

Rafif Tri Hartanto

SIB 2A

26

Jobsheet 7 PWL

Praktikum 1: Implementasi Authentication

1.

```
PWL_POS > config > auth.php
3   return [
49
50       | All authentication drivers have a user provider
51       | users are actually retrieved out of your database
52       | mechanisms used by this application to persist
53
54       | If you have multiple user tables or models you
55       | sources which represent each model / table. They
56       | be assigned to any extra authentication guards
57
58       | Supported: "database", "eloquent"
59
60       | */
61
62       'providers' => [
63           'users' => [
64               'driver' => 'eloquent',
65               'model' => App\Models\UserModel::class,
```

2.

```
PWL_POS > app > Models > UserModel.php > ...
1   <?php
2
3   namespace App\Models;
4
5   use Illuminate\Database\Eloquent\Model;
6   use Illuminate\Database\Eloquent\Factories\HasFactory;
7   use Illuminate\Database\Eloquent\Relations\BelongsTo;
8   use Illuminate\Foundation\Auth\User as Authenticatable; // IMPLEMENTASI AUTHENTICATABLE
9
10  class UserModel extends Authenticatable
11  {
12      use HasFactory;
13
14      protected $table = 'm_user';
15      protected $primaryKey = 'user_id';
16      protected $fillable = ['username', 'password', 'nama', 'level_id', 'created_at', 'updated_at'];
17
18      protected $hidden = ['password'];
19
20      protected $casts = ['password' => 'hashed'];
21
22      /** relasi ke tabel level */
23      public function level(): BelongsTo
24      {
25          return $this->belongsTo(LevelModel::class, 'level_id', 'level_id');
26      }
27
28  }
```

3.

```
PWL_POS > app > Http > Controllers > AuthController.php > ...
1  <?php
2
3  namespace App\Http\Controllers;
4  |
5  use Illuminate\Http\Request;
6  use Illuminate\Support\Facades\Auth;
7
8
9  class AuthController extends Controller
10 {
11     public function login()
12     {
13         if(Auth::check()){ // jika sudah login, maka redirect ke halaman home
14             return redirect('/');
15         }
16         return view('auth.login');
17     }
18
19     public function postlogin(Request $request)
20     {
21         if($request->ajax() || $request->wantsJson()){
22             $credentials = $request->only('username', 'password');
23             if (Auth::attempt($credentials)) {
24                 return response()->json([
25                     'status' => true,
26                     'message' => 'Login Berhasil',
27                     'redirect' => url('/')
28                 ]);
29             }
30             return response()->json([
31                 'status' => false,
32                 'message' => 'Login Gagal'
33             ]);
34         }
35     }
36 }
```

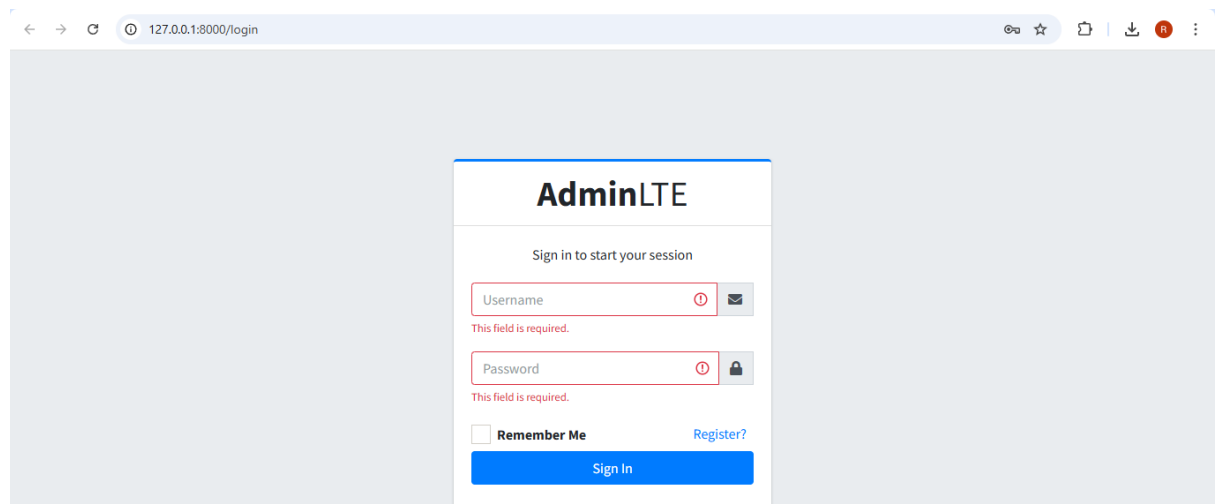
4.

```
PWL_POS > resources > views > auth > login.blade.php > html
1  <!DOCTYPE html>
2  <html lang="en">
3  <head>
4      <meta charset="utf-8">
5      <meta name="viewport" content="width=device-width, initial-scale=1">
6      <title>Login Pengguna</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=fallb
10
11      <!-- Font Awesome -->
12      <link rel="stylesheet" href="{{ asset('adminlte/plugins/fontawesome-free/css/all.min.css') }}">
13
14      <!-- iCheck bootstrap -->
15      <link rel="stylesheet" href="{{ asset('adminlte/plugins/icheck-bootstrap/icheck-bootstrap.min.css') }}">
16
17      <!-- SweetAlert2 -->
18      <link rel="stylesheet" href="{{ asset('adminlte/plugins/sweetalert2-theme-bootstrap-4/bootstrap-4.min.css') }}">
19
20      <!-- Theme style -->
21      <link rel="stylesheet" href="{{ asset('adminlte/dist/css/adminlte.min.css') }}">
22  </head>
23  <body class="hold-transition login-page">
24      <div class="login-box">
25          <div class="card card-outline card-primary">
26              <div class="card-header text-center">
27                  <a href="{{ url('/') }}" class="h1"><b>Admin</b></a>
28              </div>
29              <div class="card-body">
30                  <p class="login-box-msg">Sign in to start your session</p>
31
32                  <form action="{{ url('login') }}" method="POST" id="form-login">
```

5.

```
PWL_POS > routes > web.php > ...
1  <?php
2
3  use App\Http\Controllers\UserController;
4  use App\Http\Controllers\SupplierController;
5  use App\Http\Controllers\BarangController;
6  use App\Http\Controllers\AuthController;
7  use App\Http\Controllers\KategoriController;
8  use App\Http\Controllers\LevelController;
9  use App\Http\Controllers>WelcomeController;
10 use Illuminate\Support\Facades\Route;
11
12 /*
13 |-----|
14 | Web Routes
15 |-----|
16 |
17 | Here is where you can register web routes for your application. These
18 | routes are loaded by the RouteServiceProvider and all of them will
19 | be assigned to the "web" middleware group. Make something great!
20 |
21 */
22
23 Route::pattern('id', '[0-9]+');
24
25 Route::get('login', [AuthController::class, 'login'])->name('login');
26 Route::post('login', [AuthController::class, 'postlogin']);
27 Route::post('logout', [AuthController::class, 'logout'])->middleware('auth');
28
```

6.



127.0.0.1:8000/login

AdminLTE

Sign in to start your session

Username ⓘ ⓘ

This field is required.

Password ⓘ ⓘ

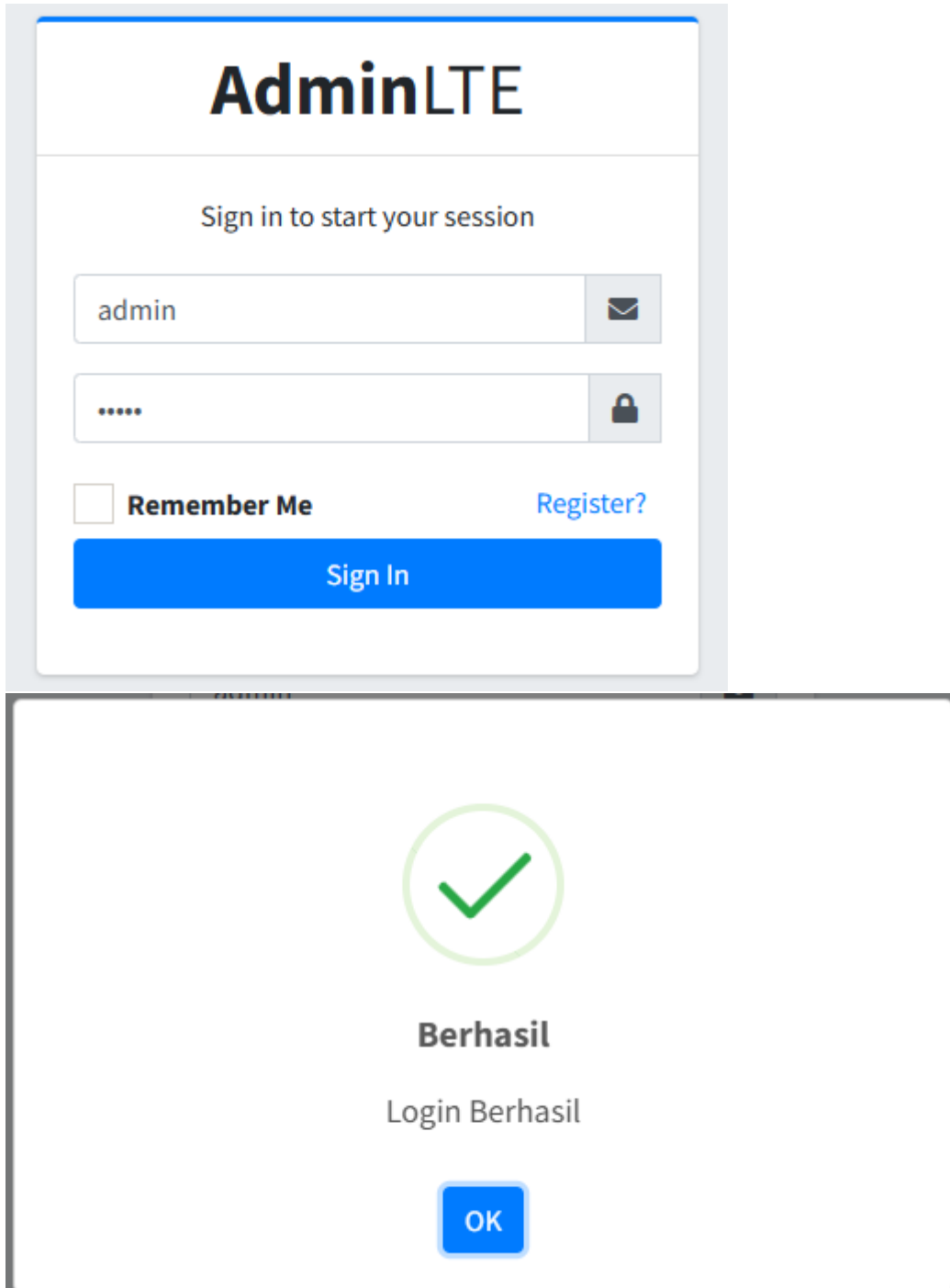
This field is required.

☐ Remember Me [Register?](#)

[Sign In](#)

Tugas 1

1.



The image displays two screenshots of the AdminLTE application. The top screenshot shows the login page with the title 'AdminLTE' and the instruction 'Sign in to start your session'. It features input fields for a username (containing 'admin') and a password (masked with dots), each with an icon (envelope and lock respectively). Below these are a 'Remember Me' checkbox and a 'Register?' link. A prominent blue 'Sign In' button is at the bottom. The bottom screenshot shows a success message with a large green checkmark icon, the text 'Berhasil' (Successful), and 'Login Berhasil' (Login Successful). A blue 'OK' button is centered at the bottom.

AdminLTE

Sign in to start your session

admin

.....

☐ **Remember Me** [Register?](#)

Sign In

Berhasil

Login Berhasil

OK

2.



Praktikum 2: Implementasi Authorization di Laravel dengan Middleware

1.

```
10 class UserModel extends Authenticatable
11 {
12     use HasFactory;
13
14     protected $table = 'm_user';
15     protected $primaryKey = 'user_id';
16     protected $fillable = ['username', 'password', 'nama', 'level_id', 'created_at', 'updated_at'];
17
18     protected $hidden = ['password'];
19
20     protected $casts = ['password' => 'hashed'];
21
22     public function level(): BelongsTo
23     {
24         return $this->belongsTo(LevelModel::class, 'level_id', 'level_id');
25     }
26     /**
27      * Mendapatkan nama role
28      */
29     public function getRoleName(): string
30     {
31         return $this->level->level_nama;
32     }
33
34     /**
35      * Cek apakah user memiliki role tertentu
36      */
37     public function hasRole($role): bool
38     {
39         return $this->level->level_kode == $role;
40     }
```

2.

```
PS C:\laragon\www\PWL_25\PWL_POS> php artisan make:middleware AuthorizeUser
```

INFO Middleware [C:\laragon\www\PWL_25\PWL_POS\app\Http\Middleware\AuthorizeUser.php] created successfully.

3.

```
PWL_POS > app > Http > Middleware > AuthorizeUser.php > PHP Intelephense > AuthorizeUser > handle
1  <?php
2
3  namespace App\Http\Middleware;
4
5  use Closure;
6  use Illuminate\Http\Request;
7  use Symfony\Component\HttpFoundation\Response;
8
9  class AuthorizeUser
10 {
11     /**
12      * Handle an incoming request.
13      *
14      * @param  \Closure(\Illuminate\Http\Request): (\Symfony\Component\HttpFoundation\Response)  $next
15      */
16     public function handle(Request $request, Closure $next, $role = ''): Response
17     {
18         $user = $request->user(); // ambil data user yg login
19         if ($user->hasRole($role)) { // fungsi user() diambil dari UserModel.php
20             return $next($request); // cek apakah user punya role yg diinginkan
21         }
22
23         // jika tidak punya role, maka tampilkan error 403
24         abort(403, 'Forbidden. Kamu tidak punya akses ke halaman ini');
25     }
26 }
27
28
```

4.

```

55     protected $middlewareAliases = [
56         'auth' => \App\Http\Middleware\Authenticate::class,
57         'authorize' => \App\Http\Middleware\AuthorizeUser::class, // middleware yg kita buat
58         'auth.basic' => \Illuminate\Auth\Middleware\AuthenticateWithBasicAuth::class,
59         'auth.session' => \Illuminate\Session\Middleware\AuthenticateSession::class,
60         'cache.headers' => \Illuminate\Http\Middleware\SetCacheHeaders::class,
61         'can' => \Illuminate\Auth\Middleware\Authorize::class,
62         'guest' => \App\Http\Middleware\RedirectIfAuthenticated::class,
63         'password.confirm' => \Illuminate\Auth\Middleware\RequirePassword::class,
64         'precognitive' => \Illuminate\Foundation\Http\Middleware\HandlePrecognitiveRequests::class,
65         'signed' => \App\Http\Middleware\ValidateSignature::class,
66         'throttle' => \Illuminate\Routing\Middleware\ThrottleRequests::class,
67         'verified' => \Illuminate\Auth\Middleware\EnsureEmailIsVerified::class,
68     ];
69 }

```

5.

		level_id	level_kode	level_nama	created_at	updated_at
<input type="checkbox"/>	Edit Copy Delete	1	ADM	Administrator	NULL	NULL
<input type="checkbox"/>	Edit Copy Delete	2	MNG	Manager	NULL	NULL
<input type="checkbox"/>	Edit Copy Delete	3	STF	Staff/Kasir	NULL	NULL

6.

```

30 Route::middleware('auth')->group(function () {
31     Route::get('/', [WelcomeController::class, 'index']);
32
33     Route::middleware(['authorize:ADM'])->group(function(){
34         Route::group(['prefix' => 'barang'], function () {
35             Route::get('/create',[BarangController::class, 'create']);
36             Route::post('/',[BarangController::class, 'store']);
37             Route::get('/create_ajax',[BarangController::class, 'create_ajax']);
38             Route::post('/ajax',[BarangController::class, 'store_ajax']);
39             Route::get('/{id}/edit',[BarangController::class, 'edit']);
40             Route::put('/{id}',[BarangController::class, 'update']);
41             Route::get('/{id}/edit_ajax',[BarangController::class, 'edit_ajax']);
42             Route::put('/{id}/update_ajax',[BarangController::class, 'update_ajax']);
43             Route::delete('/{id}',[BarangController::class, 'destroy']);
44             Route::get('/{id}/delete_ajax',[BarangController::class, 'confirm_ajax']);
45             Route::delete('/{id}/delete_ajax',[BarangController::class, 'delete_ajax']);
46         });

```

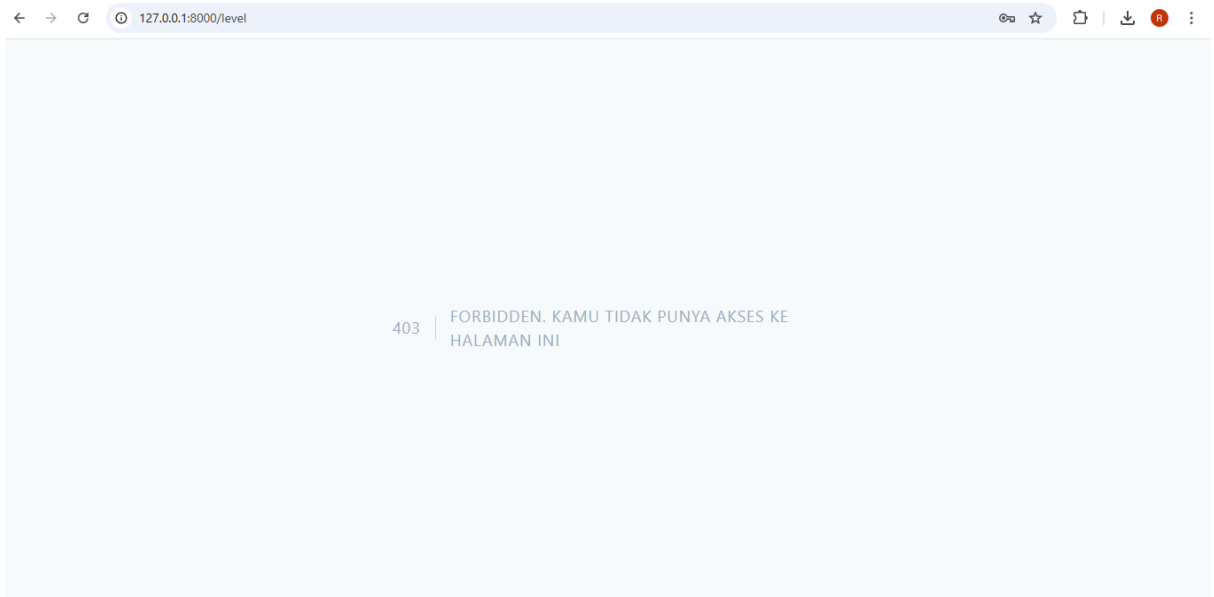
7.

AdminLTE

Sign in to start your session

☐ Remember Me
 [Register?](#)

Sign In



Tugas 2

1. Jika menggunakan authorize, pada sebuah route, akses ke halaman tersebut hanya dapat dilakukan oleh user dengan peran tertentu. Jika user tidak memiliki peran yang sesuai, maka akses akan ditolak dan user akan diarahkan ke halaman error 403

Praktikum 3: Implementasi Multi-Level Authorization di Laravel dengan Middleware

1.

```
27      /**
28       * Mendapatkan nama role
29       */
30      public function getRoleName(): string
31      {
32          return $this->level->level_nama;
33      }
34
35      /**
36       * Cek apakah user memiliki role tertentu
37       */
38      public function hasRole($role): bool
39      {
40          return $this->level->level_kode == $role;
41      }
42      /**
43       * Mendapatkan kode role
44       */
45      public function getRole()
46      {
47          return $this->level->level_kode;
48      }
49
50  }
```

2.

```
PWL_POS > app > Http > Middleware > AuthorizeUser.php > PHP Intelephense > AuthorizeUser > handle
1  <?php
2
3  namespace App\Http\Middleware;
4
5  use Closure;
6  use Illuminate\Http\Request;
7  use Symfony\Component\HttpFoundation\Response;
8
9  class AuthorizeUser
10 {
11     /**
12      * Handle an incoming request.
13      *
14      * @param  \Closure(\Illuminate\Http\Request): (\Symfony\Component\HttpFoundation\Response)
15      */
16     public function handle(Request $request, Closure $next, ... $roles): Response
17     {
18         $user_role = $request->user()->getRole(); // Ambil data level_kode dari user yang login
19
20         if (in_array($user_role, $roles)) { // Cek apakah level_kode user ada di dalam array role
21             return $next($request); // Jika ada, maka lanjutkan request
22         }
23
24         // jika tidak punya role, maka tampilkan error 403
25         abort(403, 'Forbidden. Kamu tidak punya akses ke halaman ini');
26     }
27 }
```

3.

```
Route::middleware(['authorize:ADM, MNG'])->group(function(){
    Route::group(['prefix' => 'barang'], function () {
        Route::get('/create',[BarangController::class, 'create']);
        Route::post('/',[BarangController::class, 'store']);
        Route::get('/create_ajax',[BarangController::class, 'create_ajax']);
        Route::post('/ajax',[BarangController::class, 'store_ajax']);
        Route::get('/{id}/edit',[BarangController::class, 'edit']);
        Route::put('/{id}',[BarangController::class, 'update']);
        Route::get('/{id}/edit_ajax',[BarangController::class, 'edit_ajax']);
        Route::put('/{id}/update_ajax',[BarangController::class, 'update_ajax']);
        Route::delete('/{id}',[BarangController::class, 'destroy']);
        Route::get('/{id}/delete_ajax',[BarangController::class, 'confirm_ajax']);
        Route::delete('/{id}/delete_ajax',[BarangController::class, 'delete_ajax']);
    });
});
```


Tugas 3

1. AAAA

- Admin

AdminLTE

Sign in to start your session

☐ Remember Me

Register?

Sign In

127.0.0.1:8000/barang

PWL - Starter Code

Home Contact

Search

Dashboard

Data Pengguna

Level User

Data User

Data Barang

Kategori Barang

Data Supplier

Data Barang

Data Transaksi

Stok Barang

Transaksi Penjualan

Daftar Barang

Home / Barang

Daftar barang yang terdaftar dalam sistem

Tambah Tambah Ajax

Filter: - Semua -

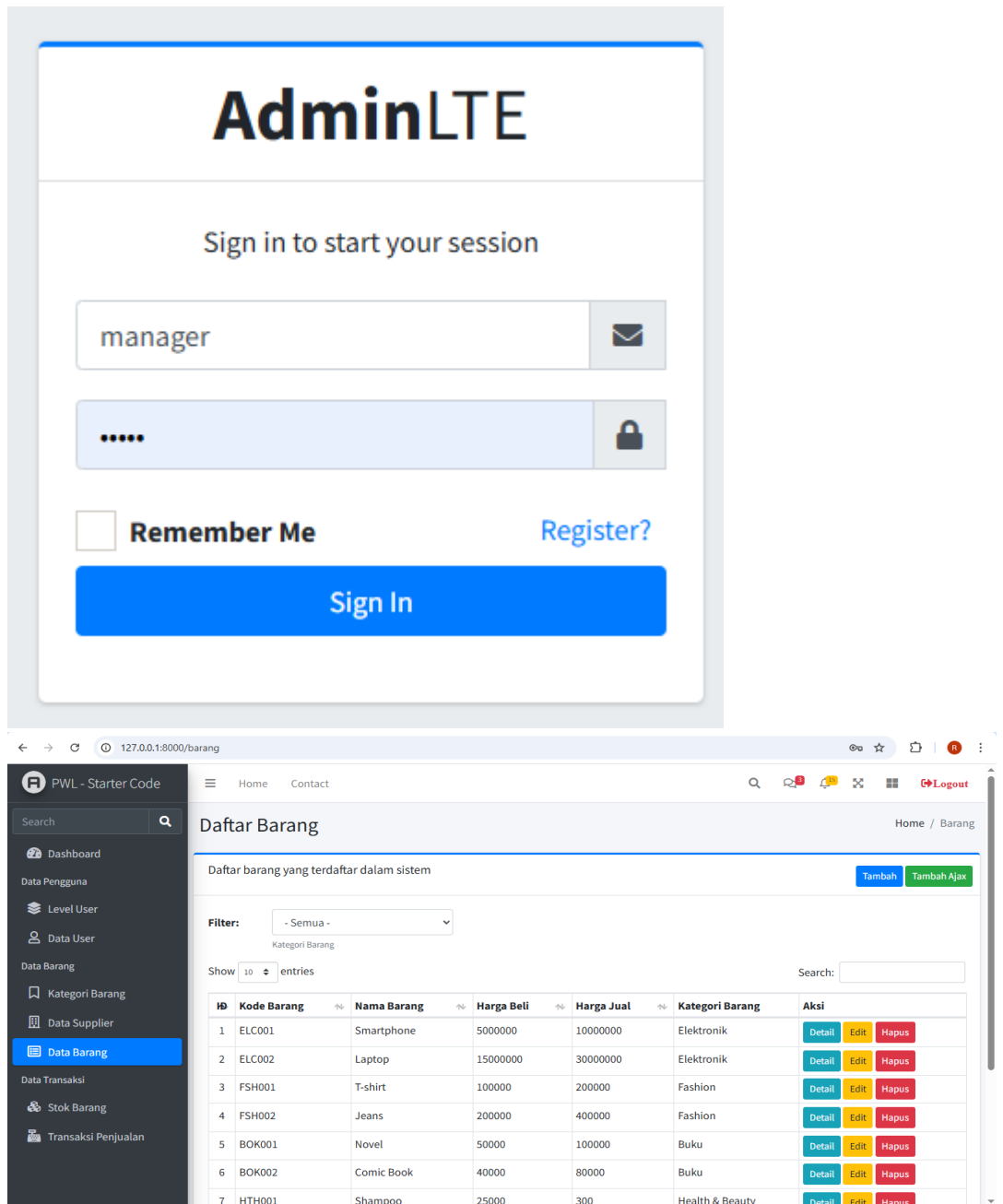
Kategori Barang

Show 10 entries

Search:

ID	Kode Barang	Nama Barang	Harga Beli	Harga Jual	Kategori Barang	Aksi
1	ELC001	Smartphone	5000000	10000000	Elektronik	Detail Edit Hapus
2	ELC002	Laptop	15000000	30000000	Elektronik	Detail Edit Hapus
3	FSH001	T-shirt	100000	200000	Fashion	Detail Edit Hapus
4	FSH002	Jeans	200000	400000	Fashion	Detail Edit Hapus
5	BOK001	Novel	50000	100000	Buku	Detail Edit Hapus
6	BOK002	Comic Book	40000	80000	Buku	Detail Edit Hapus
7	HTH001	Shampoo	25000	300	Health & Beauty	Detail Edit Hapus

- Manager



- Staff

AdminLTE

Sign in to start your session

✉

🔒

☐ **Remember Me**
[Register?](#)

Sign In

403

FORBIDDEN. KAMU TIDAK PUNYA AKSES KE HALAMAN INI

2. Laporan
- 3.

```
Route::middleware(['authorize:ADM'])->group(function(){
    Route::group(['prefix' => 'barang'], function () {
        Route::get('/',[BarangController::class, 'create']);
        Route::post('/',[BarangController::class, 'store']);
        Route::get('/create_ajax',[BarangController::class, 'create_ajax']);
        Route::post('/ajax',[BarangController::class, 'store_ajax']);
        Route::get('/{id}/edit',[BarangController::class, 'edit']);
        Route::put('/{id}',[BarangController::class, 'update']);
        Route::get('/{id}/edit_ajax',[BarangController::class, 'edit_ajax']);
        Route::put('/{id}/update_ajax',[BarangController::class, 'update_ajax']);
        Route::delete('/{id}',[BarangController::class, 'destroy']);
        Route::get('/{id}/delete_ajax',[BarangController::class, 'confirm_ajax']);
        Route::delete('/{id}/delete_ajax',[BarangController::class, 'delete_ajax']);
    });

    Route::group(['prefix' => 'user'], function () {
        Route::get('/',[UserController::class, 'index']);
        Route::post('/list',[UserController::class, 'list']);
        Route::get('/create',[UserController::class, 'create']);
        Route::post('/',[UserController::class, 'store']);
        Route::get('/create_ajax',[UserController::class, 'create_ajax']);
        Route::post('/ajax',[UserController::class, 'store_ajax']);
        Route::get('/{id}',[UserController::class, 'show']);
        Route::get('/{id}/edit',[UserController::class, 'edit']);
        Route::put('/{id}',[UserController::class, 'update']);
        Route::get('/{id}/edit_ajax',[UserController::class, 'edit_ajax']);
        Route::put('/{id}/update_ajax',[UserController::class, 'update_ajax']);
        Route::delete('/{id}',[UserController::class, 'destroy']);
        Route::get('/{id}/delete_ajax',[UserController::class, 'confirm_ajax']);
        Route::delete('/{id}/delete_ajax',[UserController::class, 'delete_ajax']);
    });
});
```

```

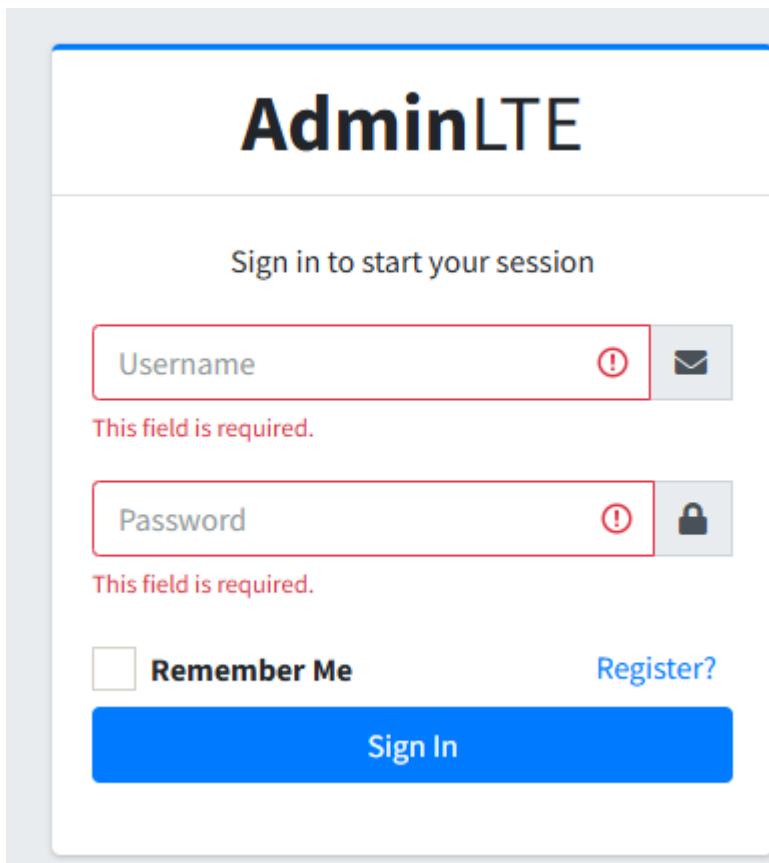
Route::middleware(['authorize:ADM,MNG'])->group(function(){
    Route::group(['prefix' => 'supplier'], function () {
        Route::get('/',[SupplierController::class, 'index']);
        Route::post('/list',[SupplierController::class, 'list']);
        Route::get('/{id}',[SupplierController::class, 'show']);
    });
});

Route::middleware(['authorize:ADM,MNG'])->group(function(){
    Route::group(['prefix' => 'barang'], function () {
        Route::get('/',[BarangController::class, 'index']);
        Route::post('/list',[BarangController::class, 'list']);
        Route::get('/{id}',[BarangController::class, 'show']);
    });
});

```

Tugas 4

1. Form registrasi
- 2.



The image shows a login form for AdminLTE. It features a white card with a blue border on a light gray background. The card has a title 'AdminLTE' in bold black font. Below the title is the text 'Sign in to start your session'. There are two input fields: 'Username' and 'Password'. Both fields have a red border and a red exclamation mark icon, indicating they are required. The 'Username' field also has an envelope icon, and the 'Password' field has a lock icon. Below each field is the text 'This field is required.' in red. There is a checkbox labeled 'Remember Me' and a link 'Register?' in blue. At the bottom is a large blue button labeled 'Sign In'.

AdminLTE

Manager



manager66



Manager 66



.....



.....



Have an account? [Login](#)

Register



Registration Successful

Registration Success.

OK

AdminLTE

Sign in to start your session

manager66
xxxxxxxxxx



.....

☐

Remember Me

[Register?](#)

Sign In



Berhasil

Login Berhasil

OK