

GESTIÓN DEL BLOG ROL EDITOR

Proyecto LaraTech :: by JLuis Dev

Curso: Laravel Página Web Administrable | Módulo: Rol Editor

www.codea.app



Objetivo

- Permitir que el **Rol Editor** acceda al módulo de **Posts** únicamente si tiene los permisos correspondientes, optimizando el control de acceso.
- Reutilizar la gestión de posts del Rol Admin



Permisos para Posts:

- **view services:** index(), show() .-Permite visualizar la lista de posts y acceder al módulo de posts en el dashboard.
- **create services:** create(),store() .-Permite crear nuevos posts en el sistema.
- **edit services:** edit(),update() .-Permite editar la información de un posts existente.
- **delete services:** destroy() .-Permite eliminar posts del sistema.



Rutas

Agrupamos las rutas bajo un middleware que permite el acceso a **Admin** y **Editor**.

```
use App\Http\Controllers\Admin\PostController;  
Route::middleware(['role:admin|editor'])->group(function () {  
    Route::resource('post', PostController::class)  
});
```



CONTROLADOR

En el constructor de Admin/PostController

- `$this->middleware('permission:view posts')->only(['index', 'show']);`
- `$this->middleware('permission:create posts')->only(['create', 'store']);`
- `$this->middleware('permission:edit posts')->only(['edit', 'update']);`
- `$this->middleware('permission:delete posts')->only(['destroy']);`



Agregar en

- editor.dashboard
- admin.post.index

```
@can('view posts')  
  <a href="{{ route('admin.post.index') }}" class="btn btn-primary mt-3">  
    Gestión posts  
  </a>  
@endcan
```



Resumen

El Rol Editor podrá acceder al módulo de **Gestión de Posts** para crear, visualizar, editar y eliminar posts de manera controlada, siempre que cuente con los permisos asignados por el Administrador.



FIN

Curso completo : [Laravel Página Web Administrable](#)

