

# Guía de instalación

Versión 1.0 – Agosto 2023



# CIIFEN

CIIFEN - <https://ciifen.org/> - [info-ciifen@ciifen.org](mailto:info-ciifen@ciifen.org)

## Contenido

1.	Introducción .....	3
2.	Requisitos previos .....	3
2.1	Sistemas operativos .....	3
2.2	Herramientas necesarias .....	3
2.3	Requisitos del sistema.....	5
3.	Pasos para la instalación .....	6

# 1. Introducción

Este manual ha sido diseñado para guiarte a través de cada etapa del proceso de instalación del sistema de digitalización de bandas pluviográficas. Este recurso proporcionará instrucciones detalladas, consejos y mejores prácticas para garantizar una implementación exitosa.

## 2. Requisitos previos

### 2.1 Sistemas operativos

El sistema de digitalización de bandas pluviográficas ha sido diseñado con flexibilidad en mente y puede ser instalado en una variedad de sistemas operativos. Ya sea que estés utilizando OSX, Windows o Linux, el proceso de instalación y configuración se adapta a tu plataforma preferida. Nuestro objetivo es asegurar que todos los usuarios, independientemente del sistema operativo que utilicen, puedan aprovechar al máximo las capacidades de digitalización y almacenamiento de datos pluviométricos que ofrece el sistema.

### 2.2 Herramientas necesarias

Para asegurar la correcta instalación y funcionamiento del sistema de digitalización de bandas pluviográficas, se requiere el uso de ciertas tecnologías y herramientas clave. A continuación, se detallan las tecnologías esenciales y sus respectivas versiones recomendadas:

**Python (versión 3.9):** Python es un lenguaje de programación fundamental para el procesamiento y la gestión de datos. Se necesita para ejecutar ciertos scripts y procesos del sistema.

**Angular (versión 13.3.3):** Angular es un marco de trabajo de desarrollo web que facilita la creación de interfaces de usuario interactivas y eficientes. Es esencial para la interfaz de usuario del sistema.

Node.js (versión 18.13.0): Node.js es una plataforma que permite ejecutar JavaScript en el servidor. Se utiliza para administrar el backend y la lógica del sistema.

PostgreSQL (versión 15.3): PostgreSQL es un sistema de gestión de bases de datos relacional potente y de código abierto. Es utilizado para el almacenamiento seguro y eficiente de los datos capturados por el sistema.

A continuación, se presenta una tabla que resume las versiones recomendadas de estas tecnologías:

Tecnología	Versión Recomendada
Python (con Pip3)	3.9 o superior
Angular	13.3.3 o superior
Node.js	18.13.0 o superior
PostgreSQL	15.3 o posterior

Se recomienda encarecidamente utilizar estas versiones para garantizar la compatibilidad y el funcionamiento óptimo del sistema. Antes de la instalación, asegúrate de contar con las versiones adecuadas de estas tecnologías y realiza las actualizaciones necesarias si es necesario.

## 2.3 Requisitos del sistema

El sistema de digitalización de bandas pluviográficas tiene requisitos específicos de hardware para asegurar su funcionamiento eficiente. Se recomienda contar con los siguientes componentes mínimos para garantizar un rendimiento óptimo:

Componente	Requisito Mínimo
Memoria RAM	4 GB
Procesador	Dual-core a 3.5 GHz

Estos requisitos están diseñados para permitir que el sistema maneje eficazmente la digitalización y almacenamiento de datos pluviométricos. Asegurarse de cumplir con estos requerimientos contribuirá a una experiencia de usuario fluida y eficiente durante la interacción con el sistema de digitalización de bandas pluviográficas.

Es importante tener en cuenta que, si se dispone de hardware con especificaciones superiores a las mínimas mencionadas, el sistema podría alcanzar un rendimiento aún mejor y una mayor capacidad para manejar grandes volúmenes de datos. Considerar la posibilidad de utilizar hardware más potente puede ser ventajoso en situaciones donde se espera una alta carga de trabajo y se busca una optimización continua del rendimiento del sistema.

### 3. Pasos para la instalación

Con los pre-requisitos cumplidos se puede dar paso al proceso de instalación.

1. Cree de una base de datos en PostgreSQL, esta base de datos debe llamarse dbciifen con el siguiente comando.

```
$ CREATE DATABASE dbciifen;
```

```
postgres=# CREATE DATABASE dbciifen;  
CREATE DATABASE
```

2. Descargue las carpetas de los proyectos de backend y frontend

```
admin@xen-VMware-Virtual-Platform:~/Digitalizador$ ls -la  
total 16  
drwxrwxr-x  4 admin admin 4096 Aug 14 14:40 .  
drwxr-x-- 15 admin admin 4096 Aug 14 14:39 ..  
drwxrwxr-x  6 admin admin 4096 Aug 13 16:58 ciifen_digitalizador_backend-main  
drwxrwxr-x  3 admin admin 4096 Aug 12 18:15 ciifen_digitalizador_frontend-main
```

3. En la carpeta backend, se genera un entorno virtual para sobre este ejecutar el proyecto. Utilizando los siguientes comandos.

```
$ python3 -m venv venv
```

```
$ source venv/bin/activate
```

```
admin@xen-VMware-Virtual-Platform:~/Digitalizador/ciifen_digitalizador_backend-main$ python3 -m venv venv  
admin@xen-VMware-Virtual-Platform:~/Digitalizador/ciifen_digitalizador_backend-main$ source venv/bin/activate  
(venv) admin@xen-VMware-Virtual-Platform:~/Digitalizador/ciifen_digitalizador_backend-main$
```

4. A continuación, se instalan los requisitos del proyecto utilizando los siguientes comandos:

```
$ pip install -r requirements/base.txt
```

```
$ pip install -r requirements/dev.txt
```

```
$ pip install -r requirements/linters.txt
```

```
(venv) admin@xen-VMware-Virtual-Platform:~/Digitalizador/ciifen_digitalizador_backend-main$ pip install -r requirements/base.txt  
Collecting asgiref==3.7.2  
  Downloading asgiref-3.7.2-py3-none-any.whl (24 kB)  
Collecting Django==3.2.19  
  Downloading Django-3.2.19-py3-none-any.whl (7.9 MB)  
    7.9/7.9 MB 12.0 MB/s eta 0:00:00  
Collecting django-cors-headers==3.7.0  
  Downloading django_cors_headers-3.7.0-py3-none-any.whl (12 kB)  
Collecting django-environ==0.10.0  
  Downloading django_enviro-0.10.0-py2.py3-none-any.whl (19 kB)  
Collecting django-rest-framework==3.12.4  
  Downloading django-rest-framework-3.12.4-py3-none-any.whl (957 kB)  
    957.7/957.7 kB 66.6 MB/s eta 0:00:00  
Collecting django-rest-framework-simplejwt==5.2.2  
  Downloading django-rest-framework-simplejwt-5.2.2-py3-none-any.whl (85 kB)  
    85.4/85.4 kB 32.8 MB/s eta 0:00:00  
Collecting psycopg2-binary==2.9.3  
  Downloading psycopg2-binary-2.9.3.tar.gz (380 kB)  
    380.6/380.6 kB 78.8 MB/s eta 0:00:00  
  Preparing metadata (setup.py) ... done  
Collecting PyJWT==2.7.0  
  Downloading PyJWT-2.7.0-py3-none-any.whl (22 kB)  
Collecting pytz==2023.3  
  Downloading pytz-2023.3-py2.py3-none-any.whl (502 kB)  
    502.3/502.3 kB 72.2 MB/s eta 0:00:00  
Collecting sqlparse==0.4.4  
  Downloading sqlparse-0.4.4-py3-none-any.whl (41 kB)  
    41.2/41.2 kB 17.6 MB/s eta 0:00:00
```

5. A continuación, se configura dentro del archivo config/settings.py las credenciales de acceso a la base de datos.

```
101 DB_USER = env.str("DB_USER", "postgres")
102 DB_HOST = env.str("DB_HOST", "localhost")
103 DB_PASSWORD = env.str("DB_PASSWORD", "postgres")
104 DB_NAME = env.str("DB_NAME", "dbciifen")
105 DB_ENGINE = env.str("DB_ENGINE", "django.db.backends.postgresql")
106 DB_PORT = env.str("DB_PORT", "5432")
```

6. Con los prerequisites instalados, se procede con la migración de la base de datos con el siguiente comando.

\$ python3 manage.py migrate

```
(venv) admin@xen-VMware-Virtual-Platform:~/Digitalizador/ciifen_digitalizador_backend-main$ python3 manage.py migrate
BASE_DIR: /home/admin/Digitalizador/ciifen_digitalizador_backend-main
MEDIA_ROOT: /home/admin/Digitalizador/ciifen_digitalizador_backend-main/media/
Operations to perform:
  Apply all migrations: admin, auth, authtoken, configuracion, contenttypes, inventario, pluviograma, 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 admin.0003_logentry_add_action_flag_choices... 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 auth.0010_alter_group_name_max_length... OK
  Applying auth.0011_update_proxy_permissions... OK
  Applying auth.0012_alter_user_first_name_max_length... OK
  Applying authtoken.0001_initial... OK
  Applying authtoken.0002_auto_20160226_1747... OK
  Applying authtoken.0003_tokenproxy... OK
  Applying configuracion.0001_initial... OK
  Applying configuracion.0002_rename_userprofile_detalleusuario... OK
  Applying inventario.0001_initial... OK
  Applying inventario.0002_mantenimiento... OK
  Applying inventario.0003_auto_20230721_2211... OK
  Applying inventario.0004_auto_20230809_1740... OK
  Applying pluviograma.0001_initial... OK
  Applying pluviograma.0002_remove_pluviograma_imagen... OK
  Applying pluviograma.0003_auto_20230812_1712... OK
  Applying sessions.0001_initial... OK
```

7. A continuación, se crea el primer superusuario para poder empezar con el uso del sistema utilizando el comando:

\$ python3 manage.py createsuperuser

Y se continua con la creación de la cuenta

```
(venv) admin@xen-VMware-Virtual-Platform:~/Digitalizador/ciifen_digitalizador_backend-main$ python3 manage.py createsuperuser
BASE_DIR: /home/admin/Digitalizador/ciifen_digitalizador_backend-main
MEDIA_ROOT: /home/admin/Digitalizador/ciifen_digitalizador_backend-main/media/
Username (leave blank to use 'admin'):
```

Email address: admin@ciifen.com

Password:

Password (again):

This password is too short. It must contain at least 8 characters.

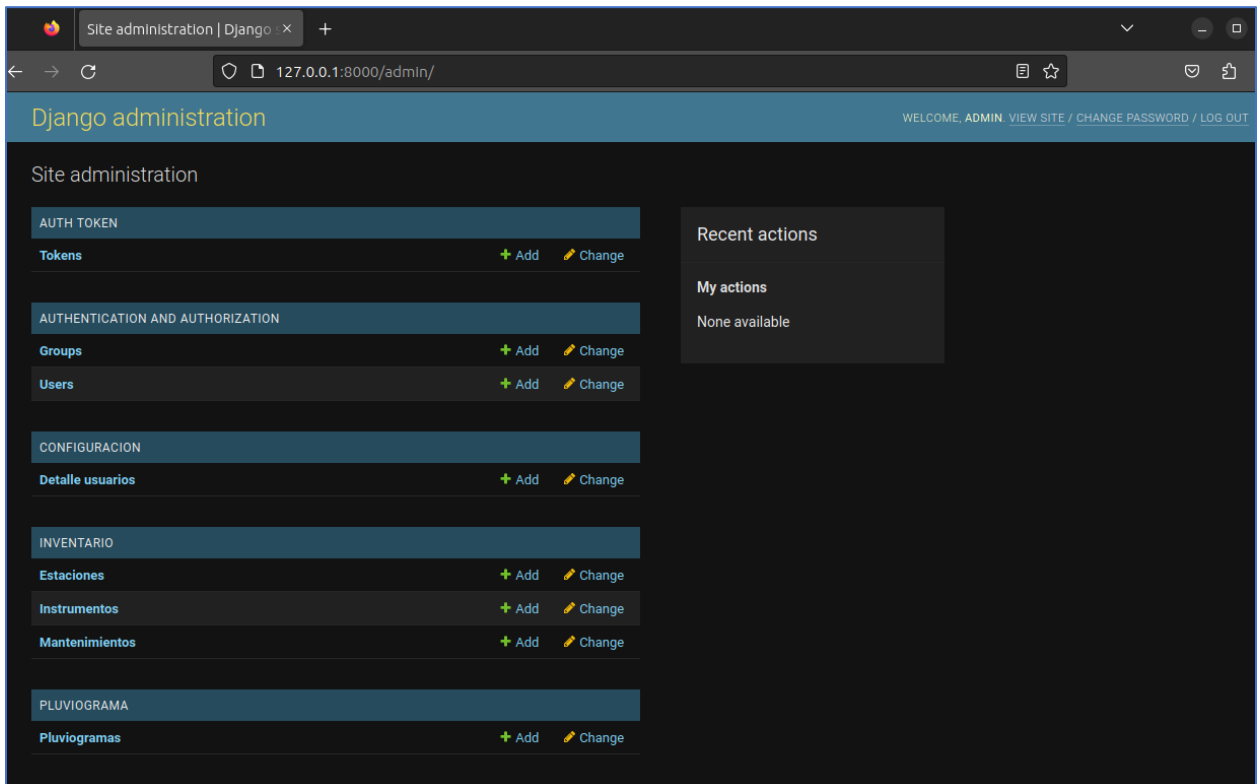
This password is too common.

This password is entirely numeric.

Bypass password validation and create user anyway? [y/N]: y

Superuser created successfully.

8. Para iniciar el backend y tener el panel de control, se utiliza el comando:
- \$ python3 manage.py runserver




Y se levantará la ruta admin donde se pueden crear más crear más usuarios, además también se encuentran los endpoints que utilizará el frontend.

9. Ahora para levantar el frontend, dentro de la carpeta raíz de este, se deben ejecutar los comandos:
- \$ npm install --legacy-peer-deps
- \$ ng serve



on | Django x CIIFEN x +

127.0.0.1:4200/auth/login ☆



# CIIFEN

**Usuario**

**Contraseña**

Acceder