

AWS Design Assignment - 1 (Design)

- **Problem Statement:** Designing a Real-time Data Ingestion Pipeline for a Flight Booking Application using AWS Services

- **Background:**

Air travel is one of the most frequently used modes of transportation globally. With the rise of technology, the majority of air ticket bookings have moved online. A hypothetical airline company, 'FlyHigh Airways', aims to modernize its flight booking application by implementing a real-time data ingestion pipeline to handle and process user bookings, cancellations, and search queries. This would not only allow the company to have real-time insights into their bookings but also offer a more personalized experience to their users based on their behaviors.

- **Requirements:**

- **Data Ingestion:** The system should be able to ingest real-time data from the flight booking application. This includes:
 - New bookings
 - Cancellations
 - Search queries made by users (destinations, dates, etc.)
- **Data Processing:** As the data is ingested in real-time, the system should be capable of processing this data for various use cases, such as:

- Real-time analytics
- Alerts on high demand routes
- Inventory updates based on bookings/cancellations
- **Storage:** The system should ensure data durability and be able to store large volumes of data without any significant delay.
- **Scalability:** As 'FlyHigh Airways' plans to expand its operations, the architecture should be scalable enough to accommodate the increasing number of users and data.
- **AWS Only:** The entire architecture should be based on AWS services.
- **Deliverables:**
 - **Architecture Diagram:** A detailed architecture diagram that showcases the real-time data ingestion pipeline using AWS services. Ensure every component in the architecture is labelled.
- **Assumptions:** List down any assumptions made while designing the architecture. This could include assumptions on data volume, rate of data ingestion, specific AWS service limits, or other constraints.
- **Scalability Justification:**
 - Provide a brief explanation on:
 - Why you believe your design is scalable.
 - How it handles peak loads or increased traffic.

- How it ensures minimal latency during real-time data processing.

- **Tips:**

- Think about the entry points for the data, i.e., where it originates from (mobile apps, web apps, etc.)
- Consider AWS services like Kinesis for real-time data streaming, Lambda for event-driven processing, and S3 or Redshift for storage.
- Scalability and fault-tolerance should be at the forefront of your design. Consider services that auto-scale and provide built-in redundancy.
- AWS offers various managed services. Leverage them to minimize operational overhead and focus on the application's business logic.