# Comandos empleados para el tablero Dash

**ssh -i "C:\Users\braya\OneDrive - universidadean.edu.co\Escritorio\UniversidadDeLosAndes\Tercer Semestre\Despliegue de soluciones analíticas\Semana2\Actvidad Indivudual\taller\_dos.pem" ubuntu@18.208.165.40**

* Descripción: Establece una conexión SSH con un servidor remoto cuya dirección IP es 18.208.165.40.
* Propósito: Inicia una sesión SSH utilizando la clave privada especificada para autenticarse en el servidor con el usuario "ubuntu".

**ssh -i Nuevo\_Par.pem ubuntu@IP**

* Descripción: Similar al primer comando, establece una conexión SSH con un servidor remoto, pero utiliza una clave diferente (Nuevo\_Par.pem) y la dirección IP debe reemplazarse por la dirección real del servidor.
* Propósito: Inicia una sesión SSH en el servidor especificado utilizando la clave privada indicada y el usuario "ubuntu".

**sudo apt update**

* Descripción: Actualiza la lista de paquetes disponibles en los repositorios de software.
* Propósito: Asegura que el sistema tenga información actualizada sobre los paquetes disponibles para instalación.

**sudo apt install python3-pip**

* Descripción: Instala el paquete de gestión de paquetes de Python (pip) para la versión 3 de Python.
* Propósito: Permite la instalación fácil y gestionada de paquetes y módulos de Python en el sistema.

**PATH=$PATH:/home/ubuntu/.local/bin**

* Descripción: Añade el directorio /home/ubuntu/.local/bin al final de la variable de entorno PATH.
* Propósito: Permite que los ejecutables instalados en este directorio puedan ser ejecutados desde cualquier ubicación en el sistema.

**pip install tox**

* Descripción: Utiliza pip para instalar el paquete "tox".
* Propósito: tox es una herramienta que simplifica y automatiza las pruebas y la gestión del entorno virtual en proyectos Python.

**get clone repo**

* Descripción: No es un comando válido. Parece haber un error tipográfico. Debería ser **git clone repo**.
* Propósito: Clona un repositorio Git llamado "repo" en el directorio actual.

**cd tablero**

* Descripción: Cambia el directorio actual al directorio llamado "tablero".
* Propósito: Se mueve al directorio donde se encuentra el código fuente del proyecto.

**pip install -r requirements.txt**

* Descripción: Utiliza pip para instalar todas las dependencias listadas en el archivo "requirements.txt".
* Propósito: Asegura que todas las bibliotecas y versiones necesarias para el proyecto estén instaladas.

**export API\_URL="18.208.165.40"**

* Descripción: Establece la variable de entorno API\_URL con el valor "18.208.165.40".
* Propósito: Define la URL de la API para que la aplicación pueda acceder a los servicios necesarios.

**python3 dashboard\_dash.py**

* Descripción: Ejecuta el script "dashboard\_dash.py" utilizando la versión 3 de Python.
* Propósito: Inicia la aplicación de dashboard\_dash, presumiblemente un servicio web o aplicación basada en Python.

# Comandos empleados para API

**ssh -i Nuevo\_Par.pem ubuntu@IP**

* Descripción: Establece una conexión SSH con un servidor remoto utilizando la clave privada "Nuevo\_Par.pem" y el usuario "ubuntu". La dirección IP debe ser reemplazada por la dirección real del servidor.
* Propósito: Inicia una sesión SSH en el servidor especificado para realizar operaciones en un entorno remoto.

**sudo apt update**

* Descripción: Actualiza la lista de paquetes disponibles en los repositorios de software del sistema.
* Propósito: Asegura que el sistema tenga información actualizada sobre los paquetes disponibles antes de realizar nuevas instalaciones.

**sudo apt install python3-pip**

* Descripción: Instala el paquete de gestión de paquetes de Python (pip) para la versión 3 de Python.
* Propósito: Permite la instalación y gestión de paquetes de Python en el sistema.

**PATH=$PATH:/home/ubuntu/.local/bin**

* Descripción: Añade el directorio /home/ubuntu/.local/bin al final de la variable de entorno PATH.
* Propósito: Permite la ejecución de comandos y scripts instalados en este directorio desde cualquier ubicación en el sistema.

**pip install tox**

* Descripción: Utiliza pip para instalar el paquete "tox".
* Propósito: tox es una herramienta que simplifica y automatiza las pruebas y la gestión del entorno virtual en proyectos Python.

**sudo apt-get install tox**

* Descripción: Instala la herramienta tox utilizando el sistema de gestión de paquetes APT.
* Propósito: Asegura que la herramienta tox esté disponible en el sistema.

**get clone repo**

* Descripción: No es un comando válido. Debería ser **git clone repo**.
* Propósito: Clona un repositorio Git llamado "repo" en el directorio actual.

**cd modelo\_clasificacion**

* Descripción: Cambia el directorio actual al directorio llamado "modelo\_clasificacion".
* Propósito: Se mueve al directorio donde se encuentra el código fuente del modelo de clasificación.

**tox run -e test\_app**

* Descripción: Ejecuta las pruebas del entorno virtual "test\_app" utilizando tox.
* Propósito: Realiza pruebas del modelo de clasificación en el entorno virtual especificado.

**python3 -m build**

* Descripción: Utiliza el módulo "build" de Python 3 para construir el paquete del modelo.
* Propósito: Crea un paquete distribuible del modelo de clasificación.

**cd**

* Descripción: Cambia el directorio actual al directorio raíz del usuario.
* Propósito: Regresa al directorio raíz del usuario después de realizar operaciones en otros directorios.

**cp /home/ubuntu/modelo\_clasificacion/dist/model\_Predicci\_n\_Clasificaci\_n\_de\_Productos-0.0.1-py3-none-any.whl /home/ubuntu/modelo\_clasificacion\_api/model-pkg/**

* Descripción: Copia el archivo de paquete (.whl) generado del modelo a un directorio específico.
* Propósito: Mueve el paquete construido a un directorio donde será accesible para otros procesos o servicios.

**cd modelo\_clasificacion\_api**

* Descripción: Cambia el directorio actual al directorio llamado "modelo\_clasificacion\_api".
* Propósito: Se mueve al directorio donde se encuentra el código fuente de la API del modelo de clasificación.

**tox run -e test\_app**

* Descripción: Ejecuta las pruebas del entorno virtual "test\_app" utilizando tox.
* Propósito: Realiza pruebas de la API de aplicación en el entorno virtual especificado.

**tox run -e run**

* Descripción: Ejecuta la aplicación de la API de aplicación utilizando tox.
* Propósito: Inicia la aplicación de la API de aplicación después de realizar pruebas y asegurarse de que todo esté en orden.