

# Flight Delay Analytics

## Project Summary and Description

People don't want to encounter delays when taking a flight. The stress of spending a night at an airport due to weather or general airport delays is a harrowing experience, especially for people traveling with children. Our solution estimates the delays you would encounter at any airport in the USA, with a specific airline, or on a specific route around the country, while providing visualization of the requested information.

## Creative Component

A good creative component to add would be having the map be interactive, specifically when it comes to interacting with airports. The map would visually display the applied filters such as desired delay time, destination, etc., and users would be able to see airport specific information. Clicking on an airport would display the top incoming/outgoing flights that fit the user's criteria in a list, and the city's information would be emphasized.

We could implement the map visually with WebGL or other API's that integrate with React, and have listeners to display filtered and highlighted information when cities are selected.

## Usefulness

Flight departure times can be variable and inconsistent across different cities, airlines, times of month, etc. This application is useful because it utilizes historical flight delays to enable users to better plan out their flights. When booking a flight, departure time is often inaccurate, and unexpected delays can disrupt travel plans, leading to missed connections, wasted time, and added stress. Our application will help single out flight delay patterns that users can use to make more conscious flight decisions, whether it's opting for an earlier departure time, choosing a different airline, or planning for potential delays in transit. Our users will be able to access detailed visualizations of aggregated flight data which they will be able to filter based on their specific needs.

MiseryMap is a similar application that also provides visualization of flight delays across the US. Our application differs in 2 ways: users will be able to (1) create a watchlist of flights they want to take in the future, and (2) interact with the data using filters. We will try to model

our map functionality using the following resource as inspiration:

<https://vega.github.io/vega/tutorials/airports/>

## Realness/Data Description

- USA Flight Information Dataset:  
[https://www.kaggle.com/datasets/patrickzel/flight-delay-and-cancellation-dataset-2019-2023?select=flights\\_sample\\_3m.csv](https://www.kaggle.com/datasets/patrickzel/flight-delay-and-cancellation-dataset-2019-2023?select=flights_sample_3m.csv)
  - Source: from a Kaggle dataset that has pulled 5 years of flight data from the US Department of Transportation's Bureau of Transportation Statistics website
  - File type: CSV
  - Dimensions: [29380334, 32]
  - Content: dataset containing information about US flights from August 2019 to August 2023 including flight information, origin and destination information, and general delay information.
- City Information Dataset: <https://simplemaps.com/data/world-cities>
  - Source: from a world cities database found through a collaboration between the National Geospatial Intelligence Agency, the US Census Bureau, USGS, and NASA
  - File type: CSV
  - Dimensions: [~43000, 11]
  - Content: dataset contains information about 43,000 cities around the world. We will only be utilizing the latitude, longitude, and population for the US cities in this dataset that are also included in the flights dataset.

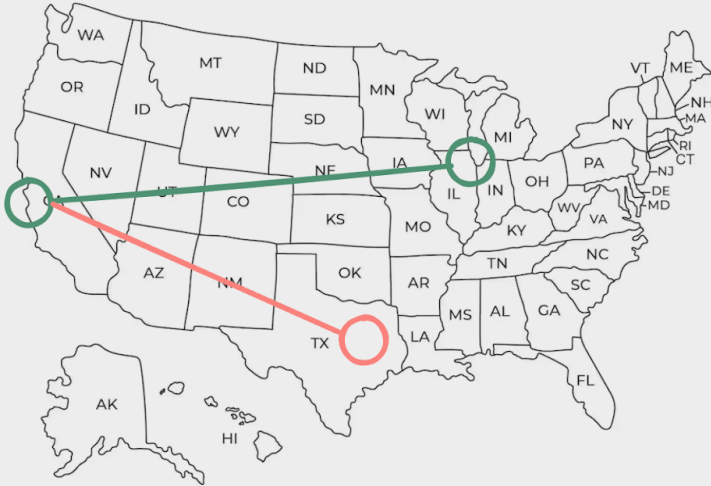
## Functionality

- Users can see the expected (average) delay between two cities
- Users can insert/delete with newer delay times between two cities as see fit (based on personal flight experience)
- Flight watchlist
  - Users can insert preferred flights into a list of favorites

- Users can remove flights from the watchlist
- Filters (selections highlighted on map):
  - Filtered results will be highlighted on the map
  - Allow users to apply multiple filters simultaneously
  - Filter options:
    - Airline: only show flights offered by specified airline
    - Departure delay: only show flights within specified delay threshold (min and max range)
    - Origin/destination cities: only display flights from specified origin and destination
    - Time of month/day: only display flights occurring at specified time
    - Population: only display flights within a specified population threshold

## Low-Fidelity UI Mockup

### Flight Delay Tracker



Flights Matching Filters

Airline

Origin

Destination

## **Project Work Distribution**

The entire team will be responsible for designing the structure and relationships in our SQL database. From there, we have two large subtasks to tackle: the watchlist of flights that each user creates for their own monitoring purposes and the filtering of our dataset by specific parameters (airline, time, origin, destination, flight #, etc). Team members can be assigned specific queries to write and test (ex. one task could be a date filter where a member is tasked with implementing and testing their filter). After writing all necessary endpoints in our backend and ensuring that they interact properly with our database, we will start on the frontend and break up this larger task into subtasks once again for team members to take on. Single member subtasks could include working on the filter component, the map component, and the watchlist component in our frontend.