Airline Backend System

Objective:

We need to build a backend system that can support different features for an airline company. Our end user is going to be someone who wants to book flights and query about flights, so we need a robust system to actually help them give the best experience possible. This document is solely going to focus on the backend part of the system. We want to prepare the whole backend, keeping in mind that the code base should be as maintainable as possible.

Requirements:

- A user should be able to search for flights from one place to another.
  - User should be able to mention the source and destination details.
  - User should be able to select date of the journey.
    - [V2] User should be able to search for return flights and multi city flights.
  - User should be able to select the class of the flights [Non mandatory].
  - User should be able to select the number of seats they want to book [Non mandatory].
  - Now based on the above data, we will list down the flights.
  - We should show our users the best available flights at the top based on time period of the flight and the price.
  - We need to support pagination so that we can list chunks of flights at one point of time.
  - We should support filters for flights based on price, departure time, and duration, airline, and stops.
    - [V2] We can add support for more filters.

- A user should be able to book a flight, considering that user is registered on the platform.
  - User should be able to cancel a booking, and then based on the criteria we can initiate a refund for them.
  - User should be able to request and book excess luggage for every flight.
- For making a booking, the user has to make a payment [dummy].
- Tracking flight prices should be possible, and the user should be notified about any price drops or delays.
- User should be able to list their previous and upcoming flights using PNR.
- User should be able to download boarding pass if they have done online check-in.
- Online check in mechanisms should be supported.
- Notifications via email for completing online check-in before 3 hours of departure.

- ➔ Notifications to users about any flight delay.
- ➔ User should be able to review the flight journey if and only if they have booked a flight.
  - ◆ Review mechanism should involve star rating along with a comment.
  - ◆ While listing any flight, we should also display the review of the flight.
- ➔ User should be able to authenticate to our system using email and password.
  - ◆ [V2] Support ticketing, where users can raise their queries.
- ➔ Listing FAQ, which will be static data.
  - ◆ [V2] prepare seat selection.
- ➔ Coupons for discounts and offers.
- ➔ While making a booking, a person can reserve more than one seats with one login ID.

Non Functional Requirements:

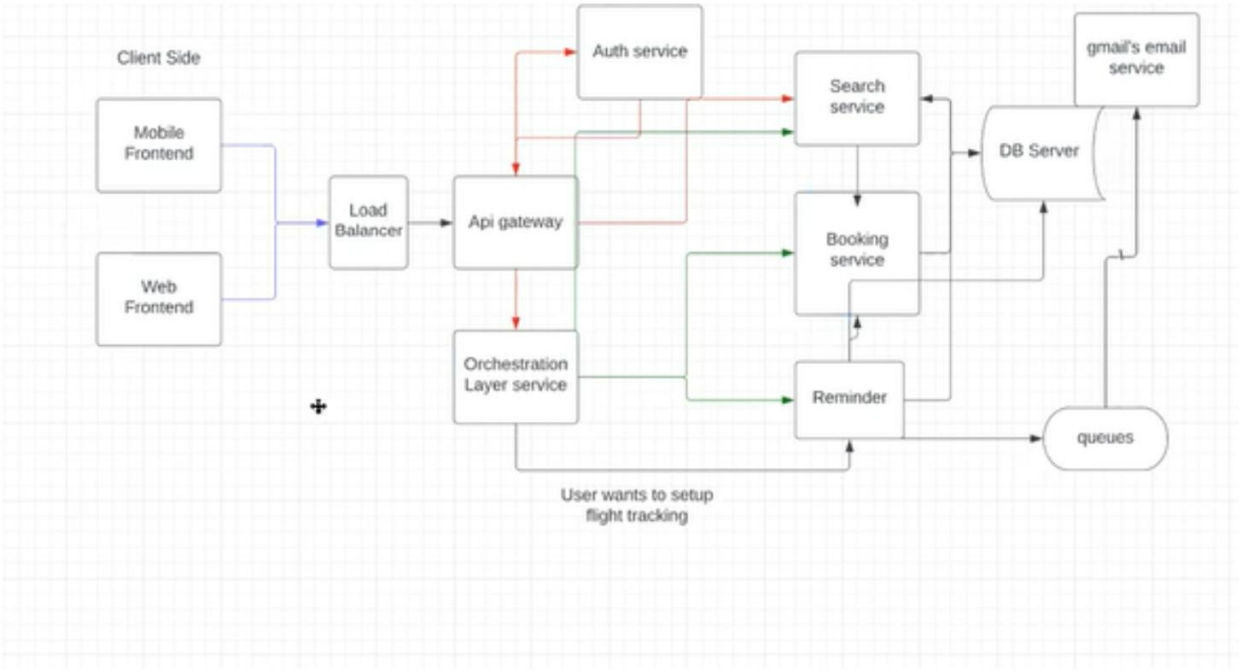
- ➔ We can expect that we are going to have more flight searches than flight bookings.
- ➔ The system needs to be accurate in terms of booking.
- ➔ Expect that we will have approximately 1,00,000 total users; 5,00,000 bookings might come up in one quarter.
- ➔ So in one day, we can expect 2000 bookings.
- ➔ The system should be capable of scaling up to at least 3x the current estimated traffic.
- ➔ The system should handle real time updates to flight prices before user makes the final booking.
- ➔ Concurrency should be handled, and using RDBMS should be a good solution.

Capacity Estimation

- ➔ Storage estimates -
  - ◆ For upcoming 5 years, there will be 80,00,000 bookings and 2,00,000 Users, Considering all the user records and booking records take 10 MB of data, overall 10 TB of memory should be fine for our initial pilot run.
- ➔ Traffic estimates - If we consider 10:1 as the search: booking ratio, then max we expect 150000 search queries a day (~ 2 query/s).

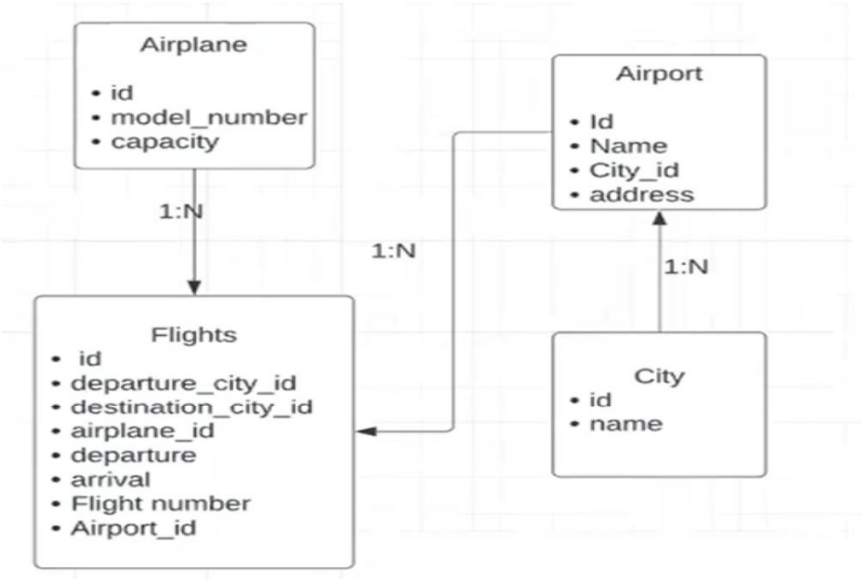
Architecture (High Level Design)

- ➔ The architecture of this project is based on microservices model, which has an orchestration layer to handle multiple microservices.
- ➔ Each microservice has MVC architecture internally.
- ➔ A block diagram of the architecture is given below :-



Search and Flight Service

- ➔ Create Flights
- ➔ Delete Flights
- ➔ Update Flights
- ➔ Search for flights
  - ◆ Based on multiple filtering criteria, we can search for flights.
  - ◆ Pagination.
- ➔ ER Diagram:



Flight Search database

Code of MicroServices:

- Search and Flight Services: https://github.com/Key-runnnnnnn/Flights-and-Search-service.git
- Authentication Service (JWT): https://github.com/Key-runnnnnnn/Auth_Service.git
- Flights Booking Service: https://github.com/Key-runnnnnnn/Flight-Booking-Services.git
- Reminder Service: https://github.com/Key-runnnnnnn/Remainder-Service.git
- API Gateway: https://github.com/Key-runnnnnnn/Flights-Api-Gateway.git
- API of API_Gateway/home: {Current Shutdown in AWS}.
- AutoScaling AWS Shell Script: