Pestaña 1

Guía para instalar los dispositivos y su respectivo entorno

Indice

[**GymGPT- API — LOCAL 1**](#_nm2yji84rspj)

[Instalación de proyecto 1](#_mkeyudjk1mij)

[Creación de Ambiente Virtual 1](#_ijwyrile9dl4)

[Instalación de dependencias 2](#_484erpi25bjf)

[Configuración de variables de entorno 2](#_225e119lgpoj)

[Verificación de datos 3](#_oahmk9f0w9uf)

[Ejecución del proyecto 3](#_6fqfsouxlx1e)

[**GymGPT- API — PythonAnywhere 4**](#_6mglp25awka1)

[Creación de cuenta 4](#_qxkus9d335lr)

[Inserción de repositorio 5](#_u20gibmxmujp)

[Creación de aplicación 6](#_g4svz275i5m0)

[Creación de entorno 7](#_zht3lb9k4k25)

[Configuración de variables de entorno 7](#_sbpmruhfhtb7)

[Configuración de WSGI 8](#_jzyvi4t2vdon)

[Ejecución del sistema 9](#_gnm06bda4m2p)

[**GymGPT- FRONT — LOCAL 9**](#_yq9zc4ej9p5l)

# GymGPT- API — LOCAL

## Instalación de proyecto

1. **Descargar el archivo ZIP:**Descarga el archivo ZIP que contiene el código del proyecto.
2. **Extraer los archivos:**Extrae el contenido del archivo ZIP en una carpeta de tu preferencia utilizando cualquier herramienta de descompresión como WinRAR, 7-Zip o similar.  
   “Ejemplo: C:\Proyectos\GymGPT-API.”

## Creación de Ambiente Virtual

1. **Abrir la terminal:**Navega a la carpeta donde extrajiste el proyecto.bash  
   cd C:\Proyectos\GymGPT-API
2. **Crear el entorno virtual:**Ejecuta el siguiente comando para crear un ambiente virtual:bash  
   Copiar código  
   python -m venv env

Esto creará una carpeta llamada env que contendrá el ambiente virtual.

1. **Activar el ambiente virtual:**

* **En Windows:** bash  
   .\env\Scripts\activate
* **En macOS/Linux:** bash  
   source env/bin/activate

## Instalación de dependencias

1. **Actualizar pip:**Asegúrate de tener la última versión de pip instalada:  
   pip install --upgrade pip
2. **Instalar las dependencias del proyecto:**Las dependencias están definidas en el archivo requirements.txt. Instálalas con el siguiente comando:  
   pip install -r GymGPT/requirements.txt

## Configuración de variables de entorno

Actualmente el sistema esta configurado para funcionar con las variables de entorno segun lo necesites

1. Modifica el manage.py: En este caso, nosotros modificamos el archivo Manage.py que se encuentra en el directorio de

/GymGPT/API/manage.py

La razón por la que modificamos este archivo es que no es un archivo en constante movimiento, existen otros que puedes modificar, tal como el WSGI ya que son privados y permiten modificar tus propias variables

*from* dotenv *import* load\_dotenv

*# Carga el archivo .env*

load\_dotenv()

*# Define aquí tus variables de entorno*

os.environ['SECRET\_KEY'] = 'SECRET\_KEY'

os.environ['OPENAI\_API\_KEY'] = 'OPENAI\_KEY'

os.environ['PORT'] = 'YOURPORT'

os.environ['HOST'] = 'localhost'

os.environ['DB\_NAME'] = 'DATABASENAME'

os.environ['DB\_USER'] = 'USER'

os.environ['DB\_PASSWORD'] = 'PASSWORD'

os.environ['DEBUG'] = 'True'

*# SMTP SERVICE:*

os.environ['EMAIL\_HOST\_USER'] = 'EMAIL\_HOST\_USER'

os.environ['EMAIL\_HOST\_PASSWORD'] = 'EMAIL\_HOST\_PASSWORD'

## Verificación de datos

1. Verifica que Settings.py lleve los datos que definiste en manage.py
2. Verifica que las respuestas funcionen correctamente con POSTMAN
3. Asegura haber migrado los archivos con el migrate, ejemplo:



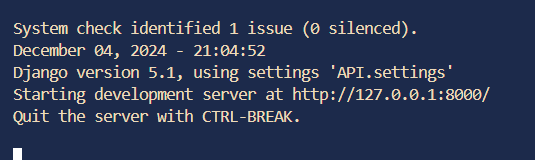
1. Asegura que tienes la versión correcta de Python, en este proyecto se usó 3.10.2

## Ejecución del proyecto

1. Para ejecutar el proyecto necesitas tener todos los pasos anteriores correctamente, para ejecutarlo necesitas ejecutar un comando en tu consola de desarrollo



El sistema ejecutará todos los sistemas necesarios y generara la aplicación en el puerto 8080



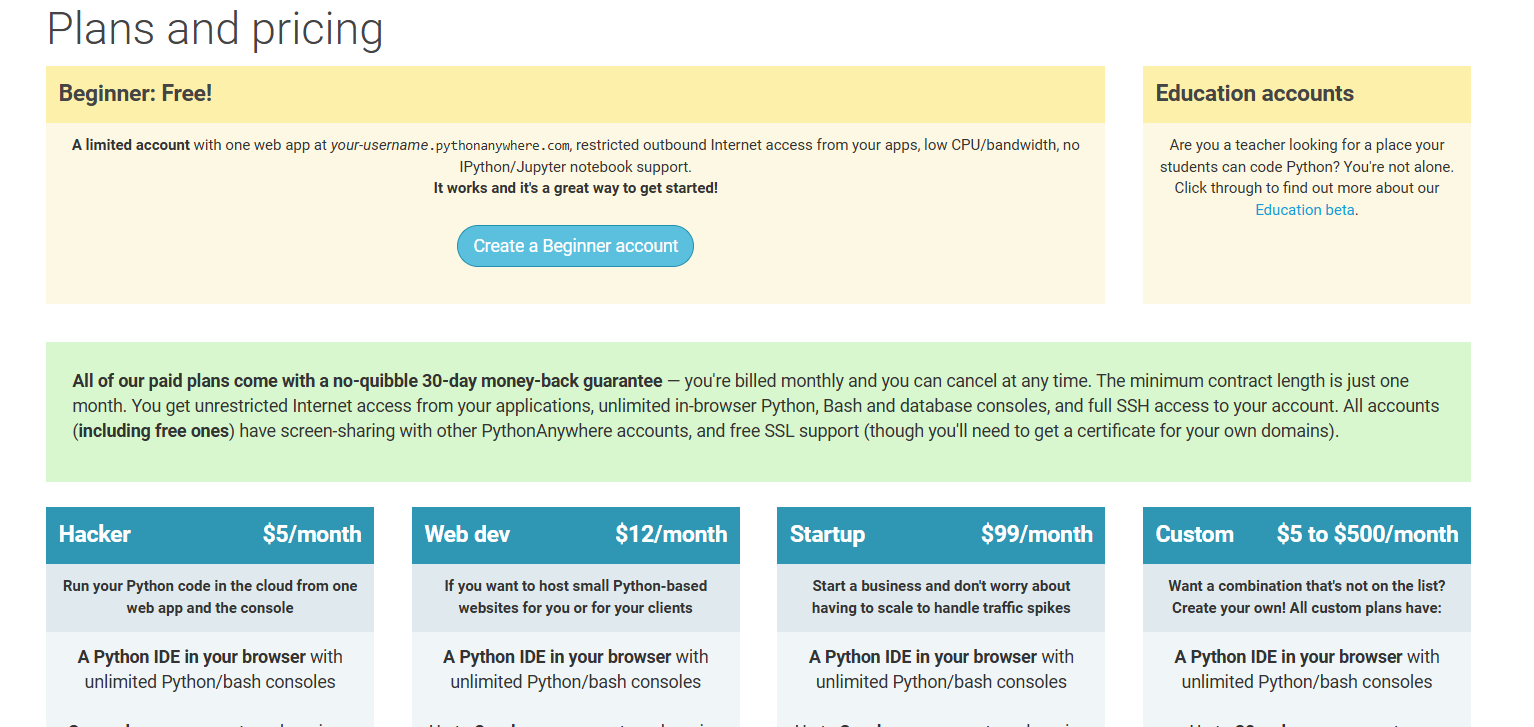
## 

# 

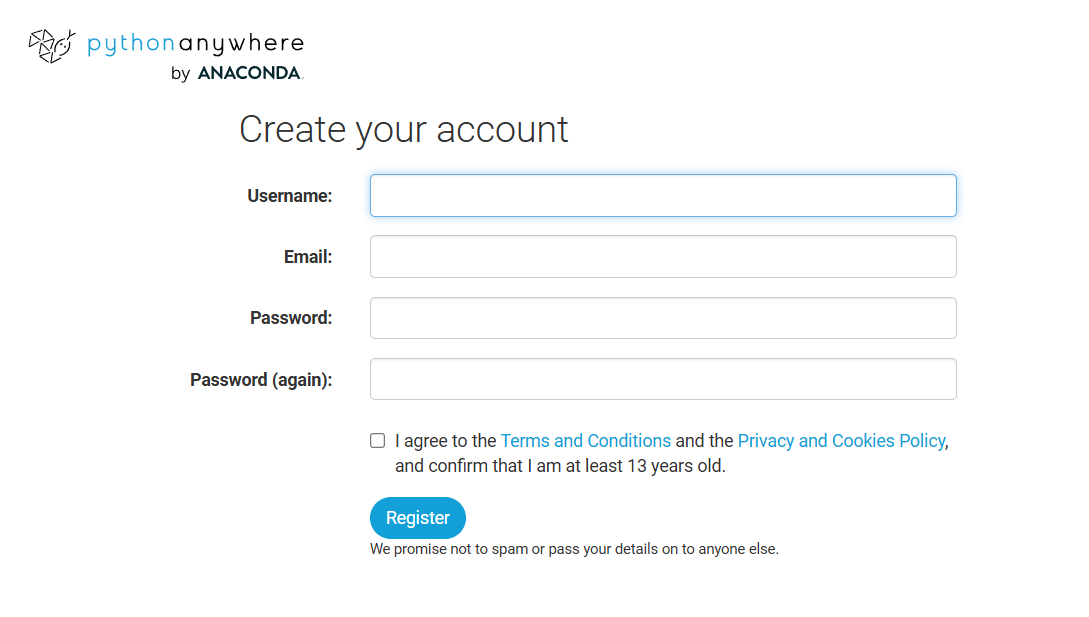
# GymGPT- API — PythonAnywhere

## Creación de cuenta

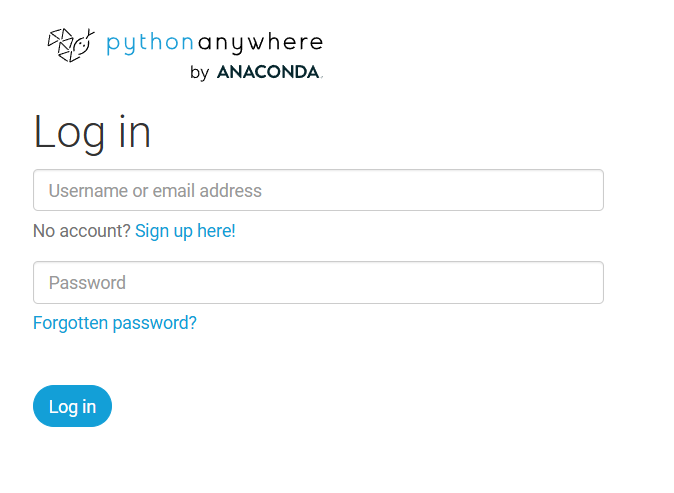
Para iniciar el uso de PythonAnywhere necesitas tener una cuenta creada

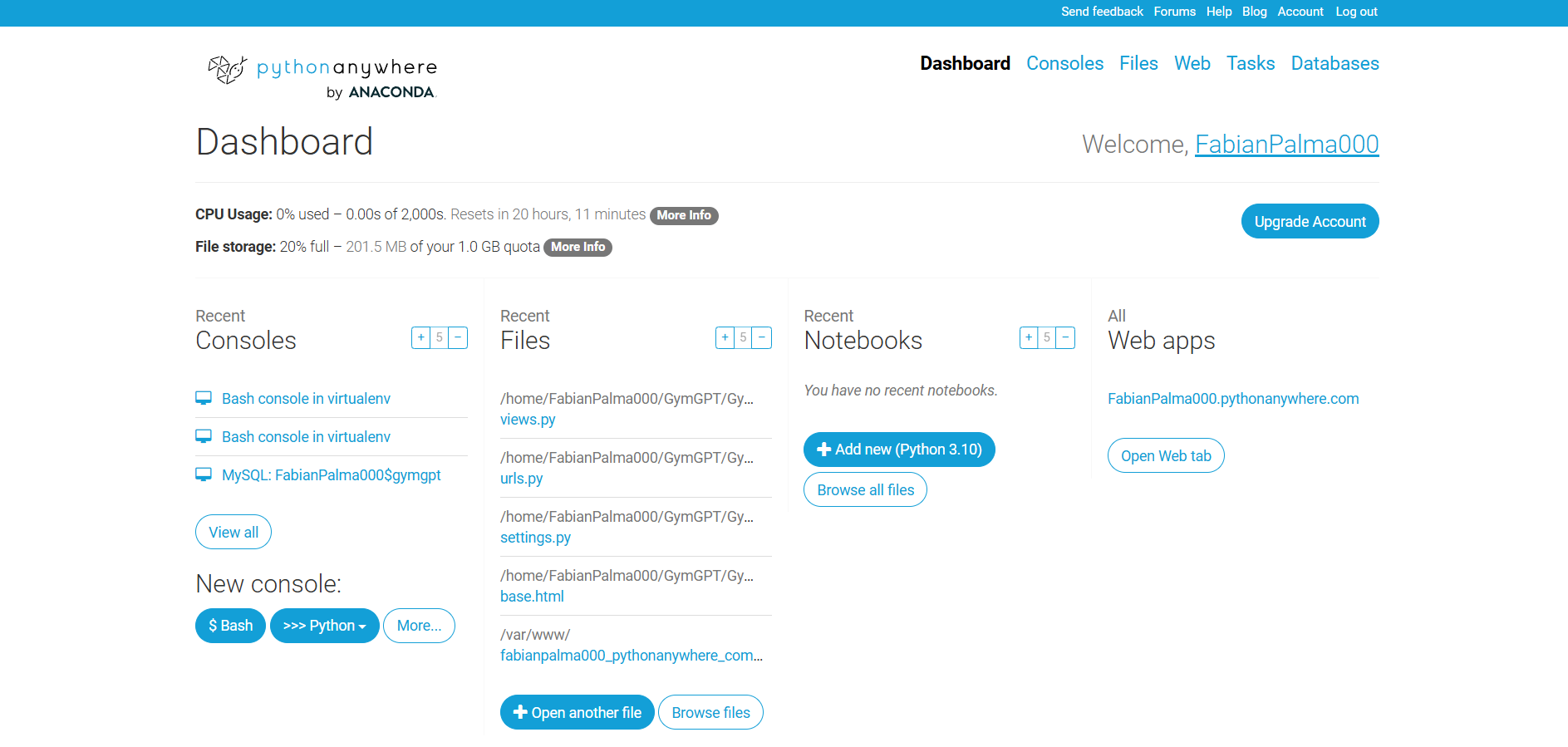


Seleccionamos la cuenta gratuita si no necesitas mayores habilidades en el desarrollo de la Aplicación.



Luego iniciar sesión en la cuenta correspondiente

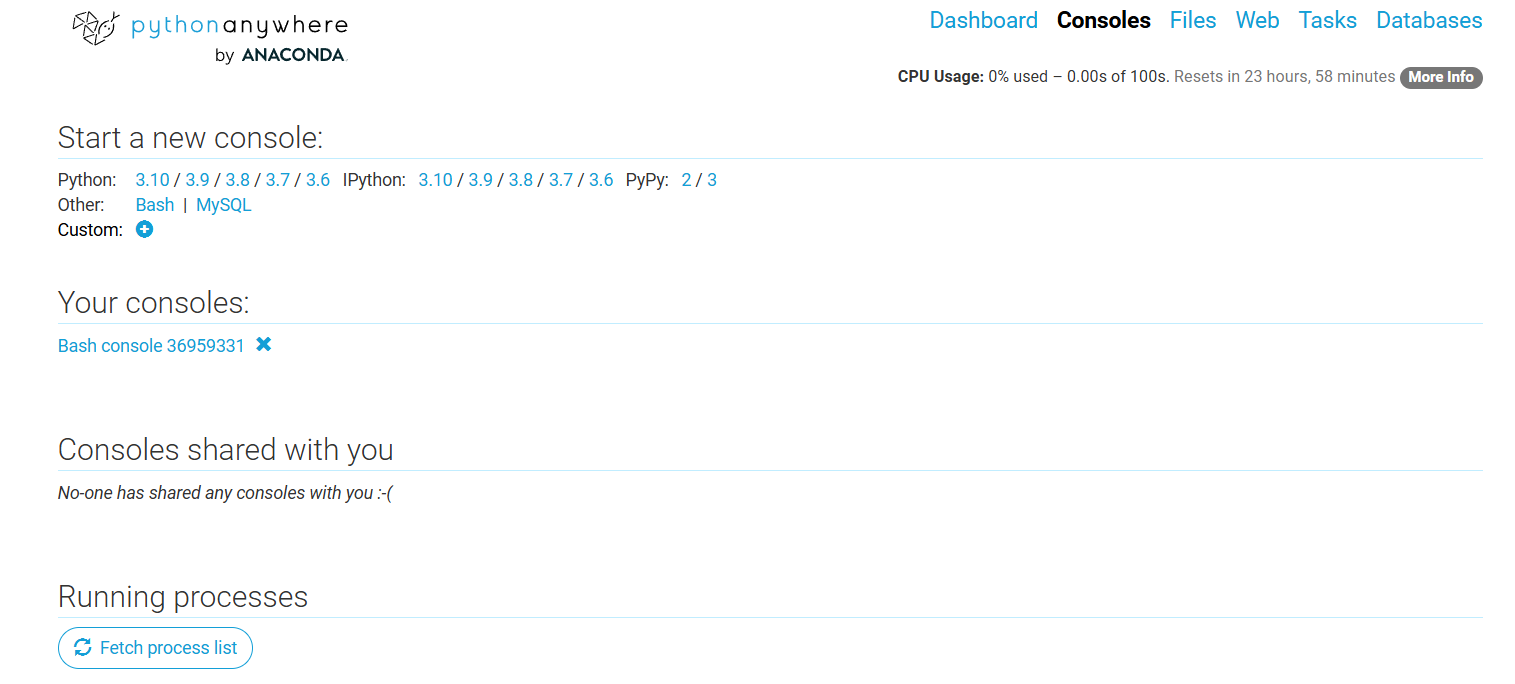
Tras ingresar a tu cuenta te redirige al Dashboard de tu cuenta;



## Inserción de repositorio

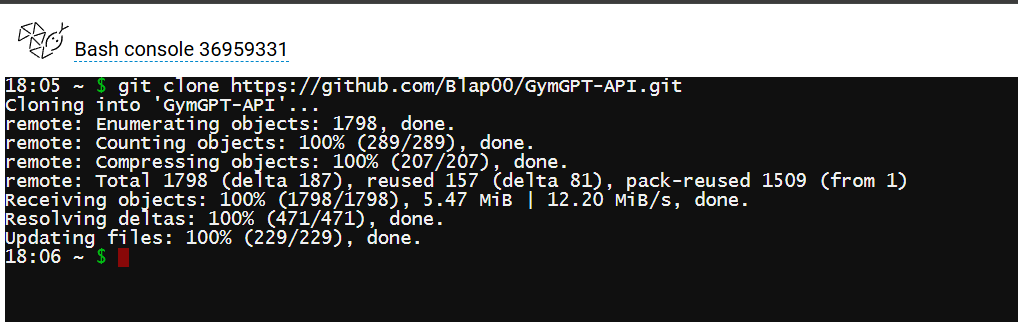
Para insertar tu repositorio sobre el sistema de PythonAnywhere necesitaras seguir los pasos de;

1. Acceder a la pestaña de “**Consoles**”

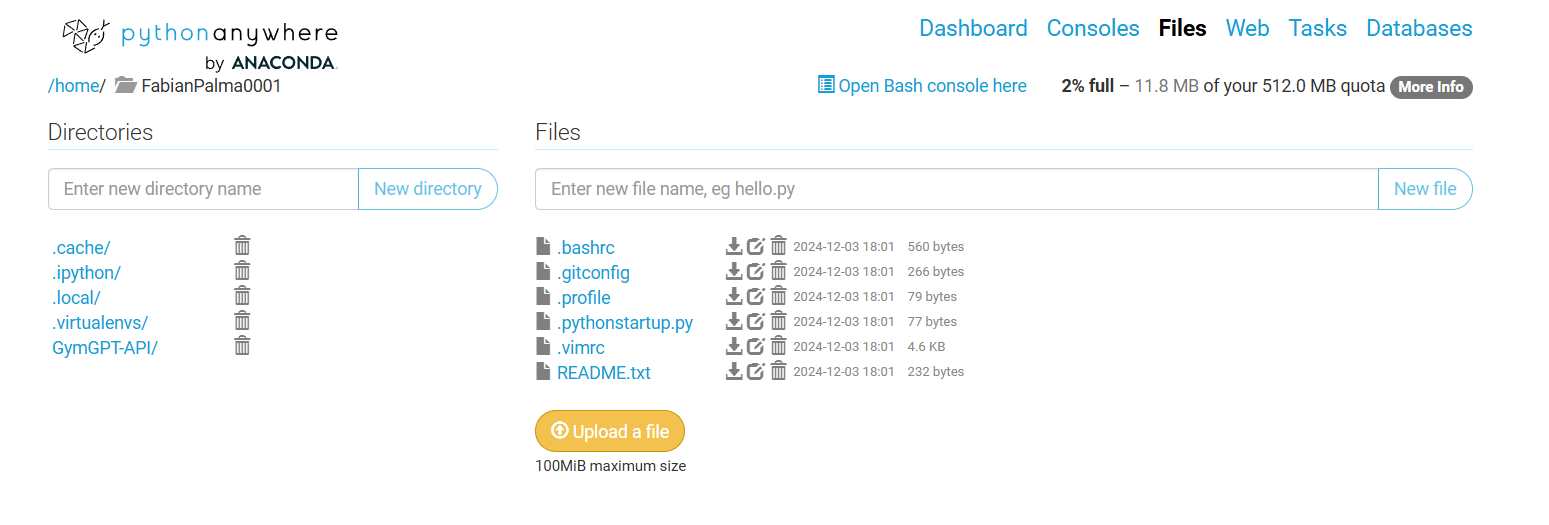


1. Ejecutar una nueva consola BASH
2. Ejecutar el comando de clonación de tu repositorio de Github

“Git Clone <URL HTTP DE GITHUB>”

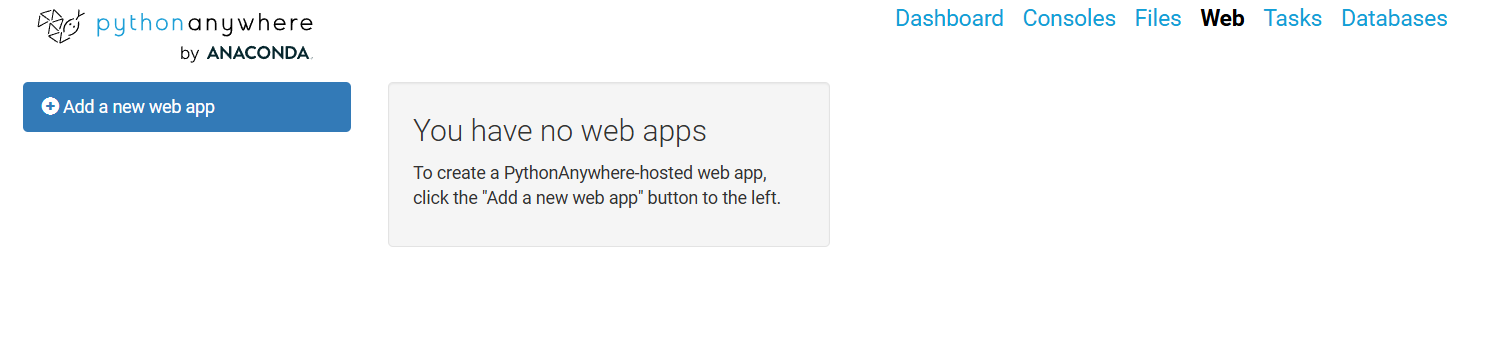


1. Verificar que los archivos se te hayan generado correctamente en la ventana de **“Files”**

****

## Creación de aplicación

Luego de verificar los datos, vamos a iniciar la nueva aplicación en PythonAnywhere

1. Nos dirigimos a la pestaña con nombre de **web**, la cual nos menciona
2. Configuramos todo de la web aplicación con una configuración manual y seleccionamos la versión de Python 3.10
3. Listo tenemos la web App generada en PythonAnywhere

## Creación de entorno

Dentro de la consola en la que importamos los datos vamos a generar nuestro entorno de desarrollo, esto con el nombre que definamos, con el código “mkvirtualenv –python=python3.10 <Nombre de nuestra aplicación>”

Luego iniciamos la instalación de dependencias establecidas en nuestro archivos requirements.txt con pip install -r GymGPT/requirements.txt

Luego de la instalación, configuramos en nuestra pestaña de **WEB**  el directorio de nuestro entorno, estos suelen estar en /home/<Username>/.virtualenvs/<Nombre de nuestra aplicación>

## Configuración de variables de entorno

1. Para configurar nuestras variables de entorno necesitaremos ingresar a la consola de desarrollo
2. Ingresamos el comando para editar en ‘nano’ con el comando nano ~/.bashrc en el cual añadiremos nuestros valores de contraseñas con el siguiente formato

export SECRET\_KEY='SECRETKEY DJANGO'

export OPENAI\_API\_KEY='KEY'

export PORT='PORT'

export HOST='HOST DATABASE'

export DB\_NAME='NAME DATABASE'

export DB\_USER='USER DATABASE'

export DB\_PASSWORD='PASS DATABASE'

export DEBUG='False'

1. Además de añadirles el formato en nuestras variables de entorno, luego tenemos que configurarlas en el WSGI, que se verá en el siguiente paso.

## Configuración de WSGI

Ingresamos a la pestaña **WEB** de nuestro PythonAnywhere en el cual descenderemos hasta encontrar el WSGI, el cual modificaremos con nuestros datos

de una manera similar a la que hicimos al configurar el manage.py de manera local pondremos aqui los datos correspondientes

*import* os

*import* sys

*# Asegúrate de que el path a tu proyecto esté en sys.path*

path = '/home/FabianPalma000/GymGPT/GymGPT-API/GymGPT/API'

*if* path not in sys.path:

sys.path.append(path)

*# Establece la variable DJANGO\_SETTINGS\_MODULE con el archivo de configuración de Django*

os.environ['DJANGO\_SETTINGS\_MODULE'] = 'API.settings'

*# Configura las variables de entorno aquí*

*# Configura el entorno Django*

os.environ.setdefault('DJANGO\_SETTINGS\_MODULE', 'GymGPT.settings')

*# Define aquí tus variables de entorno*

os.environ['SECRET\_KEY'] = ''

os.environ['OPENAI\_API\_KEY'] = ''

os.environ['PORT'] = ''

os.environ['HOST'] = ''

os.environ['DB\_NAME'] = ''

os.environ['DB\_USER'] = ''

os.environ['DB\_PASSWORD'] = ''

os.environ['DEBUG'] = 'False'

*# SMTP SERVICE:*

os.environ['EMAIL\_HOST\_USER'] = ''

os.environ['EMAIL\_HOST\_PASSWORD'] = ''

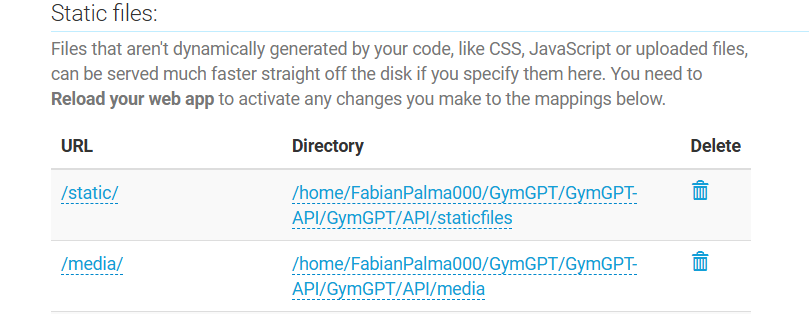
*# Cargar la aplicación WSGI*

*from* django.core.wsgi *import* get\_wsgi\_application

application = get\_wsgi\_application()

## Ejecución del sistema

1. Antes de ejecutar el sistema en PythonAnywhere necesitamos establecer los directorios de los archivos Static y Media que utilizaremos en el sitio, según lo que tengo configurado, declaramos las variables de Static según nuestro directorio de staticfiles



1. Luego de seleccionar los archivos correctamente, podemos actualizar el sitio y verificar su funcionamiento según lo teníamos en Local.

# GymGPT- FRONT — LOCAL

Ejecutar GymGPT-front en nuestro dispositivo local es más sencillo que el de la API, ya que este es ejecutado dentro de su propio entorno.

1. El primer paso es descargar el archivo .zip que contiene toda la información y acceder con la consola a la carpeta correspondiente, “cd Frontend-IONIC”
2. Después hay que instalar las dependencias con el comando de “NPM INSTALL”
3. Luego hay que ejecutar la aplicación con el comando de “Ionic serve” puesto a que las variables de entorno ya están junto al sistema de Ionic