



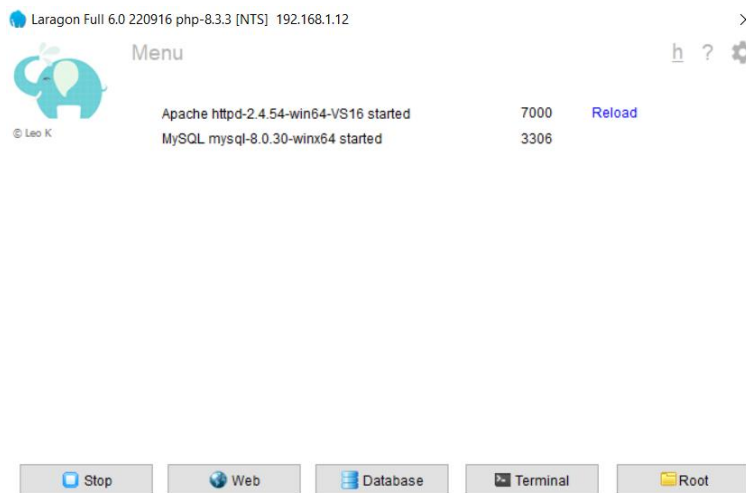
Jurusan Teknologi Informasi Politeknik Negeri Malang
Jobsheet-9: AUTHENTICATION, MIDDLEWARE
Mata Kuliah : Pemrograman Web Lanjut

Nama	:	JIHAN KARUNIA PUTRI
NIM	:	2241720031
Kelas	:	TI - 2H

LEMBAR JAWABAN JOBSHEET 9

Praktikum – Auth dan Middleware

a. Start Laragon



b. 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
```

c. Jalankan “npm run dev”

```
PS C:\laragon\www\PWL_POS> npm run dev
> dev
> vite

VITE v5.1.6 ready in 849 ms
→ Local: http://localhost:5173/
→ Network: use --host to expose
→ press h + enter to show help

LARAVEL v10.47.0 plugin v1.0.2
→ APP_URL: http://localhost
```



Jurusan Teknologi Informasi Politeknik Negeri Malang
Jobsheet-9: AUTHENTICATION, MIDDLEWARE
Mata Kuliah : Pemrograman Web Lanjut

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

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

```
'providers' => [
    'users' => [
        'driver' => 'eloquent',
        'model' => App\Models\UserModel::class,
    ],
],
```

2. Membuat middleware dengan perintah pada cmd atau terminal

```
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()



Jurusan Teknologi Informasi Politeknik Negeri Malang

Jobsheet-9: AUTHENTICATION, MIDDLEWARE

Mata Kuliah : Pemrograman Web Lanjut

```
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      */
17     public function handle(Request $request, Closure $next, $roles): Response
18     {
19         // Cek sudah login atau belum. Jika belum kembali ke halaman login
20         if (!Auth::check()) {
21             return redirect('login');
22         }
23         // Simpan data user pada variabel $user
24         $user = Auth::user();
25
26         // Jika user memiliki level sesuai pada kolom pada lanjutkan request
27         if ($user->level_id == $roles) {
28             return $next($request);
29         }
30
31         // Jika tidak memiliki akses maka kembalikan ke halaman login
32         return redirect('login')->with('error', 'Maaf anda tidak memiliki akses');
33     }
34 }
```

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 > ...
7  class Kernel extends HttpKernel
8
9  {
10     /**
11      * The middleware aliases.
12      *
13      * @var array
14      */
15     protected $middlewareAliases = [
16         'auth' => \App\Http\Middleware\Authenticate::class,
17         'auth.basic' => \Illuminate\Auth\Middleware\AuthenticateWithBasicAuth::class,
18         'auth.session' => \Illuminate\Session\Middleware\AuthenticateSession::class,
19         'cache.headers' => \Illuminate\Http\Middleware\SetCacheHeaders::class,
20         'can' => \Illuminate\Auth\Middleware\Authorize::class,
21         'guest' => \App\Http\Middleware\RedirectIfAuthenticated::class,
22         'password.confirm' => \Illuminate\Auth\Middleware\RequirePassword::class,
23         'precognitive' => \Illuminate\Foundation\Http\Middleware\HandlePrecognitiveRequests::class,
24         'signed' => \App\Http\Middleware\ValidateSignature::class,
25         'throttle' => \Illuminate\Routing\Middleware\ThrottleRequests::class,
26         'verified' => \Illuminate\Auth\Middleware\EnsureEmailIsVerified::class,
27         // Tambahkan middleware alias dengan nama cek_login
28         'cek_login' => \App\Http\Middleware\Cek_login::class,
29     ];
30 }
```

5. Membuat Controller Auth.



Jurusan Teknologi Informasi Politeknik Negeri Malang
Jobsheet-9: AUTHENTICATION, MIDDLEWARE
Mata Kuliah : Pemrograman Web Lanjut

```
PS C:\laragon\www\PWL_POS> php artisan make:controller AuthController  
  
INFO Controller [C:\laragon\www\PWL_POS\app\Http\Controllers\AuthCont  
roller.php] created successfully.
```

6. Kemudian buat kode AuthController.php

```
<?php  
  
namespace App\Http\Controllers;  
  
use App\Models\UserModel;  
use Illuminate\Http\Request;  
use Illuminate\Support\Facades\Auth;  
use Illuminate\Support\Facades\Hash;  
use Illuminate\Support\Facades\Validator;  
  
class AuthController extends Controller  
{  
    public function index ()  
    {  
        // Kita ambil data user lalu simpan pada variable $user  
        $user = Auth::user();  
  
        // Kondisi jika user nya ada  
        if ($user) {  
            // Jika user nya memiliki level admin  
            if ($user->level_id == '1') {  
                return redirect()->intended('admin');  
            }  
            // Jika user nya memiliki level manager  
            else if ($user->level == '2') {
```



Jurusan Teknologi Informasi Politeknik Negeri Malang
Jobsheet-9: AUTHENTICATION, MIDDLEWARE
Mata Kuliah : Pemrograman Web Lanjut

```
        return redirect()->intended('manager');
    }
}

return view('login');
}

public function proses_login(Request $request)
{
    // Kita buat validasi pada saat tombol login di klik, validasi
    nya username & password wajib di isi
    $request->validate([
        'username' => 'required',
        'password' => 'required'
    ]);

    // Ambil data request username & password saja
    $credential = $request->only('username', 'password');
    // Cek jika data username dan password valid (sesuai) dengan data
    if (Auth::attempt($credential)) {

        // Kalau berhasil simpan data user ya di variabel $user
        $user = Auth::user ();

        // Cek lagi jika level user admin maka arahkan ke halaman
admin
        if ($user->level_id == '1') {
            //dd($user->level_id);
            return redirect()->intended('admin');
        }
    }
}
```



Jurusan Teknologi Informasi Politeknik Negeri Malang
Jobsheet-9: AUTHENTICATION, MIDDLEWARE
Mata Kuliah : Pemrograman Web Lanjut

```
// Tapi jika level user nya user biasa maka arahkan ke
halaman user

else if ($user->level_id == '2') {
    return redirect()->intended('manager');
}

// Jika belum ada role maka ke halaman /
return redirect()->intended('/');
}

// Jika ga ada data user yang valid maka kembalikan lagi ke
halaman login, pastikan kirim pesan error juga kalau login gagal ya
return redirect('login')
    ->withInput()
    ->withErrors(['login_gagal' => 'Pastikan kembali username dan
password yang dimasukan sudah benar']);
}

public function register(){
    // Tampilan halaman form register
    return view('register');
}

// Aksi form register
public function proses_register(Request $request)
{
    // Kita buat validasi nih buat proses register, validasinya yaitu
    semua field wajib di isi, validasi username itu harus unique atau tidak
    boleh duplicate username ya

    $validator = Validator :: make($request->all(), [
        'nama' => 'required',
        'username' => 'required |unique:m_user',
```



Jurusan Teknologi Informasi Politeknik Negeri Malang
Jobsheet-9: AUTHENTICATION, MIDDLEWARE
Mata Kuliah : Pemrograman Web Lanjut

```
        'password' => 'required'
    });

    // Kalau gagal kembali ke halaman register dengan munculkan pesan
error
    if ($validator->fails()) {
        return redirect('/register')
            ->withErrors($validator)
            ->withInput();
    }

    // Kalau berhasil isi level & hash passwordnya ya biar secure
    $request['level_id'] = '2';
    $request['password'] = Hash::make($request->password);

    // Masukkan semua data pada request ke table user
    UserModel::create($request->all());

    // Kalo berhasil arahkan ke halaman login
    return redirect()->route('login');
}

public function logout(Request $request)
{
    // Logout itu harus menghapus session nya
    $request->session()->flush();

    // Jalan kan juga fungsi logout pada auth
    Auth::logout();

    // Kembali kan ke halaman login
}
```




Jurusan Teknologi Informasi Politeknik Negeri Malang
Jobsheet-9: AUTHENTICATION, MIDDLEWARE
Mata Kuliah : Pemrograman Web Lanjut

```
        return Redirect('login');  
    }  
}
```

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

```
Route::get('login', [AuthController::class, 'index'])->name('login');  
Route::get('register', [AuthController::class, 'register'])->name('register');  
Route::post('proses_login', [AuthController::class, 'proses_login'])->name('proses_login');  
Route::get('logout', [AuthController::class, 'logout'])->name('logout');  
Route::post('proses_register', [AuthController::class, 'proses_register'])->name('proses_register');  
  
// Kita atur juga untuk middleware menggunakan group pada routing  
// didalamnya terdapat group untuk mengecek kondisi login  
// jika user yang login merupakan admin maka akan diarahkan ke AdminController  
// jika user yang login merupakan manager maka akan diarahkan ke UserController  
Route::group(['middleware' => ['auth']], function () {  
    Route::group(['middleware' => ['cek_login:1']], function () {  
        Route::resource('admin', AdminController::class);  
    });  
    Route::group(['middleware' => ['cek_login:2']], function () {  
        Route::resource('manager', ManagerController::class);  
    });  
});
```

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

```
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

```
app > Http > Controllers > AdminController.php > ...  
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     {  
11         return view('admin');  
12     }  
13 }
```




Jurusan Teknologi Informasi Politeknik Negeri Malang
Jobsheet-9: AUTHENTICATION, MIDDLEWARE
Mata Kuliah : Pemrograman Web Lanjut

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

```
PS C:\laragon\www\PWL_POS> php artisan make:controller ManagerController  
  
INFO Controller [C:\laragon\www\PWL_POS\app\Http\Controllers\ManagerController.php] created successfully.
```

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

```
app > Http > Controllers > ManagerController.php > ...  
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     {  
11         return view('manager');  
12     }  
13 }
```

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

```
@extends('adminlte::auth.auth-page', ['auth_type' => 'login'])  
  
@section('adminlte_css_pre')  
  
<link rel="stylesheet" href="{{ asset('vendor/icheck-bootstrap/icheck-bootstrap.min.css') }}">  
  
@stop  
  
@php($login_url = View::getSection('login_url') ??  
config('adminlte.login_url', 'login'))  
  
@php($register_url = View::getSection('register_url') ??  
config('adminlte.register_url', 'register'))  
  
@php($password_reset_url = View::getSection('password_reset_url') ??  
config('adminlte.password_reset_url', 'password/reset'))
```



Jurusan Teknologi Informasi Politeknik Negeri Malang
Jobsheet-9: AUTHENTICATION, MIDDLEWARE
Mata Kuliah : Pemrograman Web Lanjut

```
@if (config('adminlte.use_route_url', false))
    @php($login_url = $login_url ? route($login_url) : '')
    @php($register_url = $register_url ? route($register_url) : '')
    @php($password_reset_url = $password_reset_url ?
route($password_reset_url) : '')
@else
    @php($login_url = $login_url ? url($login_url) : '')
    @php($register_url = $register_url ? url($register_url) : '')
    @php($password_reset_url = $password_reset_url ?
url($password_reset_url) : '')
@endif

@section('auth_header', __('adminlte::adminlte.login_message'))

@section('auth_body')
    @error('login_gagal')

        <div class="alert alert-warning alert-dismissible fade show"
role="alert">

            <span class="alert-inner-text"><strong>Warning!</strong>
{{ $message }}</span>

            <button type="button" class="close" data-dismiss="alert" aria-
label="Close">

                <span aria-hidden="true">&times;</span>

            </button>

        </div>
    @enderror

    <form action="{{ url('proses_login') }}" method="post">

        @csrf

        {{-- Email field --}}
```



Jurusan Teknologi Informasi Politeknik Negeri Malang
Jobsheet-9: AUTHENTICATION, MIDDLEWARE
Mata Kuliah : Pemrograman Web Lanjut

```
<div class="input-group mb-3">

    <input type="text" name="username" class="form-control
@error('username') is-invalid @enderror"
        value="{{ old('username') }}" placeholder="Username"
autofocus>

    <div class="input-group-append">
        <div class="input-group-text">
            <span class="fas fa-envelope
{{ config('adminlte.classes_auth_icon', '') }}"></span>
        </div>
    </div>

    @error('username')
    <span class="invalid-feedback" role="alert">
        <strong>{{ $message }}</strong>
    </span>
    @enderror
</div>

{{-- Password field --}}
<div class="input-group mb-3">
    <input type="password" name="password"
        class="form-control @error('password') is-invalid
@enderror"
        placeholder="{{ __('adminlte::adminlte.password') }}">
    <div class="input-group-append">
        <div class="input-group-text">
            <span class="fas fa-lock
{{ config('adminlte.classes_auth_icon', '') }}"></span>
        </div>
    </div>
</div>
```



Jurusan Teknologi Informasi Politeknik Negeri Malang
Jobsheet-9: AUTHENTICATION, MIDDLEWARE
Mata Kuliah : Pemrograman Web Lanjut

```
@error('password')

<span class="invalid-feedback" role="alert">

    <strong>{{ $message }}</strong>

</span>

@enderror

</div>

{{-- Login field --}}

<div class="row">

    <div class="col-7">

        <div class="icheck-primary"
title="{{ __('adminlte::adminlte.remember_me_hint') }}">

            <input type="checkbox" name="remember" id="remember"
{{ old('remember') ? 'checked' : '' }}>

            <label for="remember">

                {{ __('adminlte::adminlte.remember_me') }}

            </label>

        </div>

    </div>

    <div class="col-5">

        <button type="submit"

            class="btn btn-block
{{ config('adminlte.classes_auth_btn', 'btn-flat btn-primary') }}">

            <span class="fas fa-sign-in-alt"></span>

            {{ __('adminlte::adminlte.sign_in') }}

        </button>

    </div>

</div>
```



Jurusan Teknologi Informasi Politeknik Negeri Malang
Jobsheet-9: AUTHENTICATION, MIDDLEWARE
Mata Kuliah : Pemrograman Web Lanjut

```
</form>

@stop

@section('auth_footer')
    @if($register_url)
        <p class="my-0">
            <a href="{{ route('register') }}">
                {{ __('adminlte::adminlte.register_a_new_membership') }}
            </a>
        </p>
    @endif
@stop
```

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

```
@extends('adminlte::auth.auth-page', ['auth_type' => 'register'])

@php($login_url = View::getSection('login_url') ??
config('adminlte.login_url', 'login'))

@php($register_url = View::getSection('register_url') ??
config('adminlte.register_url', 'register'))

@if (config('adminlte.use_route_url', false))
    @php($login_url = $login_url ? route($login_url) : '')
    @php($register_url = $register_url ? route($register_url) : '')
@else
    @php($login_url = $login_url ? url($login_url) : '')
    @php($register_url = $register_url ? url($register_url) : '')
@endif
```



Jurusan Teknologi Informasi Politeknik Negeri Malang
Jobsheet-9: AUTHENTICATION, MIDDLEWARE
Mata Kuliah : Pemrograman Web Lanjut

```
@section('auth_header', __('adminlte::adminlte.register_message'))

@section('auth_body')
    <form action="{{ route('proses_register') }}" method="post">
        @csrf
        {{-- Nama field --}}
        <div class="input-group mb-3">
            <input type="text" name="nama" class="form-control
@error('nama') is-invalid @enderror"
                value="{{ old('nama') }}" placeholder="Nama" autofocus>
            <div class="input-group-append">
                <div class="input-group-text">
                    <span class="fas fa-user
{{ config('adminlte.classes_auth_icon', '') }}"></span>
                </div>
            </div>
            @error('nama')
                <span class="invalid-feedback" role="alert">
                    <strong>{{ $message }}</strong>
                </span>
            @enderror
        </div>

        {{-- Username field --}}
        <div class="input-group mb-3">
            <input type="text" name="username" class="form-control
@error('username') is-invalid @enderror"
                value="{{ old('username') }}" placeholder="Username"
                autofocus>
```



Jurusan Teknologi Informasi Politeknik Negeri Malang
Jobsheet-9: AUTHENTICATION, MIDDLEWARE
Mata Kuliah : Pemrograman Web Lanjut

```
<div class="input-group-append">
    <div class="input-group-text">
        <span class="fas fa-user
{{ config('adminlte.classes_auth_icon', '') }}"></span>
    </div>
</div>

@error('username')
    <span class="invalid-feedback" role="alert">
        <strong>{{ $message }}</strong>
    </span>
@enderror
</div>

{{-- Password field --}}
<div class="input-group mb-3">
    <input type="password" name="password" class="form-control
@error('password') is-invalid @enderror"
        placeholder="{{ __('adminlte::adminlte.password') }}">
    <div class="input-group-append">
        <div class="input-group-text">
            <span class="fas fa-lock
{{ config('adminlte.classes_auth_icon', '') }}"></span>
        </div>
    </div>
    @error('password')
        <span class="invalid-feedback" role="alert">
            <strong>{{ $message }}</strong>
        </span>
    @enderror
</div>
```




Jurusan Teknologi Informasi Politeknik Negeri Malang
Jobsheet-9: AUTHENTICATION, MIDDLEWARE
Mata Kuliah : Pemrograman Web Lanjut

```
</div>

    {{-- Register button --}}

    <button type="submit" class="btn btn-block
    {{ config('adminlte.classes_auth_btn', 'btn-flat btn-primary') }}">

        <span class="fas fa-user-plus"></span>

        {{ __('adminlte::adminlte.register') }}

    </button>

</form>

@stop

@section('auth_footer')

    <p class="my-0">

        <a href="{{ route('login') }}">

            {{ __('adminlte::adminlte.i_already_have_a_membership') }}

        </a>

    </p>

@stop
```

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



Jurusan Teknologi Informasi Politeknik Negeri Malang

Jobsheet-9: AUTHENTICATION, MIDDLEWARE

Mata Kuliah : Pemrograman Web Lanjut

```
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' }}
13         </div>
14         <div class="card-body">
15             <h1>Login Sebagai:
16                 {{ Auth::user()->level_id == 1 ? 'Admin' : 'Manager' }}</h1>
17             <a href="{{ route('logout') }}">Logout</a>
18         </div>
19     </div>
20 </div>
21 @endsection
22
23 @push('js')
24 @endpush
```

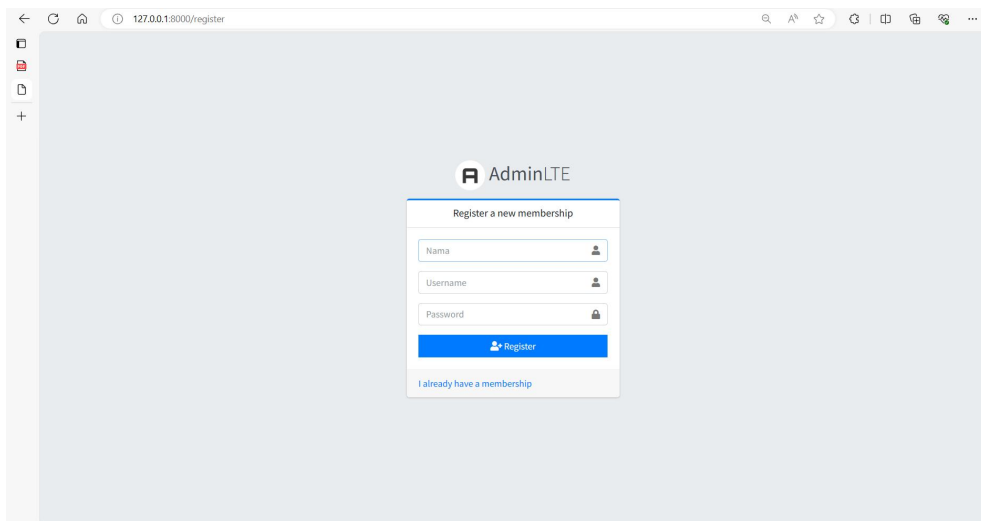
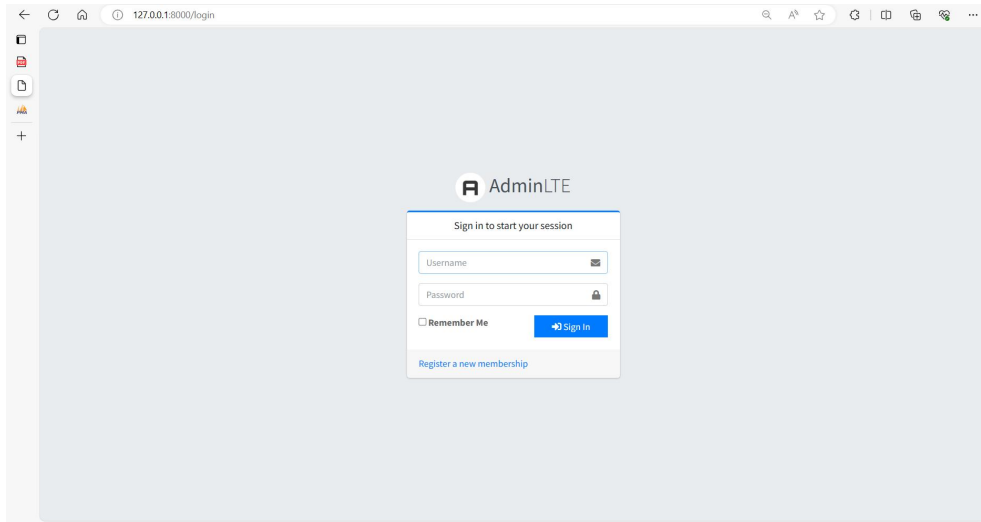
16. Buatlah halaman register dengan membuka folder resources/views dengan nama file manager.blade.php. Lalu kita isi filenya

```
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
```



Jurusan Teknologi Informasi Politeknik Negeri Malang
Jobsheet-9: AUTHENTICATION, MIDDLEWARE
Mata Kuliah : Pemrograman Web Lanjut

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





Jurusan Teknologi Informasi Politeknik Negeri Malang
Jobsheet-9: AUTHENTICATION, MIDDLEWARE
Mata Kuliah : Pemrograman Web Lanjut

