

Tarea 1. Servir contenidos desde Django

María Victoria Santiago Alcalá

I. Descripción

Se ha seguido el guion de dicha tarea, instalando docker, docker-compose y las demás herramientas necesarias para el buen funcionamiento de esta tarea 1.

Las instrucciones llevadas a cabo para la creación de este proyecto son las siguientes:

```
$ docker-compose run web django-admin.py startproject my_website . #Crea el proyecto
```

```
$ sudo chown -R $USER:$USER . #Para cambiar los permisos
```

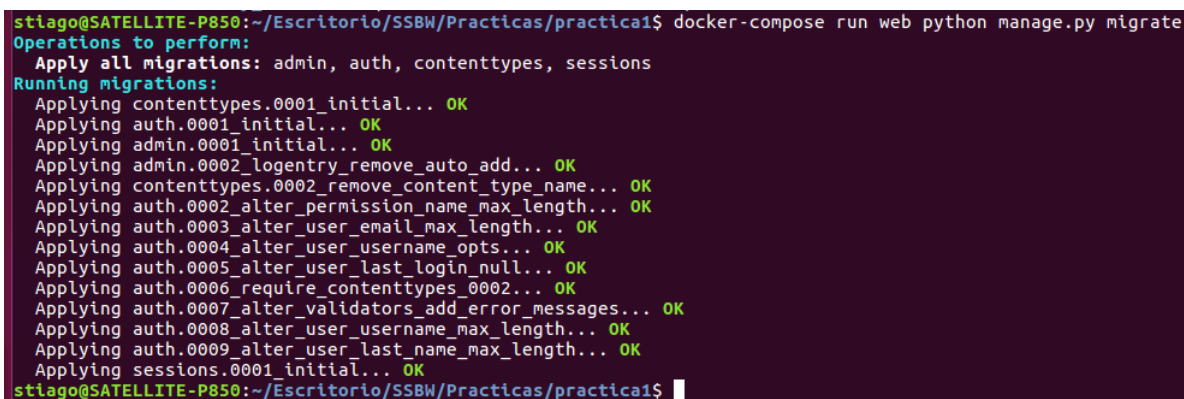
```
$ docker-compose up #Iniciamos el contenedor
```

Seguidamente se han creado los directorios “static” y “templates” los cuales contendrán los archivos de estilo css, las imágenes, javascripts, archivos html, etc.

Se ha configurado el fichero settings.py añadiendo los correspondientes path a templates y a static, además se ha añadido el nombre de nuestra aplicación al apartado “INSTALLED_APPS”.

A continuación, se ha iniciado la base de datos y se ha creado el usuario administrador ejecutando los siguientes comandos:

```
$ docker-compose run web python manage.py migrate #Ejecutar a cada modificacion de la DB
```



```
stlago@SATELLITE-P850:~/Escritorio/SSBW/Practicas/practica1$ docker-compose run web python manage.py migrate
Operations to perform:
  Apply all migrations: admin, auth, contenttypes, sessions
Running migrations:
  Applying contenttypes.0001_initial... OK
  Applying auth.0001_initial... OK
  Applying admin.0001_initial... OK
  Applying admin.0002_logentry_remove_auto_add... OK
  Applying contenttypes.0002_remove_content_type_name... OK
  Applying auth.0002_alter_permission_name_max_length... OK
  Applying auth.0003_alter_user_email_max_length... OK
  Applying auth.0004_alter_user_username_opts... OK
  Applying auth.0005_alter_user_last_login_null... OK
  Applying auth.0006_require_contenttypes_0002... OK
  Applying auth.0007_alter_validators_add_error_messages... OK
  Applying auth.0008_alter_user_username_max_length... OK
  Applying auth.0009_alter_user_last_name_max_length... OK
  Applying sessions.0001_initial... OK
stlago@SATELLITE-P850:~/Escritorio/SSBW/Practicas/practica1$
```

```
$ docker-compose run web python manage.py createsuperuser
```

Finalmente tras configurarlo todo lanzamos nuestro contenedor “docker-compose up” y accediendo al enlace http://localhost:8080/users_dashboard/ podremos ver nuestra aplicación funcionando.

Si bien queremos entrar al panel de administración, solo tendremos que acceder al enlace <http://localhost:8080/admin> e introducir nuestras credenciales.

En el siguiente apartado de contenidos y con las siguientes capturas de pantalla se describe el proceso seguido mas detalladamente.

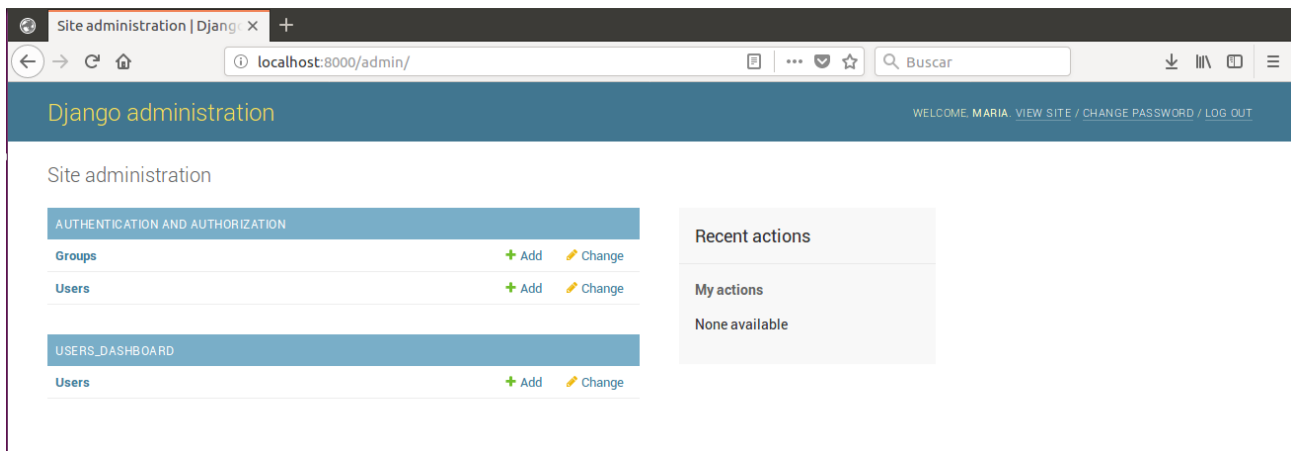
```
stiago@SATELLITE-P850: ~/Escritorio/SSBW/Practicas/practica1
stiago@SATELLITE-P850:~/Escritorio/SSBW/Practicas/practica1$ docker-compose up
Starting practica1_web_1 ... done
Attaching to practica1_web_1
web_1 | Performing system checks...
web_1 |
web_1 | System check identified no issues (0 silenced).
web_1 | February 23, 2018 - 23:36:06
web_1 | Django version 2.0.2, using settings 'my_website.settings'
web_1 | Starting development server at http://0.0.0.0:8000/
web_1 | Quit the server with CONTROL-C.
```

II. Contenidos

Inicialmente tras abrir <http://localhost:8080/> tenemos dos opciones, abrir el panel de administrador o abrir la aplicación.

1. Administrador

Introducimos nuestras credenciales y nos llevara a la siguiente pagina en la cual si hemos introducido bien nuestro modelo, por ejemplo en mi caso, podremos crear dandole un nombre y un apellido.

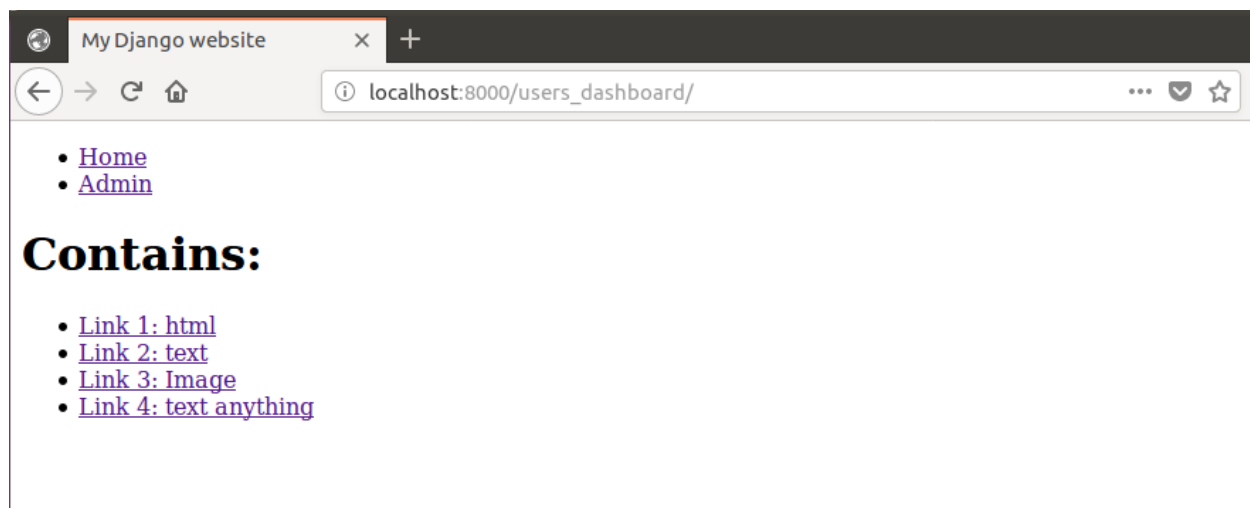


The screenshot shows a web browser window with the address bar at `localhost:8000/admin/users_dashboard/user/add/`. The page title is "Django administration" and the user is logged in as "MARIA". The breadcrumb trail is "Home > Users_Dashboard > Users > Add user". The form has two input fields: "Name:" and "Surname:". At the bottom right, there are three buttons: "Save and add another", "Save and continue editing", and "SAVE".

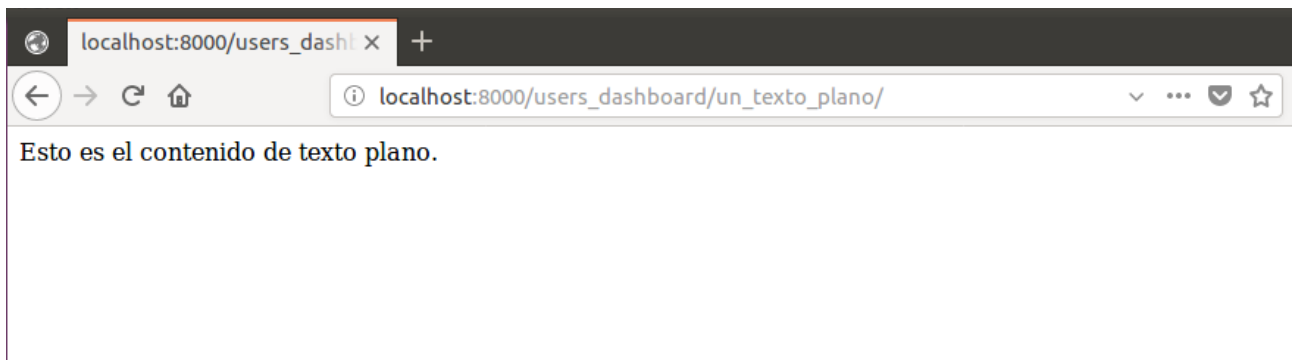
2. Abrir la aplicación.

En esta fase inicial la dashboard unicamente va a mostrar los tipos de contenidos que se han relatado en el guión:

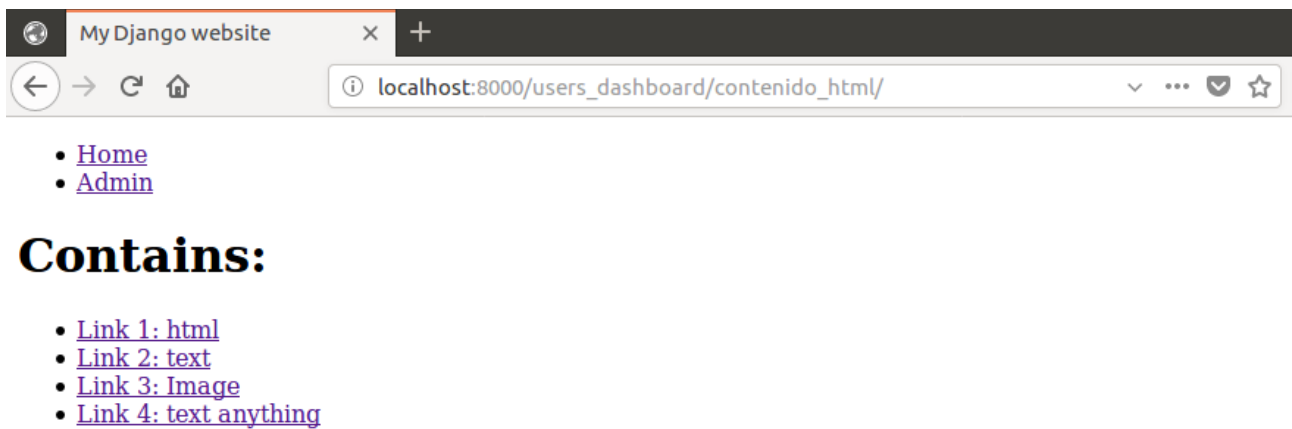
a) La primera vista o vista general se ha llamado “Home” y se corresponde con el siguiente enlace http://localhost:8080/users_dashboard desde la cual se puede acceder a todos los demás contenidos ademas de también dar la posibilidad de acceder a la administración pulsando “Admin”.



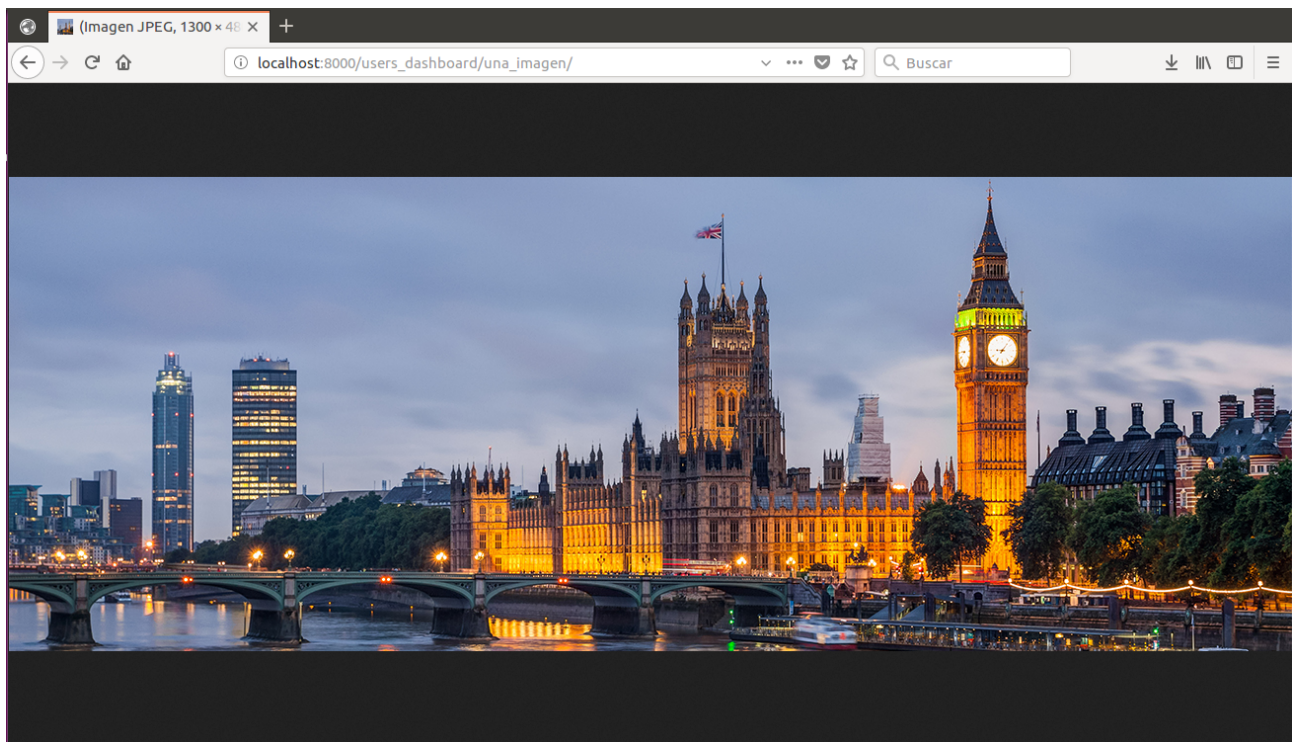
b) Contenido de texto plano con la url: http://localhost:8000/users_dashboard/un_texto_plano/



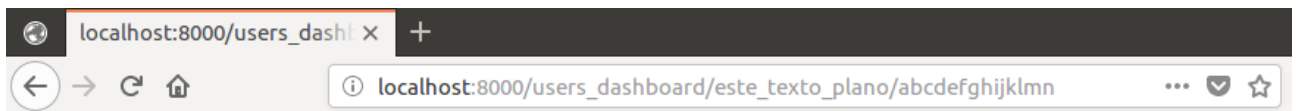
c) Contenido de html con la url: http://localhost:8000/users_dashboard/contenido_html/



d) Contenido de imagen, para visualizarla en el navegador con la url:
http://localhost:8000/users_dashboard/una_imagen/



e) Cualquier contenido tomado de la url('lo que sea' (tomado de la url)):
http://localhost:8000/users_dashboard/este_texto_plano/<text>



Tenemos este texto plano introducido en la url: abcdefghijklmn