

## Instalar PostgreSQL y PgAdmin (o Dbeaver)

Tienen dos opciones: o utilizan docker (con docker-compose) para ejecutar la imagen que se encuentra subida a la plataforma moodle ó pueden descargar el motor PostgreSQL y PgAdmin e instalarlos.

### Opción 1: Descarga e instalación del motor PostgreSQL y PgAdmin

#### En windows:

##### Instalación de PostgreSQL

1. Descargar desde el sitio oficial el motor de PostgreSQL <https://www.postgresql.org/download/> seleccionando el sistema operativo Windows.
2. Ejecutar el archivo de instalación descargado y seguir las instrucciones (dejemos las configuraciones por defecto)
3. Durante la instalación establecer una contraseña para el usuario *postgres*
4. Para iniciar el servicio PostgreSQL se puede hacer manualmente desde el Administrador de servicios de windows o reiniciar la PC para que se ejecute automáticamente.

##### Instalación de PgAdmin

1. Descargar la plataforma de administración PgAdmin en el siguiente enlace: <https://www.pgadmin.org/download/#> ó si quieren utilizar otra plataforma podrán descargar Dbeaver en su Community edition: <https://dbeaver.io/download/>
2. Ejecutar el archivo de instalación descargado y seguir las instrucciones (dejemos las configuraciones por defecto)
3. Al finalizar la instalación, PgAdmin debería abrir automáticamente en el navegador web. Si no se abre, podés buscar pgAdmin en el menú de inicio y ejecutarlo.

#### En Ubuntu:

##### Instalación de PostgreSQL:

1. Abrir una terminal y usando el gesto de paquetes apt, actualizar la lista de paquetes con “sudo apt update”.
2. Instalar PostgreSQL con el comando “sudo apt install postgresql postgresql-contrib” (postgresql-contrib es un paquete opcional que agrega más funcionalidades, no es necesario para la instalación)
3. Ahora PostgreSQL está instalado y el servicio será iniciado automáticamente, podés verificar el estado con el comando “sudo systemctl status postgresql”.
4. Configurar PostgreSQL: Crear una contraseña para el usuario *postgres* con el comando “sudo -i -u postgres” cambiamos a la cuenta del usuario postgres y con el comando “psql” ingresamos la consola de PostgreSQL. Ahí cambiamos la contraseña del usuario *postgres* con el comando “ALTER USER postgres PASSWORD ‘la\_contraseña’ (podés salir con el comando \q)
5. Para permitir conexiones remotas hay que editar el archivo de configuración de PostgreSQL
  - con vim o nano: “vim /etc/postgresql/<aca\_tu\_versión\_de\_postgresql>/main/postgresql.conf”.
  - Dentro de ese archivo en la línea #listen\_addresses cambiar a: “listen\_addresses = ‘\*’.
  - Abrir el archivo pg\_hba.conf con “vim /etc/postgresql/<aca\_tu\_versión\_de\_postgresql>/main/pg\_hba.conf y al final del archivo agregar la línea para permitir todas las conexiones: “host all all 0.0.0.0/0 md5”
- Guardar los cambios y reiniciar el servicio con “sudo systemctl restart postgresql”.

##### Instalación de PgAdmin

1. Actualizar los paquetes con “sudo apt update”
2. Instalar PgAdmin con “sudo apt install pgadmin4” (durante la instalación se va a pedir configurar el usuario “admin” y la contraseña de ingreso)

3. Una vez instalado, se puede acceder desde el navegador a la URL: <http://localhost/pgadmin4>, iniciar sesión y configurar el servidor.

## Opción 2: Docker

¿Qué es Docker (<https://docs.docker.com/get-started/overview/>)? Es una tecnología que permite la creación, ejecución y la posibilidad de compartir aplicaciones en forma rápida y sencilla que corren en contenedores (entornos aislados).

Un contenedor es una unidad que incluye todas las herramientas y dependencias que necesita una aplicación para funcionar, independientemente del entorno en el que es ejecutada.

¿Que nos proporciona Docker?

Facilidad para el desarrollo de aplicaciones: se puede crear un entorno de desarrollo consistente para las aplicaciones (asegurando que la aplicación se va a comportar de la misma manera en distintos entornos, lo que evita problemas relacionados con los diferentes sistemas operativos y las distintas configuraciones).

Aislamiento de aplicaciones: cada contenedor ejecuta su propia instancia de la aplicación, lo que permite evitar conflictos con otras aplicaciones o sistemas en el mismo entorno.

Portabilidad: se puede ejecutar en cualquier sistema que tenga Docker instalado, lo que facilita la migración y el despliegue en los distintos entornos.

Escalabilidad

Despliegue rápido

Compartir aplicaciones fácilmente

Docker compose (<https://docs.docker.com/compose/>): es una herramienta que permite no sólo definir las aplicaciones multi contenedor sino gestionar estas aplicaciones (definir, configurar y ejecutar múltiples contenedores Docker)

En Windows y Mac Docker compose se instala como parte de Docker Desktop (<https://docs.docker.com/desktop/install/windows-install/>), y en Linux se puede instalar por separado.

Instalar Docker Desktop: <https://docs.docker.com/desktop/install/windows-install/>

1. En moodle tendrán disponible una imagen que contiene PostgreSQL y PgAdmin, descargar el archivo y almacenar en un directorio creado a tal fin.

2. En el archivo docker-compose.yml modificar las variables de entorno:

- Reemplazar como usuario de PgAdmin donde dice "[npramos@untdf.edu.ar](mailto:npramos@untdf.edu.ar)", colocar la dirección de correo que vayan a usar para acceder (o usuario).
- Reemplazar "123456" por la clave que vayan a usar para acceder a PgAdmin.
- En postgres reemplazar la contraseña "123456" por la contraseña que vayan a utilizar para acceder a PostgreSQL.

3. En el directorio que crearon y donde colocaron la imagen descargada ejecutar: "docker-compose up -d".

4. Verificar que el contenedor fue correctamente iniciado con el comando "docker ps".

5. Ingresá a PgAdmin en el navegador web, de acuerdo a la configuración que tengas en el archivo docker-compose.yml (localhost:5051 que está enmascarando el puerto 80).

## En Windows y Ubuntu (con o sin docker)

### Configuración de PgAdmin:

1. Iniciar sesión ingresando las credenciales (por defecto el usuario es admin) y la contraseña es la que se configuró durante la instalación de PgAdmin.
2. Agregar un servidor, en el panel izquierdo hacer clic con el botón derecho en “Servers” y seleccionar “Create” y luego “Server”.
3. Configurar el servidor: en la pestaña “General”, en el campo “Name” pondrán un nombre descriptivo para el servidor. En la pestaña “Connections” se configurarán los siguientes campos:
  - Host name/address: Ingresá la dirección IP o el nombre del host del servidor PostgreSQL al que vayas a conectarte (si PostgreSQL está en la misma máquina que estás usando podés usar "localhost", si instalaste utilizando la imagen docker entonces podés usar la resolución interna de nombres con el host name “postgres” y el puerto 5432)
  - Port: El puerto por defecto de PostgreSQL es 5432 (en el caso de haber usado la opción de instalación con docker ver cómo es la resolución de nombres respecto del puerto, interno al contenedor el puerto será el 5432).
  - Maintenance database: Ingresá el nombre de la base de datos que vas a usar (generalmente se usa "postgres" para la base de datos de administración)
  - Username: Ingresá el nombre de usuario de PostgreSQL (por defecto, se puede usar "postgres")    Password: Ingresá la contraseña del usuario de PostgreSQL (o del usuario seleccionado)
4. Conectar al servidor: en el panel izquierdo vas a poder ver el servidor que creaste, al hacer clic en él vas a poder conectarte al servidor.