



# Pruebas de copias de seguridad y recuperación

Estado	Done
Fecha Inicio	@13 de octubre de 2023
Nº ID	CAL-19
Fecha última edición	@12 de noviembre de 2023 21:03

## Objetivos:

- Confirmar que se están realizando copias de seguridad regulares de la base de datos y los archivos del sitio web.
- Verificar que se puede restaurar el sitio web a partir de estas copias de seguridad en caso de fallo.

## Copias de seguridad de la base de datos

Qué opciones de copias de seguridad encontramos en Odoo?

## Descargar la base de datos

The screenshot shows the Odoo web interface. At the top, there's a navigation bar with the Odoo logo and links for 'Aplicaciones', 'Comunidad', and 'Precios'. Below this is a section titled 'Databases' with a 'Create' button. A table lists databases, with 'impetucorp' (Impetu S.A.S.) selected. A context menu is open over the 'impetucorp' entry, showing options: 'Duplicar', 'Renombrar', 'Descargar' (highlighted in yellow), 'Nombres de dominio', 'Categorías', and 'Compartir'.

Descargamos la información que se esta almacenando y la verificamos, hemos obtenido una carpeta llamada filestore y un archivo de tipo sql.

El archivo que se ha descargado se puede obtener [AQUI](#)

» Descargas » impetupruebas.dump

Nombre	Tipo	Tamaño comprimido	Protegido ...	Tamaño	Relación	Fecha de modificación
filestore	Carpeta de archivos					01/11/2023 21:11
dump.sql	Archivo SQL	6.748 KB	No	56.952 KB	89%	03/11/2023 9:10

La carpeta **"filestore"** en Odoo almacena archivos relacionados con los registros y datos almacenados en la base de datos. En otras palabras, es un directorio donde se almacenan los archivos adjuntos y binarios que están asociados con los registros en Odoo, tales como: imágenes, documentos, archivos adjuntos, logotipos, entre otros

impetupruebas.dump » filestore

Nombre	Tipo	Tamaño comprimido	Protegido ...	Tamaño	Relación	Fecha de modificación
00	Carpeta de archivos					14/07/2023 2:36
0a	Carpeta de archivos					02/08/2023 14:03
0b	Carpeta de archivos					14/07/2023 2:33
0c	Carpeta de archivos					14/07/2023 2:36
0d	Carpeta de archivos					14/07/2023 2:36
0e	Carpeta de archivos					14/07/2023 2:36
01	Carpeta de archivos					04/08/2023 16:14
1a	Carpeta de archivos					14/07/2023 16:15
1b	Carpeta de archivos					11/10/2023 16:12
1c	Carpeta de archivos					03/10/2023 16:12
1d	Carpeta de archivos					24/10/2023 2:06
1e	Carpeta de archivos					28/07/2023 16:11
1f	Carpeta de archivos					14/07/2023 2:36
02	Carpeta de archivos					21/08/2023 16:12
2a	Carpeta de archivos					14/07/2023 2:36
2b	Carpeta de archivos					10/10/2023 16:11
2c	Carpeta de archivos					23/09/2023 16:12
2d	Carpeta de archivos					11/10/2023 6:54
2e	Carpeta de archivos					14/07/2023 7:18
2f	Carpeta de archivos					26/09/2023 16:11
03	Carpeta de archivos					14/07/2023 2:36
3a	Carpeta de archivos					25/09/2023 16:12
3b	Carpeta de archivos					11/10/2023 5:26
3c	Carpeta de archivos					07/09/2023 16:11
3d	Carpeta de archivos					14/07/2023 16:15
3e	Carpeta de archivos					14/07/2023 2:36
3f	Carpeta de archivos					01/11/2023 0:07

El archivo **"dump.sql"** en Odoo es un respaldo o volcado de la base de datos de Odoo en formato SQL. Contiene una copia de la estructura de la base de datos y de todos los datos almacenados en ella en un momento específico. En otras palabras, es una representación completa de la base de datos de Odoo en un punto dado en el tiempo.

El archivo **"dump.sql"** incluye información sobre tablas, campos, índices, claves primarias y todos los registros almacenados en la base de datos. Se puede utilizar este archivo para realizar copias de seguridad de la base de datos, migrar datos entre diferentes instancias de Odoo, restaurar la base de datos en caso de fallos o pérdida de datos, y realizar tareas de importación o exportación de datos.

```
dump.sql
Archivo  Editar  Ver

--
-- PostgreSQL database dump
--

-- Dumped from database version 14.9 (Ubuntu 14.9-0ubuntu0.22.04.1)
-- Dumped by pg_dump version 14.9 (Ubuntu 14.9-0ubuntu0.22.04.1)

SET statement_timeout = 0;
SET lock_timeout = 0;
SET idle_in_transaction_session_timeout = 0;
SET client_encoding = 'UTF8';
SET standard_conforming_strings = on;
SELECT pg_catalog.set_config('search_path', '', false);
SET check_function_bodies = false;
SET xmloption = content;
SET client_min_messages = warning;
SET row_security = off;

--
-- Name: unaccent; Type: EXTENSION; Schema: -; Owner: -
--

CREATE EXTENSION IF NOT EXISTS unaccent WITH SCHEMA public;

--
-- Name: EXTENSION unaccent; Type: COMMENT; Schema: -; Owner: -
--

COMMENT ON EXTENSION unaccent IS 'text search dictionary that removes accents';

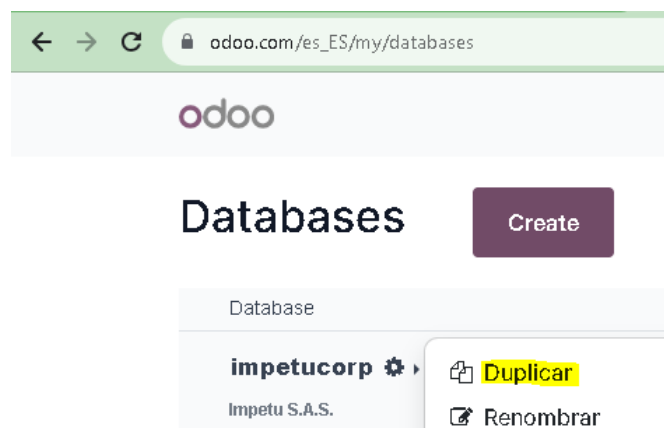
SET default_tablespace = '';

SET default_table_access_method = heap;

--
-- Name: _ir_translation; Type: TABLE; Schema: public; Owner: -
--

CREATE TABLE public._ir_translation (
  id integer NOT NULL,
  name character varying NOT NULL,
  res_id integer,
  lang character varying,
  type character varying,
  src text,
  value text,
  module character varying,
  state character varying,
  comments text
```

## Duplicar la base de datos



Duplicar base de datos

Escoger un nombre

impetupruebas

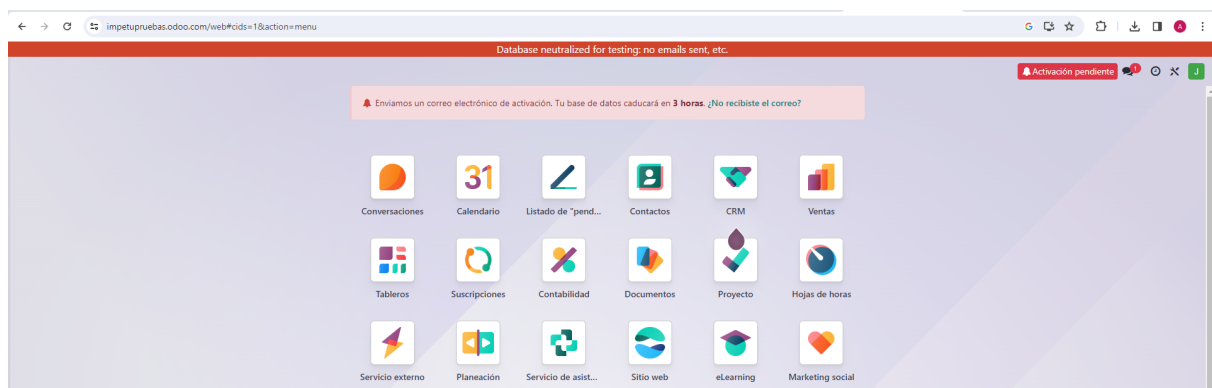
☒ Con fines de pruebas

Si se encuentra seleccionado, las integraciones adicionales, los correos electrónicos salientes y las acciones programadas se van a desactivar.

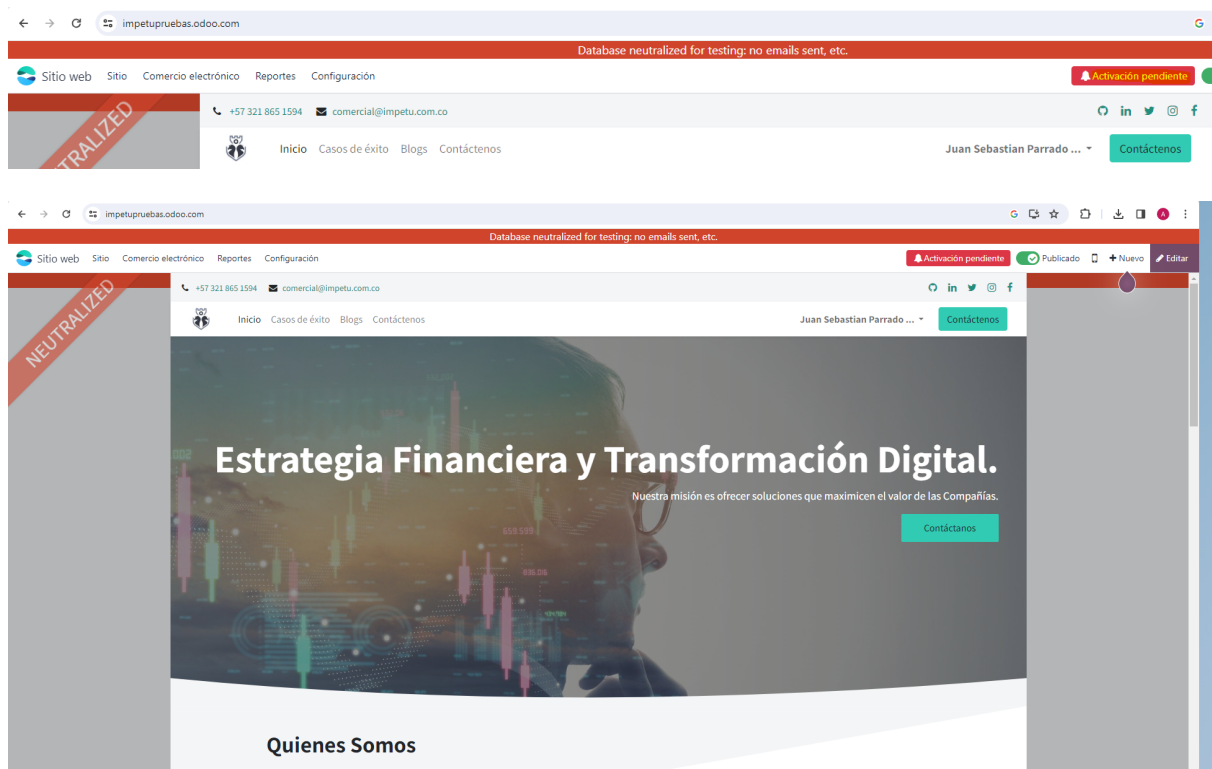
Duplicar

Cancelar

Al seleccionar la opción de duplicar la bases de datos, podemos dar clic en conectar, en ese momento se establece la conexión con las aplicaciones de odoo y se nos muestra un mensaje indicando que el sitio esta configurado para testing

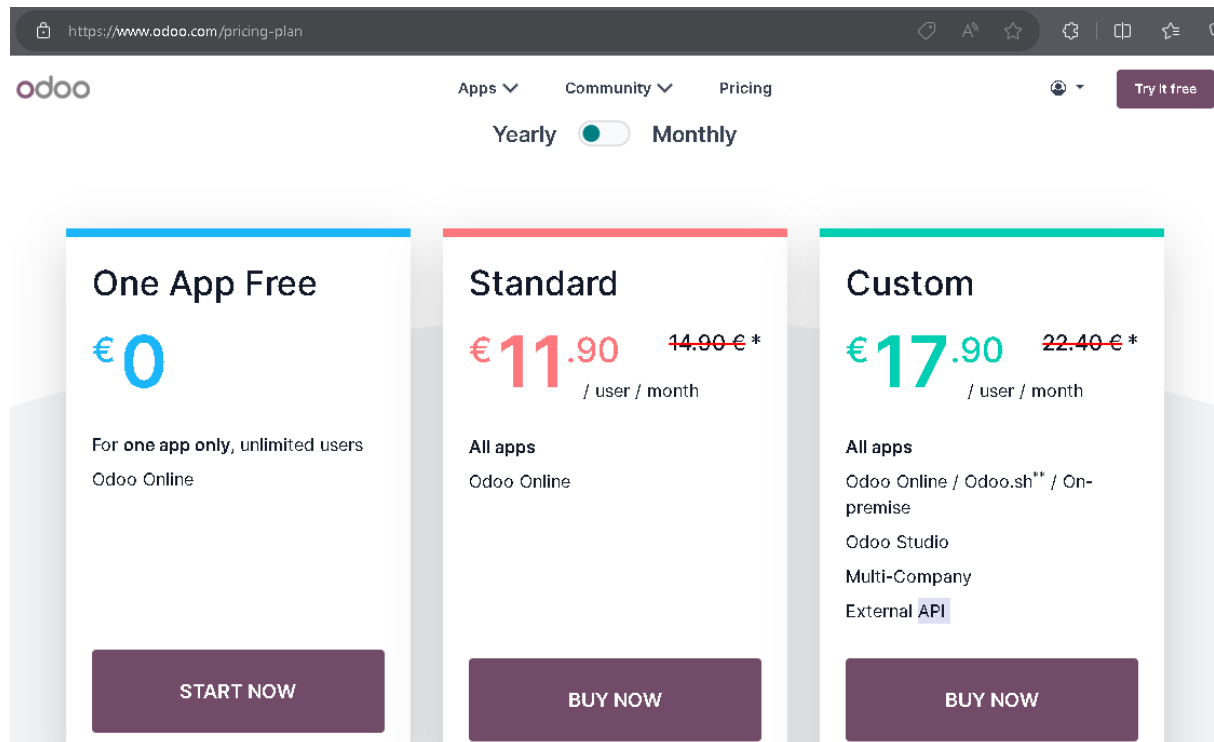


Al elegir Sitio web observamos que el sitio se encuentra también en un área de testing



Para realizar otras pruebas es necesario activar la zona de pruebas, en esta zona de pruebas podremos personalizar las aplicaciones a usar en el sitio web y a su vez editar las páginas web del mismo, en la versión gratuita que se adquirió

para el sitio <https://www.impetu.com.co/> esta información se conservará por solo 3 días, para almacenarla por un período mayor será necesario adquirir un plan de pago adicional, en el plan de pago adicional contaremos con odoo. sh.



### Cómo funciona Odoo.Sh?

Odoo SH es la plataforma oficial de alojamiento de Odoo, diseñada para dar cabida a las necesidades de los desarrolladores. Como un servicio de plataforma (PaaS) combinado con una infraestructura como servicio (IaaS), ofrece una solución completa para el desarrollo, pruebas y despliegue de aplicaciones basadas en Odoo.

Tal como está diseñada la plataforma, su aplicación aportará mayores beneficios a aquellos usuarios que buscan una solución todo-en-uno para sus necesidades de desarrollo de software. Con [Odoo.sh](https://www.odoo.sh), se podrán realizar despliegues en producción de manera sencilla, lanzar un servidor de pruebas y probar nuestros desarrollos, todo integrado estrechamente con GitHub. Además, esta plataforma brinda la posibilidad de instalar módulos de la comunidad e incluso agregar aplicaciones de terceros disponibles en la tienda de aplicaciones de Odoo.

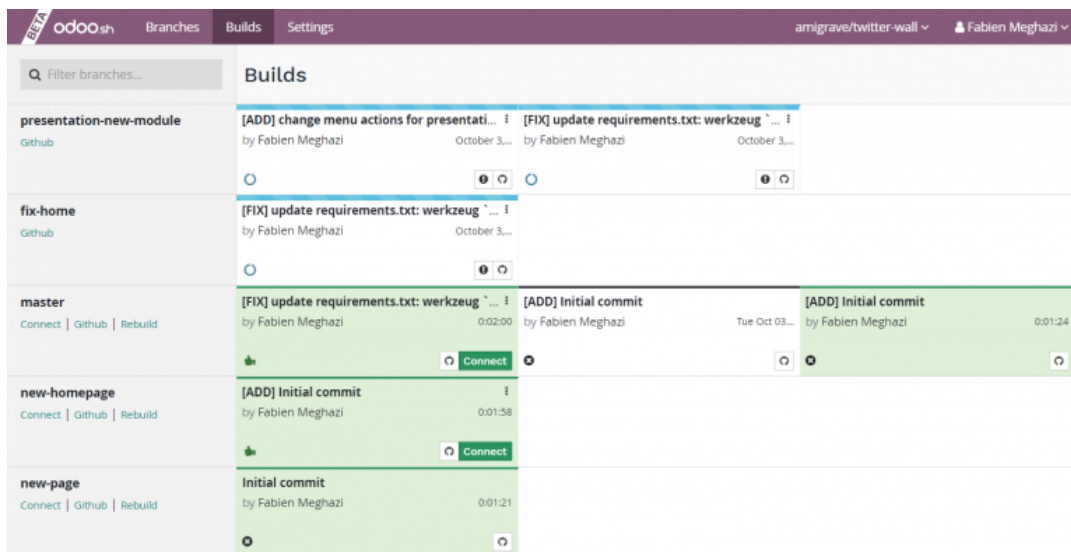
[Odoo.sh](https://www.odoo.sh) proporciona una variedad de herramientas y características ayudarán a desarrollar nuevas aplicaciones y funcionalidades personalizadas, incluyendo:

- Integración con Github
- Un editor en línea
- Integración continua (tu propio «runbot»)
- Un entorno de desarrollo y pruebas
- Gestión de ramas (seguimiento, bifurcación, fusión, etc.)
- Pruebas automatizadas de cada nuevo commit

Esto implica que es una alternativa a los despliegues de Odoo On-premise que tienen su entorno de desarrollo gestionado por personal propio. De manera que dependiendo de las necesidades [Odoo.sh](https://www.odoo.sh) puede suponer un ahorro en recursos, económicos y de personal, frente a la gestión de nuestra propia infraestructura.

Una de las características destacadas es la capacidad de realizar pruebas en cada rama de características (feature branch) con un solo clic, facilitando el proceso de pruebas y garantizando que cada función opere correctamente antes

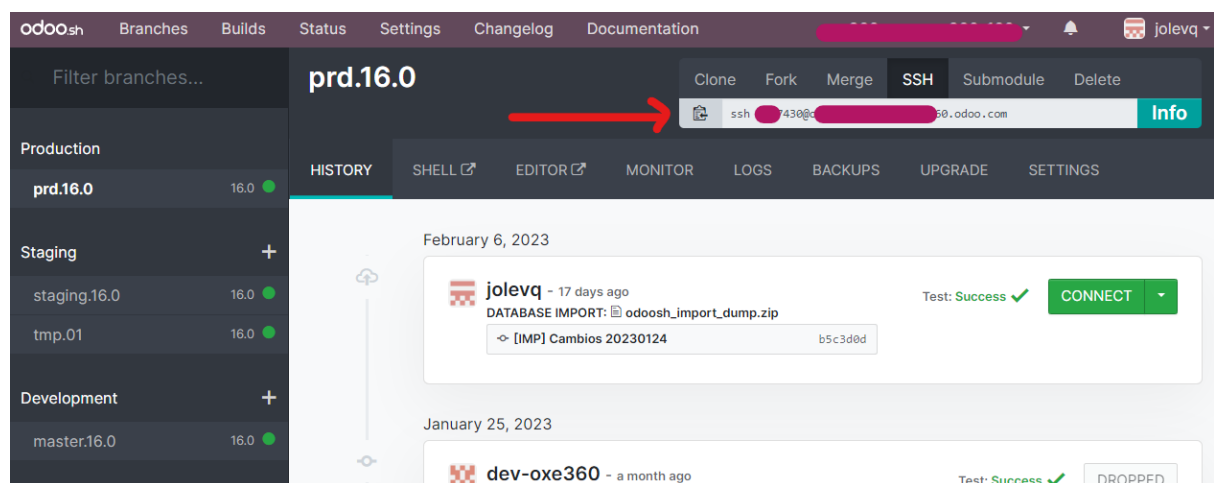
de su paso a producción.



## Cómo usar los backups de base de datos de Odoo.Sh?

### Paso 1: Acceder al servidor de Odoo

En primer lugar, los usuarios deben acceder al servidor de Odoo para descargar el backup de base de datos. Esto se puede hacer a través de la interfase web o por línea de comandos SSH. Para el último caso se requiere haber configurado previamente la llave pública (archivo id\_rsa.pub) del servidor destino en el perfil del usuario con el que se realiza la operación.



### Paso 2: Copiar la base de datos a local

Una vez que los usuarios tengan el acceso al servidor de [Odoo.sh](#), deben copiar la base de datos a través de la línea de comandos. Esto se puede hacer utilizando el comando "scp" (secure copy). Por ejemplo, el backup diario que genera [Odoo.sh](#) siempre se encuentra en la ruta "/home/odoo/backup.daily/" si el nombre de la base de datos es "my\_database" y mi acceso al servidor [Odoo.sh](#) del paso anterior es "user@serverodoosh", el siguiente comando se puede utilizar para copiar la este backup de base de datos comprimido:

```
scp -r user@serverodoosh:/home/odoo/backup.daily/mydatabase_daily.sql.gz /path/to/local/folder
```

### Paso 3: Copiar la base de datos a ubicación destino

Una vez que los usuarios hayan copiado la base de datos a su ordenador local, deben ubicarla en la ruta destino para realizar el import a la base de datos postgres. En nuestro ejemplo, consideramos que nuestra base de datos postgres se

encuentra dentro de un contenedor Docker "db" dentro del servidor local. Utilizar un contenedor Docker para instalar postgres es muy conveniente, pues solo toma unos minutos y es bastante eficiente. [Puedes encontrar la imagen oficial de Docker para Postgres aquí](#). El comando para copiar la base de datos de local a el contenedor Docker de postgres, asumiendo que el nombre del contenedor es "db" sería:

```
docker cp /path/to/local/folder/mydatabase_daily.sql.gz db:/docker-entrypoint-initdb.d/mydatabase_daily.sql.gz
```

El nombre del archivo "mydatabase\_daily.sql.gz" en el destino podría ser diferente, pero lo mantenemos por simplicidad. Adicionalmente, como el archivo backup está en formato comprimido .gz es necesario descomprimirlo para poder utilizarlo.

```
docker exec -it db sh -c "gunzip -f /docker-entrypoint-initdb.d/mydatabase_daily.sql.gz"
```

con el parámetro "-f" nos aseguramos de reemplazar cualquier archivo anterior en el caso que este sea un proceso recurrente.

#### **Paso 4: Preparar el restore (opcional)**

Si estamos utilizando Dockers en tu entorno para la base de datos postgres, y este es un entorno que también tiene instalado odoo, es muy probable que también tengamos contenedores de odoo y nginx. Asegúrate de detener los contenedores antes de hacer la restauración pues se requiere borrar y volver a crear una base de datos vacía en donde se ejecute el restore.

Para apagar tus contenedores odoo y nginx puedes utilizar los comandos

```
docker stop odoo nginx db
```

Luego para eliminar y volver a crear la base de datos dentro del contenedor "db" puedes usar este comando:

```
echo "DROP DATABASE IF EXISTS my_database; CREATE DATABASE my_database;" | PGPASSWORD=odoo psql -h localhost -U odoo -d postgres
```

Como en el paso anterior se detuvo el contenedor "db" para cerrar todas las sesiones, recuerda encenderlo nuevamente antes de borrar y crear la base de datos.

#### **Paso 5: Restaurar el backup de base de datos**

Una vez que se tiene el backup de la base de datos en el contenedor de la base de datos, el siguiente paso es restaurar el backup en la base de datos vacía creada en el Paso 4. Para esto, se debe ejecutar el siguiente comando:

```
docker exec -u postgres db psql my_database odoo -f docker-entrypoint-initdb.d/mydatabase_daily.sql
```

Este comando ejecuta el cliente psql dentro del contenedor de la base de datos y se conecta a la base de datos "my\_database". Luego, ejecuta el script SQL del backup utilizando la opción -f de psql.

Este paso puede tardar algunos minutos dependiendo del tamaño del backup y de la velocidad del servidor. Una vez que la restauración ha finalizado, se tiene una base de datos idéntica a la base de datos original. Como referencia, en un servidor Medim de AWS EC2, una base de datos que en [Odoo.sh](#) ocupa 13.5GB podría demorar unos 10 a 15mins.

#### **Paso 6: Activar los clientes (opcional)**

Finalmente, es necesario iniciar los contenedores de Odoo y Nginx que en nuestro ejemplo eran utilizados como capa de aplicación. Dependerá del uso que se le de a esta restauración de base de datos, debemos asegurarnos de contar con los accesos necesarios para poder acceder a ella. Para esto encender los contenedores Odoo y Nginx de nuestro, se deben ejecutar los siguientes comandos:

```
docker start odoo
docker start nginx
```

### **Paso 7: Cierre y Limpieza**

Especialmente si este es un proceso recurrente, siempre es importante mantener "la casa limpia" y preparada para la siguiente ejecución. Se recomienda eliminar los archivos grandes descargados o moverlos a una ubicación de almacenamiento diferente. Esto facilitará mucho la vida cuando querremos asegurarnos de que estamos utilizando el backup adecuado.

### **Script de restauración**

Un script de restauración de odoo tendría un formato similar al siguiente:

```
#Script restore_database.sh
#!/bin/bash

cd /mi/ruta/local
echo "Copiar backup desde OdooSH"
scp miusuario@miserverodoosh.odoo.com:/home/odoo/backup.daily/my_database_daily.sql.gz .
echo "Copiar backup al contenedor DB"
docker cp ./my_database_daily.sql.gz db:/docker-entrypoint-initdb.d/my_database_daily.sql.gz
echo "Descomprimir backup en el contenedor DB"
docker exec -it db sh -c "gunzip -f /docker-entrypoint-initdb.d/my_database_daily.sql.gz"
echo "detener contenedores"
docker stop odoo nginx db
echo "iniciar contenedor DB"
docker start db
sleep 30s
echo "borrar base de datos y crearla de nuevo"
echo "DROP DATABASE IF EXISTS my_database; CREATE DATABASE my_database;" | PGPASSWORD=odoo psql -h localhost -U odoo -d postgres
echo "Restaurar backup en la base de datos"
docker exec -u postgres db psql my_database odoo -f docker-entrypoint-initdb.d/my_database_daily.sql
echo "Iniciar contenedores"
docker start odoo
docker start nginx
echo "Borrar backup"
rm *_daily*
```