**Project Membuat Aplikasi Gallery Foto ( UKK Paket III )**

**SMK Telkom Lampung**

**Step 2 – Menerapkan Proses Autentikasi Pada Aplikasi**

Berdasarkan kebutuhan Anda untuk menerapkan proses autentikasi pada aplikasi Gallery Foto Anda, dengan mengingat bahwa Anda sebelumnya telah menambahkan kolom username dan alamat pada tabel users, kita akan menyesuaikan langkah-langkah implementasi autentikasi sesuai dengan struktur tabel users Anda. Ini termasuk penyesuaian pada model User dan proses pendaftaran untuk mengakomodasi kolom baru tersebut.

1. **Menyesuaikan Model User**

Pertama, kita perlu memastikan bahwa model User mencakup kolom username dan alamat yang Anda tambahkan. Buka file model User yang terletak di app/Models/User.php dan sesuaikan seperti berikut:

***<?php***

**namespace** **App\Models;**

*// use Illuminate\Contracts\Auth\MustVerifyEmail;*

**use** Illuminate**\**Database**\**Eloquent**\**Factories**\**HasFactory**;**

**use** Illuminate**\**Foundation**\**Auth**\**User **as** *Authenticatable***;**

**use** Illuminate**\**Notifications**\**Notifiable**;**

**use** Laravel**\**Sanctum**\**HasApiTokens**;**

class **User** *extends* *Authenticatable*

{

**use** HasApiTokens**,** HasFactory**,** Notifiable**;**

*/\*\**

*\* The attributes that are mass assignable.*

*\**

*\* @var array<int, string>*

*\*/*

*protected* $fillable **=** [

        'name'**,**

        'email'**,**

        'password'**,**

        'username'**,** *//ditambahkan*

        'alamat' *//ditambahkan*

    ]**;**

*/\*\**

*\* The attributes that should be hidden for serialization.*

*\**

*\* @var array<int, string>*

*\*/*

*protected* $hidden **=** [

        'password'**,**

        'remember\_token'**,**

    ]**;**

*/\*\**

*\* The attributes that should be cast.*

*\**

*\* @var array<string, string>*

*\*/*

*protected* $casts **=** [

        'email\_verified\_at' **=>** 'datetime'**,**

        'password' **=>** 'hashed'**,**

    ]**;**

}

Dengan menambahkan username dan alamat ke array $fillable, Anda memungkinkan atribut-atribut tersebut untuk ditugaskan secara massal, yang berguna saat proses pendaftaran.

1. **Membuat Controller Autentikasi**

Buat AuthController yang akan menangani logika untuk pendaftaran, login, dan logout.

php artisan make:controller AuthController

1. **Implementasi Pendaftaran**

Tambahkan method di AuthController untuk menampilkan formulir pendaftaran dan untuk menangani proses pendaftaran.

Silahkan tuliskan method dibawah ini :

***<?php***

**namespace** **App\Http\Controllers;**

**use** App**\**Models**\**User**;**

**use** Illuminate**\**Support**\**Facades**\**Auth**;**

**use** Illuminate**\**Support**\**Facades**\**Hash**;**

**use** Illuminate**\**Http**\**Request**;**

class **AuthController** *extends* *Controller*

{

*public* function signupForm()

    {

**return** view('auth.signup')**;**

    }

*public* function signup(Request **$request**)

    {

        $request**->**validate([

            'name' **=>** 'required|string|max:255'**,**

            'email' **=>** 'required|string|email|max:255|unique:users'**,**

            'password' **=>** 'required|string|min:8'**,**

            'username' **=>** 'required|string|max:255|unique:users'**,**

            'alamat' **=>** 'required|string'**,**

        ])**;**

        $user **=** User**::**create([

            'name' **=>** $request**->***name***,**

            'email' **=>** $request**->***email***,**

            'password' **=>** Hash**::**make($request**->***password*)**,**

            'username' **=>** $request**->***username***,**

            'alamat' **=>** $request**->***alamat***,**

        ])**;**

        Auth**::**login($user)**;**

**return** redirect()**->**route('home')**;**

    }

}

1. **Mendefinisikan Route**

Definisikan route untuk pendaftaran di routes/web.php.

***<?php***

**use** Illuminate**\**Support**\**Facades**\**Route**;**

**use** App**\**Http**\**Controllers**\**AuthController**;**

**use** App**\**Http**\**Controllers**\**HomeController**;**

*/\**

*|--------------------------------------------------------------------------*

*| Web Routes*

*|--------------------------------------------------------------------------*

*|*

*| Here is where you can register web routes for your application. These*

*| routes are loaded by the RouteServiceProvider and all of them will*

*| be assigned to the "web" middleware group. Make something great!*

*|*

*\*/*

Route**::**get('/'**,** [HomeController**::class,** 'index'])**->**name('home')**;** *//Mendefinisikan route untuk home*

Route**::**get('/home'**,** [HomeController**::class,** 'index'])**->**name('home')**;** *//Mendefinisikan route untuk home*

Route**::**get('signup'**,** [AuthController**::class,** 'signupForm'])**->**name('auth.signupForm')**;** *//Route untuk menampilkan halaman pendaftaran*

Route**::**post('signup'**,** [AuthController**::class,** 'signup'])**->**name('auth.signup')**;** *//Route untuk menangani proses pendaftaran*

1. **Membuat View Pendaftaran**

Buat view signup.blade.php di dalam folder resources/views/auth. Pastikan form di view mengirim data ke route pendaftaran yang telah Anda definisikan.

Silahkan tulis kode dibawah ini didalam file signup.blade.php anda :

**@extends**('layouts.app')

**@section**('title'**,** 'Signup')

**@section**('content')

<div *class***=**"container">

    <div *class***=**"row justify-content-center">

        <div *class***=**"col-md-8">

            <div *class***=**"card">

                <div *class***=**"card-header">{{ \_\_('Register') }}</div>

                <div *class***=**"card-body">

                    <form *method***=**"POST" *action***=**"{{ route('auth.signup') }}">

**@csrf**

                        <div *class***=**"form-group row">

                            <label *for***=**"name" *class***=**"col-md-4 col-form-label text-md-right">{{ \_\_('Name') }}</label>

                            <div *class***=**"col-md-6">

                                <input ***id*=**"name" *type***=**"text" *class***=**"form-control **@error**('name') is-invalid **@enderror**" *name***=**"name" *value***=**"{{ old('name') }}" *required* *autocomplete***=**"name" *autofocus*>

**@error**('name')

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

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

                                    </span>

**@enderror**

                            </div>

                        </div>

                        <div *class***=**"form-group row">

                            <label *for***=**"email" *class***=**"col-md-4 col-form-label text-md-right">{{ \_\_('E-Mail Address') }}</label>

                            <div *class***=**"col-md-6">

                                <input ***id*=**"email" *type***=**"email" *class***=**"form-control **@error**('email') is-invalid **@enderror**" *name***=**"email" *value***=**"{{ old('email') }}" *required* *autocomplete***=**"email">

**@error**('email')

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

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

                                    </span>

**@enderror**

                            </div>

                        </div>

                        <div *class***=**"form-group row">

                            <label *for***=**"username" *class***=**"col-md-4 col-form-label text-md-right">{{ \_\_('Username') }}</label>

                            <div *class***=**"col-md-6">

                                <input ***id*=**"username" *type***=**"text" *class***=**"form-control **@error**('username') is-invalid **@enderror**" *name***=**"username" *value***=**"{{ old('username') }}" *required* *autocomplete***=**"username">

**@error**('username')

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

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

                                    </span>

**@enderror**

                            </div>

                        </div>

                        <div *class***=**"form-group row">

                            <label *for***=**"alamat" *class***=**"col-md-4 col-form-label text-md-right">{{ \_\_('Address') }}</label>

                            <div *class***=**"col-md-6">

                                <textarea ***id*=**"alamat" *class***=**"form-control **@error**('alamat') is-invalid **@enderror**" *name***=**"alamat" *required*>{{ old('alamat') }}</textarea>

**@error**('alamat')

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

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

                                    </span>

**@enderror**

                            </div>

                        </div>

                        <div *class***=**"form-group row">

                            <label *for***=**"password" *class***=**"col-md-4 col-form-label text-md-right">{{ \_\_('Password') }}</label>

                            <div *class***=**"col-md-6">

                                <input ***id*=**"password" *type***=**"password" *class***=**"form-control **@error**('password') is-invalid **@enderror**" *name***=**"password" *required* *autocomplete***=**"new-password">

**@error**('password')

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

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

                                    </span>

**@enderror**

                            </div>

                        </div>

                        <div *class***=**"form-group row mb-0">

                            <div *class***=**"col-md-6 offset-md-4">

                                <button *type***=**"submit" *class***=**"btn btn-primary">

                                    {{ \_\_('Register') }}

                                </button>

                            </div>

                        </div>

                    </form>

                </div>

            </div>

        </div>

    </div>

</div>

**@endsection**

1. **Membuat Layouts**

Dalam contoh di atas, saya menggunakan layout app yang biasanya sudah termasuk dalam instalasi Laravel sebagai bagian dari Blade templating. Anda mungkin perlu menyesuaikan @extends('layouts.app') sesuai dengan struktur layout aplikasi Anda. Pastikan Anda juga telah mendefinisikan route auth.signup di file route web Anda.

Formulir ini menggunakan Bootstrap untuk styling (yang biasanya termasuk dalam scaffold autentikasi Laravel) dan menyertakan validasi kesalahan untuk setiap field untuk memberikan umpan balik kepada pengguna jika terjadi kesalahan saat pengisian form. Pastikan Anda telah menambahkan validasi yang sesuai di AuthController untuk menangani data masukan ini secara aman.

Anda dapat membuat file layout baru di dalam direktori resources/views/layouts bernama app.blade.php. Jika direktori layouts belum ada, silahkan buat terlebih dahulu.

Berikut adalah contoh sederhana dari file app.blade.php yang mencakup Bootstrap, Font Awesome, dan jQuery:

Silahkan tulis kode berikut ini didalam file app.blade.php :

<!DOCTYPE *html*>

<html *lang***=**"en">

<head>

    <meta *charset***=**"UTF-8">

    <meta *name***=**"viewport" *content***=**"width=device-width, initial-scale=1.0">

    <title>**@yield**('title'**,** 'Your App Name')</title>

*<!-- Bootstrap CSS -->*

    <link *rel***=**"stylesheet" *href***=**"https://stackpath.bootstrapcdn.com/bootstrap/4.5.2/css/bootstrap.min.css">

*<!-- Font Awesome -->*

    <link *rel***=**"stylesheet" *href***=**"https://cdnjs.cloudflare.com/ajax/libs/font-awesome/5.15.1/css/all.min.css">

*<!-- Custom CSS -->*

    <style>

        body {

            padding-top**:** 5rem**;**

        }

*.starter-template* {

            padding**:** 3rem 1.5rem**;**

            text-align**:** center**;**

        }

    </style>

</head>

<body>

<nav *class***=**"navbar navbar-expand-md navbar-dark bg-dark fixed-top">

    <a *class***=**"navbar-brand" *href***=**"#">Photo Gallery</a>

    <button *class***=**"navbar-toggler" *type***=**"button" *data-toggle***=**"collapse" *data-target***=**"#navbarsExampleDefault" *aria-controls***=**"navbarsExampleDefault" *aria-expanded***=**"false" *aria-label***=**"Toggle navigation">

        <span *class***=**"navbar-toggler-icon"></span>

    </button>

    <div *class***=**"collapse navbar-collapse" ***id*=**"navbarsExampleDefault">

        <ul *class***=**"navbar-nav mr-auto">

            <li *class***=**"nav-item active">

                <a *class***=**"nav-link" *href***=**"#">Home <span *class***=**"sr-only">(current)</span></a>

            </li>

            <li *class***=**"nav-item">

                <a *class***=**"nav-link" *href***=**"/album">Album</a>

            </li>

            <li *class***=**"nav-item">

                <a *class***=**"nav-link" *href***=**"/signup">Signup</a>

            </li>

        </ul>

    </div>

</nav>

<main *role***=**"main" *class***=**"container">

    <div *class***=**"starter-template">

**@yield**('content')

    </div>

</main>

*<!-- jQuery and Bootstrap Bundle (includes Popper) -->*

<script *src***=**"https://code.jquery.com/jquery-3.5.1.slim.min.js"></script>

<script *src***=**"https://stackpath.bootstrapcdn.com/bootstrap/4.5.2/js/bootstrap.bundle.min.js"></script>

</body>

</html>

Dalam template ini, kami menggunakan Bootstrap 4.5.2 dan Font Awesome 5.15.1 dari CDN untuk memastikan kami memiliki versi yang stabil dan terpercaya. jQuery juga dimasukkan untuk kemungkinan penggunaan skrip yang memerlukannya di masa mendatang.

Untuk menggunakan layout ini dalam view Anda, cukup gunakan @extends('layouts.app') di awal file Blade view Anda dan masukkan konten spesifik halaman dalam @section('content'). Contoh penggunaannya akan seperti ini:

@extends('layouts.app')

@section('title', 'Page Title')

@section('content')

    <h1>Page Content Here</h1>

*<!-- Konten lainnya -->*

@endsection

Ini akan menginjeksi konten spesifik halaman ke dalam layout yang telah ditentukan, memungkinkan Anda untuk dengan mudah mempertahankan konsistensi dan mengurangi duplikasi markup HTML di seluruh aplikasi Anda.

1. **Mendefinisikan Route untuk Halaman Utama**

Jika anda mengikuti modul ini dari awal pastinya sudah mendefinisikan route untuk halaman utama (Home) didalam web.php, anda dapat memerika baris kode berikut didalam file web.php :

use Illuminate\Support\Facades\Route;

use App\Http\Controllers\HomeController; // Pastikan menggunakan HomeController yang tepat

Route::get('/', [HomeController::class, 'index'])->name('home');

Jika sudah ada file diatas berarti anda telah mendefinisikan route untuk halaman utama atau home, silahkan lanjutkan Langkah berikutnya.

1. **Membuat Controller HomeController**

Buat HomeController dengan menggunakan Artisan command jika belum ada:

php artisan make:controller HomeController

Dalam HomeController, tambahkan method index yang akan menampilkan view halaman utama:

Tulis kode beriktu didalam file HomeController.php :

***<?php***

**namespace** **App\Http\Controllers;**

**use** Illuminate**\**Http**\**Request**;**

class **HomeController** *extends* *Controller*

{

*public* function index()

    {

*// Sementara kita akan mengirimkan array kosong, nanti akan diisi dengan data foto*

**return** view('home'**,** ['photos' **=>** []])**;**

    }

}

1. **Membuat View untuk Halaman Utama**

Buat file view baru di resources/views/home.blade.php. Di dalam file ini, Anda bisa menyiapkan struktur HTML untuk menampilkan daftar foto. Untuk sementara, kita akan membuat tampilan dasar:

**@extends**('layouts.app')

**@section**('title'**,** 'Home Page')

**@section**('content')

<div *class***=**"container">

    <h1>Welcome to the Photo Gallery</h1>

    <div *class***=**"row">

**@forelse**($photos **as** $photo)

            <div *class***=**"col-md-4">

                <div *class***=**"card mb-4 shadow-sm">

                    <img *src***=**"{{ $photo**->***url* }}" *class***=**"bd-placeholder-img card-img-top" *width***=**"100%" *height***=**"225">

                    <div *class***=**"card-body">

                        <p *class***=**"card-text">{{ $photo**->***description* }}</p>

                    </div>

                </div>

            </div>

**@empty**

            <p>No photos to display.</p>

**@endforelse**

    </div>

</div>

**@endsection**

Dalam contoh di atas, saya menggunakan struktur Bootstrap untuk layout, namun Anda bisa menyesuaikannya sesuai dengan kebutuhan desain Anda. Bagian @forelse digunakan untuk iterasi setiap foto yang diterima dari controller. Jika belum ada foto, maka akan menampilkan "No photos to display."

Di tahap berikut nya kita akan menampilkan data photo dari dalam database yang telah di simpan oleh pengguna.

Silahkan coba buka halaman home dengan menjalankan server terlebih dahulu dengan cara menjalankan command berikut didalam terminal :

*php artisan serve*

silahkan akses url : <http://127.0.0.1:8000> maka akan muncul tampilan seperti berikut ini :

A screenshot of a computer

Description automatically generated

Anda juga dapat mencoba mengakses halaman signup dan melakukan pendaftaran sendiri, silahkan akses url berikut ini http:127.0.0.1:8000/signup maka akan tampil seperti berikut ini, silahkan isikan data pendaftaran untuk mencoba apakah aplikasi dapat berjalan dengan baik.

A screenshot of a computer

Description automatically generated

1. **Membuat dan Mendefinisikan Halaman Login**

Untuk membuat halaman login, Anda akan melalui beberapa langkah yang mirip dengan pembuatan halaman pendaftaran. Langkah-langkah ini termasuk membuat view untuk formulir login, mendefinisikan route yang sesuai, dan menambahkan metode di AuthController untuk menangani proses login.

1. Mendefinisikan Route untuk Login

Pertama, definisikan route untuk menampilkan formulir login dan juga untuk menangani proses login di routes/web.php. Anda mungkin sudah memiliki route ini jika Anda mengikuti langkah-langkah sebelumnya:

Tambahkan kode dibawah ini didalam file web.php

Route::get('login', [AuthController::class, 'loginForm'])->name('auth.loginForm');

Route::post('login', [AuthController::class, 'login'])->name('auth.login');

1. Menambahkan Metode di AuthController

Tambahkan dua metode di AuthController: satu untuk menampilkan formulir login dan satu lagi untuk menangani proses login.

*public* function loginForm()

    {

**return** view('auth.login')**;**

    }

*public* function login(Request **$request**)

    {

        $credentials **=** $request**->**validate([

            'email' **=>** ['required'**,** 'email']**,**

            'password' **=>** ['required']**,**

        ])**;**

**if** (Auth**::**attempt($credentials)) {

            $request**->**session()**->**regenerate()**;**

**return** redirect()**->**intended('home')**;**

        }

**return** back()**->**withErrors([

            'email' **=>** 'The provided credentials do not match our records.'**,**

        ])**->**onlyInput('email')**;**

    }

Pastikan untuk sudah mengimpor Illuminate\Http\Request dan Illuminate\Support\Facades\Auth di awal file.

1. Membuat View Login

Buat file view baru di resources/views/auth/login.blade.php. Di dalam file ini, Anda akan membuat formulir login yang mengirimkan data ke route login yang telah Anda definisikan.

**@extends**('layouts.app')

**@section**('title'**,** 'Login')

**@section**('content')

<div *class***=**"container">

    <div *class***=**"row justify-content-center">

        <div *class***=**"col-md-8">

            <div *class***=**"card">

                <div *class***=**"card-header">{{ \_\_('Login') }}</div>

                <div *class***=**"card-body">

                    <form *method***=**"POST" *action***=**"{{ route('auth.login') }}">

**@csrf**

                        <div *class***=**"form-group row">

                            <label *for***=**"email" *class***=**"col-md-4 col-form-label text-md-right">{{ \_\_('E-Mail Address') }}</label>

                            <div *class***=**"col-md-6">

                                <input ***id*=**"email" *type***=**"email" *class***=**"form-control **@error**('email') is-invalid **@enderror**" *name***=**"email" *value***=**"{{ old('email') }}" *required* *autocomplete***=**"email" *autofocus*>

**@error**('email')

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

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

                                    </span>

**@enderror**

                            </div>

                        </div>

                        <div *class***=**"form-group row">

                            <label *for***=**"password" *class***=**"col-md-4 col-form-label text-md-right">{{ \_\_('Password') }}</label>

                            <div *class***=**"col-md-6">

                                <input ***id*=**"password" *type***=**"password" *class***=**"form-control **@error**('password') is-invalid **@enderror**" *name***=**"password" *required* *autocomplete***=**"current-password">

**@error**('password')

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

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

                                    </span>

**@enderror**

                            </div>

                        </div>

                        <div *class***=**"form-group row mb-0">

                            <div *class***=**"col-md-8 offset-md-4">

                                <button *type***=**"submit" *class***=**"btn btn-primary">

                                    {{ \_\_('Login') }}

                                </button>

**@if** (Route**::**has('auth.signupForm'))

                                    <a *class***=**"btn btn-link" *href***=**"{{ route('auth.signupForm') }}">

                                        {{ \_\_('Don\'t have an account? Sign Up') }}

                                    </a>

**@endif**

                            </div>

                        </div>

                    </form>

                </div>

            </div>

        </div>

    </div>

</div>

**@endsection**

Dalam contoh di atas, form login sederhana telah dibuat menggunakan Bootstrap untuk styling. Formulir ini termasuk field untuk email dan password, serta tombol submit untuk mengirimkan formulir. Jika terjadi error validasi, error akan ditampilkan di bawah field yang bersangkutan.

Setelah langkah-langkah ini, Anda seharusnya memiliki halaman login yang berfungsi penuh yang mengizinkan pengguna untuk masuk ke aplikasi Anda. Setelah berhasil login, pengguna akan diarahkan ke halaman utama yang telah Anda tentukan sebelumnya.

Silahkan dicoba untuk melakukan login menggunakan data yang sebelumnya telah didaftarkan dengan menjalankan *php artisan serv* terlebih dahulu dan akses url berikut ini :

<http://127.0.0.1:8000/login> maka akan menampilkan halaman seperti dibawah ini :

A screenshot of a computer

Description automatically generated