**CREACIÓN DE VISTAS Y CONFIGURACIÓN URLs**

* El modelo MVT permitirá que al realizar una petición de un usuario, Django compruebe primero la URL, si es correcta enviará al view.
* Ese view decidirá reenviar a un template(página), o incluso si accederá a datos (modelo) antes de mostrar la información en el navegador.

Templates

Views

URL

dispatcher

SALIDA

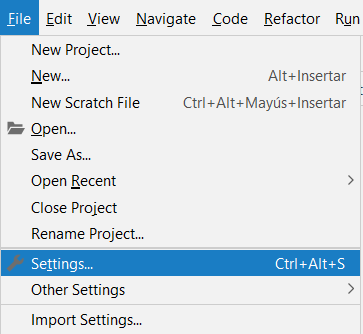
Models

Vamos a configurar el proyecto para poder realizar una primera petición a nuestra aplicación.

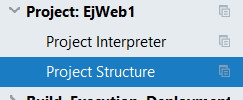
**CREACIÓN DE VISTA**

1. **Antes de comenzar a crear los componentes, como estamos creando subcarpetas, vamos a indicarle al editor que la carpeta proyecto1 es nuestra carpeta de código.**

Pulsa file, settings

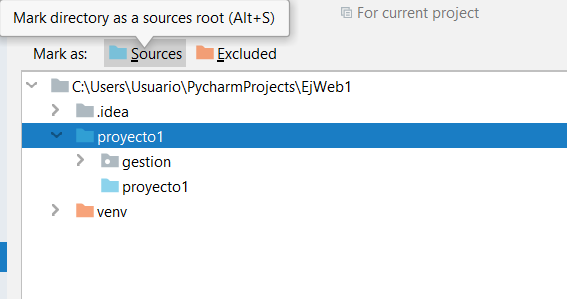


Project Structure

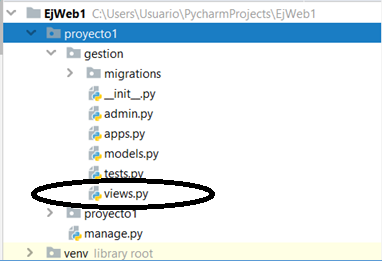


Seleccionamos la carpeta de nivel superior llamada proyecto1 y pulsamos en Sources.

Después pulsamos en el botón aplicar.



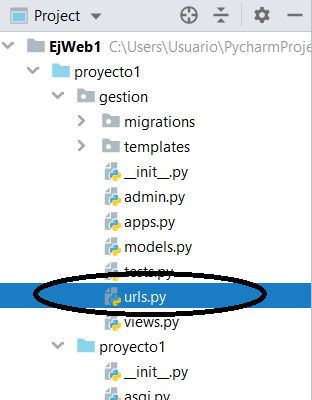
1. Abrimos el archivo **views.py** de la aplicación **gestion** e incluimos el siguiente código:



**from** django.http **import** HttpResponse  
  
  
  
**def** index(request):  
 **return** HttpResponse(**"Página de inicio"**)

**CREACIÓN DE URLS**

1. Abrimos el archivo **urls.py** de la aplicación **gestion** e incluimos el siguiente código:



**from** django.urls **import** path  
  
**from** gestion **import** views  
  
urlpatterns=[  
 path(**''**,views.index,name=**'index'**)  
]

Parámetros método path:

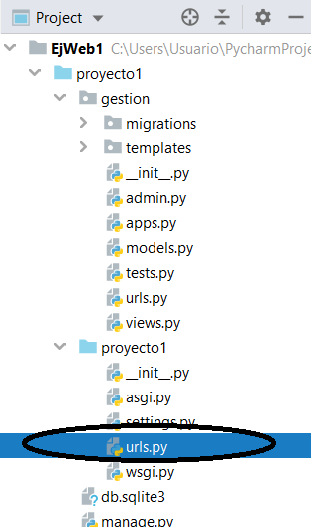
Primero: Cuando se dirija a la url ‘’ (vacía).

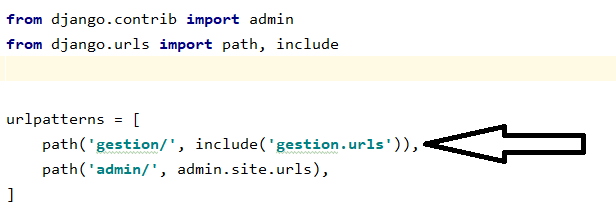
Segundo: Le diremos que vaya a index de view.

Tercero: Nombre que le damos.

path(**''**,views.index,name=**'index'**)

1. Abrimos el archivo **urls.py** del proyecto **proyecto1** e incluimos el siguiente código:





**from** django.contrib **import** admin  
**from** django.urls **import** path, include  
  
  
urlpatterns = [  
 **path('gestion/', include('gestion.urls')),**  
 path(**'admin/'**, admin.site.urls),  
]

Veremos que ya tenemos configurado la url de admin para poder acceder a la consola de administración que Django trae configurada.

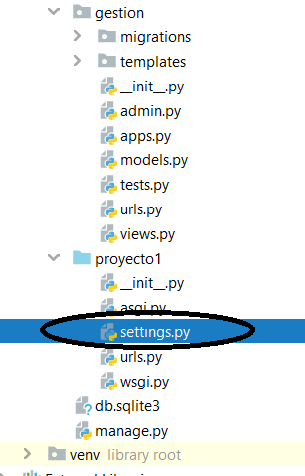
**Parámetros del método path:**

**path('gestion/', include('gestion.urls')),**

Al solicitar en la dirección url **'gestion/',** que nos enrute todo el tráfico del fichero urls de nuestra aplicación gestion.

1. Por último, antes de ejecutar tendremos que registrar nuestra aplicación para que pueda ser ejecutada en el archivo **settings.py.**

Añadimos a la lista **INSTALLED\_APPS** nuestra aplicación llamada gestion.



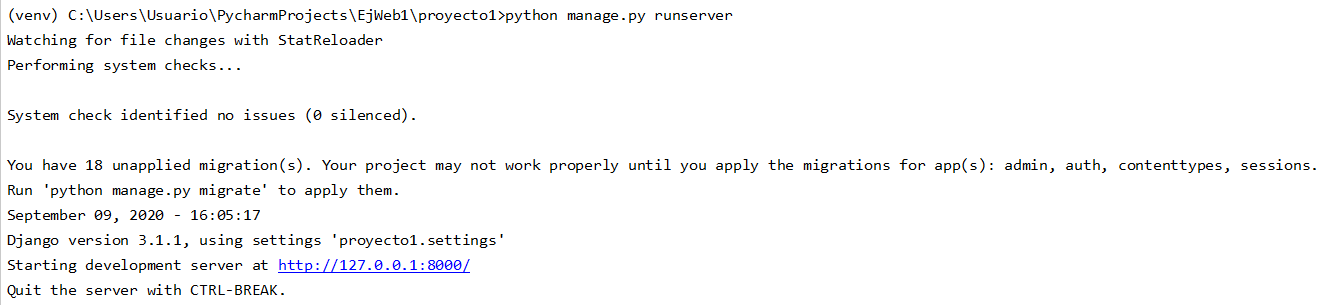


INSTALLED\_APPS = [  
 **'django.contrib.admin'**,  
 **'django.contrib.auth'**,  
 **'django.contrib.contenttypes'**,  
 **'django.contrib.sessions'**,  
 **'django.contrib.messages'**,  
 **'django.contrib.staticfiles'**,  
 **'gestion'**]

**EJECUTAR APLICACIÓN**

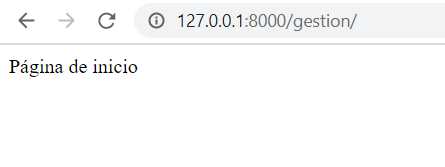
En el terminal ejecutamos el comando:

**python manage.py runserver**



Ya podemos solicitar la url que hemos configurado

**http://127.0.0.1:8000/gestion/**



**EJECUTAR APLICACIÓN DESDE EL IDE**

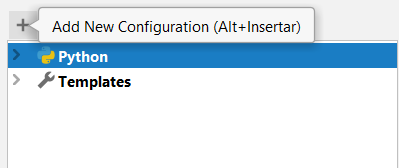
También podemos configurar el IDE para poder ejecutar al pulsar el icono  y no tener que ejecutar el comando “python manage.py runserver”

desde el terminal.

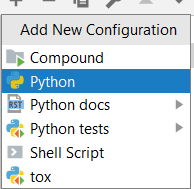
Pulsamos en editar configuración



Seleccionamos el icono + para poder añadir una nueva configuración



Elegimos Python

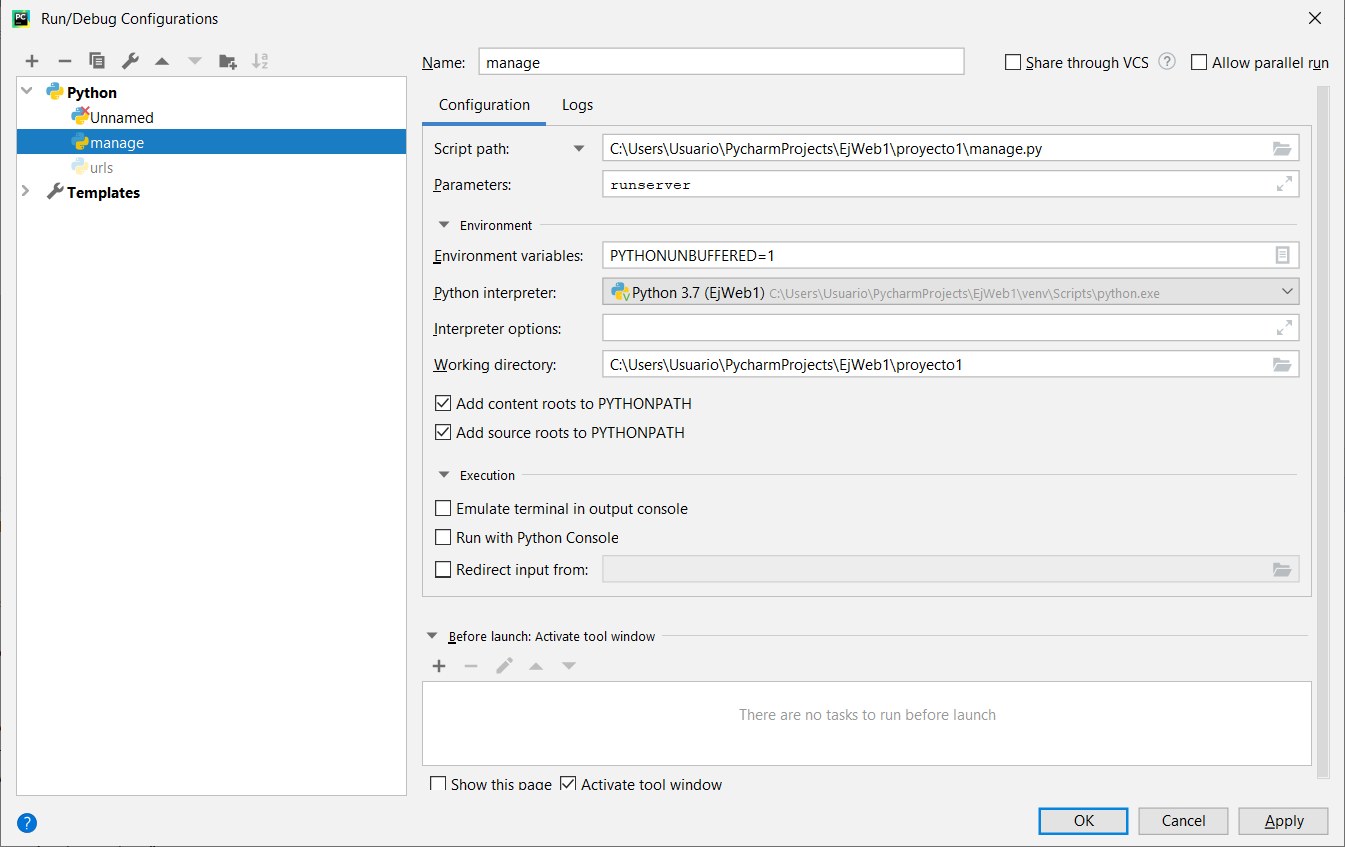


Y rellenamos las siguientes opciones:

Name: Nombre que queramos para poder localizarlo

Script path: C:\Users\Usuario\PycharmProjects\EjWeb1\proyecto1\manage.py(es la ruta del archivo manage.py de nuestro proyecto)

Parameters: runserver



Ya podemos ejecutar el servidor desde el propio editor.

