



Manual de Instalación del Tablero

En este manual, se detallará el proceso que debe seguir el usuario que desee usar el tablero que genera las predicciones de los resultados del ICFES. Se debe tener en cuenta que el usuario, puede desplegar este tablero ya sea de manera local, o en una máquina virtual. Además, en el momento de realizar este manual el tablero ya se encuentra desplegado en una máquina virtual y puede acceder a partir de la siguiente dirección en su navegador: <http://54.89.193.67:8050/>. (la ip puede cambiar ya que estamos en los ambientes de aws academy)

El primer paso que debe hacer ya sea si lo está haciendo de manera local o en una máquina virtual es crear un ambiente virtual de Python en el cual se descargarán posteriormente todas las dependencias necesarias para usar el tablero.

- `python3 -m venv env` (Crear el ambiente virtual)
- `source env/bin/activate` (Activar el ambiente virtual)

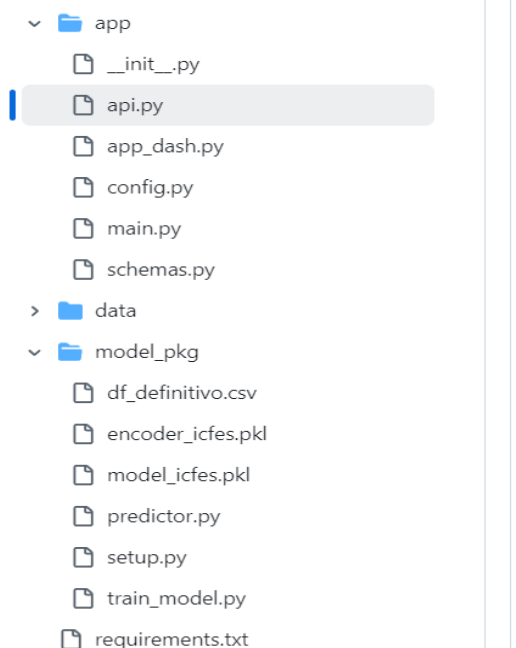
Una vez de creado el ambiente virtual y activado, se procede a crear una nueva carpeta donde usted va a clonar el repositorio que tiene el proyecto para desplegar el tablero.

```
(env-api) ubuntu@ip-172-31-27-178:~$ mkdir proyecto
```

Una vez creada, debe entrar dentro de la carpeta y clonar el repositorio, el cual es público y se encuentra en el siguiente link: <https://github.com/felipemiad/Prueba2/tree/main>. El siguiente comando le permitirá clonar el repositorio:

```
(env-api) ubuntu@ip-172-31-27-178:~/proyecto$ git clone https://github.com/felipemiad/Prueba2.git
```

Al ejecutar ese comando, se creará una nueva carpeta con el nombre del repositorio a la cual debe acceder. Debe tener en cuenta que el proyecto tiene la siguiente estructura de carpetas:



- En la carpeta `app`, se encuentra tanto los scripts para desplegar la API, así como el script para desplegar el tablero.

- La carpeta data, tiene el dataset utilizado para el entrenamiento del modelo.
- La carpeta model_pkg, es la carpeta que tiene el modelo empaquetado, y los scripts para entrenar el modelo y generar las predicciones.

Una vez entra dentro de la carpeta generada al clonar el repositorio, lo primero que debe hacer es instalar las dependencias para usar el proyecto, las cuales se encuentran dentro del archivo “requirements.txt”.

```
(env-api) ubuntu@ip-172-31-27-178:~/proyecto/Prueba2$ pip install -r "requirements.txt"
```

Posterior a la instalación de las dependencias, es importante que usted defina el directorio raíz del proyecto para evitar posibles errores al ejecutar alguno de los scripts ubicados en cualquiera de las carpetas.

```
(env-api) ubuntu@ip-172-31-27-178:~/proyecto/Prueba2$ export PYTHONPATH=/home/ubuntu/proyecto/Prueba2:$PYTHONPATH
```

Despliegue de la API.

Lo primero que se debe hacer antes de desplegar el tablero, es desplegar la API la cual es la encargada de realizar las predicciones a partir de los datos de entrada del tablero. Para desplegar la API, lo primero que debe hacer usted es generar los archivos de extensión .pkl que se encuentran en la carpeta model_pkg. Para generar estos archivos, usted debe ejecutar el script llamado “train_model.py” (En caso de no ejecutar este paso, muy posiblemente al desplegar la API le saldrá un error)

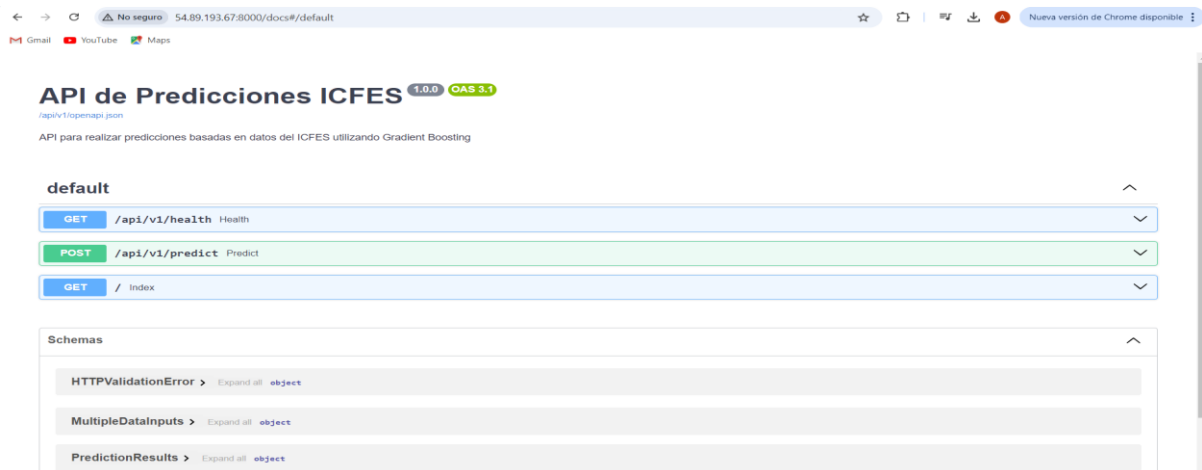
```
(env-api) ubuntu@ip-172-31-27-178:~/proyecto/Prueba2$ python3 model_pkg/train_model.py
```

Al ejecutar este Script usted esta empaquetando nuevamente el modelo, y el encoder, por tanto, ya puede proceder al siguiente paso el cual se encarga de desplegar la API.

Para desplegar la API, debe ahora ejecutar el script llamado “main.py”. Al ejecutar este script usted tendrá como salida la confirmación del despliegue de la API y el servidor en el que está desplegada.

```
(env-api) ubuntu@ip-172-31-27-178:~/proyecto/Prueba2$ python3 app/main.py
2024-11-25 05:33:23.822 | INFO | app.config:setup_app_logging:32 - Logging configurado correctamente.
2024-11-25 05:33:23.824 | WARNING | _main__<module>:59 - Ejecutando en modo desarrollo. No usar esta configuración en producción.
INFO: Started server process [1525]
INFO: Waiting for application startup.
INFO: Application startup complete.
INFO: Uvicorn running on http://0.0.0.0:8000 (Press CTRL+C to quit)
```

Para comprobar su despliegue correcto, acceda a la dirección indicada. En caso de que este utilizando una máquina virtual ponga la IP Pública de su máquina seguido de los dos puntos y el puerto 8000 (IP:8000, tenga en cuenta que debe tener habilitado el puerto para poder acceder a dicha dirección en su navegador). Al ingresar a su navegador debe mostrarle la API desplegada de la siguiente manera:



Debe dejar el servidor activo, por tanto, no cierre su terminal ni detenga la ejecución del servidor.

Despliegue del Tablero

Una vez desplegada la API, usted debe desplegar el tablero. Para ello, debe abrir una terminal nueva ya sea desde el local o desde su máquina virtual. Debe ingresar nuevamente a la carpeta raíz del proyecto, y, por último, usted debe ejecutar el script que se encarga de desplegar el tablero el cual se llama “app_dash.py”.

```
(env-api) ubuntu@ip-172-31-27-178:~/proyecto/Prueba2$ python3 app/app_dash.py
Dash is running on http://0.0.0.0:8050/

* Serving Flask app 'app_dash'
* Debug mode: on
```

Fíjese que el servidor se está desplegando en otro puerto diferente al de la API, en este caso es el puerto 8050. Por tanto, usted puede acceder al tablero a través de la IP de su máquina virtual indicando el puerto 8050 en el navegador de la siguiente manera: IP:8050 (Debe habilitar ese puerto en el grupo de seguridad de su máquina virtual).

Recuerde el servidor tanto de la API como del tablero deben estar activos para que usted pueda interactuar con el tablero y ver los resultados. Si sigue los procesos correctamente, se espera que al entrar al navegador con la dirección del tablero vea lo siguiente:



Para entender cómo puede interactuar con el tablero y ver los resultados de las predicciones realizadas por el modelo, puede consultar el manual de usuario del tablero.