**Step 1: Create a Controller**

php artisan make:controller PageController

* *This creates app/Http/Controllers/PageController.php*

**Step 2: Define Methods in Controller**

Modify PageController.php:

namespace App\Http\Controllers;

use Illuminate\Http\Request;

class PageController extends Controller

{

public function home()

{

return view('welcome');

}

public function about()

{

return "This is the About page";

}

public function contact()

{

return view('contact');

}

}

**Step 3: Define Routes in routes/web.php:**

use App\Http\Controllers\PageController;

Route::get('/', [PageController::class, 'home']);

Route::get('/about', [PageController::class, 'about']);

Route::get('/contact', [PageController::class, 'contact']);

* **Laravel API Routes (RESTful API Development)**

**Example: Create API for User Management**

**Step 1: Create an API Controller**

php artisan make:controller Api/UserController –api

* *This will generate a controller in app/Http/Controllers/Api/UserController.php.*

**Step 2: Define Methods in UserController.php**

namespace App\Http\Controllers\Api;

use App\Models\User;

use Illuminate\Http\Request;

use App\Http\Controllers\Controller;

use Illuminate\Support\Facades\Hash;

class UserController extends Controller

{

// Get all users

public function index()

{

return response()->json(User::all(), 200);

}

// Create a new user

public function store(Request $request)

{

$request->validate([

'name' => 'required',

'email' => 'required|unique:users',

'password' => 'required|min:6'

]);

$user = User::create([

'name' => $request->name,

'email' => $request->email,

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

]);

return response()->json($user, 201);

}

// Get a single user

public function show($id)

{

$user = User::find($id);

return $user ? response()->json($user, 200) : response()->json(['message' => 'User not found'], 404);

}

// Update user

public function update(Request $request, $id)

{

$user = User::find($id);

if (!$user) return response()->json(['message' => 'User not found'], 404);

$user->update($request->all());

return response()->json($user, 200);

}

// Delete user

public function destroy($id)

{

$user = User::find($id);

if (!$user) return response()->json(['message' => 'User not found'], 404);

$user->delete();

return response()->json(['message' => 'User deleted successfully'], 200);

}

}

**Step 3: Define API Routes in routes/api.php**

use App\Http\Controllers\Api\UserController;

Route::get('/users', [UserController::class, 'index']); // Get all users

Route::post('/users', [UserController::class, 'store']); // Create user

Route::get('/users/{id}', [UserController::class, 'show']); // Get single user

Route::put('/users/{id}', [UserController::class, 'update']); // Update user

Route::delete('/users/{id}', [UserController::class, 'destroy']); // Delete user

* **3. Middleware for API Security**

**Step 1: Create Middleware**

php artisan make:middleware ApiAuthMiddleware

**Step 2: Modify app/Http/Middleware/ApiAuthMiddleware.php**

namespace App\Http\Middleware;

use Closure;

use Illuminate\Http\Request;

class ApiAuthMiddleware

{

public function handle(Request $request, Closure $next)

{

if ($request->header('API\_KEY') !== env('API\_KEY')) {

return response()->json(['error' => 'Unauthorized'], 401);

}

return $next($request);

}

}

**Step 3: Register Middleware in app/Http/Kernel.php**

protected $routeMiddleware = [

'apiauth' => \App\Http\Middleware\ApiAuthMiddleware::class,

];

**Step 4: Apply Middleware to API Routes**

Modify routes/api.php:

Route::middleware('apiauth')->group(function () {

Route::get('/users', [UserController::class, 'index']);

Route::post('/users', [UserController::class, 'store']);

});