**Implementation Steps**

**Milestone 1: Environment & Prerequisites**

1. **Provision Azure Resources**
   * Azure Data Lake Storage Gen2 (Raw, Curated, Analytics zones)
   * Azure Data Factory (ADF) or Synapse Pipelines for orchestration
   * Azure Databricks (for transformation & ML)
   * Azure Key Vault (secrets/connection strings)
   * Azure Purview / Microsoft Purview (governance & lineage)
   * Event Hubs / Kafka (for real-time streams)
   * Power BI (for visualization)
2. **Networking & Security**
   * Private endpoints for ADLS, Databricks, and ADF
   * Role-based access (RBAC) setup
   * Service principals for ADF & Databricks

**Milestone 2: Ingestion Layer**

Datasets we have:

* Flight Operations (flight\_operations.csv)
* Aircraft Telemetry (aircraft\_telemetry.csv)
* Passenger Bookings (passenger\_bookings.csv)
* Passenger Master (passenger\_master.csv)
* Crew Assignments (crew\_assignments.csv)
* Weather Data (weather\_data.csv)
* Maintenance Logs (maintenance\_logs.csv)
* Governance Logs (governance\_logs.csv)
* Data Quality Checks (data\_quality\_checks.csv)

**Steps:**

1. **Raw Landing Zone Setup**
   * Create containers in ADLS: /raw/flight\_ops/, /raw/telemetry/, /raw/bookings/, etc.
   * Enable **hierarchical namespace** for folder structure.
2. **Batch Ingestion**
   * Use **ADF Copy Activity** to pull CSV files into /raw.
   * Schedule pipelines (e.g., hourly for operations, daily for passengers).
3. **Streaming Ingestion (Telemetry + Bookings)**
   * Stream aircraft\_telemetry.csv (simulated) via Kafka/Event Hubs → ADLS /raw/streaming/.
   * Capture passenger\_bookings.csv changes via Change Data Capture (CDC) or API.
4. **Metadata-driven Framework**
   * Store file metadata (dataset name, frequency, delimiter, schema) in a **control table** (SQL DB or ADLS JSON).
   * ADF pipelines read control metadata → dynamically load datasets.
5. **Initial Validation**
   * Row count, null checks, schema validation during ingestion.
   * Store logs in /raw/logs/ and Purview.

**Milestone 3: Transformation & Curation Layer**

1. **Bronze (Raw) → Silver (Curated)**
   * Use **Databricks Auto Loader** for streaming data.
   * Standardize column names, formats (date, time, codes).
   * Join Passenger Bookings with Passenger Master.
2. **Silver → Gold (Analytics-Ready)**
   * Aggregate flight delays, telemetry anomalies, passenger booking trends.
   * Create star schema / wide tables for analytics.

**Milestone 4: Governance & Quality Layer**

1. **Data Quality Rules**
   * Apply validation rules from data\_quality\_checks.csv.
   * Store results in curated zone /curated/quality/.
2. **Data Lineage**
   * Register datasets in **Purview**.
   * Enable lineage from ingestion → curation → reporting.
3. **Access Control**
   * PII masking for passenger data (GDPR/CCPA compliance).
   * Row-level security for crew vs passenger data.