1. Objetivo General

El objetivo de este primer proyecto es que, a partir de la idea general de tu tesis o proyecto de investigación, definas con claridad:

1. **La pregunta de negocio o de investigación** que vas a resolver.
2. La **organización** de tu proyecto basándote en la **metodología CRISP-DM**.
3. Un **análisis exploratorio** de los datos (EDA) que ya tengas o que planees recopilar.
4. Un **plan de limpieza y manipulación** de datos (Data Wrangling) para obtener la mejor calidad posible de la información.
5. La **estructura** de tu repositorio de GitHub con buenas prácticas de organización y versionado de tu trabajo.

Este avance te permitirá sentar las bases para profundizar en fases posteriores de modelado y validación.

---

2. Propuesta y Organización CRISP-DM

La metodología **CRISP-DM** (Cross Industry Standard Process for Data Mining) consiste en las siguientes fases:

1. **Business Understanding (Entendimiento del negocio/problema)**
    - Define claramente cuál es el problema de negocio o de investigación.
    - Describe el contexto: ¿cuál es la hipótesis principal?, ¿qué necesitas comprobar o resolver?, ¿cuál es tu pregunta de investigación?
    - Explica **el valor** que proporcionará tu producto de datos MVP (¿por qué es relevante para tu institución, empresa o investigación?).
2. **Data Understanding (Entendimiento de los datos)**
    - Explica de dónde provienen tus datos (fuentes, tipo de datos, frecuencia de actualización, etc.).
    - Describe de manera general las variables y su posible relevancia o relación con el problema.
    - Identifica los posibles desafíos: datos faltantes, duplicados, inconsistencias, calidad y confiabilidad.
3. **Data Preparation (Preparación de los datos)**
    - Lista las tareas de limpieza y transformación necesarias (Data Wrangling).
    - Documenta la extracción de datos y la manipulación para llegar a la forma deseada.

4. **Modeling (Modelado)**
5. **Evaluation (Evaluación)**
6. **Deployment (Despliegue)**

El primer proyecto se centra principalmente en las **primeras tres fases**: (1) Entendimiento del problema/negocio, (2) Entendimiento de los datos (EDA) y (3) Data Preparation (Data Wrangling).

---

3. Entendimiento del Problema y de los Datos (EDA)

3.1 Formulación del Problema

- Describe en un párrafo tu **problema principal**, con foco en los objetivos y preguntas que vas a responder.
- Destaca la **relevancia** del problema en el contexto de tu disciplina o industria.

3.2 Exploratory Data Analysis (EDA)

- **Descripción de las variables**: Identifica las variables clave y explica por qué son importantes para el problema.
- **Visualizaciones iniciales**: Genera gráficas simples (histogramas, boxplots, scatter plots, etc.) que ayuden a entender la distribución y las relaciones entre variables.
- **Estadísticas descriptivas**: Muestra medidas como media, mediana, moda, rangos, desviaciones estándar para variables relevantes.
- **Identificación de outliers y valores faltantes**: Documenta la cantidad de datos nulos, atípicos y cómo podrían afectar el análisis.

En esta sección, no solo muestres resultados, sino **analiza** tus hallazgos. ¿Hay patrones claros? ¿Existen sesgos? ¿Qué insights iniciales puedes extraer?

---

4. Data Wrangling

Durante el Data Wrangling o preparación de datos, debes:

1. **Limpiar** los datos:
    - Eliminar o imputar valores faltantes (o justificar por qué se mantienen).

    - Manejar outliers o valores atípicos de manera razonable (cap, remove, transform).
2. **Transformar** los datos:
    - Ajustar tipos de variables (fechas, categóricas, numéricas).
    - Crear variables derivadas o nuevas características que puedan aportar valor al análisis.
3. **Integrar** datos de diferentes fuentes (en caso de que tengas más de un dataset).
4. **Documentar** cada paso de limpieza y transformación. Esto es importante para la **reproducibilidad** de tu trabajo y para que puedas explicar por qué tomaste ciertas decisiones.

---

5. Estructura de tu Repositorio de GitHub

A continuación, se recomienda una **organización de carpetas** típica en proyectos de ciencia de datos. La idea es que cada carpeta tenga un propósito claro. Puedes adaptar la estructura según la naturaleza de tu proyecto, pero mantén la consistencia.

```
├── data/
|   ├── raw/            <- Datos originales sin modificación.
|   ├── clean/        <- Datos de transición, tras algunas limpiezas parciales.
|   └── curated/      <- Datos limpios/listos para el modelado o análisis final.
|
├── notebooks/
|   ├── 01_EDA.ipynb    <- Notebook de análisis exploratorio de datos.
|   └── 02_DataWrangling.ipynb  <- Notebook con el proceso de limpieza.
|
├── src/
```

```
|   ├── data_prep.py    <- Scripts Python para funciones de limpieza, wrangling, etc.

|   └── utils.py        <- Funciones de utilería, constantes globales, etc.

|

├── reports/

|   ├── figures/        <- Gráficas y visualizaciones relevantes para el reporte.

|   └── eporte.pdf  <- Reporte final (o README) sobre avances y resultados.

|

├── .gitignore

├── README.md          <- Descripción general del proyecto, objetivos y uso.

└── requirements.txt    <- Lista de librerías y versiones necesarias.
```

Descripción breve de cada carpeta:

1. **data/**: Mantén separados los datos en bruto (`raw`) de los datos procesados (`curated`).
2. **notebooks/**: Crea notebooks ordenados por número o por tema para EDA, data wrangling y cualquier otro paso relevante de tu pipeline.
3. **src/**: Aquí se guardan los scripts Python que contienen funciones o módulos reutilizables.
4. **reports/**: Aquí irán tus informes, visualizaciones y resultados.
    - También puedes incluir `figures/` para almacenar y referenciar imágenes (gráficas, diagramas, etc.).
5. **README.md**: Debe ofrecer una visión general clara del proyecto (propósito, estructura, instrucciones de uso).
6. **.gitignore**: Lista de archivos y carpetas que no quieres incluir en el control de versiones (por ejemplo, datos grandes, archivos temporales, etc.).
7. **requirements.txt**: Indicando las librerías y versiones que utilizas para que otros (o tú mismo, en el futuro) puedan reproducir el entorno.

---

6. Entregables

1. **Documento con la Propuesta y Organización CRISP-DM**
    - Explica la idea general (Business Understanding).
    - Indica la disponibilidad de datos (Data Understanding).
    - Describe el plan de Data Preparation y las motivaciones de cada paso.
2. **Notebook de EDA**
    - Con visualizaciones y conclusiones iniciales.
    - Incluye estadísticas descriptivas y detección de valores atípicos/faltantes.
3. **Notebook de Data Wrangling**
    - Documenta claramente los pasos de limpieza.
    - Explica cada decisión (por qué imputaste, por qué eliminaste filas, etc.).
4. **Estructura del Repositorio de GitHub**
    - Que esté organizada según la plantilla o estructura propuesta.
    - Incluir un README.md explicando el proyecto y cómo reproducirlo.

---

7. Recomendaciones Finales

- **Versiona tu trabajo** frecuentemente en GitHub. Esto te servirá para mantener un historial de cambios y poder revertir si algo sale mal.
- Sé **organizado y claro** con los nombres de tus notebooks, scripts y datos. Es importante la claridad para otros (y para tu "yo" futuro).
- Asegúrate de **limitar la cantidad de datos sensibles** que subes a GitHub. Si tu proyecto maneja información confidencial, asegúrate de anonimizar o no exponer dicha información.
- Utiliza **notebooks** para exploración y reportes rápidos; sin embargo, evita que tu notebook crezca de forma desproporcionada (usa scripts modulares en la carpeta `src/` para funciones repetitivas).

---

8. Fecha de Entrega y Forma de Evaluación

- Se evaluará:
    1. Claridad y **profundidad** en la definición del problema y su relevancia.
    2. Correcta aplicación del **framework CRISP-DM**.
    3. Calidad de la **EDA** (visualizaciones, análisis y conclusiones).
    4. Rigor y justificación de los pasos de **Data Wrangling**.

5. Organización y claridad del **repositorio en GitHub**.