

Guía de Despliegue de Django en CloudPanel

INTRODUCCIÓN

Este documento proporciona instrucciones detalladas para desplegar un proyecto Django en CloudPanel. Siga los pasos en el orden indicado para asegurar una instalación exitosa.

REQUISITOS PREVIOS

- Acceso SSH a su servidor VPS con privilegios de root
- Un dominio configurado en CloudPanel
- Proyecto Django funcional con la estructura proporcionada

PASO 1: PREPARACIÓN DEL ENTORNO

1.1 Acceso al Servidor

Conéctese al servidor VPS mediante SSH como usuario root. Una vez conectado, navegue al directorio raíz de su proyecto utilizando el siguiente comando:

```
cd /home/[tu-dominio]/htdocs/www.[tu-dominio].com
```

PASO 2: CONFIGURACIÓN DEL ENTORNO VIRTUAL

2.1 Instalación de Requisitos Previos

Instale Python y las herramientas necesarias con el siguiente comando:

```
apt install -y python3.10-venv python3-pip
```

2.2 Creación del Entorno Virtual

Cree un nuevo entorno virtual ejecutando:

```
python3 -m venv env
```

2.3 Activación del Entorno Virtual

Active el entorno virtual con el comando:

```
source env/bin/activate
```

Nota: Cuando el entorno virtual está activo, verá (env) al inicio de su línea de comandos.

PASO 3: INSTALACIÓN DE DEPENDENCIAS

3.1 Instalación de Requerimientos del Proyecto

Instale todas las dependencias listadas en su archivo requirements.txt:

```
pip install -r requirements.txt
```

3.2 Instalación de Django y Gunicorn

Instale los paquetes necesarios para el servidor:

```
pip install django gunicorn psycopyg2-binary
```

PASO 4: CONFIGURACIÓN DE LA BASE DE DATOS

4.1 Migraciones de la Base de Datos

Genere las migraciones necesarias:

```
python3 manage.py makemigrations
```

Aplique las migraciones a la base de datos:

```
python3 manage.py migrate
```

4.2 Archivos Estáticos

Recopile todos los archivos estáticos:

```
python3 manage.py collectstatic
```

Nota: Asegúrese de que STATIC_ROOT esté correctamente configurado en su archivo settings.py antes de ejecutar este comando.

PASO 5: CONFIGURACIÓN DE GUNICORN

5.1 Preparación

Primero, desactive el entorno virtual:

```
deactivate
```

5.2 Configuración del Socket de Gunicorn

Cree un nuevo archivo para el socket:

```
nano /etc/systemd/system/gunicorn.socket
```

Copie el siguiente contenido en el archivo:

```
[Unit]
```

```
Description=gunicorn socket
```

```
[Socket]
```

```
ListenStream=/run/gunicorn.sock
```

```
[Install]
```

```
WantedBy=sockets.target
```

5.3 Configuración del Servicio de Gunicorn

Cree el archivo de servicio:

```
nano /etc/systemd/system/gunicorn.service
```

Copie y adapte el siguiente contenido según su proyecto:

```
[Unit]
```

```
Description=gunicorn daemon
```

```
Requires=gunicorn.socket
```

```
After=network.target
```

```
[Service]
```

```
User=[tu-usuario]
```

```
Group=www-data
```

```
WorkingDirectory=/home/[tu-usuario]/htdocs/www.[tu-dominio]
```

```
ExecStart=/home/[tu-usuario]/htdocs/www.[tu-dominio]/env/bin/gunicorn \
    --access-logfile - \
    --workers 3 \
    --bind unix:/run/gunicorn.sock \
    [nombre_proyecto].wsgi:application
```

[Install]

WantedBy=multi-user.target

5.4 Activación de Gunicorn

Inicie y habilite el socket de Gunicorn:

```
systemctl start gunicorn.socket
systemctl enable gunicorn.socket
```

PASO 6: CONFIGURACIÓN DE NGINX EN CLOUDPANEL

6.1 Configuración del Virtual Host

Acceda a CloudPanel y siga estos pasos:

1. Vaya a la configuración del sitio
2. Seleccione la pestaña Vhost
3. Reemplace la configuración existente con el siguiente código:

```
server {
    listen 80;
    listen [::]:80;
    listen 443 ssl http2;
    listen [::]:443 ssl http2;
    {{ssl_certificate_key}}
    {{ssl_certificate}}
    server_name www.[tu-dominio];
    {{root}}

    {{nginx_access_log}}
    {{nginx_error_log}}

    if ($scheme != "https") {
        rewrite ^ https://$host$uri permanent;
    }

    location ~ /\.well-known {
        auth_basic off;
        allow all;
    }

    location = /favicon.ico {
        access_log off;
        log_not_found off;
    }
}
```

```

location /static/ {
    root /home/[tu-usuario]/htdocs/www.[tu-dominio]/[nombre_proyecto];
}

{{settings}}

index index.html;

location / {
    include proxy_params;
    proxy_pass http://unix:/run/gunicorn.sock;
}
}

```

PASO 7: FINALIZACIÓN Y VERIFICACIÓN

7.1 Configuración del Firewall

Configure el firewall para permitir el tráfico web:

```
ufw allow 'Nginx Full'
```

7.2 Reinicio de Servicios

Reinicie Nginx para aplicar los cambios:

```
systemctl restart nginx
```

7.3 Verificación

Verifique el estado de los servicios:

```
systemctl status gunicorn.socket
```

```
systemctl status nginx
```

Por último, visite su dominio en el navegador para confirmar que todo funciona correctamente.

ESTRUCTURA DEL PROYECTO

Su proyecto debe tener una estructura similar a la siguiente:

```

proyecto/
├── customer_service/
│   ├── __pycache__/
│   ├── fixtures/
│   ├── migrations/
│   ├── static/
│   ├── templates/
│   ├── admin.py
│   ├── apps.py
│   ├── models.py
│   └── serializers.py

```

```
|
| |— tests.py
| |— urls.py
| |— views.py
|— rios_sac/
| |— __pycache__/_
| |— static/_
| |— settings.py
| |— urls.py
| |— wsgi.py
|— db.sqlite3
|— manage.py
|— requirements.txt
```

NOTAS IMPORTANTES

- Reemplace [tu-usuario], [tu-dominio] y [nombre_proyecto] con los valores correspondientes a su proyecto.
- Asegúrese de que todos los archivos de configuración tengan los permisos correctos.
- Mantenga copias de seguridad de sus archivos de configuración originales.
- Verifique los logs en caso de encontrar errores durante el despliegue.

SOLUCIÓN DE PROBLEMAS

Si encuentra problemas durante el despliegue:

1. Verifique los logs de Nginx y Gunicorn para identificar errores
2. Asegúrese de que todos los paths y nombres de usuario sean correctos
3. Confirme que los permisos de archivos y directorios sean apropiados
4. Verifique que el firewall permita el tráfico necesario