



Materia:	Programación Web 2		
Nombre			
Informe:	Ejercicio: Formularios, archivos		
Alumno(s):	America Citlalli Lopez Lemus	Fecha:	28/09/25

Objetivo

Realizar una app en Django que contenga un formulario que permita cargar múltiples archivos y gestionarlos. La temática del formulario es libre y deberán implementarlo utilizando los recursos de la documentación. El formulario deberá validar que los archivos sean de tipo imagen o pdf.

Temas del plan de estudios

TEMA 2:

2.1 Desarrollo de formularios y validación de datos en aplicaciones web con Django, incluyendo el uso de errores.

Material

- Computadora con sistema operativo Windows.
- Entorno de desarrollo: Python 3.12.5, Django 5.2.5.
- Editor de código: Visual Studio Code.
- Navegador web (Chrome).
- Archivos necesarios (en mi caso): Proyecto Django (Editorial), modelo Manuscrito, plantillas HTML, archivo estático logo.jpg.

Marco Teórico

Los formularios en Django permiten recolectar y validar datos de usuario mediante la clase `forms.Form` o `forms.ModelForm`. La validación se logra definiendo campos con `required=True` o usando `blank=False` en los modelos. Los errores se muestran automáticamente en las plantillas si el formulario no pasa la validación. Además, se emplea Bootstrap para estilizar la interfaz y mejorar la experiencia del usuario.

Desarrollo

Se creó el proyecto Editorial y la aplicación manuscritos, configurando `settings.py` para archivos estáticos, después se definió el modelo Manuscrito con campos obligatorios (titulo, autor, resumen) y se aplicaron migraciones. Se implementó un `ModelForm` y `ArchivoForm` para validar y guardar datos, asegurando que los campos sean obligatorios y solo pdf e imagen validos. Se modificaron los html para incluir un formulario o tabla o confirmación dependiendo el documento con botones "Aceptar" y "Cancelar", usando Bootstrap para el diseño. Se verificó que los campos obligatorios generen errores al dejarlos vacíos o con extensiones no validas y que "Cancelar" redirija a la lista sin validar.



Resultado

Registrar Manuscrito

127.0.0.1:8000/registro/

Google, Audiomaster, Spotify, YouTube, Prime, Netflix, Disney+, Mail, Twitch, Facebook, Instagram, Twitter, Drive, Podcaster, PDF, Gmail, Correo, Pinterest, RSS, etc.

Penguin Random House Grupo Editorial

Iniciar, Registrarse

Registrar Manuscrito

Título:

Autor:

Resumen:

Archivos

Seleccionar archivos (PDF o imágenes):

Elegir archivos: Sin archivos seleccionados

Puedes seleccionar varios archivos al mismo tiempo manteniendo presionada la tecla Ctrl (o Cmd en Mac).

Registrar, Cancelar



Registrar Manuscrito

Título:
Prueba

Autor:
Prueba

Resumen:
prueba

Archivos

Seleccionar archivos (PDF o imágenes):

Elegir archivos Sin archivos seleccionados

El archivo 'diabetes.py' no tiene un formato permitido. Formatos válidos: pdf, jpg, jpeg, png, gif

Puedes seleccionar varios archivos al mismo tiempo manteniendo presionada la tecla Ctrl (o Cmd en Mac).

Registrar Cancelar

Penguin Random House Grupo Editorial

Inicio Registrar

Registrar Manuscrito

Título:
Prueba

Autor:
Prueba

Resumen:
Prueba

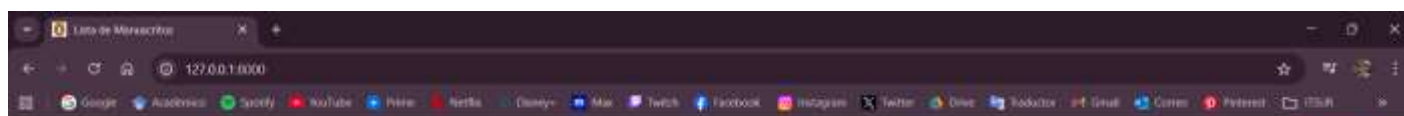
Archivos

Seleccionar archivos (PDF o imágenes):

Elegir archivos 2 archivos

Puedes seleccionar varios archivos al mismo tiempo manteniendo presionada la tecla Ctrl (o Cmd en Mac).

Registrar Cancelar



Lista de Manuscritos

Título	Autor	Resumen	Estado	Acciones			
Prueba	Prueba	Prueba	No aceptado	Editar	Eliminar	Cambiar Estado	Ver Archivo



Cambiar Estado de "Prueba"

Estado actual: No aceptado

[Cambiar a Aceptado](#) [Cancelar](#)



The first screenshot shows the 'Actualizar Manuscrito' (Update Manuscript) form. It has a header with the Penguin Random House logo and 'Inicio' and 'Registrar' buttons. The form fields are: 'Titulo:' with value 'Prueba', 'Autor:' with value 'Prueba', and 'Resumen:' with value 'Prueba'. At the bottom are 'Actualizar' and 'Cancelar' buttons.

The second screenshot shows the 'Eliminar Manuscrito' (Delete Manuscript) form. It has the same header. The form contains the text '¿Estás seguro de eliminar "Prueba"?' and 'Eliminar' and 'Cancelar' buttons at the bottom.

Conclusiones

Esta práctica reforzó el entendimiento de Django como framework para desarrollo web. La resolución de errores como el 404 fue clave para aprender sobre la estructura de carpetas y configuraciones. Se concluye que una correcta configuración de STATICFILES_DIRS y STATIC_ROOT es esencial, y se sugiere practicar más con collectstatic en entornos controlados.

Bibliografía

Working with forms | Django documentation. (s. f.). Django Project.
<https://docs.djangoproject.com/en/5.2/topics/forms/#forms-in-django>
Managing files | Django documentation. (s. f.). Django Project.
<https://docs.djangoproject.com/en/5.2/topics/files/#using-files-in-modelsm>