1. Install package

composer require spatie/laravel-permission

1. in config\app.php

    'providers' [

        Spatie\Permission\PermissionServiceProvider::class

]

1. in User.php Model

use Spatie\Permission\Traits\HasRoles ;

class User extends Authenticatable

{

    use HasApiTokens, HasFactory, Notifiable , HasRoles;

}

1. ***You should publish***[*the migration*](https://github.com/spatie/laravel-permission/blob/main/database/migrations/create_permission_tables.php.stub)*and the*[*config/permission.php config file*](https://github.com/spatie/laravel-permission/blob/main/config/permission.php)*with*

php artisan vendor:publish --provider="Spatie\Permission\PermissionServiceProvider"

1. Run the migrations

php artisan migrate

1. Create the permissions and roles controller

php artisan make:controller PermissionController

php artisan make:controller RoleController

1. To addPermissionRoRole

    public function addPermissionToRole(Request $request, Role $role)

    {

        $validatedData = $request->validate([

            'permissions' => ['required', 'array'],

            'permissions.\*' => ['string', 'exists:permissions,name' ],

        ]);

        $role->syncPermissions($validatedData);

        return response()->json([

            "message" => "Permissions added to role successfully",

        ], 201);

    }

1. AssignToleToUser

    public function addRoleToUser(Request $request, User $user)

    {

        $validatedData = $request->validate([

            'roles' => ['required', 'array'],

            'roles.\*' => ['string', 'exists:roles,name' ],

        ]);

        $user->syncRoles($validatedData);

        return response()->json([

            "message" => "Roles added to user successfully",

        ], 201);

    }

1. To protect routes by role or permission
2. This package comes with *RoleMiddleware*, *PermissionMiddleware* and *RoleOrPermissionMiddleware* middleware. In Laravel 10 you can add them in *app/Http/Kernel.php*:

    protected $middlewareAliases = [

        'role' => \Spatie\Permission\Middleware\RoleMiddleware::class,

        'permission' => \Spatie\Permission\Middleware\PermissionMiddleware::class,

        'role\_or\_permission' => \Spatie\Permission\Middleware\RoleOrPermissionMiddleware::class,

    ];

Route::post('categories', [CategoryController::class, 'store'])-> middleware(['role:admin']);