Contenido

Introducción

Para ejecutar el proyecto por primera vez, el usuario debe completar dos etapas que constan de un total de nueve pasos.

La primera etapa consiste en la construcción y despliegue del `Docker Compose` del proyecto, lo que permitirá acceder a la plataforma o a la vista inicial de la página.

La segunda etapa corresponde a la ejecución del `Dev Container`, que le permitirá al usuario editar el código según sea necesario. Todo esto se ejecutará en un entorno aislado, sin requerir configuraciones adicionales en el dispositivo local.

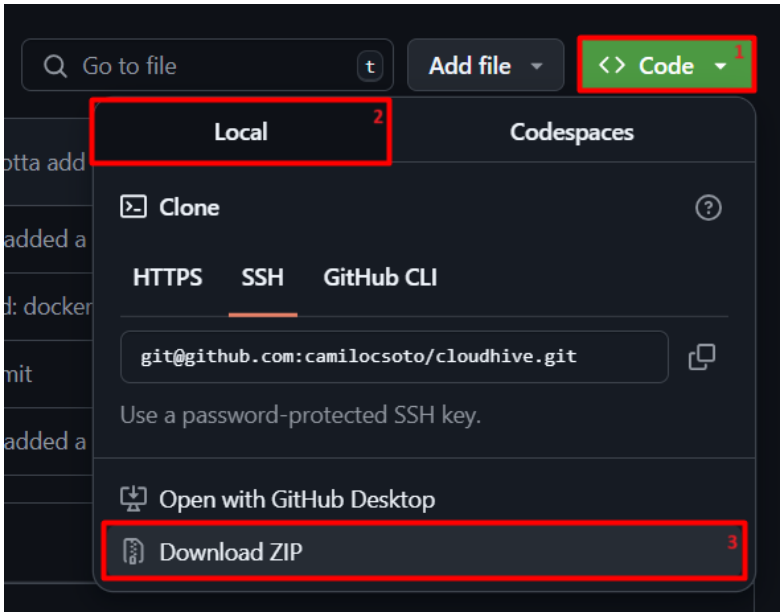
Recomendaciones

Antes de comenzar, es fundamental contar con dos herramientas instaladas en el entorno local: **Docker Desktop**, **Visual Studio Code** y una terminal de comandos **Shell** u otra. Son fáciles de instalar y esenciales para el correcto funcionamiento del proyecto.

1 etapa: construcción y levantamiento del proyecto

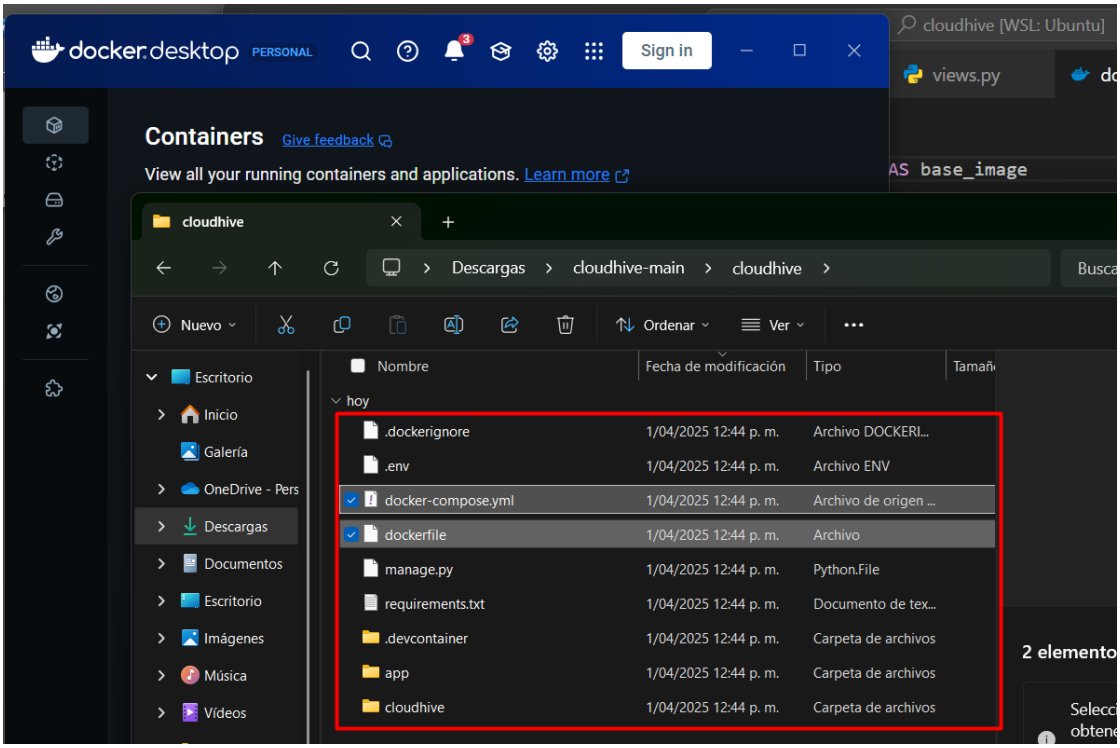
Durante esta primera etapa, usted entenderá el proceso para desplegar el proyecto en su entorno local (*localhost:8181)*

Descargar el código: Para obtener el código, diríjase al repositorio de **GitHub**, donde se encuentra alojado el proyecto, y descargue el archivo `.zip` en su PC, siguiendo los pasos que se muestran a continuación.



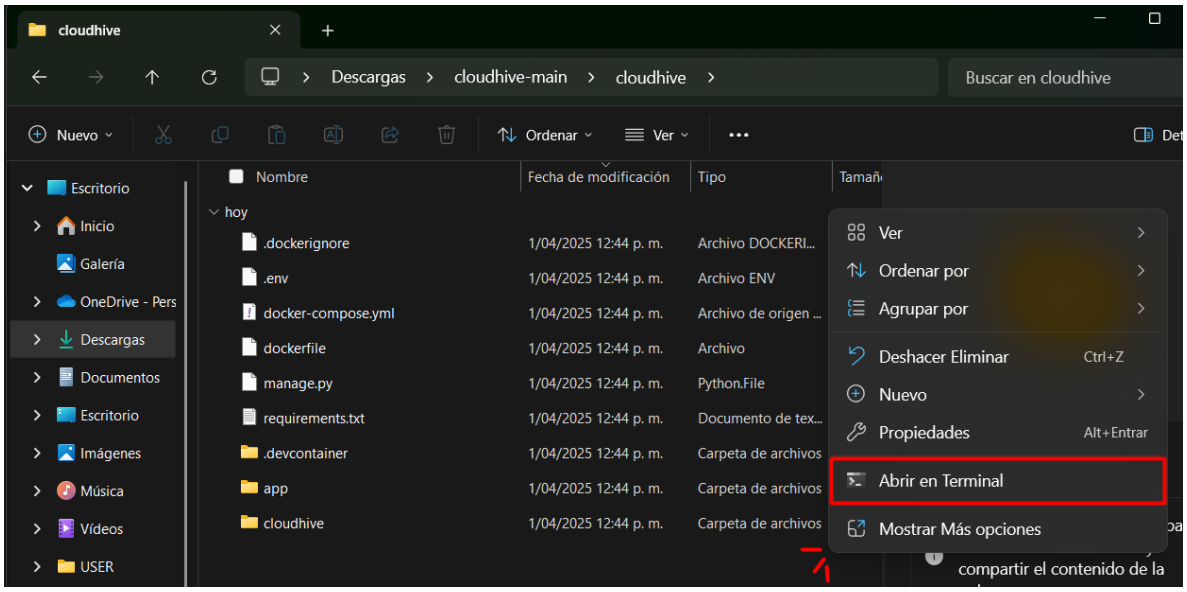
Acceder a la carpeta raíz: Descomprima el archivo `.zip` descargado y acceda a la carpeta resultante. Dentro de ella, debería encontrar archivos como `docker-compose.yml` o `Dockerfile`. Si estos archivos están visibles, significa que está en la ubicación correcta.

Nota: Asegúrese que `Docker Desktop` esté en ejecución en segundo plano; simplemente debe mantener la aplicación abierta.

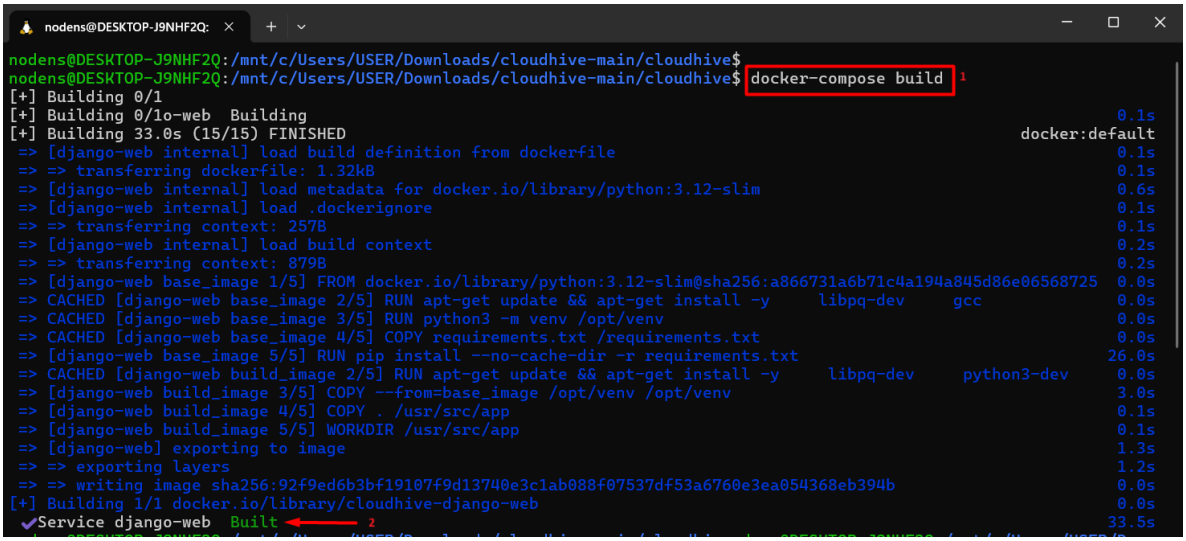


Ejecutar archivos docker en la carpeta raíz: En el directorio raíz, haga clic derecho y seleccione la opción `Abrir en terminal`.

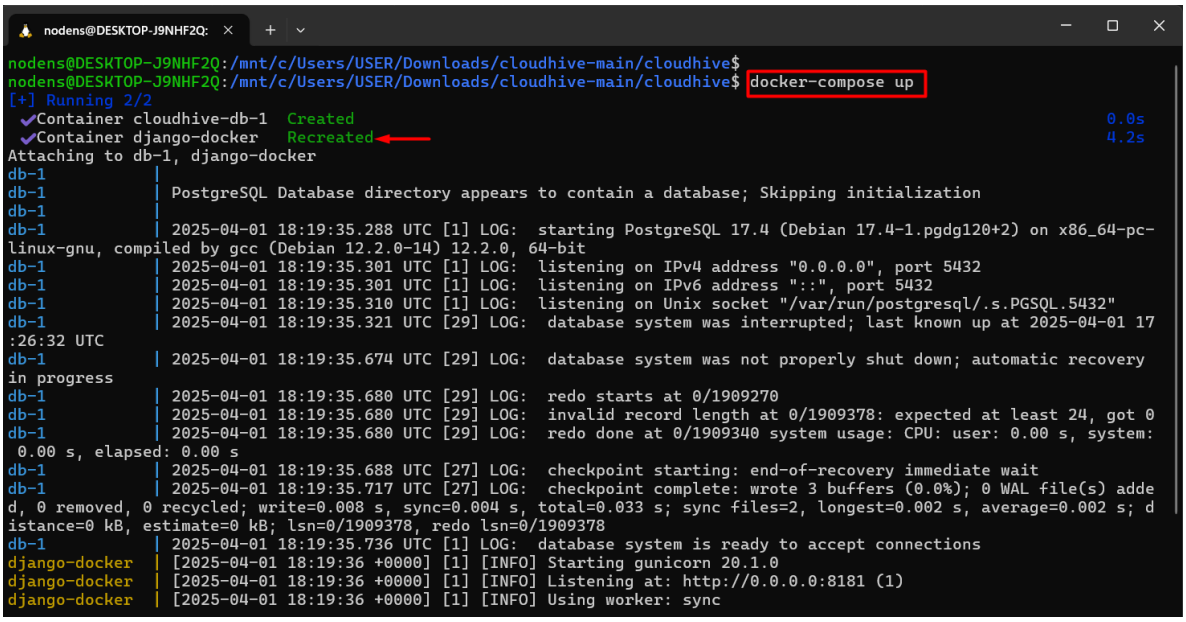
Nota: Si esta opción no está disponible, copie la ruta desde el explorador de archivos y abra la terminal de su preferencia, luego navegue hasta la ruta copiada.



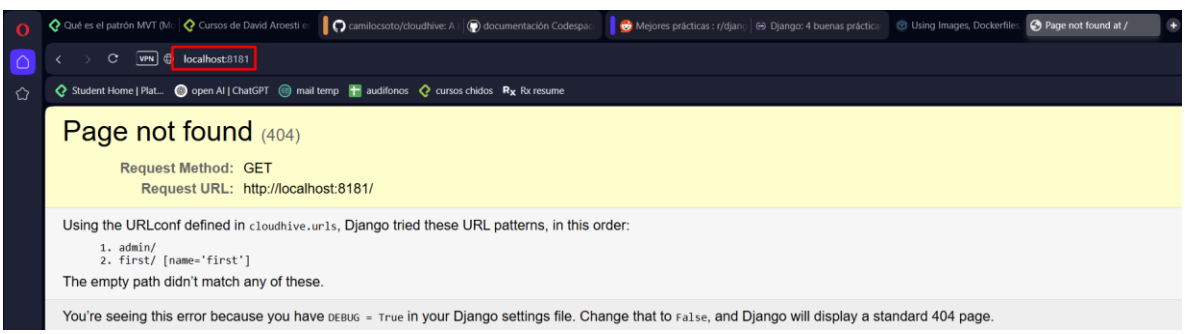
Construcción de la imagen: Para construir la imagen de docker que contiene Django y PostgreSQL, ejecute el siguiente comando en la terminal: `docker-compose build`, este proceso puede tardar entre 1 y 5 minutos.



Levantamiento del contenedor: Para iniciar el contenedor de Docker que gestionará los servicios de Django y PostgreSQL, ejecute el siguiente comando en la terminal: `docker-compose up`. Este proceso puede tardar entre 1 y 3 minutos.



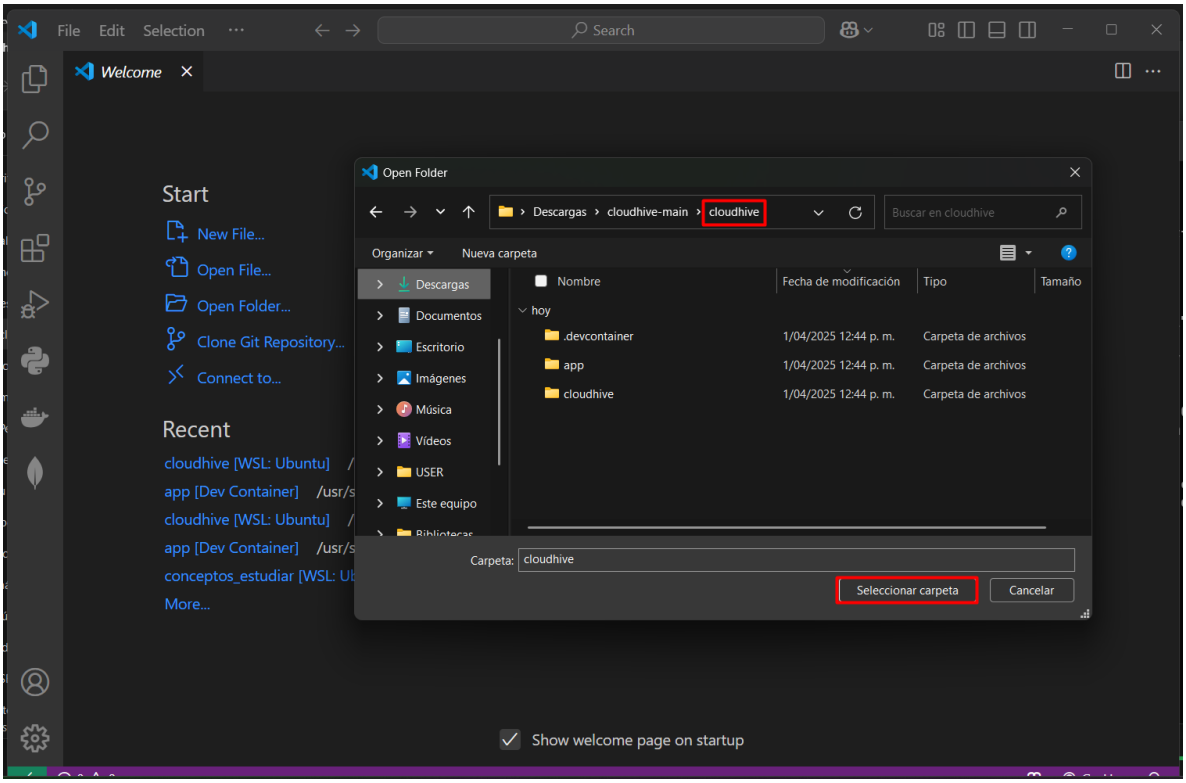
Si en un navegador accede a `localhost:8181` usted podrá ver la interfaz de django:



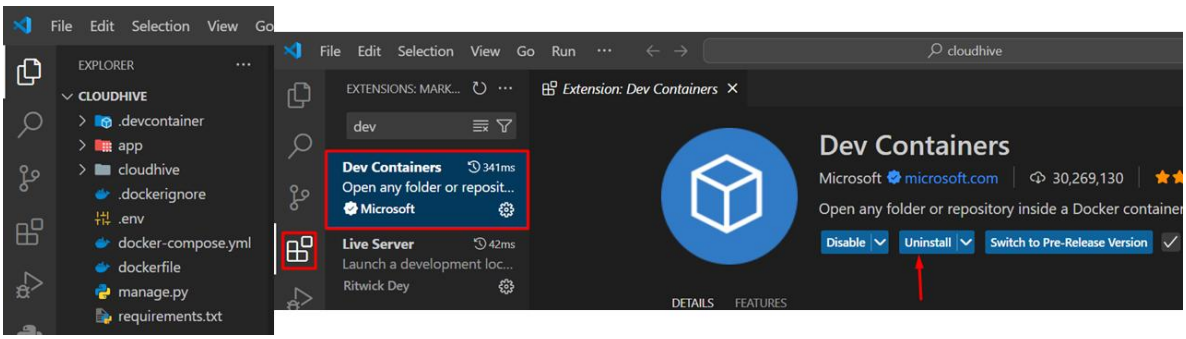
2 etapa: Edición de código en entorno aislado

En esta segunda etapa, aprenderá a modificar el código del proyecto en un entorno aislado, sin necesidad de instalar SDKs, librerías u otras dependencias, ya que todo está previamente configurado.

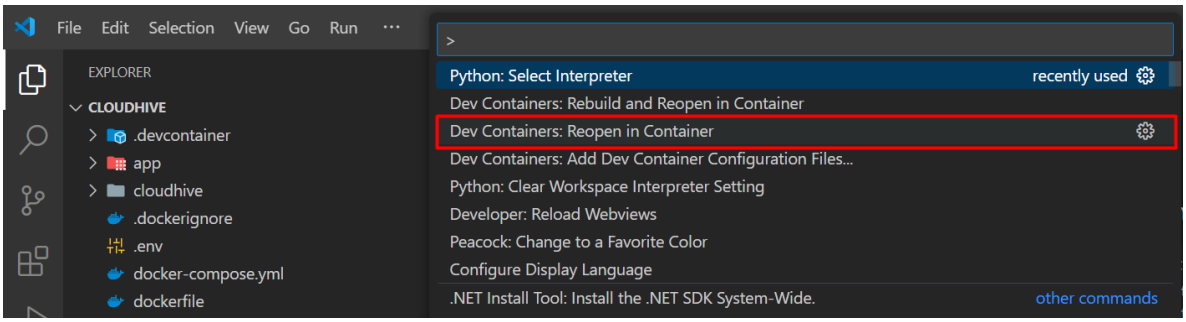
Abrir la carpeta raíz en vsc: Para ello, debe abrir su visual studio code y en la zona superior izquierda seleccionar File > Open Folder. Debe ubicar la carpeta raíz que contiene los directorios cloudhive, app y .devcontainer como se observa a continuación



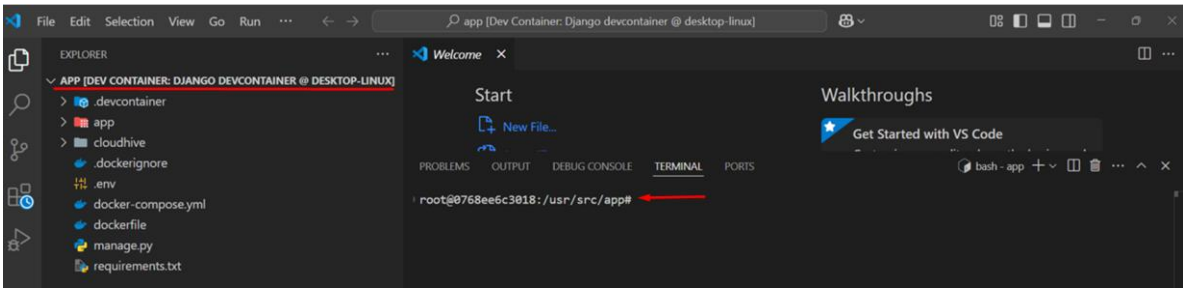
Extensión Dev Container: Una vez abierto el directorio, podrá visualizar una serie de archivos en la parte izquierda de la pantalla. A continuación, diríjase al menú izquierdo, seleccione la opción `Extensiones` y busque `Dev Containers`. Asegúrese de que esta extensión esté instalada ya que es el único requisito adicional.



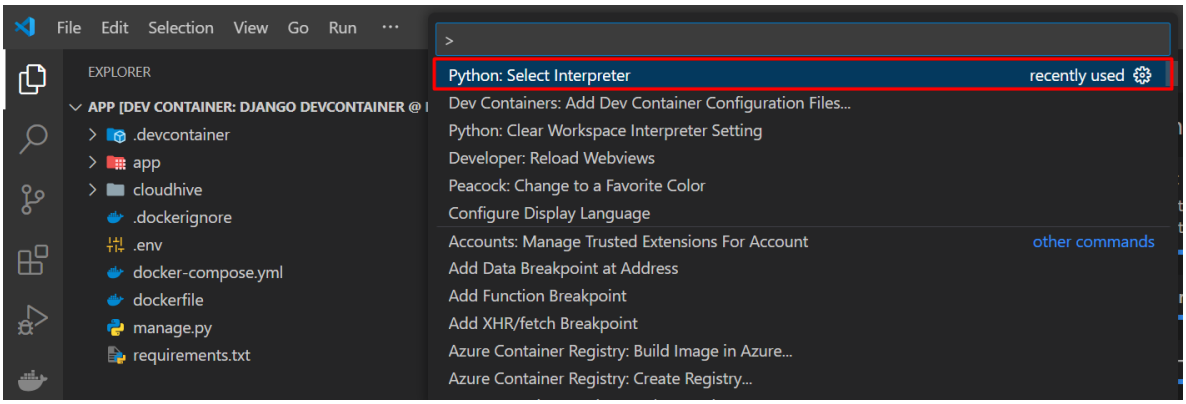
Abrir proyecto en un Dev Container: Para iniciar el entorno aislado debe presionar las teclas: `Ctrl` + `Shift` + `P`, para abrir la barra de comandos. Luego, busque y seleccione `Dev Containers: Reopen in container`. Si esta opción no aparece, puede elegir `Dev Containers: Rebuild and reopen in container` aunque este proceso puede ser más lento, ya que reconstruye la imagen de Docker y la conexión con el contenedor puede tardar un poco más en establecerse.



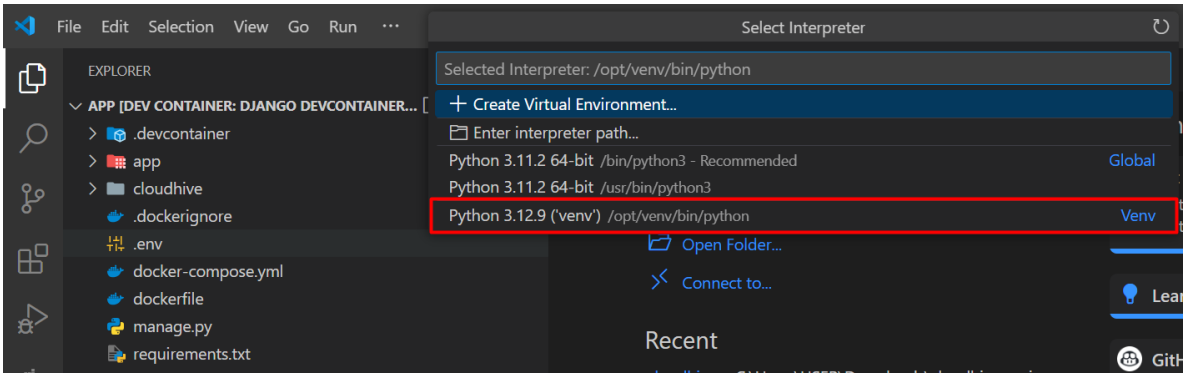
Sabrá que la conexión con el **Dev Container** se ha completado cuando el nombre del directorio raíz cambie y la terminal muestre la siguiente ruta:



Selección del intérprete de Python: Seleccione el entorno virtual donde se encuentran todas las librerías del proyecto. Para ello, presione las teclas: `Ctrl` + `Shift` + `P`, para abrir la barra de comandos, luego busque y seleccione la opción `Python: Select interpreter`



A continuación, aparecerá una ventana que indicará el entorno de Python a utilizar en el proyecto. Seleccione `Python 3.12.9 ('venv') /opt/venv/bin/python`, el entorno virtual que opera dentro del contenedor:



Al haber llegado hasta este punto, usted podrá editar lo que considere necesario en el proyecto de django + postgreSQL.

Ficha del documento

Fecha	Revisión	Autor	Revisado por
Marzo 2025	1	Juan Camilo Suárez Soto	Edward Chaparro Andrés Gil Giovanni Pinto