



Estimador de ventas OPAIN

Manual de instalación y uso



Presentado por: Grupo 1

Fecha: diciembre 1, 2023

Versión: 1.0

Tabla de Contenido

Introducción

- 1.1. Propósito del manual
- 1.2. Alcance
- 1.2. Audiencia objetivo

Inicio

- 2.1. Requerimientos del sistema
- 2.2. Instrucciones de instalación

Interfaz de usuario

- 3.1. Menú principal y funcionamiento

1. Introducción

1.1. Propósito del manual

Este manual tiene como propósito proporcionar una guía detallada y clara para utilizar eficientemente el sistema desarrollado para la predicción de ventas. Aquí encontrarás instrucciones paso a paso sobre cómo ejecutar y acceder al sistema.

1.2 Alcance

El manual cubre la descarga del código fuente, su ejecución y el acceso al sistema.

1.3 Audiencia objetivo

Este manual está diseñado para ser utilizado por el personal técnico y los usuarios del sistema. La audiencia incluye, pero no se limita a, personal de aeropuertos, analistas de datos y cualquier otro usuario que interactúe con la plataforma.

2. Inicio

2.1. Requerimientos del sistema

Antes de proceder con la instalación, asegúrate de que tu sistema cumple con los siguientes requisitos:

- Sistema Operativo: Linux
- RAM: 16 GB
- Espacio en disco: 20 GB
- Python 3.10
- Angular 16.2.8
- Node 18.10.0
- Docker (opcional)
- Docker compose (opcional)

2.2. Instrucciones de instalación

Siga las siguientes instrucciones

1. Clonar el repositorio

https://github.com/GrupoUnoOpain/ENTREGA_FINAL_OPAIN

Backend

Debido a su peso el archivo "trained_colinear_models.pkl" no fue cargado al repositorio, este archivo se puede encontrar en el directorio compartido por OPAIN

Después de descargar el archivo "trained_colinear_models.pkl", este debe ser guardado en la carpeta model del proyecto

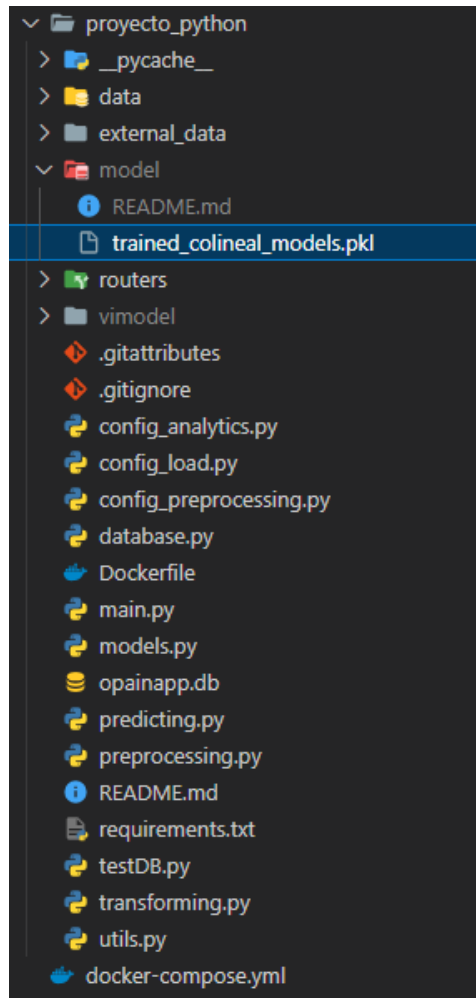


Figura 1. Proyecto Backend

Despliegue manual del backend

1. Crear el entorno virtual en la carpeta raíz del proyecto con el siguiente comando:

Python3 -m venv vimodel

2. Iniciar el entorno virtual con el siguiente comando:

source vimodel/bin/activate

3. Instalar las dependencias haciendo uso del archivo requeriments.txt que se encuentra en la raíz del proyecto, utilizando el siguiente comando:

pip3 install -r requirements.txt

4. Correr el programa utilizando el siguiente comando:

uvicorn main:app

Tan pronto el Sistema este desplegado se puede validar su funcionamiento ingresando a la siguiente url <http://localhost:8000/docs>

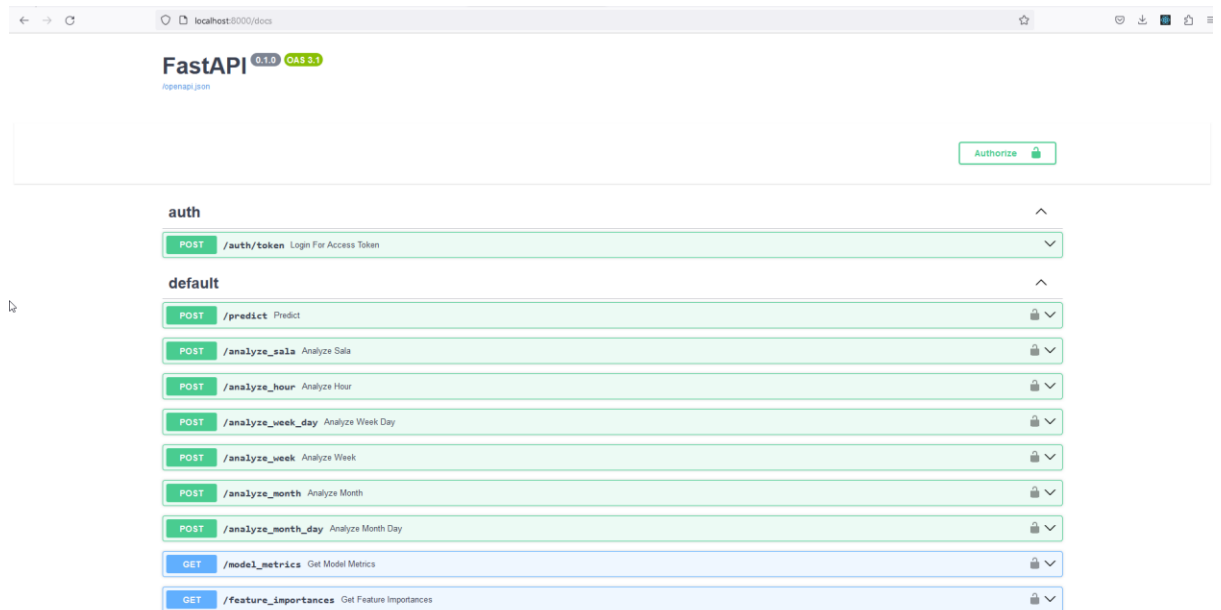


Figura 2. Documentación del API REST

Con esto finaliza el despliegue del sistema de forma manual, adicionalmente se puede desplegar el sistema haciendo uso de Docker, como se explicará a continuación.

Despliegue del backend utilizando Docker

En el proyecto se agregó el archivo Dockerfile con el cual se puede crear la imagen del proyecto backend con el comando:

docker build -t opain_backend .

Después se puede crear el contenedor con el siguiente comando:

docker run -it -p 8000:8000 opain_backend

Con esto se despliega la aplicación en el puerto 8000 que es el puerto seleccionado y configurado en el frontend.

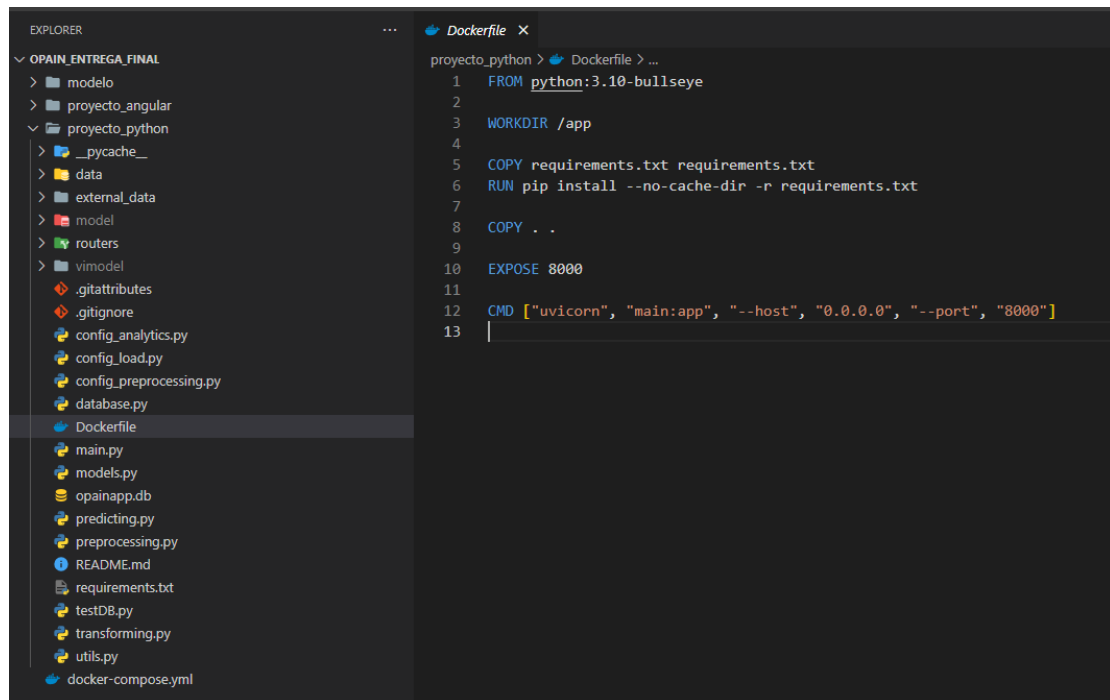


Figura 3. Dockerfile Backend

Frontend

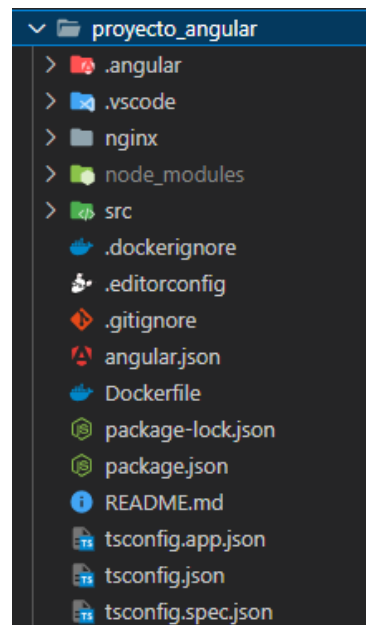


Figura 4. Proyecto Backend

Después de descargar el código fuente del frontend, se debe acceder a la carpeta raíz y ejecutar el comando:

npm install

Despliegue manual del frontend

Para ejecutar la aplicación se debe correr el comando:

ng serve --open

Tan pronto el Sistema este desplegado se puede validar su funcionamiento ingresando a la siguiente url <http://localhost:4200>

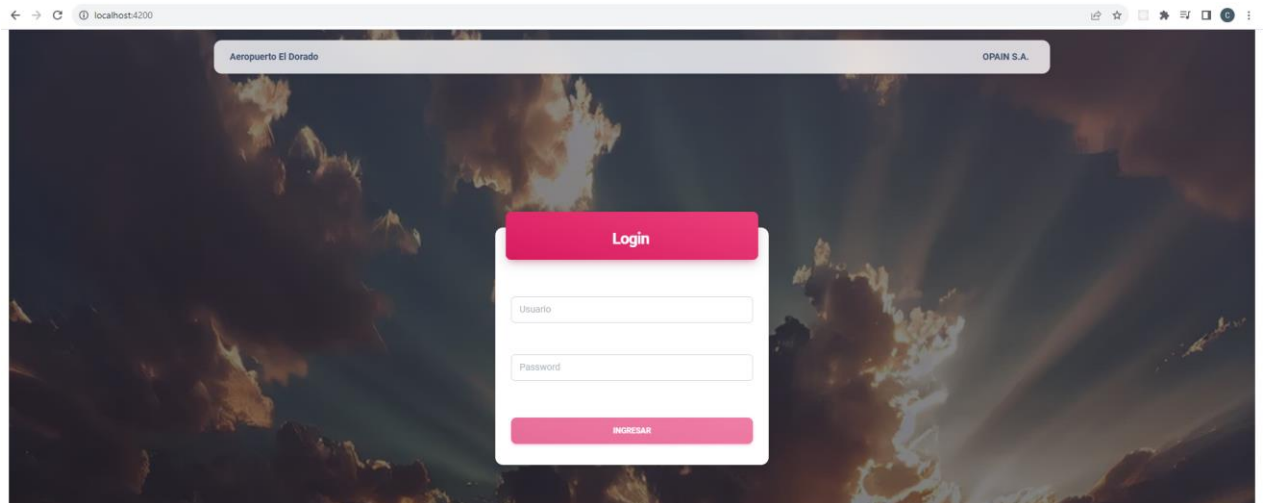


Figura 5. Interfaz Aplicación Web

Despliegue del frontend utilizando Docker

En el proyecto se agregó el archivo Dockerfile con el cual se puede crear la imagen del proyecto con el comando:

docker build -t opain_frontend .

Después se puede crear el contenedor con el siguiente comando:

docker container run -it -p 4200:4200 opain_frontend

Con esto se despliega la aplicación en el puerto 4200.

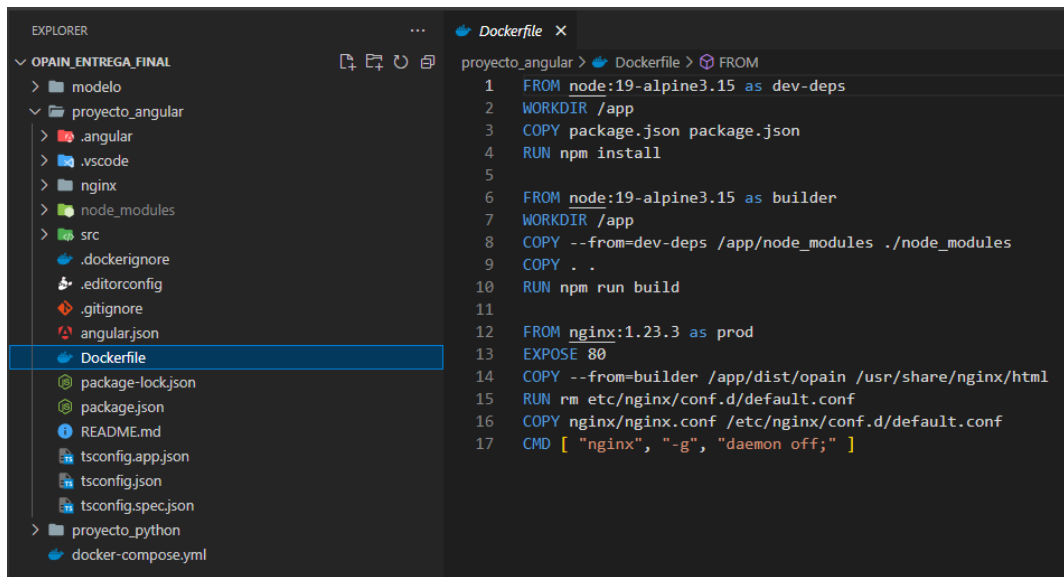


Figura 6. Dockerfile Backend

DESPLIEGUE CON DOCKER COMPOSE

Existe otra forma más sencilla de hacer el despliegue utilizando Docker compose, para esto las dos imágenes se deben subir a <https://hub.docker.com/>

Para esto se debe contar con una cuenta activa y ejecutar los siguientes comandos

Ingresar a Docker hub

Docker tag NOMBRE_IMAGEN USUARIO_DOCKER_HUB/NOMBRE_ASIGNADO

docker push NOMBRE_IMAGEN USUARIO_DOCKER_HUB/NOMBRE_ASIGNADO

Ejemplo

docker tag opain_backend USUARIO_DOCKER_HUB/opainservice

docker push USUARIO_DOCKER_HUB/opainservice

docker tag opain_frontend USUARIO_DOCKER_HUB/opainfront

docker push USUARIO_DOCKER_HUB/opainfront

Después de subir las dos imágenes se puede utilizar el archivo docker_compose.yml adjunto, haciendo la modificación del usuario y nombres de las imágenes según corresponda.

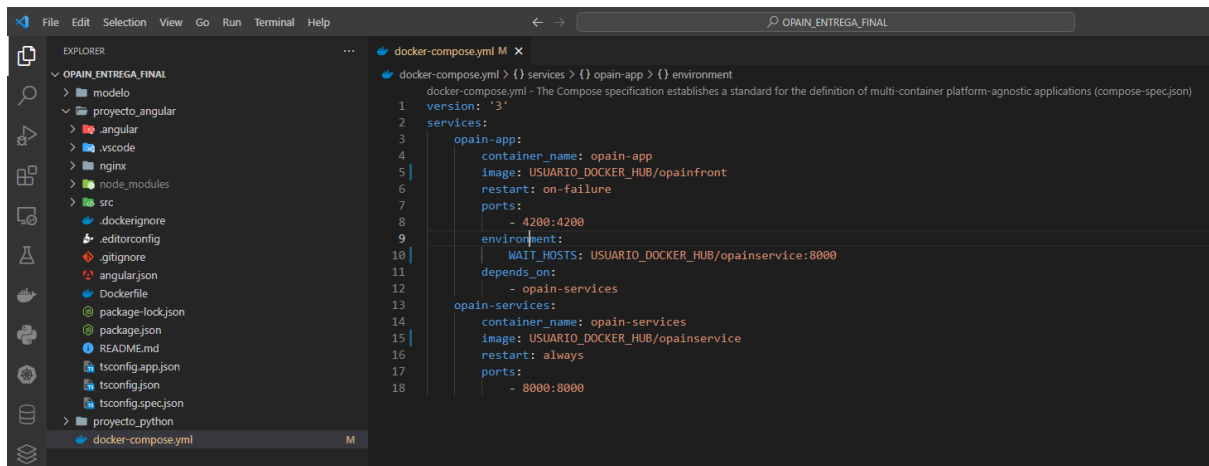


Figura 7. Archivo docker-compose.yml

Para desplegar tanto el backend como el frontend, en la carpeta donde se haya descargado el archivo “**docker-compose.yml**”, se debe ejecutar el siguiente comando que desplegara el sistema completo:

docker-compose up

3. Interfaz de usuario

3.1. Menú principal y funcionamiento

Para acceder al sistema las credenciales de acceso son:

Usuario: opain

Password: @painMM1ZQx6h5L

Se adjunta el enlace al video de YouTube el cual muestra el funcionamiento del sistema

https://www.youtube.com/watch?v=IY4jQ_enb5I&t=1s