buatlah halaman register dengan membuka folder resources/views dengan nama file manager.blade.php. Lalu kita isi filenya seperti contoh dibawah</p>	 <pre> 1 @extends('layouts.app') 2 3 {{-- Customize layout sections --}} 4 5 @section('subtitle', 'Manager') 6 @section('content_header_title', 'Home') 7 @section('content_header_subtitle', 'Manager') 8 9 @section('content') 10 &lt;div class="container"&gt; 11   &lt;div class="card"&gt; 12     &lt;div class="card-header"&gt;Tampilkan &lt;?php echo (Auth::user()-&gt;level_id == 1) ? 'Admin' : 'Manager'; ?&gt;&lt;/div&gt; 13     &lt;div class="card-body"&gt; 14       &lt;h1&gt;Login Sebagai: 15       &lt;?php echo (Auth::user()-&gt;level_id == 1) ? 'Admin' : 'Manager'; ?&gt; 16       &lt;/h1&gt; 17       &lt;a href="{{ route('logout') }}"&gt;Logout&lt;/a&gt; 18     &lt;/div&gt; 19   &lt;/div&gt; 20 &lt;/div&gt; 21 @endsection 22 23 @push('js') 24 @endpush </pre>
17.	<p>Jalankan kode diatas dan buat laporan dengan isi screenshoot hasil dan beri penjelasan pada setiap screenshoot hasil dan commit perubahan pada git.</p>	<ul style="list-style-type: none"> <li>• /register</li> </ul>  <ul style="list-style-type: none"> <li>• /register (username taken)</li> </ul>

		<div><p>A screenshot of the AdminLTE login page. The login form has fields for username (filled with 'admin'), password (filled with 'admin'), and a 'Remember Me' checkbox. The 'Login' button is highlighted in blue. Below the button is a link that says 'I already have a membership'.</p></div> <ul style="list-style-type: none"><li>• /login</li></ul> <div><p>A screenshot of the AdminLTE login page, identical to the previous one, showing the login form with 'admin' as the username and password, and the 'Login' button.</p></div> <ul style="list-style-type: none"><li>• /login (username / password salah)</li></ul> <div><p>A screenshot of the AdminLTE login page. A yellow error message box is displayed at the top, stating: 'Warning: Username dan password yang dimasukkan salah!' (Warning: Username and password entered are wrong!). The login form below shows 'admin' as the username and '*****' as the password.</p></div> <ul style="list-style-type: none"><li>• /manager</li></ul> <div><p>A screenshot of the AdminLTE manager page. The left sidebar shows the 'AdminLTE' logo and a list of menu items: Home, Pages, Account Settings, Profile, Change Password, Manage Roles, Add User, Roles, Requests, Working, and Information. The main content area is titled 'Home &gt; Manager' and contains a box with the text 'Login Sebagai: Manager' and a 'Login' link.</p></div> <ul style="list-style-type: none"><li>• /admin</li></ul>
--	--	-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------

		<div data-bbox="784 212 1427 556"></div> <div data-bbox="685 598 956 630"><p><b>Hasil test middleware</b></p></div> <div data-bbox="735 632 1396 730"><ul style="list-style-type: none"><li>• User selain level_id 1 tidak dapat mengakses /admin</li><li>• User selain level_id 2 tidak dapat mengakses /manager</li></ul></div>
--	--	---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------