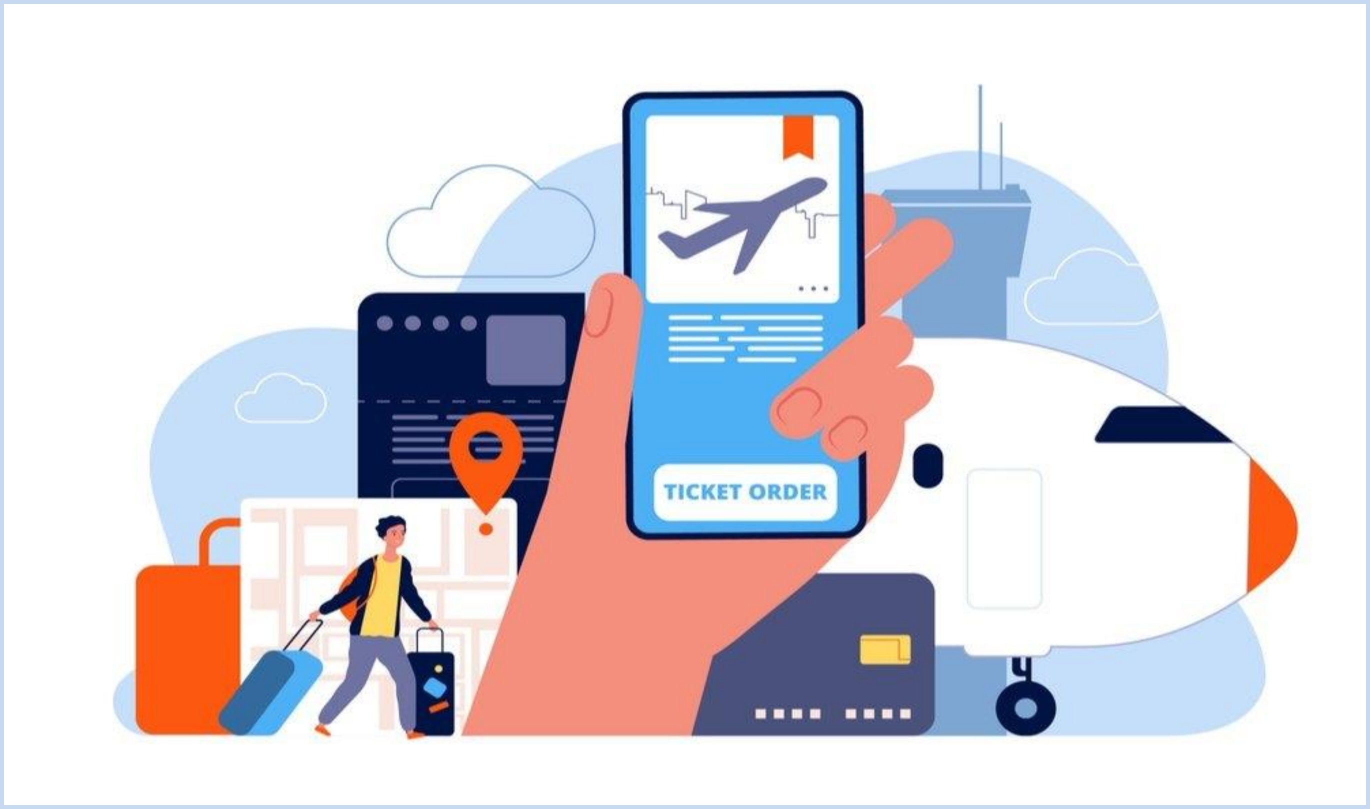
CLOUD ARCHITECTURE FOR AN AIRLINE TICKETING SYSTEM



BY: JOHN ODI ETTA

April 10th 2025

Table of Contents

MISSION

"To design and implement a scalable, resilient and secure cloud infrastructure that integrates diverse data sources to support airline operations, enhance workflows, and ensure high performance and reliability.

OBJECTIVES

Scalable Infrastructure

Design an infrastructure capable of handling fluctuations in system usage, ensuring continuous service availability during peak loads.

Data Consistency

Ensure the consistency and integrity of data across internal systems, supporting mission-critical operations like booking, scheduling, and inventory management.

Centralized Repository

Design a centralized data repository to consolidate operational data, enabling efficient reporting and real-time analytics.

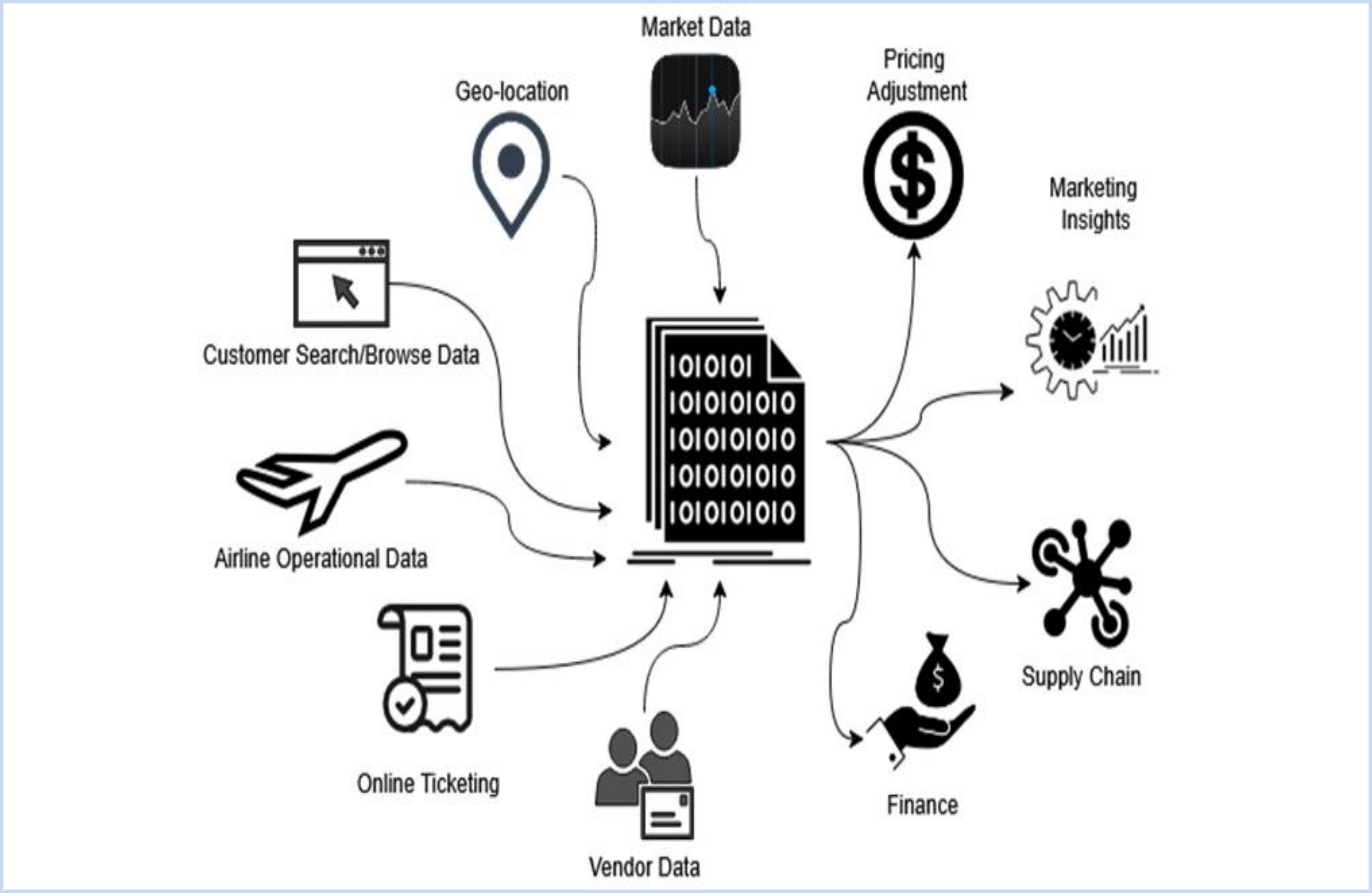
System Monitoring

Integrate monitoring, logging, and alerting tools to track system health and performance, ensuring rapid issue detection and resolution.

Support Seamless Integration with Internal and External Systems

Enable smooth interoperability between the ticketing system and other modules such as flight operations, customer management, GDS (Global Distribution Systems), and finally expose data through secure APIs for partners, agencies, and customer-facing platforms.

DESIGN AND DISCOVERY



Data Flow

DATA SOURCES

1. **Online Ticketing Data**
   Nature: Structured, Real-time
   Format: CSV, JSON
   Usage: Captures real-time booking transactions from direct sales channels, providing insights for demand forecasting, pricing strategies, and customer segmentation analysis.

2. **Vendor Data (from Online Travel Agency)**
   Nature: Structured, Batched, and Real-time
   Format: CSV, JSON
   Usage: Aggregates booking data from external vendors, enhancing demand forecasting, evaluating partnerships, and assessing market reach.

3. **Customer Search and Browsing Data**
   Nature: Unstructured, Streaming
   Format: XML, TXT, Event Logs
   Usage: Analyzes user activity on digital platforms to model customer intent, refine demand forecasts, and support dynamic pricing initiatives.

4. **Geo-location Data**
   Nature: Semi-structured, Real-time, or Streaming
   Format: JSON
   Usage: Offers real-time customer location insights, enabling regional demand analysis, personalized promotions, and route optimization.

5. **Airline Operational Database**
   Nature: Structured, Batched
   Format: JSON, CSV
   Usage: Includes operational details like schedules and inventory, crucial for capacity planning and aligning demand with supply.

6. **Historical Sales and Booking Records**
   Nature: Structured, Batched
   Format: CSV, JSON
   Usage: Archives past booking trends for training forecasting models, seasonal analysis, and performance evaluation.

7. **Competitor Pricing and Market Data**
   Nature: Semi-Structured, Batched, or Real-time
   Format: CSV, JSON
   Usage: Gathers competitor and market insights to refine pricing strategies, model demand, and optimize revenue.

Data Destination (SINK)

A. **Operational Efficiency Reports (Supply Chain)**
   Purpose: To enhance airline operations by optimizing resource utilization and minimizing inefficiencies.
   Details: Insights into flight schedules, seat occupancy, crew performance, and resource allocation highlight areas for improvement, such as overbooked flights and delays. Key metrics include turnaround times, on-time performance, and staff productivity.

B. **Customer Behavior Insights (Marketing Insights)**
Purpose: To analyze customer preferences and behavior for targeted marketing and pricing strategies.
Details: Derived from user activity, booking history, and demographics, these insights reveal customer segments, route demand trends, and decision drivers, supporting personalized offers and revenue growth.

C. **Demand Forecasting Reports (Supply Chain, Pricing Adjustment)**
Purpose: To anticipate future demand, enabling efficient capacity planning, pricing, and resource allocation.
Details: Combines historical and real-time data with external factors (weather, holidays/events) to predict demand by route, time, and segment, informing inventory and fleet management.

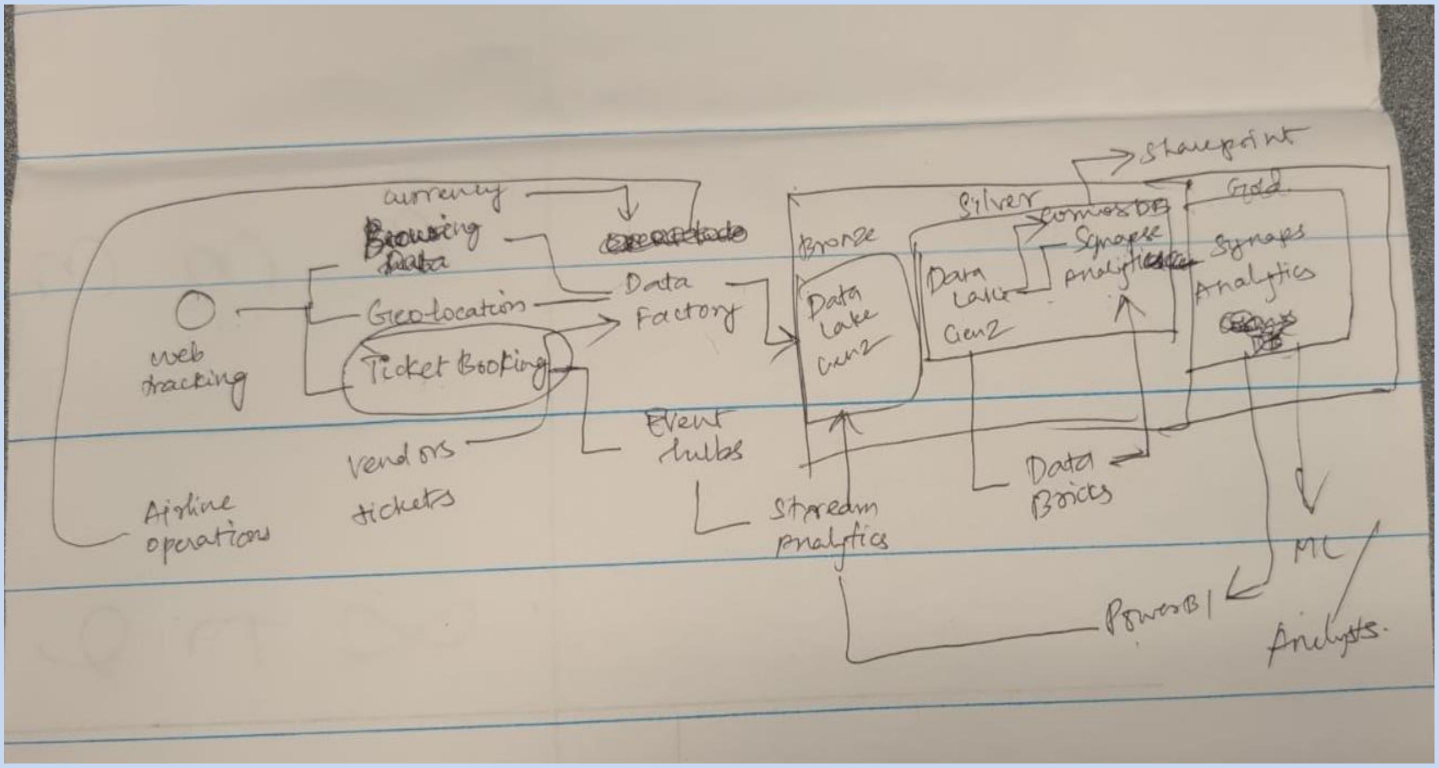
D. **Pricing Optimization Analysis (Finance, Pricing Adjustment)**
Purpose: To refine pricing strategies using competitive and market analyses.
Details: Real-time competitor trends and customer behavior data drive dynamic pricing models, ensuring competitive fares that maximize revenue and customer appeal.

E. **Route Optimization Insights (Supply Chain)**
Purpose: To optimize flight routes for efficiency and profitability.
Details: Analyze booking and competitor data to recommend route adjustments, identify high-demand opportunities, and align resources with market needs.

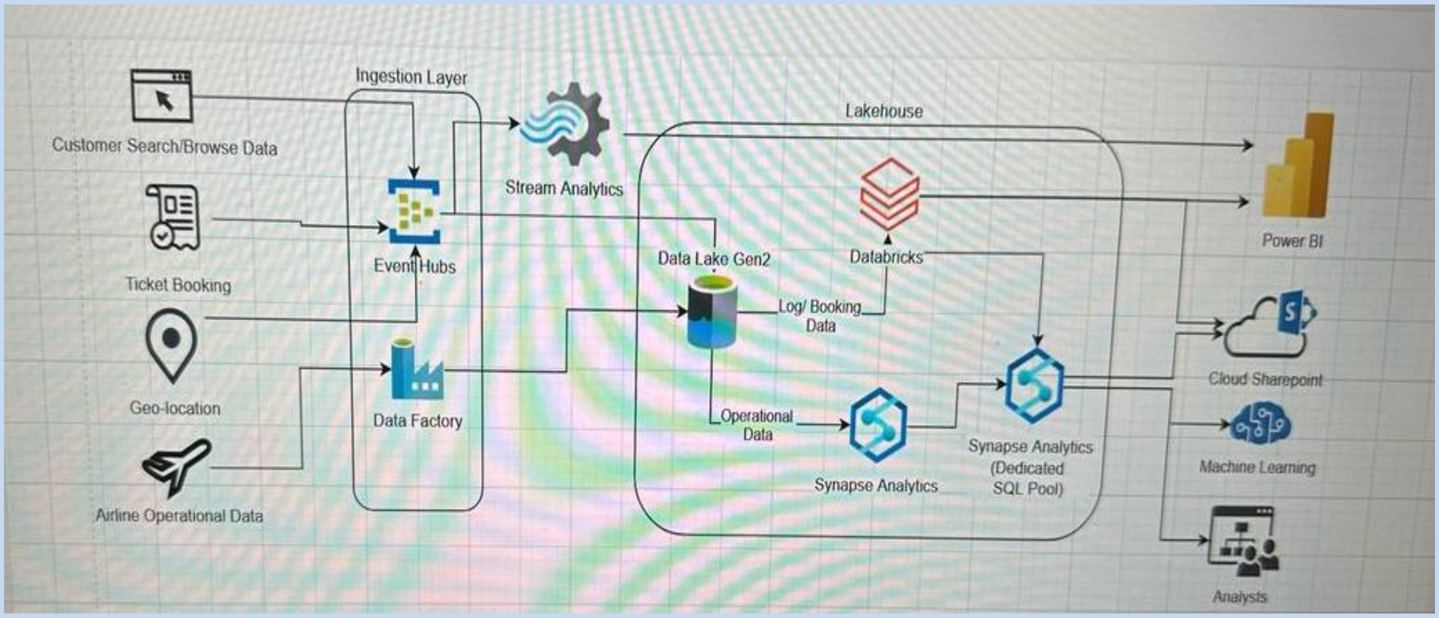
F. **Competitor Performance and Market Positioning Reports (Finance, Marketing Insights)**
Purpose: To benchmark the airline's performance and refine its competitive strategy.
Details: Evaluates competitor pricing, market share, and promotional activities to identify differentiation opportunities and enhance market positioning.

CLOUD ARCHITECTURE PHASES

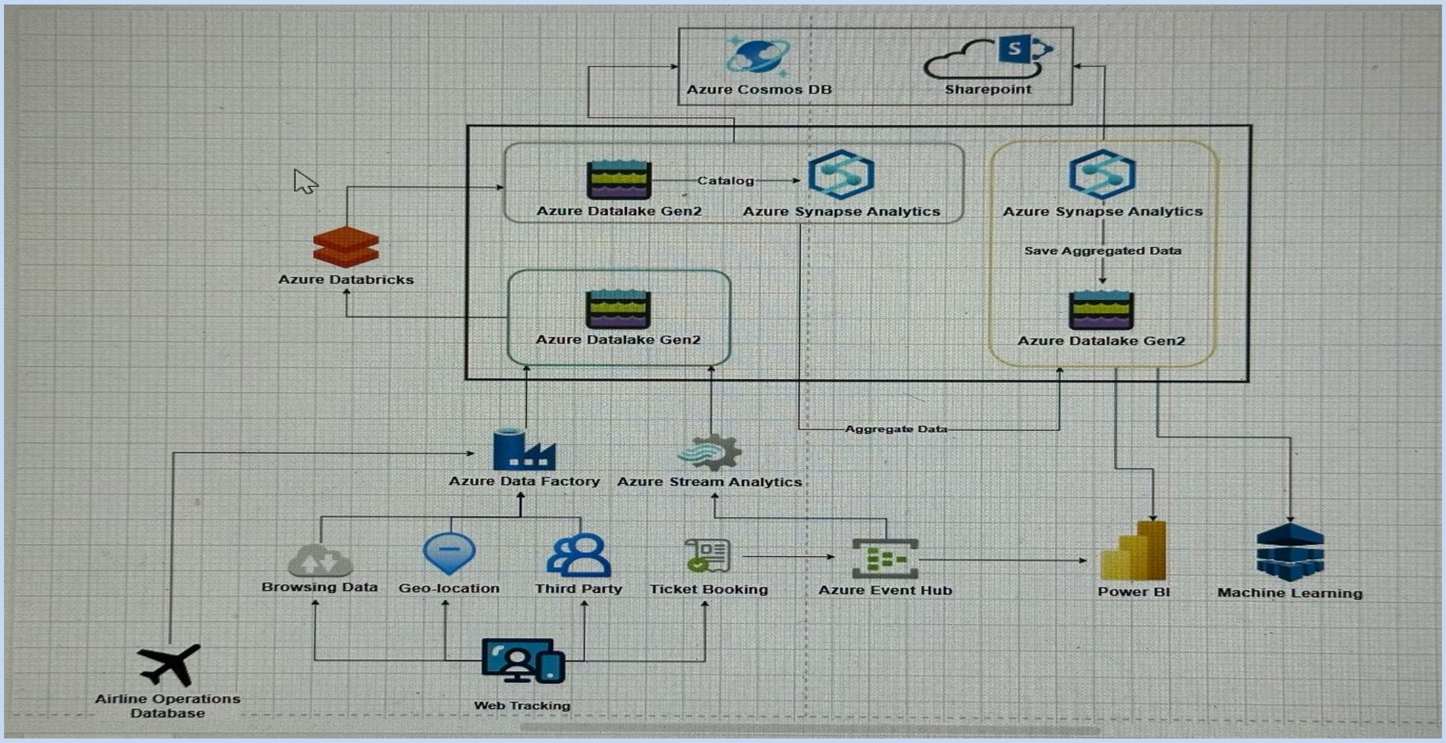
I. First Iteration.



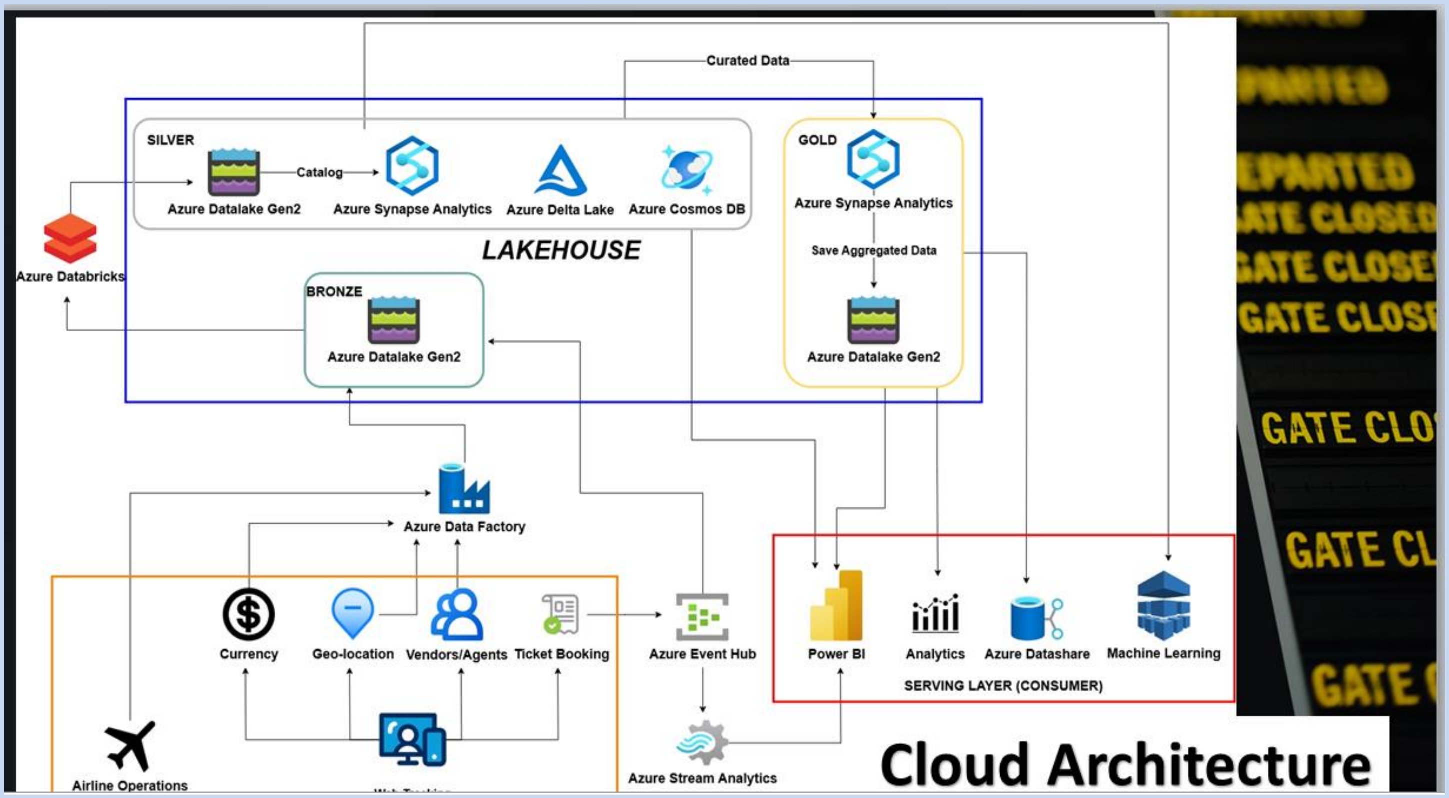
II. Second Iteration



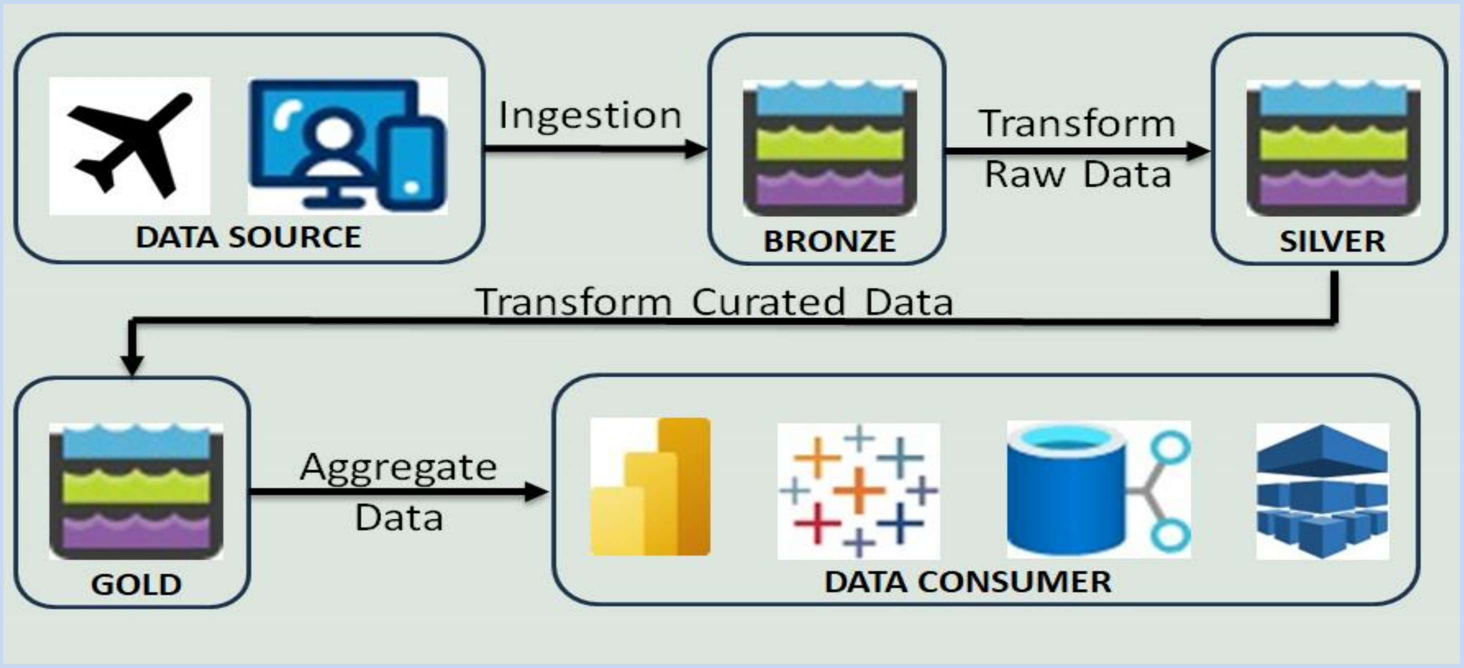
III. Third Iteration



IV. Final Iteration



PIPELINE DESIGN



Data Source Layer

Components: Airline operations, reservation platforms, customer interfaces (mobile/web apps).

Function: Raw data is generated from various transactional systems — flight bookings, customer check-ins, pricing engines, and operational logs.

Type of Data: Semi-structured (JSON, XML), structured (CSV, SQL tables), and unstructured (logs, images).

BRONZE LAYER

The bronze layer is designed to capture data in its raw, unprocessed form, without any transformations or modifications. The data is stored in its original format, allowing for Data lineage, tracking data origins and changes, Flexibility accommodating changing data structures or formats and Scalability, handling large volumes of data

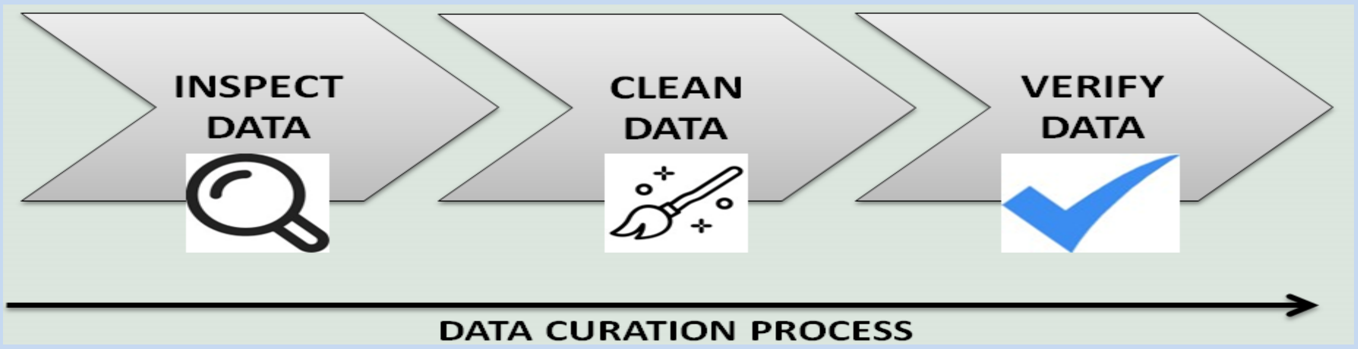
SILVER LAYER: Curated Data

The Silver layer is the second layer in the proposed architecture, where raw data from the Bronze layer is refined, transformed, and optimized for analysis. This layer provides a curated view of the data, making it more suitable for reporting, analytics, and data science applications. The curated layer ensures that;

Data is cleaned, transformed, and formatted for analysis.

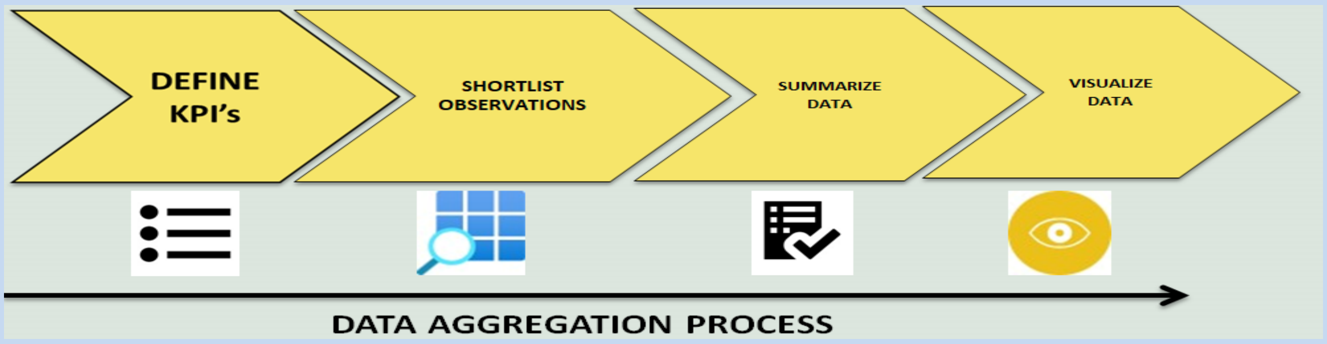
Data is optimized for querying and analysis, often using techniques like data warehousing or data marting.

Data is standardized to ensure consistency and quality.



GOLD LAYER: Aggregated Data

The Gold layer is the third and final layer of the architecture. Curated data from the silver layer is moved into this layer, where data is aggregated and transformed into actionable insights, KPI’s, and data products that drive business decisions. This layer provides a highly curated and aggregated view of the data, making it suitable for: Business Intelligence, Decision making and predictive Analysis.



PIPELINE FAILURE STRATEGY

1. Max Retries

Max Retries is a strategy used to handle transient failures in a system. When a task or request fails, the system will automatically retry the operation a specified number of times (max retries) before giving up and marking it as failed. This strategy reduces the likelihood of failures due to temporary issues, such as network glitches or service unavailability.

2. Backoff Interval

Backoff Interval is used in conjunction with Max Retries. When a task or request fails, the system will wait for a specified interval before retrying. The interval can be fixed or exponential (increasing with each retry). An advantage is that it reduces the likelihood of overwhelming the system with retries, allows for temporary issues to resolve themselves.

3. Timeout

Timeout is a strategy used to limit the amount of time a system waits for a response or completion of a task. If the task doesn't complete within the specified time, it is considered failed. Timing out prevents system hangs, reduces resource utilization, and helps detect issues. Though this can lead to false positives and the choice of timeout value can be challenging.

4. Monitoring & Alerting

This strategy used to detect and notify teams about system failures or issues. It involves tracking system performance, error rates, and other key metrics. Due to rapid detection and response to issues there is an overall reduction in downtime and improvement on system reliability

5. Pipeline Re-run

Pipeline Re-run is used to re-execute a failed pipeline or workflow either from a specific point or from the beginning. This can be useful for handling failures due to external dependencies or data issues. Pipeline re-run strategy facilitates recovery from failures, reduces manual intervention and improves overall pipeline reliability.

FAILURE IMPLEMENTATION IN ARCHITECTURE

A. Batch Ingestion Pipeline
Uses checkpointing using timestamp (last_updated and current_timestamp) to track the last successfully processed data.

"On failure, the system reprocesses data from the last successful timestamp to ensure no data is lost."

B. Streaming Ingestion Pipeline
It uses a session window for group events (customer booking sessions).

"On failure, reprocess from the most recent offset of the session's stream to ensure data continuity."

C. Curation Pipeline
Curates the accumulated data in a fixed time window. Performs dependency checks on what actions to take onSuccess or onFailure.

"*Stores the last successful window to allow resumption of processing from the failed window, ensuring no duplication or loss of data."*

D. Aggregation Pipeline
Performs fixed interval aggregation (end of day or for an hour).

"*Stores the last successful window to resume processing from the point of failure, ensuring accurate aggregation and no missing data."*

CONCLUSION

The Airlines project demonstrates how a cloud-native, data architecture can address the operational and analytical challenges of a modern airline company. By designing a robust cloud infrastructure rooted in scalability, security, and resilience, this project offers a model that aligns with industry demands for agility and real-time responsiveness.

The layered pipeline architecture, comprising bronze, silver, and gold layers ,ensure a logical, traceable flow of data from ingestion to actionable insights, enabling accurate forecasting, pricing optimization, and customer behavior analysis. The inclusion of advanced failure strategies across batch, streaming, and aggregation processes safeguards data reliability and continuity, even under stress.

In an industry where operational intelligence directly affects profitability and customer experience, this framework empowers airlines to make proactive, data-informed decisions. As cloud technologies continue to evolve, the principles embedded in this architecture will remain relevant, scalable, and adaptable to future innovations and organizational growth.