

Laporan Pemrograman WEB Lanjut  
Jobsheet 7 Authentication dan Authorization di Laravel



Politeknik Negeri Malang  
Semester 4  
2025

**NIM**

2341720082

**Nama**

Noklent Fardian Erix

**Kelas**

2A

**Jurusan**

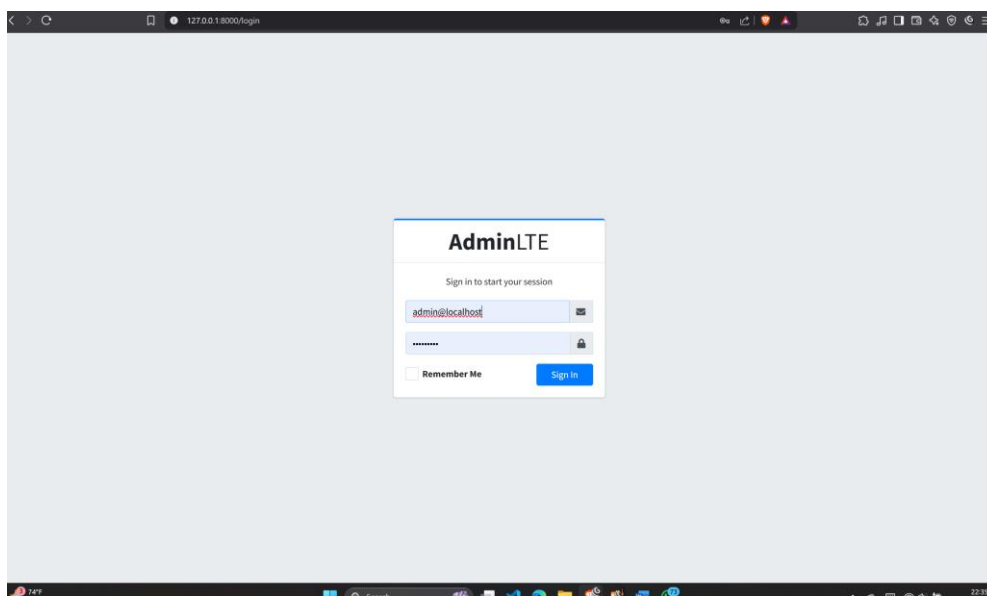
Teknologi Informasi

**Progam Studi**

D-IV Teknik Informatika

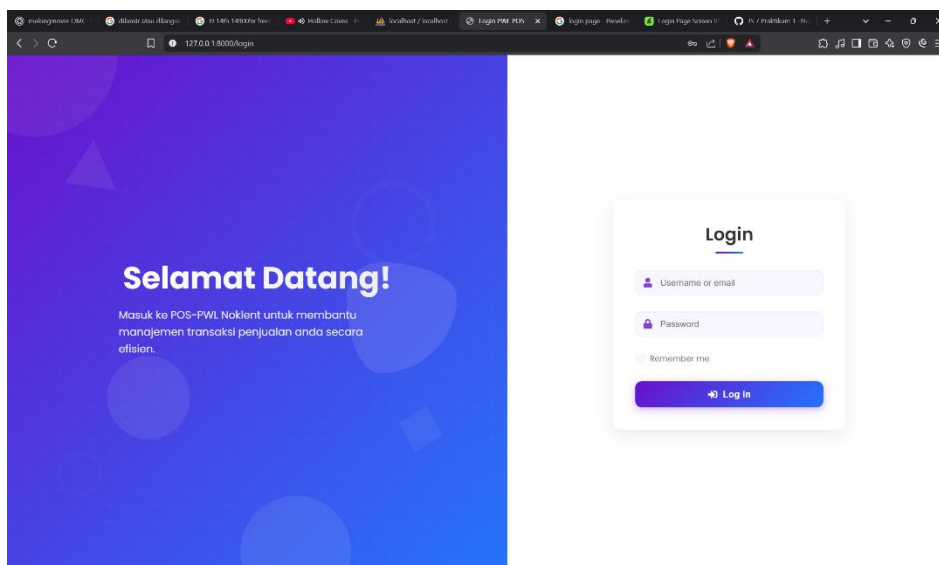
## Praktikum 1 – Implementasi Authentication :

1. Setelah saya mencoba arahan yang ada pada praktikum maka tampilan halaman login akan menjadi seperti ini



## Tugas

1. Q: Silahkan implementasikan proses login pada project kalian masing-masing  
A: Saya mengkreasikan tampilan login page dengan custom css dan jadinya menjadi seperti ini

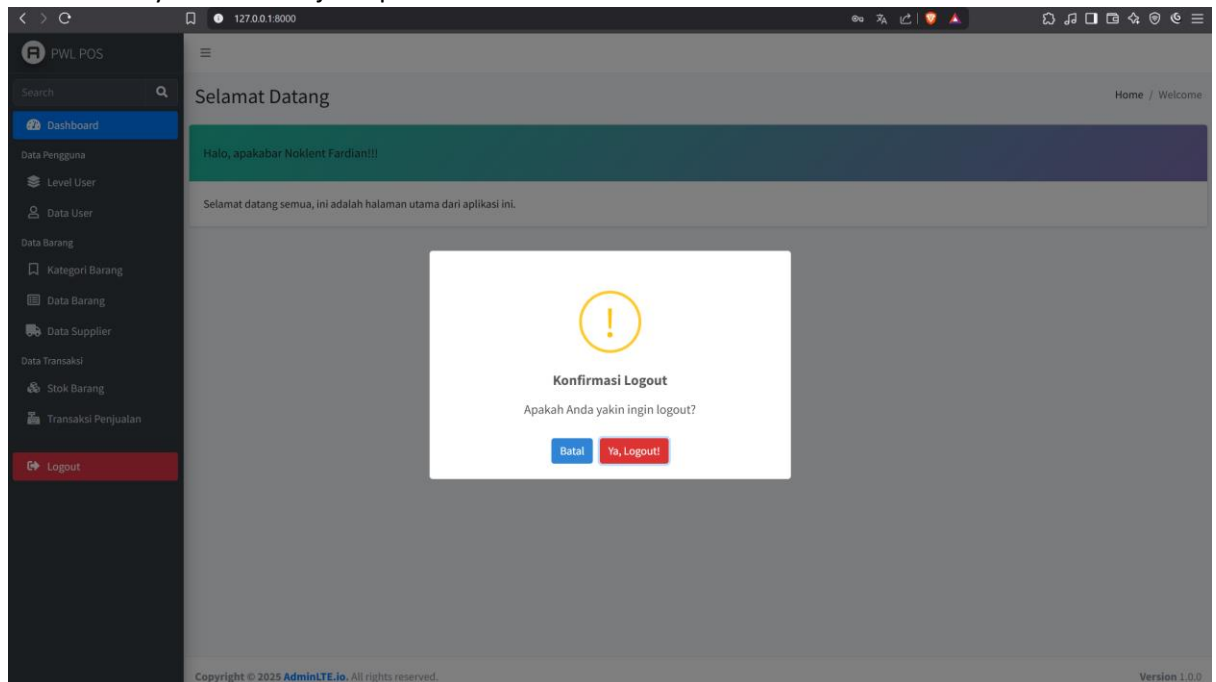


2. Q: Silahkan implementasi proses logout pada halaman web yang kalian buat

A: Tambahkan kode ini pada file sidebar.blade.php

```
1 <li class="nav-item mt-4">
2     <a href="javascript:void(0)" onclick="confirmLogout()" class="nav-link bg-danger text-white">
3         <i class="nav-icon fas fa-sign-out-alt"></i>
4         <p>Logout</p>
5     </a>
6 </li>
7 </ul>
8 </nav>
9 </div>
10 <script>
11     function confirmLogout() {
12         Swal.fire({
13             title: 'Konfirmasi Logout',
14             text: 'Apakah Anda yakin ingin logout?',
15             icon: 'warning',
16             showCancelButton: true,
17             confirmButtonColor: '#d33',
18             cancelButtonColor: '#3085d6',
19             cancelButtonText: 'Batal',
20             confirmButtonText: 'Ya, Logout!',
21             reverseButtons: true
22         }).then((result) => {
23             if (result.isConfirmed) {
24                 window.location.href = "{ url('/logout') }";
25             }
26         });
27     }
28 </script>
```

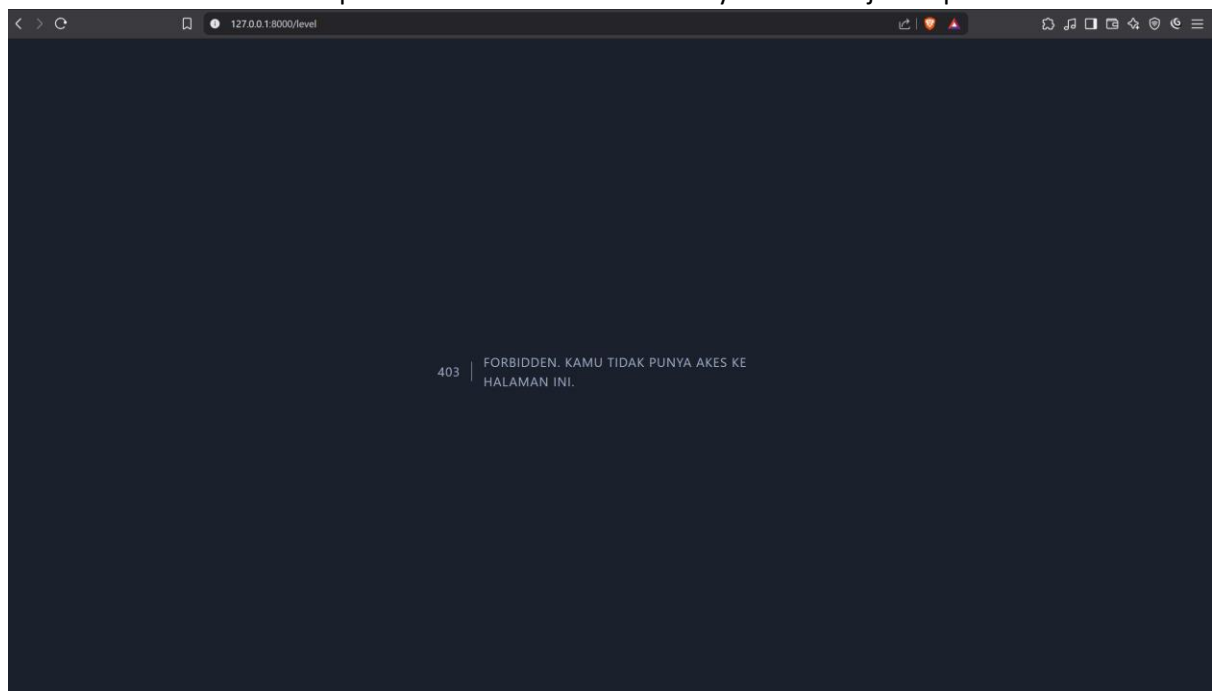
Maka Hasilnya akan menjadi seperti ini



3. Q: Amati dan jelaskan tiap tahapan yang kalian kerjakan, dan jabarkan dalam laporan
- A:
- Pertama kita menunjur UserModel menjadi targer autentikasi pada auth.php
- Kedua kita membuat UserModel akan menjadi subclass dari Authenticatable agar fungsi fungsi dari lavarel dapat kita gunakan
- Ketiga jika tombol login pada login.blade.php diklik, maka akan berjalan validasi ajax terlebih dahulu, Ketika validasi ajax telah terpenuhi maka fungsi postlogin akan dijalankan. Jika berhasil maka kita bisa memasuki system, jika tidak akan muncul pop up
- Dan jika tombol logout diklik maka fungsi logout pada AuthController akan dijalankan

### Praktikum 2 – Implementasi Authorizaton di Laravel dengan Middleware

1. Setelah kita melakukan semua al yang ada pada praktikum 2 dan kita login sebagai bukan admin dan kita akan masuk pada halaman level maka hasilnya akan menjadi seperti ini



### Tugas

1. Q: Apa yang kalian pahami pada praktikum 2 ini?
- A: Middleware authorize dapat untuk membatasi role tertentu untuk tidak dapat mengakses suatu fitur dalam system.

2. Q: Amati dan jelaskan tiap tahapan yang kalian kerjakan, dan jabarkan dalam laporan A:

Langkah Pertama : kita akan membuat fungsi `hasLevel` pada `UserModel`, fungsi tersebut digunakan untuk nantinya dipakai untuk mengecek apakah user yang login mempunyai level yang mempunyai tipe data dan value yang sama, dan fungsi ini akan mengeluarkan return `true` atau `false`.

Langkah kedua : kemudian kita membuat File baru yang bernama `AuthorizeUser` yang mempunyai fungsi `handle` yang nantinya akan dipanggil ketika user akan memasuki halaman fitur tertentu

Langkah ketiga adalah mendaftarkan dan memberi nama alias untuk Middleware `AuthorizeUser` kita dengan nama `'authorize'` hal ini memudahkan kita dalam memanggil class tersebut pada route.

Langkah Terakhir yaitu menambahkan middleware ke route seperti ini `Route::prefix('level')->middleware('authorize:ADM')->group(function () {`

Ini artinya Ketika kita ingin mengakses fitur level, Laravel akan mengecek dengan fungsi `handle` pada `AuthorizeUser` apakah level yang ada pada route `('authorize:ADM')` sama persis dengan level yang saat ini user login, jika sama maka user tersebut dalam mengakses fitur tersebut, jika tidak maka akan muncul error 403

### 3. Multi-Level Authorization di Laravel

Setelah melakukan praktikum pada praktikum 3 maka jika kita login sebagai user admin atau user manager, kita masih dapat untuk mengakses fitur barang, namun jika kita login sebagai kasir akan muncul error 403

#### Tugas

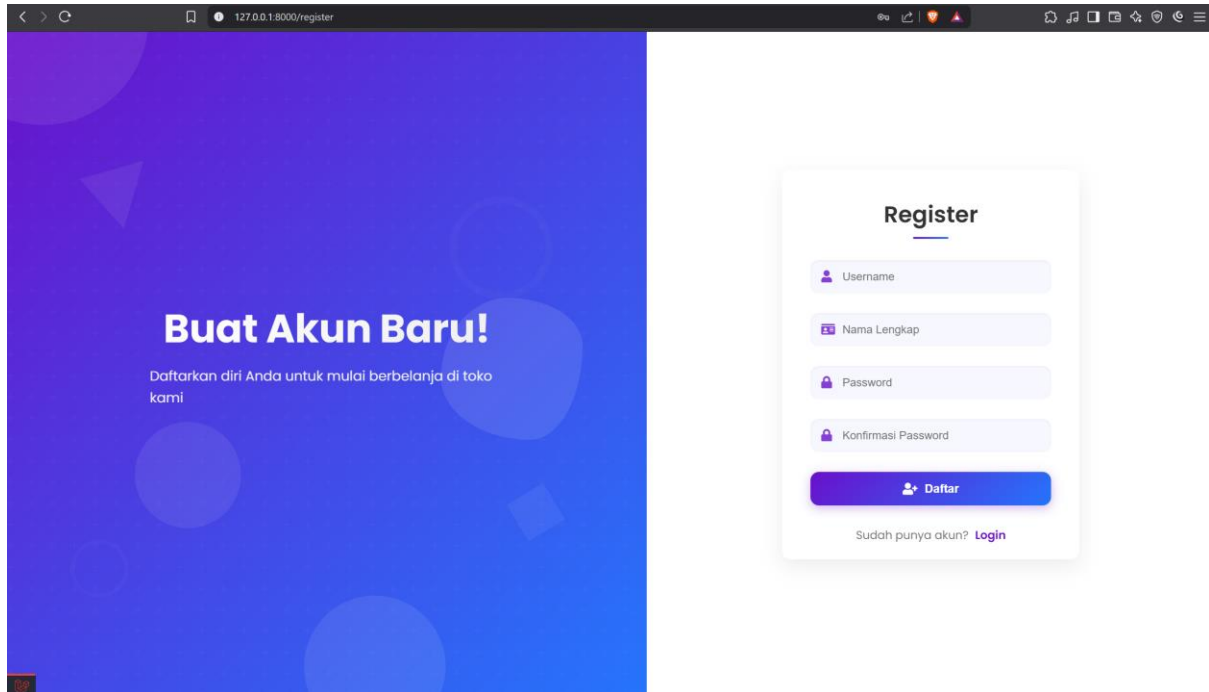
1. Q: Silahkan implementasikan multi-level authorization pada project kalian masing-masing A: Done

2. Q: Amati dan jelaskan tiap tahapan yang kalian kerjakan, dan jabarkan dalam laporan

A: Pertama middleware menerima user `level_kode` dengan cara memanggil fungsi `getLevelCode()`. Setelah itu, sistem melakukan pemeriksaan menggunakan fungsi `in_array()` dari PHP. Fungsi ini digunakan untuk memeriksa apakah level (role) pengguna ada di dalam daftar role yang diizinkan untuk mengakses fitur yang spesifik, yang dikirim ke middleware. Jika berhasil akan memanggil `$next($request)` yang berguna untuk memproses dan masuk pada controller. Jika gagal maka akan mengalami error 403

3. Q: Implementasikan multi-level authorization untuk semua Level/Jenis User dan Menu menu yang sesuai dengan Level/Jenis User  
A: Sudah ada di github, namun untuk fitur stock dan penjualan belum ada isinya

## Tugas Implementasi Form Registrasi :



127.0.0.1:8000/register

### Buat Akun Baru!

Daftarkan diri Anda untuk mulai berbelanja di toko kami

#### Register

Username

Nama Lengkap

Password

Konfirmasi Password

**Daftar**

Sudah punya akun? [Login](#)