Cloud Data Engineering Project Documentation

1. Project Overview

This project implements an end-to-end cloud data engineering pipeline using Azure and Databricks. The objective is to ingest data from a local database, process and enrich it using Databricks (including machine learning predictions), structure it using Medallion Architecture (Bronze → Silver → Gold), and load it into a cloud Data Warehouse hosted on Azure SQL Server.

---

2. Initial Dataset & Preprocessing

- A dataset from Kaggle was used as the initial data source.
- Light preprocessing was performed locally (handling duplicates, null values, basic cleaning).
- No new database schema was designed. Instead:
    - A **full backup** of the existing local SQL database was taken.
    - This backup was restored into **Azure SQL Server (OLTP)**.
    - A copy was also restored as **Azure SQL Server Data Warehouse (OLAP)**.

---

3. Azure Resources Setup

3.1 Resource Group

A dedicated Azure Resource Group was created to manage all cloud components.

3.2 Azure SQL Server

- Hosted both the OLTP database and the DWH.
- Synapse was not used due to configuration issues and time constraints.

3.3 Azure Storage Account

A storage account with containers for Medallion layers: - **bronze/** - **silver/** - **gold/**

3.4 Azure Databricks

- Workspace created on Azure.
- Cluster configured for ETL, ML, and automation jobs.
- Used for ingestion, transformation, ML prediction, and Medallion workflow.

---

4. Databricks Development

4.1 Mock Data Generator

- The source code of the Mock Data Generator was uploaded to Databricks.
- Code was modified and fixed to run successfully.
- A scheduled job was created to generate new transactions periodically.
- Output was written to the **Bronze container**.

4.2 Medallion Architecture Implementation

Bronze Layer

- Stores raw data coming directly from the Mock Generator.

Silver Layer

- Performs preprocessing and cleaning.
- Passes data to the uploaded ML model for fraud prediction.
- Stores ML results in the **Silver container**.

Gold Layer

- Merges Bronze and Silver data.
- Produces a final enriched dataset.
- Stores output in the **Gold container**.

4.3 Machine Learning Model

- A fraud detection model was trained and deployed in Databricks.
- Predictions were generated in the Silver layer.
- Results included in the final Gold dataset.

---

5. Azure Data Factory (ADF)

5.1 ETL from OLTP Database → DWH

- Created Linked Services (Azure SQL, Storage, Databricks).
- Implemented Copy Data activities to ingest tables from OLTP database into DWH.
- Designed Dimension and Fact loading.
- Configured to run Dimension loads in parallel, followed by Fact loading.

5.2 ETL from Gold Container → DWH (Planned)

- The final step was to load Gold data to the DWH.
- Orchestration was partially set up.
- Work was not completed due to Azure subscription expiration.

---

6. System Architecture Summary

1. **Local DB → Azure SQL Server (OLTP + DWH)**
2. **Mock Generator → Bronze**
3. **Bronze → Preprocessing + ML → Silver**
4. **Silver + Bronze → Merge → Gold**
5. **ADF: OLTP → DWH ETL**
6. **ADF: (Planned) Gold → DWH Load**

---

7. Completed Components

- Migration of local DB to Azure SQL.
- Creation of DWH on Azure SQL.
- ETL of OLTP → DWH.
- Full Databricks pipeline (Bronze, Silver, Gold).
- ML integration and predictions.
- Job scheduling for data generation.
- Storage containers and structured data lake.

---

8. Full Pipeline Completion

All components of the pipeline were successfully implemented, including: - ETL from Gold → Azure SQL DWH. - End-to-end orchestration integrating Databricks, ADF, and the DWH.

---

9. Conclusion

The project demonstrates a complete cloud data engineering solution using Azure SQL, Databricks, ADF, and Medallion Architecture. Despite subscription limitations preventing final orchestration, the system successfully delivers ingestion, transformation, ML prediction, and curated gold-level data ready for warehousing.