

# Guia de uso

---

Este documento describe la funcionalidad y arquitectura de los archivos Python que componen el proyecto: `utils.py`, `charts.py`, `layout.py` y `main.py`.

## ***1. data\_cleaning.py - Modelado de Datos***

Este archivo contiene la clase `ModelData`, encargada de procesar archivos Excel o CSV desde rutas predefinidas y transformarlos en DataFrames limpios.

Clases y Métodos:

### **- ModelData**

#### **`_model_ONE_data(path: str)`**

Procesa un archivo individual del tipo ONE.

Elimina filas basura, crea nombres de columnas dinámicamente y transpone los datos.

#### **`_model_CNSS_data(paths: List[str])`**

Procesa múltiples archivos del tipo CNSS.

Une los datos por fecha y reestructura columnas "año" y "mes".

#### **`_model_all_data()`**

Llama a los dos métodos anteriores y retorna un diccionario con la data modelada para 'ONE' y 'CNSS'.

#### **`model_and_save_data()`**

Guarda los DataFrames modelados como archivos Excel en la carpeta `data/processed/<institucion>/.`

## **2. *charts.py* - Generación de Gráficas**

Este archivo contiene funciones para crear visualizaciones interactivas con Plotly, basadas en filtros por año y tipo de régimen.

Funciones Principales:

**cust\_amt\_per\_capitation\_type(data, filter\_, years)**

Gráfica de línea de afiliados por tipo de régimen.

**amt\_capitation\_paid\_per\_cust\_type(data, filter\_, years)**

Gráfica de barras del número de cápitas pagadas por tipo.

**amt\_capitation\_per\_gender(data, filter\_, years)**

Gráfica de barras agrupadas del monto de cápitas por género.

**capitation\_amt\_per\_cust\_type(data, years)**

Gráfica de línea del número de cápitas dispersadas por cliente.

**pct\_money\_per\_cust\_type(data, years)**

Gráfica de barras apiladas del porcentaje del monto dispersado por tipo de cliente.

Cada función valida los años, filtra los datos y ajusta el formato del gráfico con leyendas y etiquetas.

### ***3. layout.py - Interfaz de Usuario***

Define la estructura de la aplicación Dash e integra los datos y gráficas.

Funciones:

#### **load\_data()**

Carga tres archivos de Excel y extrae los años disponibles globalmente.

#### **show\_dashboard()**

Define el layout de la interfaz:

RangeSlider: Selección de años.

RadioItems: Filtro por tipo de régimen.

Gráficas: Se actualizan en base a los filtros.

#### **- Callback de Dash:**

- La función create\_charts() genera y retorna las cinco visualizaciones definidas en charts.py según los filtros.

### ***4. main.py - Punto de Entrada del Proyecto***

Este archivo orquesta la ejecución completa del sistema.

Función Principal:

#### **main()**

Crea una instancia de ModelData y llama a model\_and\_save\_data() para procesar los datos.

Llama a load\_data() para cargar los datos procesados.

Ejecuta el dashboard llamando a show\_dashboard().

Este archivo permite iniciar el flujo completo con solo ejecutar `main.py`.

### ***Flujo General del Proyecto:***

1. Se ejecuta `main.py` para modelar, guardar y cargar los datos, y lanzar el dashboard.
2. Los datos crudos se limpian y guardan con `utils.py`.
3. `layout.py` organiza la interfaz del usuario.
4. `charts.py` genera las gráficas dinámicas.