
Guía de Instalación Geonode Desarrollo

Versión 3.3.X

CIDIS - ESPOL

17 de marzo de 2022

Qué es Geonode?

Geonode es un sistema de gestión de contenido geoespacial, una plataforma de software libre fácil de utilizar que permite además la gestión de usuarios y la creación de mapas de manera interactiva.

Geonode es un proyecto que forma parte de La Fundación Geoespacial de Código Abierto (OSGEO) que agrupa proyectos de software libre destinados a trabajar con información geoespacial.

Componentes

- Geoserver
- Pycsw
- Postgres - Postgis
- Mapstore

Instalación

Esta guía de instalación esta realizada para el desarrollo en Ubuntu 20.04 LTS para sistemas operativos diferentes se recomienda revisar la documentación oficial.

Dependencias

```
$ sudo add-apt-repository ppa:ubuntugis/ppa
$ sudo apt update -y
```

Paquetes Geonode Core

```
$ sudo apt install -y --allow-downgrades build-essential \
python3-gdal=3.3.2+dfsg-2~focal2 gdal-bin=3.3.2+dfsg-2~focal2 libgdal-dev=3.3.2+dfsg-2~focal2 \
python3.8-dev python3.8-venv virtualenvwrapper \
libxml2 libxml2-dev gettext libmemcached-dev zlib1g-dev \
libxslt1-dev libjpeg-dev libpng-dev libpq-dev \
software-properties-common build-essential \
git unzip gcc zlib1g-dev libgeos-dev libproj-dev \
sqlite3 spatialite-bin libsqlite3-mod-spatialite libsqlite3-dev
```

```
# Install Openjdk
$ sudo apt install openjdk-8-jdk-headless default-jdk-headless -y
$ sudo update-java-alternatives --jre-headless --jre --set java-1.8.0-openjdk-amd64
```

```
# Verify GDAL version
$ gdalinfo --version
$> GDAL 3.3.2, released 2021/09/01
```

```
# Verify Python version
$ python3.8 --version
$> Python 3.8.10
```

```
# Python path
$ which python3.8
$> /usr/bin/python3.8
```

```
# Verify Java version
$ java -version
$> openjdk version "1.8.0_315"
```

```
# Cleanup the packages
$ sudo apt update -y; sudo apt autoremove --purge
```

Entorno Virtual

El entorno virtual solo se debe crear la primera vez en caso de no contar con un entorno virtual, además se utiliza el editor de archivos “nano” pero puede hacer uso del editor de su elección.

```
# Create the Geo Virtual Environment (first time only)
$ export WORKON_HOME=~/.virtualenvs
$ source /usr/share/virtualenvwrapper/virtualenvwrapper.sh
$ mkvirtualenv --python=/usr/bin/python3.8 geo # Use the python path from above
```

```
#The next time you need to access the Virtual Environment just run
$ source /usr/share/virtualenvwrapper/virtualenvwrapper.sh
$ workon geo
```

```
#In order to save permanently the virtualenvwrapper environment
$ nano ~/.bashrc
# Write to the bottom of the file the following lines
$ export WORKON_HOME=~/.virtualenvs
$ source /usr/share/virtualenvwrapper/virtualenvwrapper.sh
```

Requerimientos Geonode

```
$ git clone https://github.com/UDannyf/geonode.git -b 3.3.x geonode
# Install the Python packages
$ cd geonode
$ pip install -r requirements.txt --upgrade --no-cache --no-cache-dir
$ pip install -e . --upgrade
$ pip install pygdal=="gdal-config --version`.*"
```

Instalar Postgres - Postgis

```
# Ubuntu 20.04 (focal) Postgis database Setup
$ sudo sh -c 'echo "deb http://apt.postgresql.org/pub/repos/apt/ `lsb_release -cs`-pgdg main" >>
/etc/apt/sources.list.d/pgdg.list'
$ sudo wget --no-check-certificate --quiet -O - https://www.postgresql.org/media/keys/ACCC4CF8.asc |
sudo apt-key add -
$ sudo apt update -y; sudo apt install -y postgresql-13 postgresql-13-postgis-3 postgresql-13-postgis-3-
scripts postgresql-13 postgresql-client-13
```

Crear base datos y usuario

```
#create user to settings
$ sudo service postgresql start
$ sudo -u postgres createuser -P geonode
# Use the password: geonode
#Create database geonode and geonode_data with owner geonode
$ sudo -u postgres createdb -O geonode geonode
$ sudo -u postgres createdb -O geonode geonode_data
```

Postgis Extensión

```
#Next let's create PostGIS extensions
$ sudo -u postgres psql -d geonode -c 'CREATE EXTENSION postgis;'
$ sudo -u postgres psql -d geonode -c 'GRANT ALL ON geometry_columns TO PUBLIC;'
$ sudo -u postgres psql -d geonode -c 'GRANT ALL ON spatial_ref_sys TO PUBLIC;'
$ sudo -u postgres psql -d geonode -c 'GRANT ALL PRIVILEGES ON ALL TABLES IN SCHEMA public TO geonode;'
$ sudo -u postgres psql -d geonode -c 'GRANT ALL PRIVILEGES ON ALL SEQUENCES IN SCHEMA public TO geonode;'

$ sudo -u postgres psql -d geonode_data -c 'CREATE EXTENSION postgis;'
$ sudo -u postgres psql -d geonode_data -c 'GRANT ALL ON geometry_columns TO PUBLIC;'
$ sudo -u postgres psql -d geonode_data -c 'GRANT ALL ON spatial_ref_sys TO PUBLIC;'
$ sudo -u postgres psql -d geonode_data -c 'GRANT ALL PRIVILEGES ON ALL TABLES IN SCHEMA public TO geonode;'
$ sudo -u postgres psql -d geonode_data -c 'GRANT ALL PRIVILEGES ON ALL SEQUENCES IN SCHEMA public TO geonode;'
```

Configurar pg_hba.conf

```
$ sudo nano /etc/postgresql/13/main/pg_hba.conf
# Database administrative login by Unix domain socket
local all postgres trust
# TYPE DATABASE USER ADDRESS METHOD
# "local" is for Unix domain socket connections only
local all all md5
# IPv4 local connections:
host all all 127.0.0.1/32 md5
# IPv6 local connections:
host all all ::1/128 md5
# Allow replication connections from localhost, by a user with the
# replication privilege.
local replication all peer
host replication all 127.0.0.1/32 md5
host replication all ::1/128 md5
```

Reiniciar y probar

```
# Restart PostgreSQL to make the change effective.
$ sudo service postgresql restart
# PostgreSQL is now ready. To test the configuration, try to connect to the geonode database as geonode
role.
$ psql -U postgres geonode
# This should not ask for any password
$ psql -U geonode geonode
# This should ask for the password geonode

# Repeat the test with geonode_data DB
$ psql -U postgres geonode_data
$ psql -U geonode geonode_data
```

Instalar Nodejs

```
$ sudo apt-get install nodejs npm
$ cd geonode/geonode/static
$ npm install --save-dev
```

Configurar Local_settings

```
$ cd /home/geonode/ # or to the directory containing your cloned GeoNode
$ pip install -e geonode
$ cd geonode/geonode
# Copy https://github.com/UDannyf/GeoProject.git local_settings.py to path OR
$ cp geonode/local_settings.py.geoserver.sample geonode/local_settings.py
$ nano geonode/local_settings.py

SITEURL = "http://localhost:8000/"
DATABASES = {
'default': {
    'ENGINE': 'django.db.backends.postgresql_psycopg2',
    'NAME': 'geonode',
    'USER': 'geonode',
    'PASSWORD': 'geonode',

'datastore' : {
    'ENGINE': 'django.contrib.gis.db.backends.postgis',
    'NAME': 'geonode_data',
    'USER' : 'geonode',
    'PASSWORD' : 'geonode',

CATALOGUE = {
'default': {
    'ENGINE': 'geonode.catalogue.backends.pyctsw_local',
    'USER': 'admin',
    'PASSWORD': 'admin',
}}
```

Instalar Geoserver e iniciar Geonode

```
$ cd /home/geonode/geonode

# Make migrations to database
$ DJANGO_SETTINGS_MODULE=geonode.local_settings python manage.py migrate

# Use paver
$ DJANGO_SETTINGS_MODULE=geonode.local_settings paver setup
$ DJANGO_SETTINGS_MODULE=geonode.local_settings paver sync
$ service apache2 stop
$ service tomcat7 stop
# To run geonode
$ DJANGO_SETTINGS_MODULE=geonode.local_settings paver start
# Veriffy http://localhost:8000

# To Stop geonode
$ DJANGO_SETTINGS_MODULE=geonode.local_settings paver stop
```

Nodejs

En caso de inconvenientes con la versión de nodejs, se recomienda para la versión de geonode nodejs v12

```
Node.js v12.x:

# Using Ubuntu
$ curl -fsSL https://deb.nodesource.com/setup_12.x | sudo -E bash -
$ sudo apt-get install -y nodejs
```

Recomendaciones

1. En el caso de trabajar con una aplicación adicional de django verificar las dependencias necesarias en el entrono virtual para evitar conflictos.
2. Las configuraciones del local_settings en el caso de clonar del repositorio estas ya se encuentran disponibles.
3. En caso de trabajar con una máquina virtual asignar un mínimo de 4GB de Ram y 30GB de disco para evitar conflictos.
4. La versión de MapStore2 utilizada es la 3.3.X que ha sido agregada a los requerimientos del proyecto requirements.txt.
5. Para agregar una aplicación adicional a geonode se recomienda revisar la documentación oficial, para este proyecto se agregó la aplicación “forest” que puede servir de guía para nuevas implementaciones.

Referencias

<https://www.osgeo.org>

<https://geonode.org/>

<https://github.com/GeoNode/geonode/>

<https://docs.geonode.org/en/master/install/advanced/core/index.html#ubuntu-20-04lts>

<https://master.demo.geonode.org/#/>

<https://github.com/UDannyf/geonode>

<https://github.com/GeoNode/geonode-mapstore-client>

https://training.geonode.geo-solutions.it/GN4/mapstore_client/006_ADVANCED_DEVELOPMENT.html

<https://mapstore.readthedocs.io/en/2020.02.xx/user-guide/annotations/>

<https://mapstore.geosolutionsgroup.com/mapstore/docs/api/plugins#plugins.Annotations>

<https://mapstore.readthedocs.io/en/latest/developer-guide/developing-with-mapstore-2-intro/>

<https://www.geosolutionsgroup.com/>

<https://www.geosolutionsgroup.com/technologies/geoserver/>

<https://www.geosolutionsgroup.com/technologies/mapstore/>

<https://www.geosolutionsgroup.com/technologies/geonode/>