



TEMA V

DESARROLLO DE APLICACIONES WEB EN BACKEND

*"Siempre parece imposible,
hasta que se hace"*
Nelson Mandela

AÑADIENDO CSS, IMÁGENES...

Los sitios web generalmente necesitan archivos adicionales como imágenes, CSS, archivos JavaScript que son necesarios para representar páginas web completas en un navegador. En proyectos pequeños, podemos trabajar proporcionando rutas absolutas a nuestros recursos o escribiendo funciones CSS y JavaScript en línea en los archivos HTML. Esto no solo va en contra de las mejores prácticas de codificación, sino que también se complica cuando manejamos proyectos más grandes, especialmente con múltiples aplicaciones.

En Django, a estos archivos se les llama archivos estáticos.

Django proporciona `django.contrib.staticfiles` para recopilar archivos estáticos de cada una de sus aplicaciones (y cualquier otro lugar que especifique) en una sola ubicación que se puede servir fácilmente en producción.

1. Lo primero será comprobar que aparece en settings.py
2. Debemos especificar STATIC_URL, que es la URL utilizada cuando se hace referencia a archivos estáticos. Por defecto viene incluida en settings.py pero hay que comprobarlo

```
STATIC_URL = '/static/'
```

Esta ruta significa que los archivos estáticos se almacenarán en la ubicación `http: //localhost:8000/static/`

3. Crearemos una carpeta llamada static, dentro de ella a su vez tendremos subcarpetas para imágenes, css,...
 - Si estamos trabajando con aplicaciones, en la raíz del directorio de la app creamos la carpeta static
 - Si estamos trabajando con proyecto sólo, esta carpeta irá en la raíz del proyecto.
 - Si tenemos varias aplicaciones y archivos que se llamen igual, por ejemplo index.css tenemos que tener cuidado para que no confunda los de unas aplicaciones con otras. En este caso podemos hacer otro nivel de carpetas o simplemente nombrarlas distintas. Por ejemplo cssappnombre

Otra posibilidad es que varias aplicaciones de un proyecto tengan archivos comunes, en este caso es mejor crear una carpeta static en el directorio raíz de su proyecto.

Es decir, para utilizar un lugar común para todos los archivos estáticos en el directorio de su proyecto, necesitamos configurar

STATICFILES_DIRS para informar a Django sobre nuestro nuevo directorio porque por defecto buscará en app directorios solamente.

```
STATICFILES_DIRS = (  
    os.path.join(PROJECT_ROOT, 'static'),  
    # Extra lookup directories for collectstatic to find static files  
)
```

Si estamos trabajando en aplicaciones y ponemos la carpeta static dentro de la aplicación no necesitamos configurar ninguna ruta porque django buscará en las carpetas de las aplicaciones los archivos estáticos.

4. Generamos las carpetas y almacenamos los archivos
5. Añadimos el código en nuestro html, para ello usamos las etiquetas static y url

```
{% load static %}  
<link rel="stylesheet" href="{% static 'css/index.css' %}">  
<script src="{% static 'js/functions.js' %}"></script>  

```