

Vistas

1. Podemos cambiar nuestras propias vistas y poder sustituir así la web de inicio de django.

Para ello necesitamos crear un archivo que se llame views.py, con este contenido:

```
from django.http import HttpResponse #Necesario para poder responder al cliente

#con httpresponse, podemos responder con texto plano o con html
#forma de responder al cliente cuando hace un http

def index (request): # El request captura las peticiones de los clientes
    return HttpResponse (<h1>hola mundo</h1>)
#Con esto hemos habilitado una vista, pero hay que además darla de alta, en urls.py
```

Posteriormente hay que habilitar una ruta para esta función, en el archivo urls.py (ya está creado)

```
from django.contrib import admin
from django.urls import path
from . import views # Importo nuestro módulo views, recién creado.

urlpatterns = [
    path('hola/', views.index), #indicamos que la vista la queremos mostrar en esa ruta,
    añadimos a la lista de url, la nuestra, con hola, como nombre de ruta, y nuestro nombre
    de vista.

    path('admin/', admin.site.urls),
]
```

Si volvemos a la terminal/consola, los nuevos cambios se recargarán automáticamente gracias a que el DEBUG=true. Esta característica la podemos ver dentro del archivo setting.py

Al volver al navegador veremos nuestra vista:



Interesante: [Writing your first Django app, part 1 | Django documentation | Django](#)

2. Otra forma de cambiar la vista por defecto es utilizando el paquete render

En nuestro fichero views.py, añadiremos una nueva función home, importando también previamente el paquete render.

```
from django.http import HttpResponse #Necesario para poder responde al cliente
from django.shortcuts import render #necesario para que funcione la otra forma de
visualizar una vista

#con httpresponde, podemos responder con texto plano o con html
#forma de responder al cliente cuando hace un http

def index (request): # El request captura las peticiones de los clientes
    return HttpResponse ("<h1>hola mundo</h1>")
#Con esto hemos habilitado una vista, pero hay que de alguna manera darla de alta, en
urls.py

def home (request): # Pinta una página con render, también hay que darlo de alta en urls.py
    return render(request, 'index.html') # la página index.html hay que crearla dentro del
archivo de configuración de todo proyecto de django, settings.py
```

Igual que antes, esta nueva vista hay que darla de alta en el fichero de urls.py

```
from django.contrib import admin
from django.urls import path
from . import views # Importo nuestro módulo views

urlpatterns = [
    path('hola/', views.index), #indicamos que la vista la queremos mostrar en esa ruta.
```

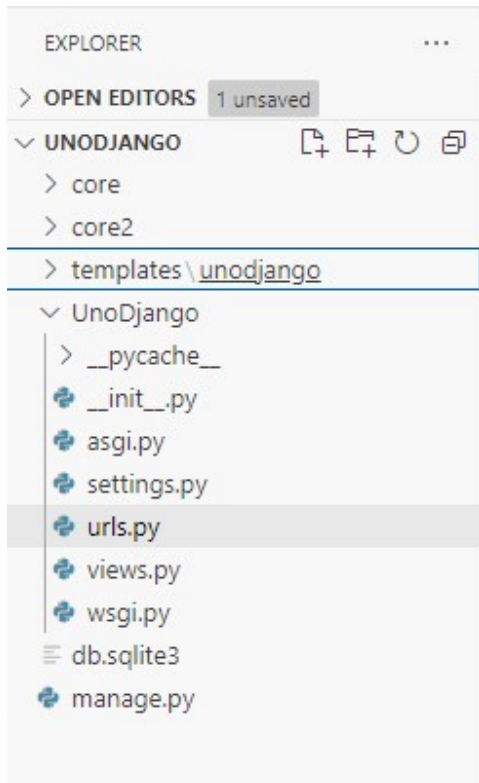
```
#Si ponemos path('', views.index), no haría falta poner en el navegador el directorio.  
path('otramas/', views.home), #nueva vista  
path('admin/', admin.site.urls),  
]
```

Hay además que dar de alta la página index.html dentro del fichero de configuración de django, dentro de settings.py. Ahí dentro hay un diccionario , TEMPLATES, y dentro de él existe la clave DIR, que es una lista, que contiene las rutas dónde puede buscar los templates, y dónde estará mi index.html

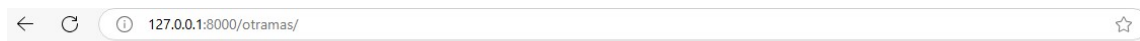
```
TEMPLATES = [  
    {  
        'BACKEND': 'django.template.backends.django.DjangoTemplates',  
        'DIRS': ['templates'], #creo directorio dentro de mi aplicación dónde tendré mi  
index.html y así no dará error en el navegador.  
        'APP_DIRS': True,  
        'OPTIONS': {  
            'context_processors': [  
                'django.template.context_processors.debug',  
                'django.template.context_processors.request',  
                'django.contrib.auth.context_processors.auth',  
                'django.contrib.messages.context_processors.messages',  
            ],  
        },  
    ],  
]
```

```
'APP_DIRS': True # implica que buscará dentro de cada aplicación la carpeta template,  
y dentro de ésta el archivo que estoy buscando  
'DIRS': [templates], #añado templates que será el directorio dentro de mi aplicación dónde  
tendré mi index.html.
```

Los templates se suelen manejar fuera de la carpeta de la aplicación, tal como se ve en la figura y dentro de esa carpeta templates, abrir una por cada aplicación que tengamos.

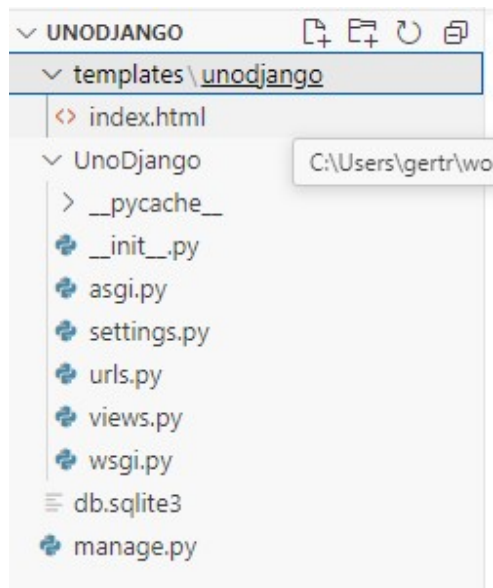


Probamos a mostra en el navegador nuestro index.html, que puede tener el código que queramos.



esto es el index

Si dentro de nuestra carpeta templates creamos una carpeta que se va a corresponder con las páginas html de nuestro proyecto, tal que así:



Para que nuestro index siga funcionando hay que cambiar en la views.py la ruta del index.html.

```
from django.http import HttpResponse #Necesario para poder responde al cliente
from django.shortcuts import render #necesario para que funcione la otra forma de
visualizar algo

#con httpresponde, podemos responder con texto plano o con html
#forma de responder al cliente cuando hace un http

def index (request): # El request captura las peticiones de los clientes
    return HttpResponse ("<h1>hola mundo</h1>")
#Con esto hemos habilitado una vista, pero hay que de alguna manera darla de alta, en
urls.py

def home (request): # Pinta una página con render, también hay que darlo de alta en urls.py
    return render(request, 'unodjango/index.html') # la página index.html hay que crearla
dentro del archivo de configuración de todo proyecto de django, settings.py
```

No hay que incorporar la carpeta templates en la ruta, pues lo busca por defecto.