

SISTEMA BALANCEADO  
ODOO + POSTGRESQL

Manual de usuario

Pablo Herrero Sánchez

SGE – 2º DAM

UD2 – ACTIVIDAD 2

Contenido

[**1.** **Preparación del entorno** 1](#_Toc56590289)

[1.1. Creación Servidor 1 & 2 1](#_Toc56590290)

[**2.** **Instalación Odoo sobre Servidor 1** 6](#_Toc56590291)

[2.1. Dependencias y paquetes 6](#_Toc56590292)

[2.2. Instalación y configuración Odoo 9](#_Toc56590293)

[2.3. Configurar Odoo como servicio 14](#_Toc56590294)

[**3.** **Instalación PostgreSQL sobre Servidor 2** 15](#_Toc56590295)

[**4.** **Comunicando los servidores** 18](#_Toc56590296)

[4.1. Diciéndole a Odoo dónde está PostgreSQL 18](#_Toc56590297)

[4.2. Prueba de funcionamiento 19](#_Toc56590298)

[**5.** **Conclusión** 20](#_Toc56590299)

A lo largo de este documento explicaremos cómo crear un sistema balanceado con **Odoo** y una base de datos **PostgreSQL**, con el que conseguiremos un mayor rendimiento del sistema.

Para facilitar el seguimiento de este manual, además de las capturas de pantalla, incluyo todas las sentencias usadas, para facilitar al usuario el posible copia-pega de las mismas.

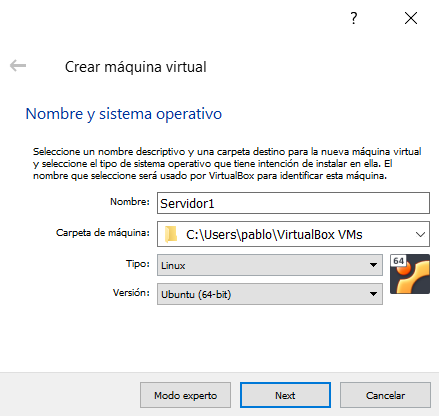
# **Preparación del entorno**

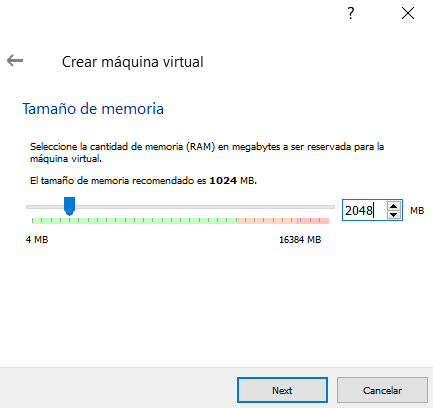
Para el propósito de este manual, prepararemos **dos máquinas virtuales con VirtualBox**, cada una de ellas hará la función de servidor, siendo **Servidor 1** donde se alojará Odoo y **Servidor 2** donde correrá la base de datos PostgreSQL.

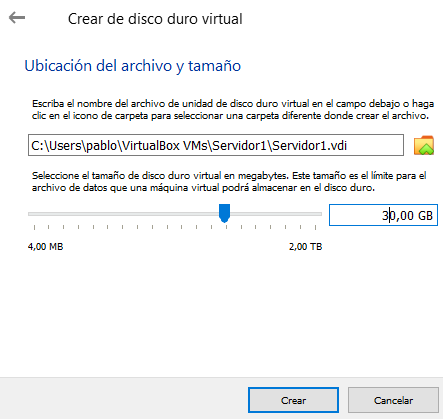
## Creación Servidor 1 & 2

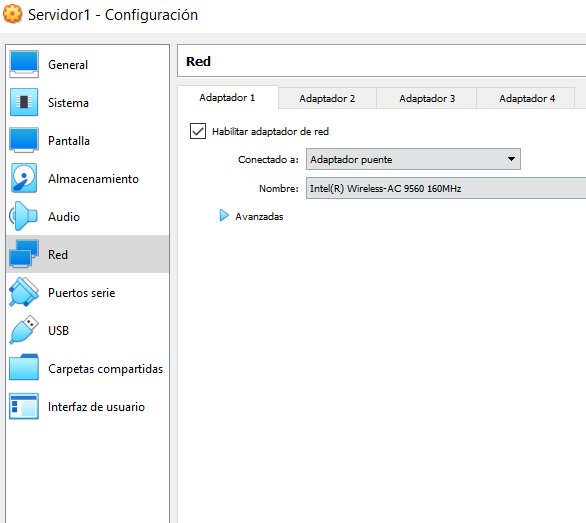
Como ambos servidores corren **Ubuntu Server 20.04.01 LTS**, la instalación del SO en ambas máquinas es exactamente igual, cambiando tan sólo el nombre que le doy al equipo. A continuación se muestra la creación de la máquina para Servidor 1, siguiendo la de Servidor 2 el mismo proceso.

1. Preparación de las **máquinas virtuales**:

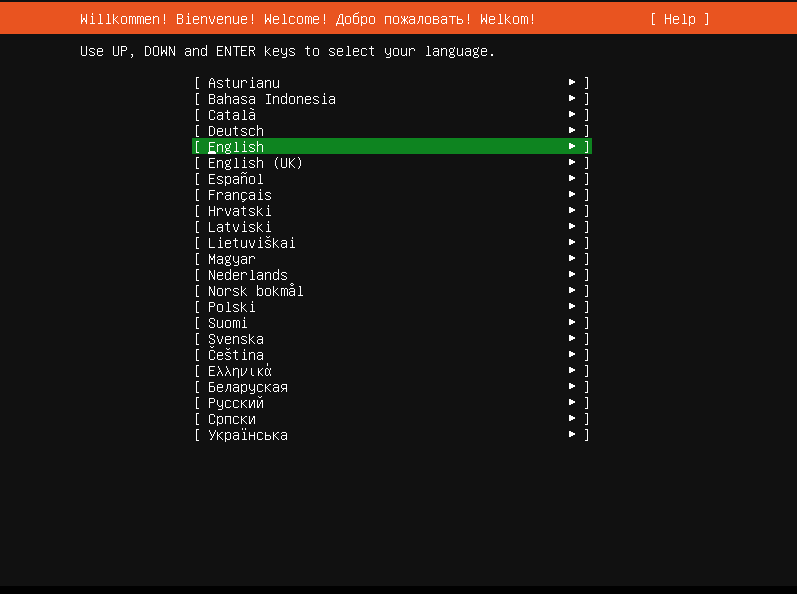


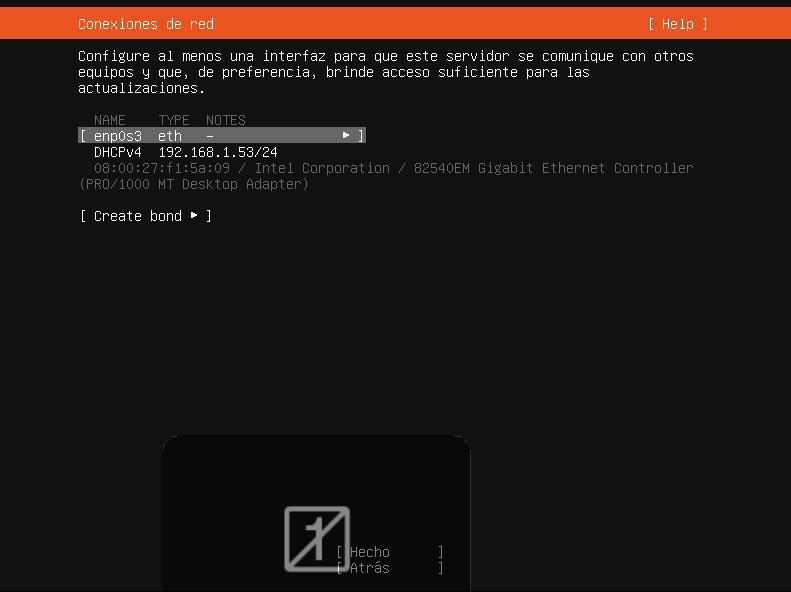


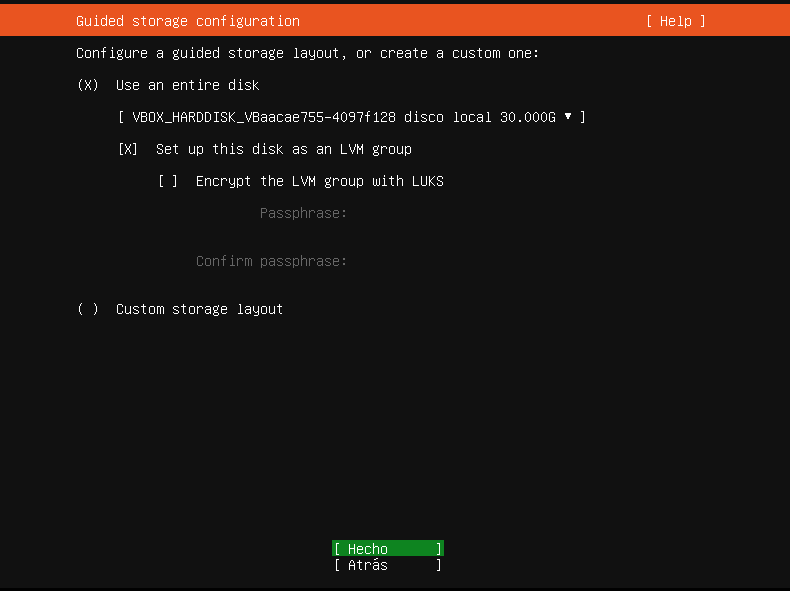


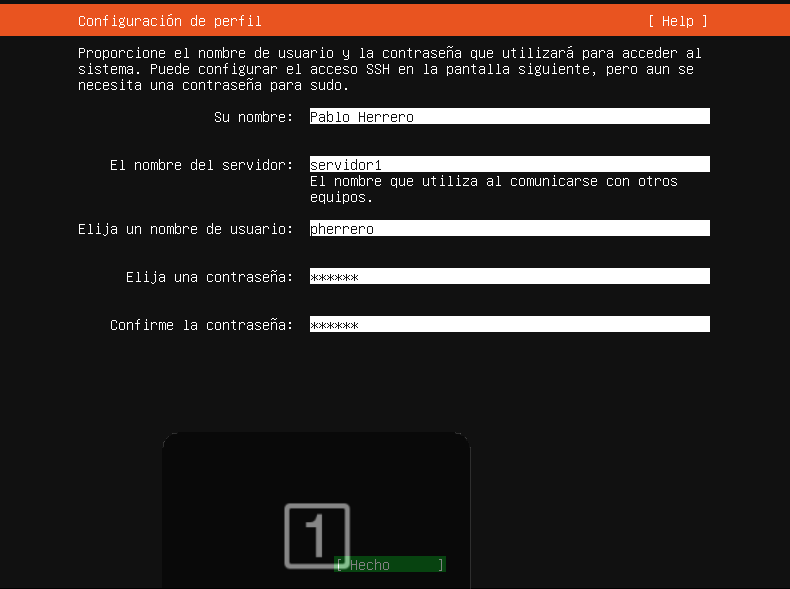


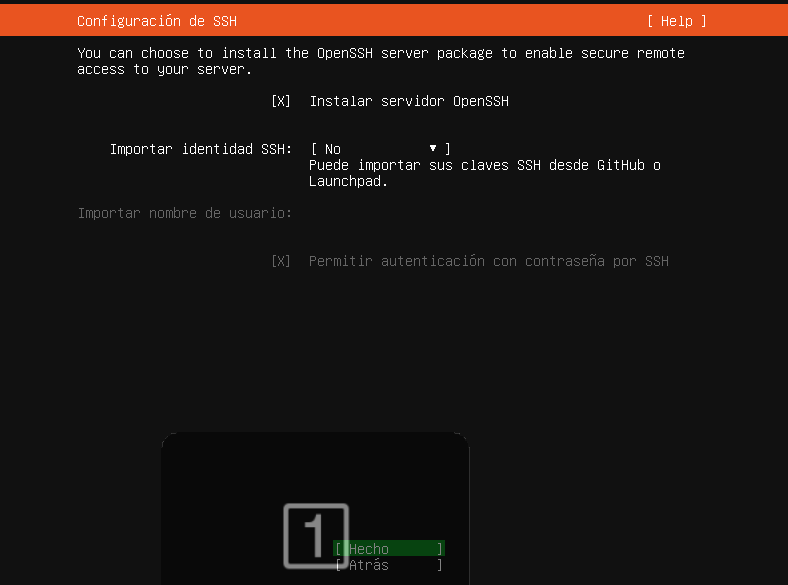
1. Instalación de **Ubuntu Server 20.04.01 LTS** sobre la máquina:



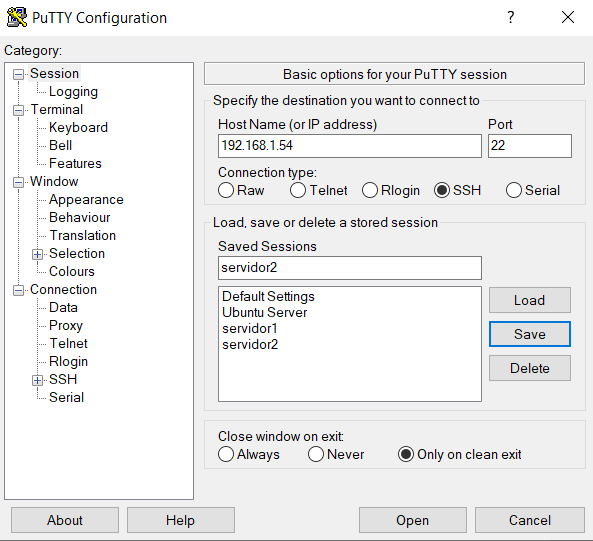








1. Conecto ambas máquinas a **PuTTY** para facilitar el resto de instalaciones:



# **Instalación Odoo sobre Servidor 1**

Para la configuración de **Odoo** en el Servidor 1 realizaremos una instalación manual del mismo:

## Dependencias y paquetes

1. Instalación de **dependencias** necesarias para Odoo, usando el comando ***sudo apt-get***. Si al descargar alguno de los paquetes nos devolviera algún error por falta de dependencias, se puede corregir con el comando ***sudo apt-get --fix-broken install***:

* *sudo apt-get install git python3 python3-pip libxml2-dev libxslt-dev libevent-dev libsasl2-dev libldap2-dev*



* *sudo apt install xfonts-base xfonts-75dpi –y*
* *sudo apt-get install -y software-properties-common*
* *sudo apt-add-repository -y "deb http://security.ubuntu.com/ubuntu bionic-security main"*
* *sudo apt-get -yq update*
* *sudo apt-get install -y libxrender1 libfontconfig1 libx11-dev libjpeg62 libxtst6 fontconfig xfonts-75dpi xfonts-base libjpeg-turbo8*

1. Instalación de **wkhtmltopdf**:

* *sudo wget* [*https://github.com/wkhtmltopdf/packaging/releases/download/0.12.6-1/wkhtmltox\_0.12.6-1.focal\_amd64.deb*](https://github.com/wkhtmltopdf/packaging/releases/download/0.12.6-1/wkhtmltox_0.12.6-1.focal_amd64.deb) [OJO, al copiar la sentencia completa en PuTTY da error “URL NOT FOUND”. Aconsejo escribir la sentencia completa en el terminal para que no dé dicho error]
* *sudo dpkg -i wkhtmltox\_0.12.6-1.focal\_amd64.deb*

1. Instalación de **npm**:

* sudo apt-get -yq install npm

Al instalar este paquete es posible que nos dé un error, solucionable con los comandos *sudo apt-get -f install* + *sudo apt-get update* + *sudo apt-get clean*.

1. Instalación de **Node.js**:

* *sudo ln -s /usr/bin/nodejs /usr/bin/node*

1. Instalación de **less-plugin-clean-css**:

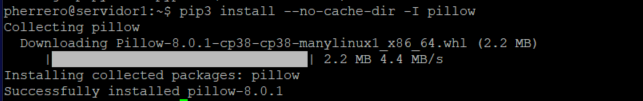
* *sudo npm install -g less less-plugin-clean-css*

1. Instalación de dependencias de **Python y pip3**:

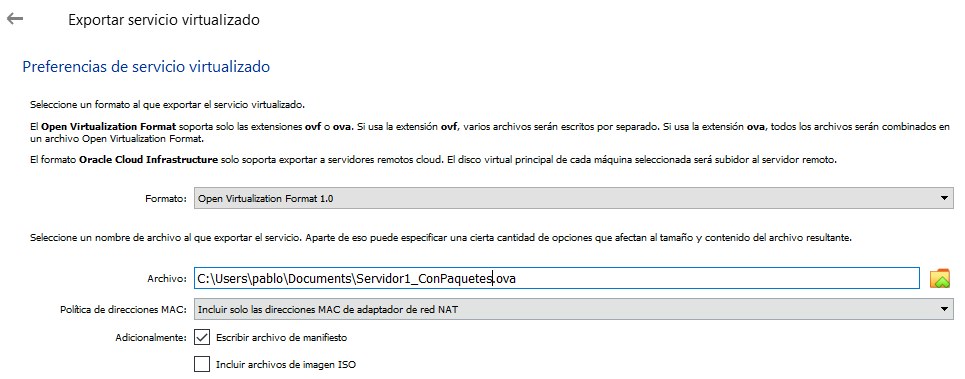
Aunque se pueden copiar y pegar la sentencia que indico a continuación, esto puede dar errores en la instalación, por lo que aconsejo, pese al tiempo que pueda llevar, instalar cada paquete individualmente.

* *sudo apt-get install python3 python3-pip fontconfig libxml2 libxml2-dev libxslt1.1 libxslt1-dev zlib1g liblcms2-2 libtiff5 tk tcl libpq5 libldap-2.4-2 libsasl2-2 libx11-6 libxext6 libxrender1 git libsasl2-dev libldap2-dev python3-passlib python3-babel python3-werkzeug python3-lxml python3-decorator python3-dateutil python3-psycopg2 python3-pil python3-psutil python3-jinja2 python3-reportlab python3-html2text python3-docutils python3-suds python3-pypdf2*

1. Instalación de **Pillow**:

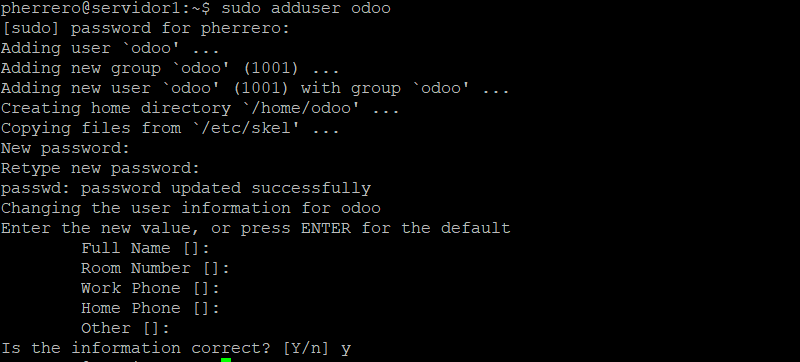
* *pip3 install --no-cache-dir -I pillow*

1. Llegado a este punto, es muy recomendable realizar una **copia de seguridad** del Servidor 1 desde VirtualBox para tener este punto de restauración en el futuro:

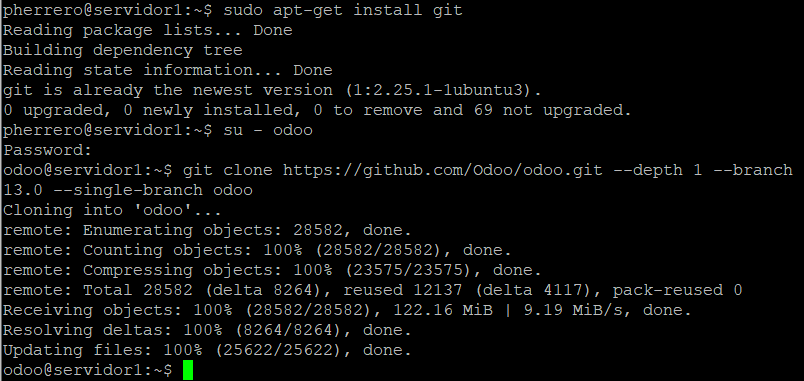


## Instalación y configuración Odoo

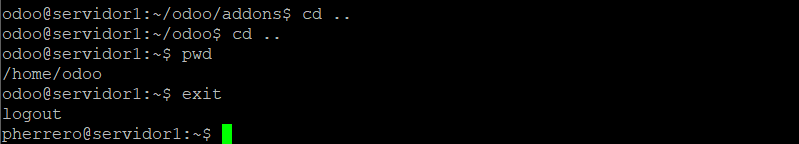
1. Crear un **usuario del sistema odoo** con los valores por defecto vacíos:

* *sudo adduser odoo*

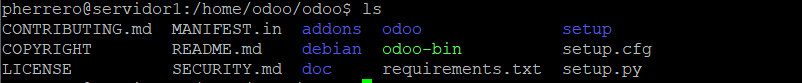
1. Descargar el **código fuente de Odoo** de su repositorio de Git, almacenándolo en el usuario creado:

* *sudo apt-get install git*
* *su – odoo git clone https://github.com/Odoo/odoo.git --depth 1 --branch 13.0 --single-branch odoo*

1. Volver al directorio de Odoo y salir del usuario Odoo:

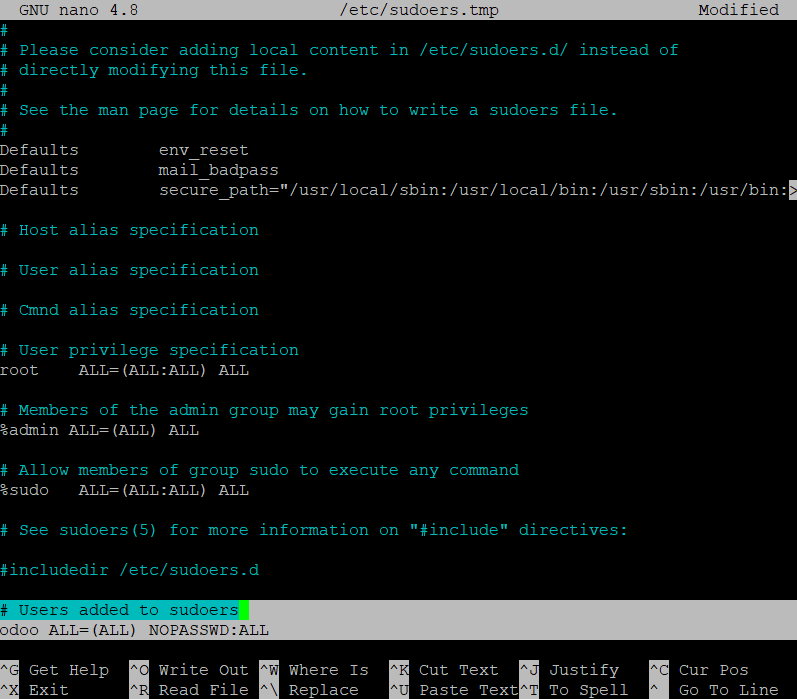


1. Instalar los requerimientos de Oddo en el archivo **requiriments.txt**. Se almacenará en /home/odoo/odoo/:

* *sudo pip3 install -r /home/odoo/odoo/requirements.txt*

1. Incluir al usuario **odoo en sudoers**.

El usuario odoo necesita acceder a requiriments.txt. Se le debe incluir en el archivo *sudoers* mediante el comando *sudo visudo* desde el usuario sudo. En este archivo es donde se configuran en un entorno Linux los permisos de los usuarios.

* *sudo visudo*
* Añadir *odoo ALL=(ALL) NOPASSWD:ALL*

Tras esta operación, logarse con el usuario odoo y comprobar el acceso a requirements.txt:

* *Nano requirements.txt*

1. Crear **odoo-server.conf**, archivo de configuración de Odoo, en el directorio del usuario *odoo*. Para ello, situarse en el directorio del usuario y crear el archivo:

* *Nano odoo-server.conf*
* Añadir este contenido al archivo:

*[options]*

*admin\_passwd = admin*

*xmlrpc = True*

*xmlrpc\_port = 8069*

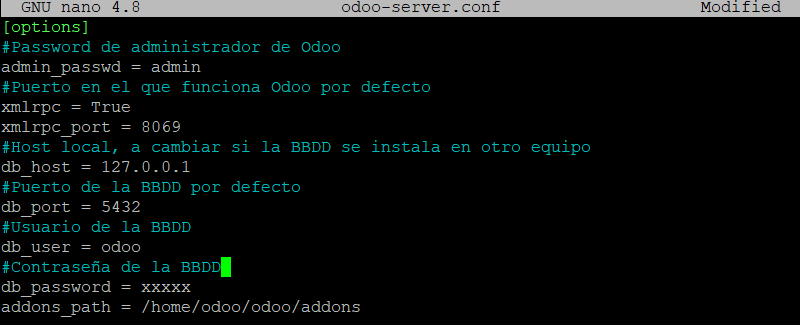
*db\_host = 127.0.0.1*

*db\_port = 5432*

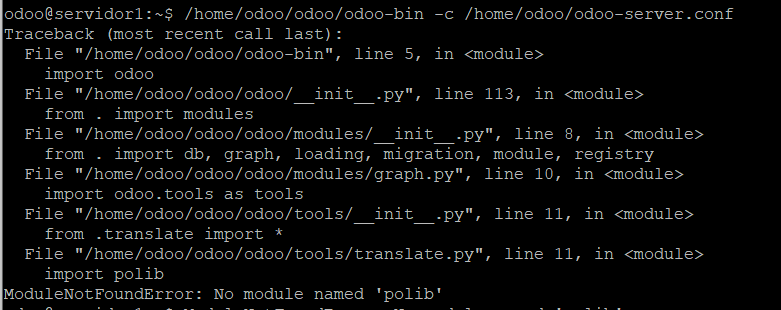
*db\_user = odoo* [usuario de la base de datos que crearemos en el siguiente apartado]

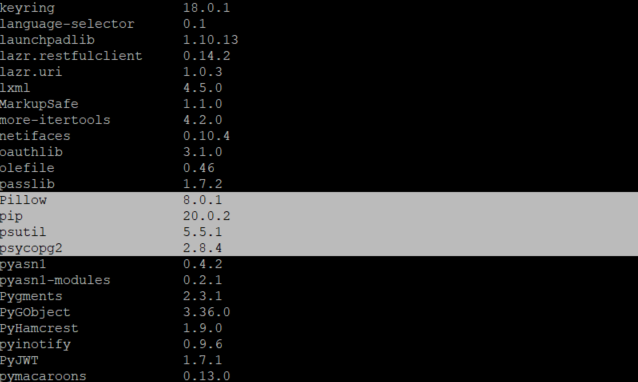
*db\_password = xxxxx* [contraseña del usuario de la base de datos]

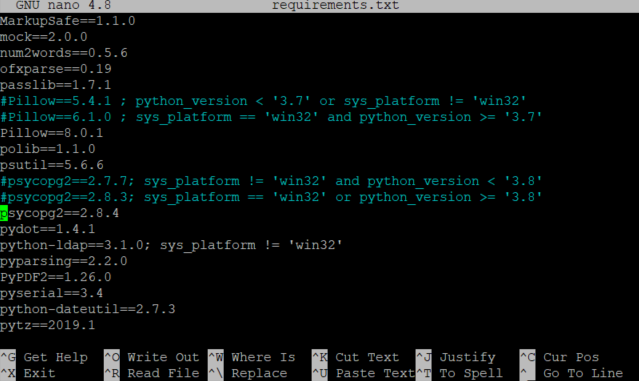
*addons\_path = /home/odoo/odoo/addons*



1. Al intentar **iniciar Odoo** es muy posible que nos devuelva este **error**:



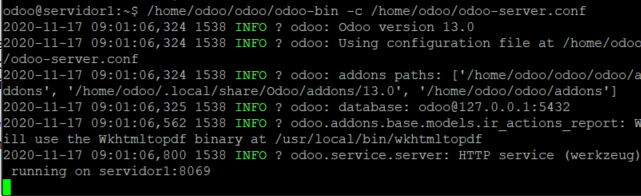
* Para solucionarlo, modificamos el achivo *requirements.txt* con los datos que nos devuelven los comandos *pip3 list* para *Pillow* y *psycopg2*



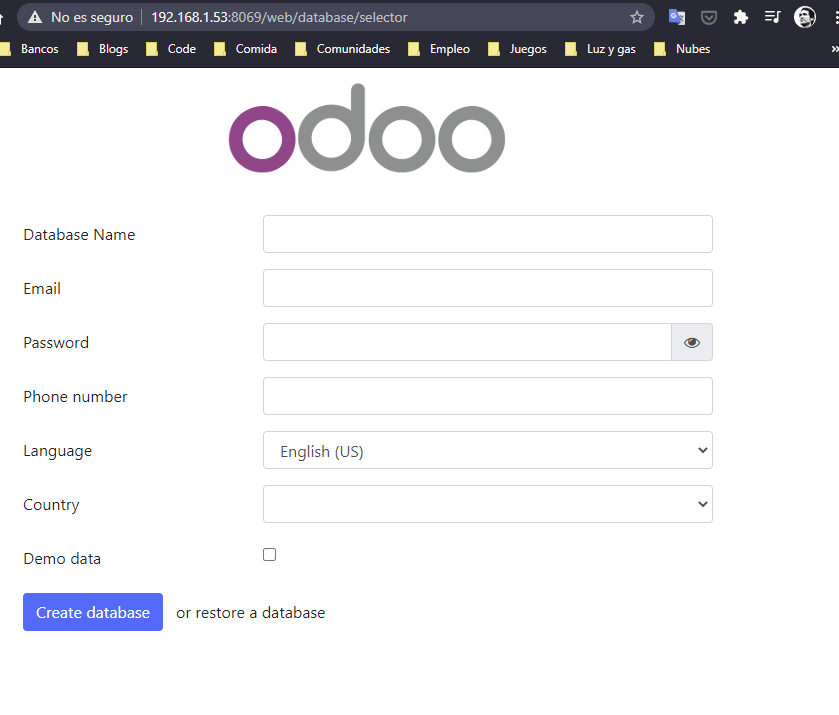
* Volvemos a instalar los requerimientos con

*sudo pip3 install -r /home/odoo/odoo/requirements.txt*

1. **Lanzamos Odoo manualmente**, estando logados como usuario odoo, y comprobamos que corre perfectamente:

* */home/odoo/odoo/odoo-bin -c /home/odoo/odoo-server.conf* 

1. Para **abrir Odoo en el navegador** del pc madre, en vez de abrir Odoo con la dirección habitual, <http://ip:8069/>, indicamos la ip del Servidor 1, en mi caso <http://192.168.1.53:8069/>



## Configurar Odoo como servicio

A continuación detallamos cómo configurar Odoo como servicio, conseguimos que se ejecute automáticamente cuando se inicia el servidor sin necesidad siquiera de tener que logarnos en la terminal y permitiéndonos, si así lo quisiéramos, seguir utilizando la consola.

1. **Creamos y editamos** el fichero **odoo13.service** en el directorio /etc/system/system/:

* *Sudo nano /etc/systemd/system/odoo13.service*



* **Añadimos** los siguientes parámetros:

*[Unit]*

*Description = Odoo*

*[Service]*

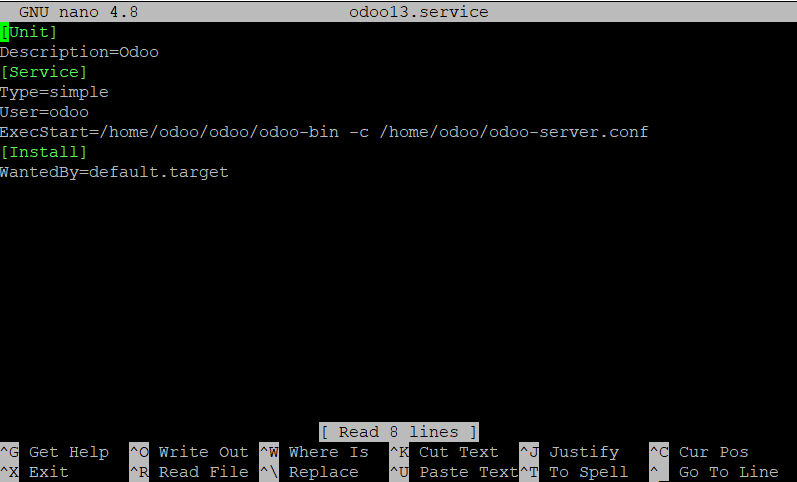
*Type=simple*

*User=odoo*

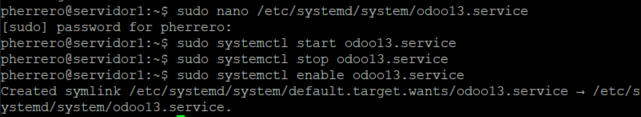
*ExecStart=/home/odoo/odoo/odoo-bin -c /home/odoo/odoo-server.conf*

*[Install]*

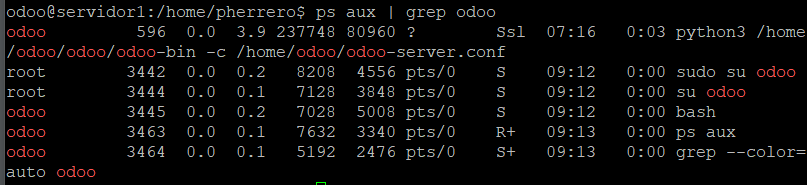
*WantedBy=default.target*



1. Le indicamos al SO que queremos que el servicio de Odoo que acabamos de crear se **arranque al inicio del sistema**, habilitando dicho servicio:

* sudo systemctl enable odoo13.service

1. **Reiniciamos el sistema y comprobamos que Odoo esté activo:**

* *sudo reboot*
* *ps aux | grep odoo*

1. Es importante recalcar los **comandos** con los que podemos **manejar el servicio de Odoo**:
   1. Iniciar: *sudo systemctl start odoo13.service*
   2. Estado: *sudo systemctl status odoo13.service*
   3. Parar: *sudo systemctl stop odoo13.service*
   4. Reiniciar: *sudo systemctl restart odoo13.service*
2. Llegado este punto, recomiendo hacer una **copia de seguridad** de la máquina virtual.

# **Instalación PostgreSQL sobre Servidor 2**

En este apartado procederemos a realizar la instalación de la base de datos PostgreSQL junto con su cliente para terminal en el Servidor 2.

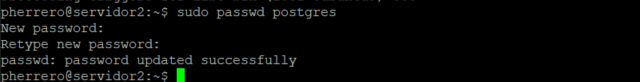
1. **Instalamos** la base de datos **PostgreSQL**:

* *sudo apt-get install postgresql*



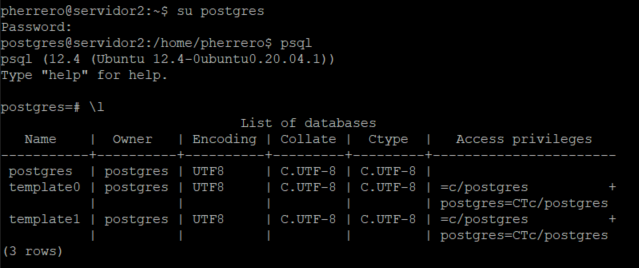
1. Asignamos una **contraseña al usuario postgres**:

* *sudo passwd postgres*



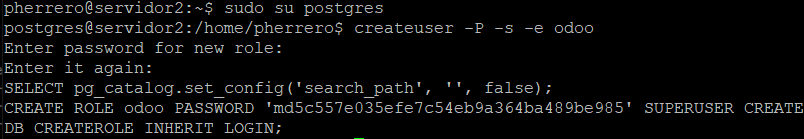
1. Nos **logamos** como postgres y **comprobamos el funcionamiento** de la base de datos:

* *su postgres*

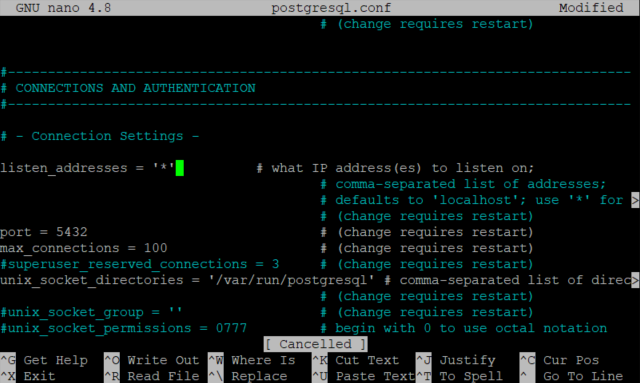


1. Logados como el usuario postgres, **creamos el usuario odoo para la base de datos:**

* *createuser -P -s -e odoo*



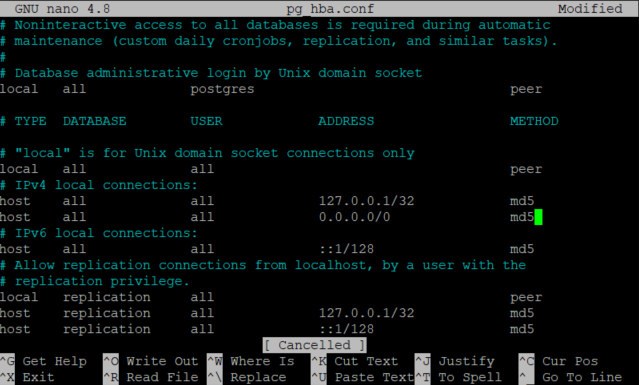
1. **Configuramos** el archivo **postgresql.conf.** Contiene los parámetros de configuración que afectan a PostgreSQL:

* Acceder a */etc/postgresql/12/main*
* Editar con *sudo nano postgresql.conf*
* Descomentar y modificar la línea *listen\_addreses*, quedando de esta forma: *listen\_addresses = ‘\*’* 

De esta forma la base de datos escucha todas las IP. Tras realizar el cambio, reiniciamos con *sudo reboot*.

1. **Configuramos** el archivo **pg\_hba.conf**:

* Nos situamos en la ruta */etc/postgresql/12/main*
* Editamos el archivo con *sudo nano pg\_hba.conf*
* Añadimos la línea *host all all 0.0.0.0/0 md5* en el apartado *IPv4*. Tras realizar el cambio, reiniciamos con *sudo reboot*:



1. **Configuramos** la **contraseña** de la base de datos.

* Nos logamos como usuario *postgres* y ejecutando el cliente *psql* aplicamos la sentencia *ALTER USER postgres PASSWORD ‘Password\_para\_base\_de\_datos’;* donde password lo sustituimos por la contraseña que queramos.

1. En este punto, aconsejo hacer una **copia de seguridad** de la máquina del Servidor 2.

# **Comunicando los servidores**

En este último apartado, comunicaremos ambos servidores y crearemos una base de datos de prueba desde Odoo para comprobar el correcto funcionamiento del sistema

## Diciéndole a Odoo dónde está PostgreSQL

1. **Detenemos el servicio Odoo en el Servidor 1**, logándonos como usuario odoo:

* *sudo systemctl stop odoo13.service*

1. **Averiguamos la IP del Servidor 2** con la sentencia *ifconfig*.
2. **Modificamos el archivo odoo-server.conf** en el **Servidor 1**:

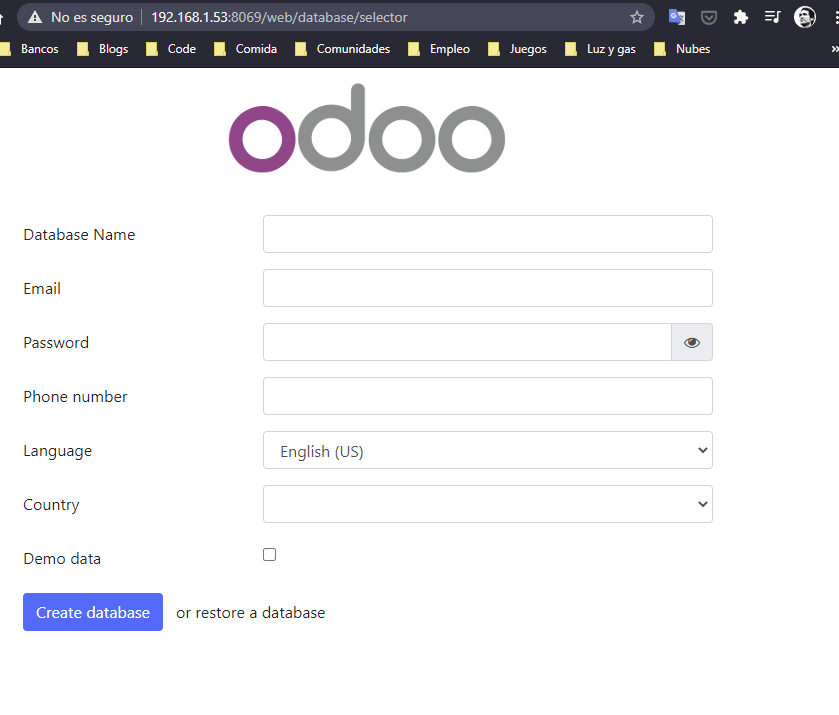
* Logándonos como usuario odoo, nos situamos en el directorio */home/odoo*.
* Editamos con *sudo nano odoo-server.conf,* indicando la ip del Servidor 2 del punto anterior.
* Verificamos el usuario de la base de datos en *db\_user* y ponemos su contraseña en *db\_password*.

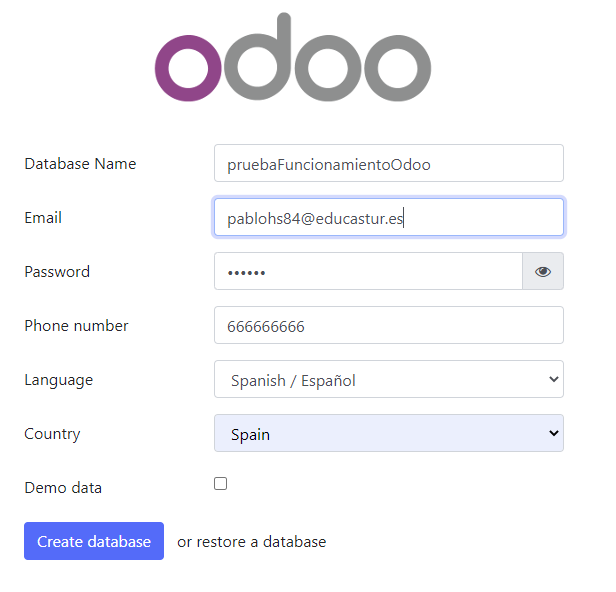
1. **Iniciamos el servicio Odoo** **en el** **Servidor 1**:

* *sudo systemctl start odoo13.service*

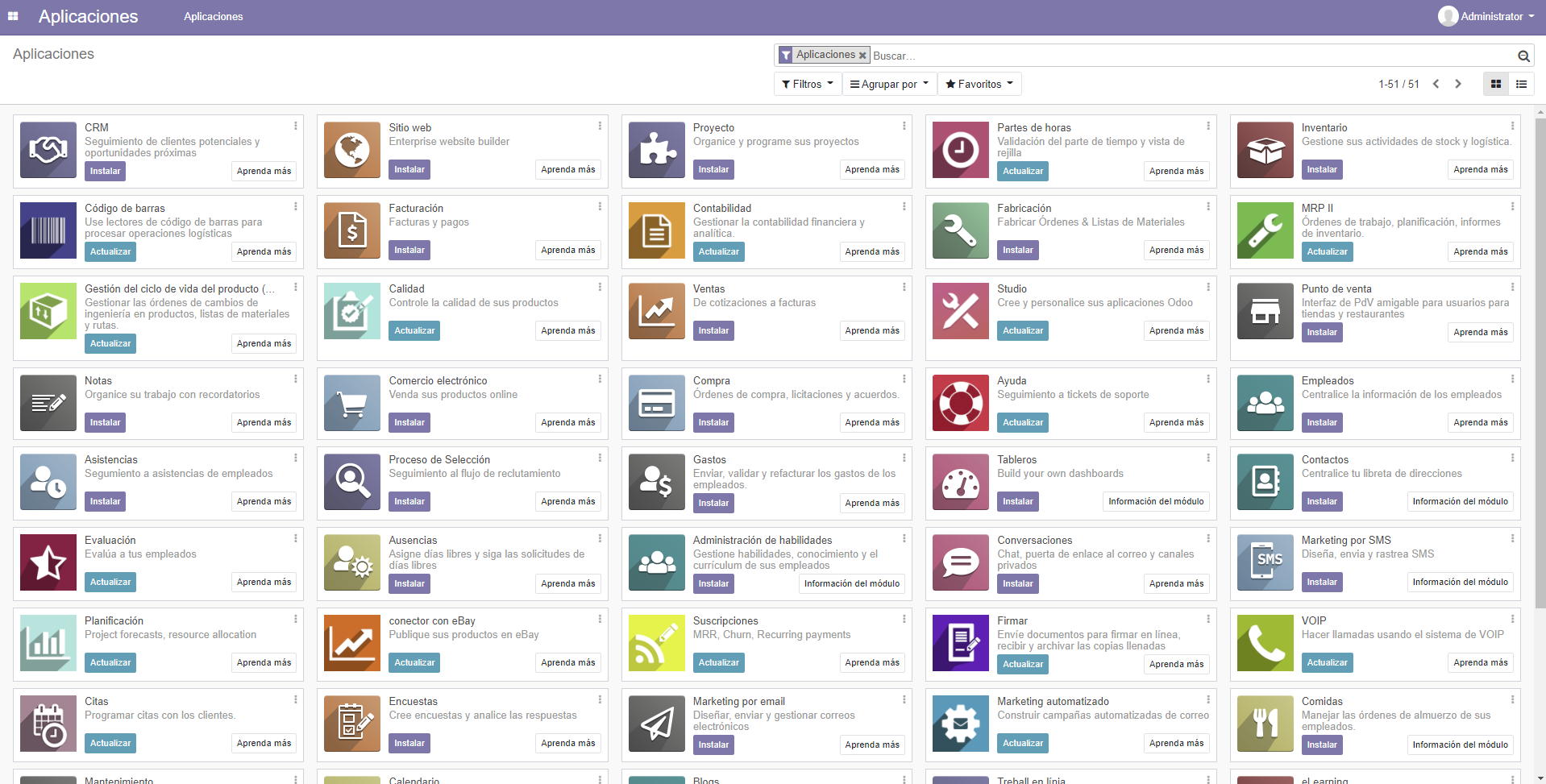
## Prueba de funcionamiento

1. Iniciamos las dos máquinas virtuales (no hay necesidad de logarnos).
2. Accedemos a Oddoo desde el navegador e introducimos los datos de la prueba:





1. Accedemos al panel de control de Odoo, verificando que el sistema funciona correctamente:



# **Conclusión**

Con los puntos indicados, ya tenemos nuestro sistema balanceado de Odoo + PostgreSQL listo para funcionar, dándonos un mayor rendimiento general en todas las operaciones que realicemos con él.

**\_**