

MANUAL TÉCNICO

Elemento	Descripción
Nombre del Proyecto:	Análisis de Desempeño y Retrasos en Vuelos de EE.UU. (2011–2021)
Metodología aplicada:	CRISP-DM
Herramientas principales:	Python, SQL Server, Power BI, Docker
Equipo:	Data Administrator – Karen Morales Data Engineer – Luis Culajay BI Analyst – Jhonatan Simon Data Analyst – Brandon Portillo

Requisitos del Sistema

Infraestructura necesaria para replicar análisis

Hardware mínimo recomendado

- CPU: 4 núcleos
- RAM: 16 GB recomendado para procesamiento
- Almacenamiento: 30 GB libres
- Conexión a internet

Software necesario

Herramienta	Versión mínima	Propósito
Python	3.14	Limpieza, EDA, ETL
SQL Server	2022	Base de datos del modelo estrella
Power BI	2025	Visualización de Dashboard
Git	2.0+	Clonar el repositorio del proyecto
Google Colab (alternativa)	—	Ejecución en la nube sin instalación local
Docker Desktop		Crear y ejecutar contenedores

Instalación de Software y Herramientas

Docker

Crear el contenedor para la base de datos

Descarga oficial: <https://www.docker.com/products/docker-desktop/>

Pasos:

1. Descarga Docker Desktop desde el enlace oficial.
2. Instalarlo normalmente (necesita permisos de administrador).
3. Una vez instalado, abre Docker Desktop y asegúrate de que el servicio esté “Running”.
4. En terminal (PowerShell o CMD), ejecuta para verificar:
`docker --version`
`docker ps`

Python

Limpieza, integración y carga de datos en SQL Server mediante scripts.

Descarga oficial: <https://www.python.org/downloads/>

Pasos:

1. Descarga Python 3.14 o superior.
2. Durante la instalación, marca la opción:
Add Python to PATH
3. Verifica la instalación:
`python --version`

Librerías Python requeridas

Ejecuta este comando en la terminal o en un notebook de Colab

Pasos:

1. Ejecutar en terminal
`pip install pandas numpy sqlalchemy pyodbc matplotlib seaborn fastparquet pyarrow`

Microsoft Power BI Desktop

Creación de dashboards interactivos conectados a SQL Server.

Descarga oficial: <https://powerbi.microsoft.com/es-es/desktop/>

Pasos:

1. Descarga e instala Power BI Desktop.

Configuración del entorno

Configuración de SQL Server con Docker

1. Descargar imagen oficial de SQL Server

Terminal:

```
docker pull mcr.microsoft.com/mssql/server:2022-latest
```

2. Crear y ejecutar el contenedor

Terminal:

```
docker run -e "ACCEPT_EULA=Y" -e "SA_PASSWORD=Manager123*" \
-p 1433:1433 --name sqlserver_bts \
-d mcr.microsoft.com/mssql/server:2022-latest
```

3. Validar que el contenedor esté corriendo

Terminal:

```
docker ps
```

El resultado debería ser algo como esto:

CONTAINER ID	IMAGE	PORTS
a1b2c3d4e5f6	mcr.microsoft.com/mssql/server:2022-latest	0.0.0.0:1433->1433/tcp

Conexión a base de datos

Conectarse al contenedor desde SSMS

1. Abrir SQL Server Management Studio
2. En la ventana de conexión ingresar valores (Server type, Server name, user, Password) y click en conectar

Conectarse desde Python

1. Instalar dependencias

Terminal:

```
pip install sqlalchemy pyodbc
```

2. Código para la conexión

```
from sqlalchemy import create_engine

# Cadena de conexión a SQL Server en Docker
engine = create_engine(
    "mssql+pyodbc://sa:Manager123@localhost,1433/BTS_Project?driver=ODBC+Driver+17+for+SQL+Server"
)

# Prueba de conexión
with engine.connect() as conn:
    result = conn.execute("SELECT @@VERSION;")
    print(result.fetchone())
```

**Si el controlador ODBC no está instalado, puedes descargarlo desde:

<https://learn.microsoft.com/en-us/sql/connect/odbc/download-odbc-driver-for-sql-server>

Creación de Modelo Estrella, ETL y visualización

1. Descargar Dataset
Análisis de Desempeño y Retrasos en Vuelos de EE.UU. (2011–2021)
2. Crear base de datos
 - a. Ejecutar DLL.sql -> Creacion de base y Esquema de estrella
 - b. Ejecutar [ETL.py](#) -> Llenado de data
3. Conexión a Power BI
 - a. Elegir Obtener datos: SQL Server Database
 - b. Ingresar credenciales: Servidor, base de datos
 - c. Cargar las tablas de hechos y dimensiones
 - d. Construir KPIs y gráficas a conveniencia.