



Nama : Dea Putri Nastiti

NIM : 2241720117

Kelas : TI 2H

JOBSHEET 9

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”

```
PS C:\laragon\www\PWL_POS> php artisan serve  
  
INFO Server running on [http://127.0.0.1:8000].  
  
Press Ctrl+C to stop the server
```

- Jalankan “npm run dev”

```
> vite  
Port 5173 is in use, trying another one...  
  
VITE v5.1.6 ready in 692 ms  
→ Local: http://localhost:5174/  
→ Network: use --host to expose  
→ press h + enter to show help  
  
LARAVEL v10.47.0 plugin v1.0.2  
→ APP URL: http://localhost
```

Jika 3 tahap diatas sudah dilakukan kemudian lakukan langkah-langkah praktikum di bawah ini

- 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

```
config > auth.php  
3 return [  
62     'providers' => [  
63         'users' => [  
64             'driver' => 'eloquent',  
65             'model' => App\Models\UserModel::class,  
66         ],  
    ],
```



2. Membuat middleware dengan perintah pada cmd atau terminal

```
PS C:\laragon\www\PWL_POS> php artisan make:middleware Cek_Login
PS C:\laragon\www\PWL_POS> php artisan make:middleware Cek_Login

INFO Middleware [C:\laragon\www\PWL_POS\app\Http\Middleware\Cek_Login.php] created successfully.
```

Setelah itu cek didalam folder app/Http/Middleware apakah sudah ada file bernama **Cek_login.php**

3. Kemudian tambahkan kode dibawah ini, letakkan di function handle()

```
app > Http > Middleware > Cek_Login.php > ...
1  <?php
2
3  namespace App\Http\Middleware;
4
5  use Closure;
6  use Illuminate\Http\Request;
7  use Illuminate\Support\Facades\Auth;
8  use Symfony\Component\HttpFoundation\Response;
9
10 class Cek_Login
11 {
12     /**
13      * Handle an incoming request.
14      *
15      * @param  \Closure(\Illuminate\Http\Request): (\Symfony\Component\HttpFoundation\Response)  $next
16      * @param  string  $roles  // Assumed roles is a string or similar. You may need to adjust based on actual use
17      */
18     public function handle(Request $request, Closure $next, $roles): Response
19     {
20         // Cek sudah login atau belum. Jika belum, kembali ke halaman login.
21         if (!Auth::check()) {
22             return redirect('login');
23         }
24
25         // Simpan data user pada variabel $user
26         $user = Auth::user();
27
28         // Jika user memiliki level sesuai pada kolom, lanjutkan request
29         if ($user->level_id == $roles) {
30             return $next($request);
31         }
32
33         // Jika tidak memiliki akses maka kembalikan ke halaman login
34         return redirect('login')->with('error', 'Maaf Anda tidak memiliki akses');
35     }
36 }
```

4. 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

```
app > Http > Kernel.php > Kernel > $middlewareAliases
7  class Kernel extends HttpKernel
55     protected $middlewareAliases = [
56         'verified' => \Illuminate\Auth\Middleware\EnsureEmailIsVerified::class,
57         //tambahkan middleware alias dengan nama cek_login
58         'cek_login' => \App\Http\Middleware\Cek_Login::class,
59     ];
```

5. 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

```
PS C:\laragon\www\PWL_POS> php artisan make:controller AuthController

INFO Controller [C:\laragon\www\PWL_POS\app\Http\Controllers\AuthController.php] created successfully.
```

Buka file **App/Http/Controllers/AuthController.php** yang sudah kita buat. Lalu kita akan isi perintah code untuk membuat proses autentikasi , verifikasi, dan logout



6. Kemudian buat kode AuthController.php seperti dibawah ini

```
app > http > Controllers > AuthController.php > AuthController > logout
1 <?php
2
3 namespace App\Http\Controllers;
4
5 use App\Models\UserModel;
6 use Illuminate\Http\Request;
7 use Illuminate\Support\Facades\Auth;
8 use Illuminate\Support\Facades\Hash;
9 use Illuminate\Support\Facades\Validator;
10
11 class AuthController extends Controller
12 {
13     public function index(){
14         //kita ambil data user lalu simpan pada variable user
15         $user = Auth::user();
16
17         //kondisi jika usernya ada
18         if($user){
19             //jika user memiliki level admin
20             if($user->level_id == '1'){
21                 return redirect()->intended('admin');
22             }
23             //jika user nya memiliki level manager
24             else if($user->level_id == '2'){
25                 return redirect()->intended('manager');
26             }
27         }
28         return view('login');
29     }
30 }
```

```
31 ~ public function proses_login(Request $request){
32     //kita buat validasi pada saat tombol login di klik
33     //validasi nya username dan password wajib di isi
34     $request->validate([
35         'username' => 'required',
36         'password' => 'required',
37     ]);
38
39     //ambil data request username dan password
40     $credential = $request->only('username', 'password');
41     //cek jika data username dan password valid (sesuai) dengan data
42     if(Auth::attempt($credential)){
43         //kalau berhasil simpan data user admin maka diarahkan ke halaman admin
44         $user = Auth::user();
45
46         //cek lagi jika level user admin maka arahkan ke halaman admin
47         if($user->level_id == '1'){
48             //dd($user->level_id);
49             return redirect()->intended('admin');
50         }
51         //tapi jika level user nya user biasa maka arahkan ke halaman user
52         else if($user->level_id == '2'){
53             return redirect()->intended('manager');
54         }
55         //jika belum ada role maka ke halaman
56         return redirect()->intended('/');
```

```
57     return redirect()->intended('/');
58
59     //jika ga ada data user yang valid maka kembalikan lagi ke halaman login
60     //tampilkan pesan error juga kalau gagal login
61     return redirect('login');
62     withInput();
63     -withErrors(['login_gagal' => 'Pastikan kembali username dan password yang dimasukkan sudah benar']);
64 }
65
66 public function register(){
67     //tampilkan view register
68     return view('register');
69 }
70
71 //aksi form register
72 public function proses_register(Request $request){
73     //kita buat validasi aksi form proses register
74     //validasi nya semua field wajib diisi
75     //validasi username dan password agar tidak ada duplicate username
76     $validator = Validator::make($request->all(), [
77         'nama' => 'required',
78         'username' => 'required|unique:users',
79         'password' => 'required'
80     ]);
81
82     //kalau gagal kembalikan ke halaman register dengan menampilkan pesan error
83     if($validator->fails()){
84         return redirect('/register')
85             -withErrors($validator)
86             -withInput();
87     }
88     //kalau berhasil isi level $ hash passwordnya ya biar secure
89     $request['level_id'] = '2';
90     $request['password'] = Hash::make($request->password);
91
92     //simpankan semua data pada request ke table user
93     UserModel::create($request->all());
94
95     //jika berhasil arahkan ke halaman login
96     return redirect()->route('login');
97 }
98 }
```



```
99
100 public function logout(Request $request){
101     //logout itu harus menghapus session nya
102     $request->session()->flush();
103
104     //jalan kan fungsi logout pada auth
105     Auth::logout();
106     //kembali ke halaman login
107     return redirect('login');
108 }
109 }
```

7. Kemudian tambahkan kode pada route pada file routes/web.php seperti kode program dibawah

```
routes > web.php > Closure > Closure
132 Route::get('login', [AuthController::class, 'index'])->name('login');
133 Route::get('register', [AuthController::class, 'register'])->name('register');
134 Route::post('proses_login', [AuthController::class, 'proses_login'])->name('proses_login');
135 Route::get('logout', [AuthController::class, 'logout'])->name('logout');
136 Route::post('proses_register', [AuthController::class, 'proses_register'])->name('proses_register');
137
138 // kita atur juga untuk middleware menggunakan group pada routing
139 // didalamnya terdapat group untuk mengecek kondisi login
140 // jika user yang login merupakan admin maka akan diarahkan ke AdminController
141 // jika user yang login merupakan manager maka akan diarahkan ke UserController
142 Route::group(['middleware' => ['auth']], function () {
143     Route::group(['middleware' => ['cek_login:1']], function () {
144         Route::resource('admin', AdminController::class);
145     });
146     Route::group(['middleware' => ['cek_login:2']], function () {
147         Route::resource('manager', ManagerController::class);
148     });
149 });
```

8. Kemudian buat file controller dengan nama AdminController sebagai halaman yang boleh diakses oleh admin saja. Jalankan perintah dibawah

```
PS C:\laragon\www\PWL_POS> php artisan make:controller AdminController

INFO Controller [C:\laragon\www\PWL_POS\app\Http\Controllers\AdminController.php] created successfully.
```

9. Buka folder App/Http/Controllers dan isi file AdminController seperti dibawah

```
app > Http > Controllers > AdminController.php > AdminController > index
1 <?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         return view('admin');
11     }
12 }
```

10. Kemudian buat file controller dengan nama ManagerController sebagai halaman yang boleh diakses oleh admin saja. Jalankan perintah dibawah

```
INFO Controller [C:\laragon\www\PWL_POS\app\Http\Controllers\ManagerController.php] created successfully.

PS C:\laragon\www\PWL_POS> php artisan make:controller ManagerController
```



11. Buka folder App/Http/Controllers dan isi file ManagerController seperti dibawah

```
app > Http > Controllers > ManagerController.php > ManagerController > index
1  <?php
2
3  namespace App\Http\Controllers;
4
5  use Illuminate\Http\Request;
6
7  class ManagerController extends Controller
8  {
9      public function index(){
10         return view('manager');
11     }
12 }
```

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

```
resources > views > login.blade.php > ...
1  @extends('adminlte:auth-page', ['auth_type' => 'login'])
2
3  @section('adminlte_css_pre')
4  <link rel="stylesheet" href="{{ asset('vendor/icheck-bootstrap/icheck-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($password_reset_url = View::getSection('password_reset_url') ?? config('adminlte.password_reset_url', 'password/reset'))
10
11
12  @if (config('adminlte.use_route_url', false))
13      @php($login_url = $login_url ? route($login_url) : '')
14      @php($register_url = $register_url ? route($register_url) : '')
15      @php($password_reset_url = $password_reset_url ? route($password_reset_url) : '')
16  @else
17      @php($login_url = $login_url ? url($login_url) : '')
18      @php($register_url = $register_url ? url($register_url) : '')
19      @php($password_reset_url = $password_reset_url ? url($password_reset_url) : '')
20  @endif
21
22  @section('auth_header', __('adminlte:adminlte.login_message'))
23
24  @section('auth_body')
25      @error('login_gagal')
26      <div class="alert alert-warning alert-dismissible fade show" role="alert">
27          <span class="alert-inner-text"><strong>Warning</strong> {{ $message }}</span>
28          <button type="button" class="close" data-dismiss="alert" aria-label="Close">
29              <span aria-hidden="true">&times;</span></button>
30      </div>
31      @enderror
32
33  <form action="{{ url('proses_login') }}" method="post">
34      @csrf
35      <div class="input-group mb-3">
36          <input type="text" class="form-control" placeholder="Email field" />
37          <input type="password" class="form-control" placeholder="Password field" />
38          <button type="submit" class="btn btn-primary" value="Login"> Login </button>
39      </div>
40  </form>
```



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>

```
38 <input type="text" name="username" class="form-control @error('username') is-invalid @enderror"
39 value="{{ old('username') }}" placeholder="Username autofocus"
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 @error('username')
46 <span class="invalid-feedback" role="alert">
47 <strong>{{ $message }}</strong>
48 </span>
49 @enderror
50 </div>
51
52 {{-- Password field --}}
53 <div class="input-group mb-3">
54 <input type="password" name="password"
55 class="form-control @error('password') is-invalid @enderror"
56 placeholder="{{ __('adminlte:adminlte.password') }}"
57 <div class="input-group-append">
58 <div class="input-group-text">
59 <span class="fas fa-lock {{ config('adminlte.classes_auth_icon', '') }}"></span>
60 </div>
61 </div>
62 @error('password')
63 <span class="invalid-feedback" role="alert">
64 <strong>{{ $message }}</strong>
65 </span>
66 @enderror
67 </div>
68
69 {{-- Login field --}}
70 <div class="row">
71 <div class="col-7">
72 <div class="checkbox-primary" title="{{ __('adminlte:adminlte.remember_me_hint') }}">
73 <input type="checkbox" name="remember" id="remember" {{ old('remember') ? 'checked' : '' }}>
74 <label for="remember">
75 {{ __('adminlte:adminlte.remember_me') }}
76 </label>
77 </div>
78 </div>
79 <div class="col-5">
80 <button type="submit"
81 class="btn btn-block {{ config('adminlte.classes_auth_btn', 'btn-flat btn-primary') }}">
82 <span class="fas fa-sign-in-alt"></span>
83 {{ __('adminlte:adminlte.sign_in') }}
84 </button>
85 </div>
86 </div>
87 </form>
88 @stop
89
90 @section('auth_footer')
91 <div class="my-0">
92 <a href="{{ route('register') }}">
93 {{ __('adminlte:adminlte.register_a_new_membership') }}
94 </a>
95 </div>
96 @endif
97 @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-page', ['auth_type' => 'register'])
2
3 @php($login_url = View::getSection('login_url') ?? config('adminlte.login_url', 'login'))
4 @php($register_url = View::getSection('register_url') ?? config('adminlte.register_url', 'register'))
5
6 @if (config('adminlte.use_route_url', false))
7 @php($login_url = $login_url ? route($login_url) : '')
8 @php($register_url = $register_url ? route($register_url) : '')
9 @else
10 @php($login_url = $login_url ? url($login_url) : '')
11 @php($register_url = $register_url ? url($register_url) : '')
12 @endif
13
14 @section('auth_header', __('adminlte:adminlte.register_message'))
15
16 @section('auth_body')
17 <form action="{{ route('proses_register') }}" method="post">
18 @csrf
19 {{-- Name field --}}
20 <div class="input-group mb-3">
21 <input type="text" name="nama" class="form-control @error('nama') is-invalid @enderror"
22 value="{{ old('nama') }}" placeholder="Nama autofocus"
23 <div class="input-group-append">
24 <div class="input-group-text">
25 <span class="fas fa-user {{ config('adminlte.classes_auth_icon', '') }}"></span>
26 </div>
27 </div>
28 @error('nama')
29 <span class="invalid-feedback" role="alert">
30 <strong>{{ $message }}</strong>
31 </span>
32 @enderror
33 </div>
```



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>

```
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 <div class="input-group-append">
40 <div class="input-group-text">
41 <span class="fas fa-user {{ config('adminlte.classes_auth_icon', '') }}"></span>
42 </div>
43 </div>
44 @error('username')
45 <span class="invalid-feedback" role="alert">
46 <strong>{{ $message }}</strong>
47 </span>
48 @enderror
49 </div>
50
51 <!-- Password field -->
52 <div class="input-group mb-3">
53 <input type="password" name="password" class="form-control @error('password') is-invalid @enderror"
54 placeholder="{{ __('adminlte::adminlte.password') }}">
55 <div class="input-group-append">
56 <div class="input-group-text">
57 <span class="fas fa-lock {{ config('adminlte.classes_auth_icon', '') }}"></span>
58 </div>
59 </div>
60 @error('password')
61 <span class="invalid-feedback" role="alert">
62 <strong>{{ $message }}</strong>
63 </span>
64 @enderror
65 </div>
66
67 <!-- Register button -->
68 <button type="submit" class="btn btn-block {{ config('adminlte.classes_auth_btn', 'btn-flat btn-primary') }}">
69 <span class="fas fa-user-plus"></span>
70 <span class="text" {{ __('adminlte::adminlte.register') }}></span>
71 </button>
72 </form>
73 @stop
74
75 @section('auth.footer')
76 <p class="my-0">
77 <a href="{{ route('login') }}">
78 <span class="text" {{ __('adminlte::adminlte.i_already_have_a_membership') }}></span>
79 </a>
80 </p>
81 @stop
```

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

```
resources > views > admin.blade.php > ...
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 <div class="container">
11 <div class="card">
12 <div class="card-header">Tampilan {{ Auth::user()->level_id == 1 ? 'Admin' : 'Manager' }}</div>
13 <div class="card-body">
14 <h1>Login Sebagai:
15 <span class="text" {{ Auth::user()->level_id == 1 ? 'Admin' : 'Manager' }}></span></h1>
16 <a href="{{ route('logout') }}">Logout</a>
17 </div>
18 </div>
19 </div>
20
21 @endsection
22
23 @push('js')
24 @endpush
```

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



KEMENTERIAN PENDIDIKAN, KEBUDAYAAN, RISET, DAN TEKNOLOGI

POLITEKNIK NEGERI MALANG

JURUSAN TEKNOLOGI INFORMASI

Jl. Soekarno Hatta No. 9, Jatimulyo, Lowokwaru, Malang 65141

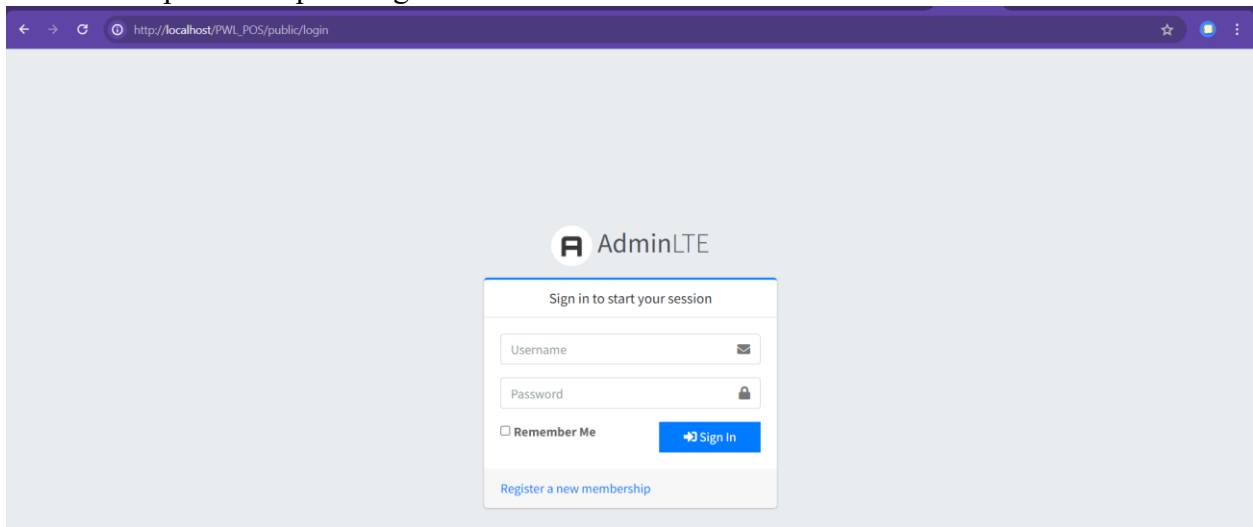
Telp. (0341) 404424 – 404425, Fax (0341) 404420

<http://www.polinema.ac.id>

```
resources > views > manager.blade.php > ...
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', 'Manager')
8
9  @section('content')
10     <div class="container">
11         <div class="card">
12             <div class="card-header">Tampilan
13                 {{ Auth::user()->level_id == 1 ? 'Admin' : 'Manager' }}
14             </div>
15             <div class="card-body">
16                 <h1>Login Sebagai:
17                     {{ Auth::user()->level_id == 1 ? 'Admin' : 'Manager' }}</h1>
18                 <a href="{{ route('logout') }}">Logout</a>
19             </div>
20         </div>
21     </div>
22 @endsection
23
24 @push('js')
25 @endpush
```

17. Jalankan kode diatas dan buat laporan dengan isi screenshoot hasil dan beri penjelasan pada setiap screenshoot hasil dan commit perubahan pada git.

- Ini merupakan tampilan login awal





KEMENTERIAN PENDIDIKAN, KEBUDAYAAN, RISET, DAN TEKNOLOGI

POLITEKNIK NEGERI MALANG

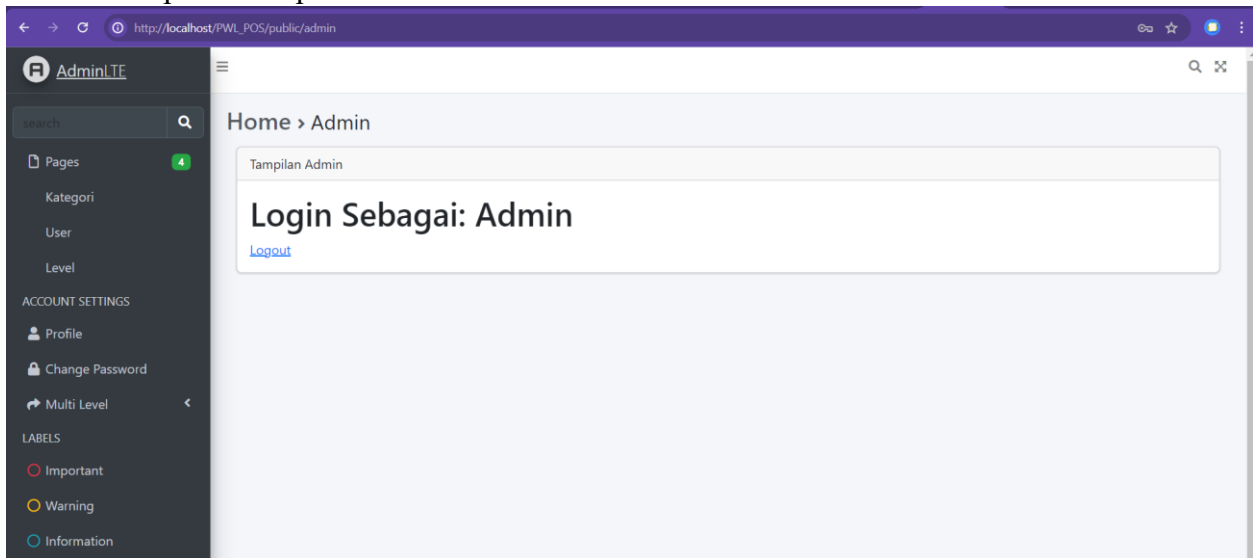
JURUSAN TEKNOLOGI INFORMASI

Jl. Soekarno Hatta No. 9, Jatimulyo, Lowokwaru, Malang 65141

Telp. (0341) 404424 – 404425, Fax (0341) 404420

<http://www.polinema.ac.id>

- Ini merupakan tampilan halaman admin



- Ini merupakan tampilan halaman manager

