

# Proyecto: Análisis y predicción basados en datos de precios de vivienda en el estado de Washington, Estados Unidos

Grupo 2: Lizeth García, Leonardo Guzmán, María Alejandra Herrera, Carlos Silva

## Manual de instalación

### 1. Configuración de máquina virtual en AWS EC2

- En AWS EC2 lanzar una instancia:

- Tipo: **t2.medium**
- AMI: **Ubuntu Server 24.04 LTS (HVM), SSD Volume Type**
- Almacenamiento: **20 GB SSD**

- Crear un par de llaves (.pem) para conexión SSH a la instancia.

- En el security group (sección de grupos de seguridad), agregar reglas:

- **TCP 8001 - Anywhere (API)**
- **TCP 8501 - Anywhere (Dashboard)**

### 2. Preparación de máquina virtual

- Conéctese a la máquina virtual usando SSH y el par de llaves respectivo a través de un terminal haciendo uso del comando '**ssh -i nombre\_del\_par\_de\_llaves.pem ubuntu@ip\_de\_la\_instancia**'.

En su computador:

```
chmod 400 nombre_llave.pem
ssh -i nombre_llave.pem ubuntu@IP_PUBLICA_EC2
```

- Una vez esté conectado al terminal de la máquina virtual, ejecute los siguientes comandos de inicialización y actualización:

```
sudo apt update
sudo apt install zip unzip
sudo apt install python3-pip
sudo apt install python3-venv -y
python3 -m venv myenv
```

- Active el ambiente virtual e instale la librería de automatización y pruebas 'tox'

```
source myenv/bin/activate
pip install tox
export PATH=$PATH:/home/ubuntu/.local/bin
```

### 3. Subir y extraer el proyecto empaquetado

- En tu PC (no dentro de EC2):

```
scp -i nombre_llave.pem api-dashboard.zip ubuntu@IP_PUBLICA_EC2:/home/ubuntu
```

- En EC2:

```
unzip api-dashboard.zip
```

- Esto creara:

```
house-pricing-api/  
tablero/  
requirements_ec2.txt
```

### 4. Instalar dependencias

- Realice la instalación de los módulos requeridos para la ejecución de la API:

```
pip install --upgrade pip  
pip install -r requirements_ec2.txt
```

### 5. Instalar el modelo empaquetado

- Instale el modelo empaquetado .whl

```
cd house-pricing-api/model-pkg  
pip install model_house_pricing-0.0.1.whl
```

### 6. Ejecución de API y tablero en la máquina virtual

- Lance el servicio de la API en uvicorn (desde EC2):

```
cd house-pricing-api  
uvicorn app.main:app --host 0.0.0.0 --port 8001
```

- Si aparece Unicorn running on <http://0.0.0.0:8001>, la API está funcionando. Y puede probarla en el navegador: [http://IP\\_PUBLICA\\_EC2:8001](http://IP_PUBLICA_EC2:8001)

### 7. Actualizar la IP en el dashboard

El dashboard no funcionará si no apunta a la API real.

- a. Abre **una segunda terminal local** y conecta nuevamente por SSH:

```
ssh -i nombre_llave.pem ubuntu@IP_PUBLICA_EC2
```

- b. Activa el entorno:

```
source myenv/bin/activate
cd tablero
nano dashboard.py
```

- c. Busca la línea:

```
API_URL = http://ip de la instancia:8001/api/v1/predict
```

- d. Reemplázala por:

```
API_URL = http://IP PUBLICA EC2:8001/api/v1/predict
```

- e. Guarda y Cierra nano:

```
Ctrl + O
Enter
Ctrl + X
```

## 8. Ejecutar el dashboard:

- a. Lance el tablero desde la carpeta tablero:

```
cd tablero
streamlit run dashboard.py --server.port 8501 --server.address 0.0.0.0
```

- b. Aparecerá:

```
You can now view your Streamlit app in your browser.
URL: http://0.0.0.0:8501
```

- c. Acceda al tablero desde el navegador ingresando <http://IP PUBLICA EC2:8501>

## 9. Navegación del dashboard:

- a. El tablero tiene **3 secciones navegables** desde la barra superior:

- Predecir precio:

Formulario donde el usuario ingresa:

- Ciudad
- Habitaciones
- Baños
- sqft
- Año construido
- Vista
- Waterfront

- Condition, etc.

Y el sistema muestra:

Precio estimado de la vivienda en USD

- Mapa de calor:

En este mapa se puede ver cuáles son las ciudades con los precios de vivienda promedio por pies cuadrados más y menos costosos. Puede navegar a lo largo del mapa, hacer zoom y ver la información de cada ciudad.

- Estadísticas descriptivas:

En esta sección se muestran gráficos para visualizar la relación entre el precio de la vivienda y los factores estructurales principales de las viviendas.

## 10. Para detener la API y el dashboard:

a. En las terminales donde están corriendo:

CTRL + C