#### **Exposición – main, init**

#### **Persona A: Manejo de Datos con Pandas y Archivos – csv, data\_manipulation**

* **Lectura y escritura de archivos CSV**
* Funciones: load\_weather\_data y save\_weather\_data
* Explicación de cómo se leen y escriben datos en archivos CSV usando Pandas.
* Manejo de errores comunes como archivo vacío o no encontrado.
* **Manejo de datos faltantes**
* Función: handle\_missing\_data
* Explicación del uso de fillna para rellenar datos faltantes con el promedio.

#### **Persona B: Visualización de Datos con Matplotlib - plot**

* **Gráficos de línea**
* Función: plot\_temperature
* Explicación de cómo crear gráficos de temperatura a lo largo del tiempo.
* Personalización de gráficos: títulos, etiquetas, leyendas y cuadrículas.
* **Gráficos de dispersión**
* Función: scatter\_humidity\_temperature
* Creación de gráficos para analizar la relación entre temperatura y humedad.
* Importancia del análisis visual en la interpretación de datos.

#### **Persona A: Consumo de APIs (Open-Meteo) – api\_consumption**

* **Obtención de datos en tiempo real**
* Función: fetch\_weather\_data
* Explicación de cómo construir una URL para consumir una API.
* Manejo de respuestas HTTP y manejo de errores en el consumo de servicios.
* **Parámetros de la API**
* Uso de parámetros como clave de API, ciudad, y unidades de medida.
* Ejemplo de la estructura de datos obtenidos (JSON).

#### **Persona B: Interfaz de Usuario con Streamlit - view**

* **Diseño de la interfaz**
* Estructura de la aplicación usando Streamlit.
* Elementos interactivos como file\_uploader, text\_input y botones.
* **Visualización interactiva**
* Uso de Streamlit para mostrar datos y gráficos de manera dinámica.
* Presentación de datos JSON obtenidos de la API en la interfaz.
* **Integración de todas las funcionalidades**
* Explicación de cómo se combinan las diferentes partes del programa para crear una aplicación completa.