

Hackathon: Intelligent Flight Scheduling

Problem Statement

Due to capacity limitations and a heavy passenger load, flight operations at busy airports, especially in **Mumbai and Delhi**, are becoming a scheduling nightmare. Controllers and operators need to find efficiency in scheduling within the constraints of the system and find means to de-congest flight traffic at these airports.

Challenge

Analyze flight routes using **Flightradar24 or FlightAware data** for a few busy airports (e.g., Mumbai, Delhi) and provide insights to support scheduling decisions.

Dataset

You are required to take **one week's worth of flight data** for Mumbai Airport (BOM) from the Flightradar24 website. Schedule information for the same time period also needs to be collected from Flightradar24.

- **Sample Data:** Flight_Data.xlsx
-

Constraints

- Flight schedules are consolidated around **peak hours** for maximum revenue generation.
 - Airport runways have **limited capacity** for takeoffs and landings.
 - Schedule disruption has a **cascading effect** on subsequent flights.
 - **Weather-related disruptions** can lead to reduced runway capacity.
-

Expectations

- This project should use open-source AI tools to analyze flight data and provide an interface to query the processed information using **NLP prompts**.
- Find the **best time to take off/land** at an airport by analyzing the difference between scheduled vs. actual times.
- Identify the **busiest time slots** at an airport that should be avoided.

- Provide a model to **tune the schedule time** for any flight and visualize its impact on delays.
 - Provide a model to **isolate flights that have the biggest cascading impact** on schedule delays.
-

Addendum: Resources

1. **Flightradar24**: <https://www.flightradar24.com/>
 2. **FlightAware**: <https://www.flightaware.com/>
 3. **Mumbai Airport Data**: <https://www.flightradar24.com/data/airports/bom>
 4. **Delhi Airport Data**: <https://www.flightradar24.com/data/airports/del>
-

Required Submission Outline

You must provide an outline of your submission that contains the following sections:

1. Proposed Solution

- A detailed explanation of your Idea/Solution/Prototype.
- A clear description of how your solution addresses the problem statement.
- An explanation of the innovation and uniqueness of your solution.

2. Technical Approach

- **Technologies Used**: List the programming languages, frameworks, libraries, and hardware you plan to use.
- **Methodology**: Describe the process for implementation, including flow charts, architecture diagrams, or images of a working prototype.

3. Feasibility and Viability

- An analysis of the feasibility of your proposed idea.
- A discussion of potential challenges and risks.
- The strategies you will use to overcome these challenges.

4. Research and References

- Provide details and links to any research papers, articles, or other works you referenced.