## CUE: EL SITIO ADMINISTRATIVO DE DJANGO I

#### Pablo Varas Salamanca

06 de noviembre de 2024, 13:49:27

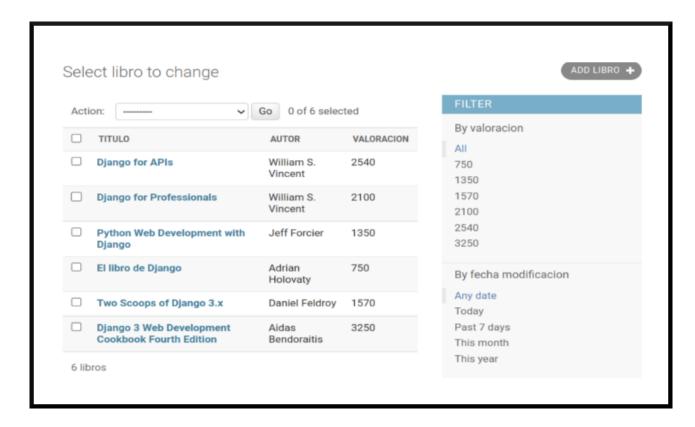
GitHub del proyecto

# DRILLING: MANIPULACIÓN DEL PANEL ADMINISTRATIVO DE DJANGO

#### **EJERCICIO:**

Para esta revisión práctica del sitio de administración de Django se debe:

- Habilitar los campos fecha\_creacion y fecha\_modificacion, y que sean de solo lectura al visualizar un determinado libro.
  - Visualizar el título, valor y valoración respectivamente.
  - Agregar un filtro según la valoración, y la fecha de modificación.



#### **DESARROLLO:**

#### 1. Modificar el sitio administrativo

■ Crear clase custom BookAdmin (admin.ModelAdmin) para modificar cómo se despliegan los valores de cada campo

```
app/admin.py
```

class BookAdmin(admin.ModelAdmin):

```
from django.contrib import admin
from .models import Book
# Register your models here.
# https://docs.djangoproject.com/en/5.1/ref/contrib/admin/#django.contrib.admin.ModelAdmin
class BookAdmin(admin.ModelAdmin):
    readonly_fields = ('fecha_creacion', 'fecha_modificacion')
    list_display = ('titulo', 'autor', 'valoracion', 'rating')
    list_filter = ('autor', 'valoracion', 'fecha_modificacion')
    def rating(self, obj):
        if obj.valoracion < 10:</pre>
            return 'Baja'
        elif 10 <= obj.valoracion <= 25:
            return 'Media
        else:
            return 'Alta
admin.site.reqister(Book, BookAdmin) # reqistro de los model separados por coma
                                 (Django company FlyC- WK SP yas ElDoc)
                      All L1
```

■ Habilitar los campos fecha\_creacion y fecha\_modificacion, y que sean de solo lectura al visualizar un determinado libro.

```
class BookAdmin(admin.ModelAdmin):
    readonly_fields = ('fecha_creacion', 'fecha_modificacion')
```

## 2. Visualizar el título, valor y valoración respectivamente.

Añadir estos campos en el atributo list\_display y para que se muestren en la lista principal de objetos en el panel administrativo.

```
list_display = ('titulo', 'valor', 'valoracion')
```

3. Agregar un filtro según la valoración, y la fecha de modificación.

```
list_filter = ('valoracion', 'fecha_modificacion')
```

### 4. Registrar el Modelo en el Panel Administrativo

Registrar la clase administrativa para que Django aplique estas configuraciones al modelo Libro.

admin.site.register(Libro, LibroAdmin)

#### 5. Activar el entorno virtual

pablo@debian:/media/DATA/Descargas/BOOTCAMP/Bootcamp/VSC/M6\_Django/S10/drilling/book\_/site\_django/book\$ source /
home/pablo/pablo-env/bin/activate

- 6. Al acceder al sitio administrativo en http://127.0.0.1:8000/admin/book/book/ despliega lo solicitado
  - los campos título, valor, valoración, en modo solo lectura
  - el filtro según la valoración, y la fecha de modificación

