

81. Ignorando los global scopes en el panel de administración con Laravel

Queremos que los productos no disponibles también aparezcan en el panel de administración para poder modificarlos, pero sin embargo no podemos hacer esto si ya los hemos ocultado con un global scope para los clientes.

La **solución** es crear un nuevo modelo de Producto específico para los administradores que herede del modelo Producto. Se hará **override** de la función **booted()** y esta se dejará vacía, para que no se aplique el global scope.

Es raro, pero según dice el chaval es lo único que más o menos va a funcionar.

En Tinker:

```
php artisan make:model PanelProduct
```

En PanelProduct:

```
<?php

namespace App\Models;

use App\Models\Product;
use Illuminate\Database\Eloquent\Factories\HasFactory;
use Illuminate\Database\Eloquent\Model;

class PanelProduct extends Product
{
    // Tiene todas las propiedades y atributos de Product

    protected static function booted() {
        // Literalmente el vacío más absoluto
    }
}
```

En ProductController:

```
use App\Models\PanelProduct;
// En vez del anterior Product
```

⚠ Cambiar cada vez que aparezca el model Product por PanelProduct.

Si Laravel asume una tabla que no existe

Hay que irse al modelo Product (no PanelProduct, ya que llega con la herencia) y añadir la línea:

```
protected $table = 'products';
```

Así lo que hacemos es que Laravel busque a todas las instancias de productos que queramos en la tabla que tiene el nombre que hemos proporcionado. Por supuesto, también de sus clases descendientes como PanelProduct.