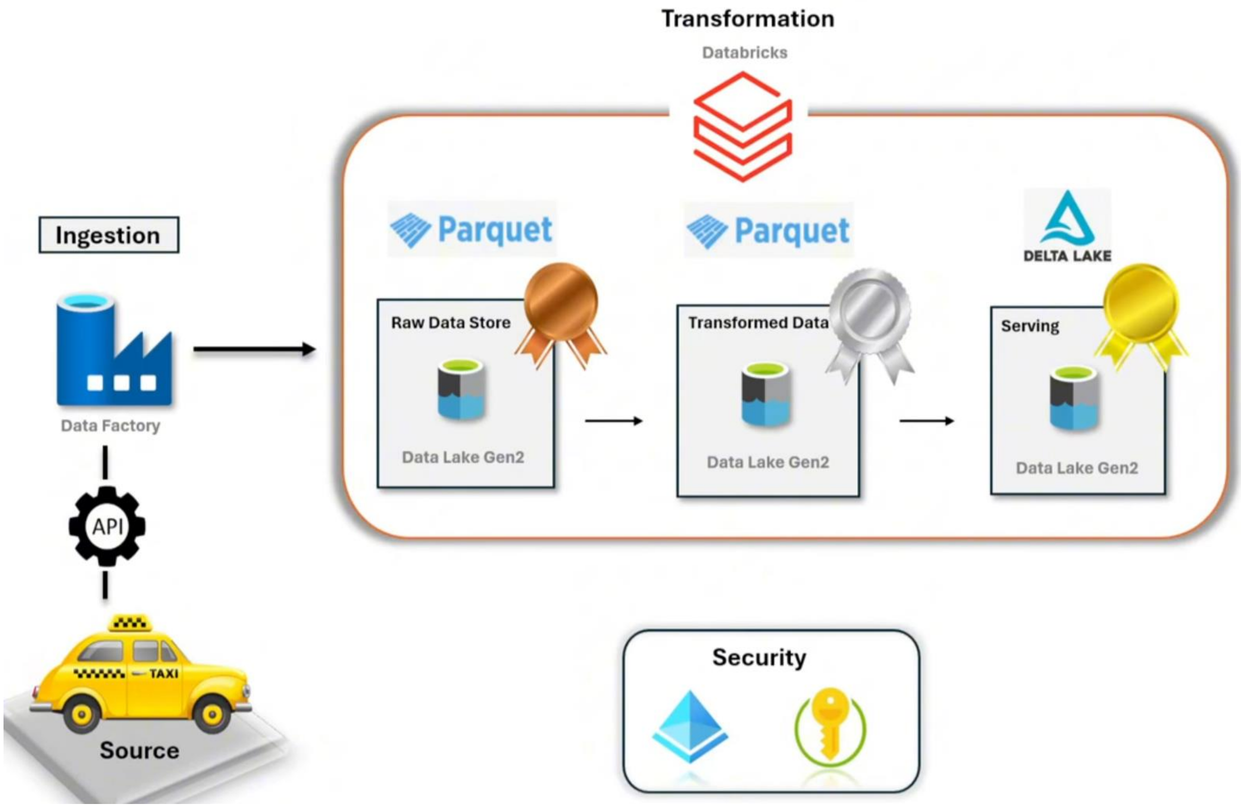
NYC Taxi Data Engineering Pipeline – Azure

Overview

This project demonstrates an **end-to-end data engineering pipeline** built using **Azure** to process **NYC Taxi Data**. The pipeline involves **data ingestion, transformation, storage, and visualization**, enabling real-time analytics on taxi trip data.



🔥 Key Features:

✅ **Automated Data Ingestion** – Using **Azure Data Factory (ADF)** to fetch data from APIs.
✅ **Data Processing & Transformation** – Using **Azure Databricks (PySpark)** to clean and enrich taxi trip data.
✅ **Medallion Architecture (Bronze → Silver → Gold)** – Structured data processing for better analytics.
✅ **Delta Lake for Versioning & Time Travel** – Ensuring historical data tracking.
✅ **BI Dashboard (Power BI)** – Analyzing key taxi trends like busiest pickup spots and average fares.

🗂️ Project Structure

bash

CopyEdit

```
NYC-Taxi-DE-Pipeline/
│── data/                  # Sample NYC taxi trip data (CSV/Parquet)
│── notebooks/             # Databricks notebooks for data processing
│── pipelines/             # Azure Data Factory pipeline configurations
│── reports/               # Power BI dashboard files
│── scripts/               # PySpark scripts for data transformation
│── README.md              # Project Documentation
```

🏙️ Understanding NYC Taxi Data

NYC Taxi Dataset contains millions of ride records, including:

- 🚖 **Pickup & drop-off locations** (latitude/longitude)
- ⏳ **Trip duration**
- 💰 **Fare amount & payment type**
- 🚘 **Total passengers per ride**
- 📅 **Trip timestamps**

Real-World Use Cases

🔷 **Traffic Analysis** – Identify congestion-prone areas.
🔷 **Fare Optimization** – Detect patterns in trip pricing.
🔷 **Peak Demand Analysis** – Find busiest ride hours.
🔷 **Urban Planning** – Improve public transport routes.

🏗 Project Workflow & Architecture

🔷 1. Data Ingestion Layer – Azure Data Factory (ADF)

- ADF **fetches taxi data from an API** and loads it into **Azure Data Lake**.
- **Pipeline is scheduled** to collect new data periodically.

🔷 2. Storage Layer – Azure Data Lake

- Raw data is stored in **Azure Data Lake Storage (ADLS Gen2)**.
- Supports **structured & unstructured data**.

🔷 3. Processing Layer – Azure Databricks (PySpark)

- **Transformations in PySpark:**
  ✅ Removing duplicates & fixing missing values
  ✅ Calculating **average fare per mile**
  ✅ Extracting peak hours & busiest pickup locations

🔷 4. Data Warehouse Layer – Delta Lake

- Data is **stored in Delta format** for versioning & rollback.
- Enables **Time Travel** – retrieve older versions if needed.

🔷 5. Analytics & Visualization Layer – Power BI

- **Dashboard Insights:**
  ✅ **Busiest pickup spots** (Times Square? JFK Airport?)
  ✅ **Peak ride hours** (Morning rush vs Late-night trips)
  ✅ **Fare per mile comparison**

🛠 Technologies Used

Technology	Purpose
Azure Data Factory	Automate data ingestion from API
Azure Data Lake (ADLS)	Store raw taxi trip data
Azure Databricks	Process & transform data using PySpark
Delta Lake	Maintain historical data & enable versioning
Power BI	Visualize insights using dashboards

🚀 Project Setup & Execution

🏗 Step 1: Set Up Azure Services

1️⃣ **Create Azure Resource Group** (like a folder for all resources).
2️⃣ **Create Azure Storage Account & Data Lake (ADLS Gen2)** to store data.
3️⃣ **Set up Azure Data Factory (ADF)** to automate ingestion.
4️⃣ **Launch Azure Databricks** and configure a cluster.

⚙ Step 2: Data Ingestion – Using Azure Data Factory

🔷 **Connect to NYC Taxi API** and schedule data extraction.
🔷 **Save files in Azure Data Lake in raw format** (Bronze Layer).

🔥 Step 3: Data Processing – Using Databricks & PySpark

✅ **Read data from Data Lake** into a PySpark DataFrame.
✅ **Clean the data** (handle missing values, remove duplicates).
✅ **Transform data** – Calculate average fare per mile.
✅ **Store processed data** in Delta Lake (Silver Layer).

📊 Step 4: Visualization – Using Power BI

✅ Connect Power BI to Delta Lake.

✅ Create dashboards showing ride trends, peak times, and fare distributions.

📈 Example SQL Queries on Delta Tables

🔹 Find Top 5 Busiest Pickup Locations

sql

CopyEdit

```sql
SELECT pickup_location, COUNT(*) AS total_rides

FROM nyc_taxi_gold

GROUP BY pickup_location

ORDER BY total_rides DESC

LIMIT 5;
```

🔹 Calculate Average Fare Per Mile

sql

CopyEdit

```sql
SELECT trip_distance, AVG(fare_amount) AS avg_fare

FROM nyc_taxi_gold

GROUP BY trip_distance

ORDER BY trip_distance DESC;
```

📊 Power BI Dashboard Insights

Key Insights:

✅ **Times Square & JFK Airport** are busiest pickup locations.
✅ **Morning rush hours (8-10 AM) & late nights (10 PM-12 AM) have peak demand**.
✅ **Shorter trips (<3 miles) have the highest ride frequency**.

🎯 Why This Project Matters?

🚕 **NYC Taxi Data is used in real-world scenarios**, such as:
✅ **Urban traffic optimization** – Helping city planners reduce congestion.
✅ **Fare policy decisions** – Understanding customer ride patterns.
✅ **Taxi business growth** – Identifying high-demand locations for drivers.