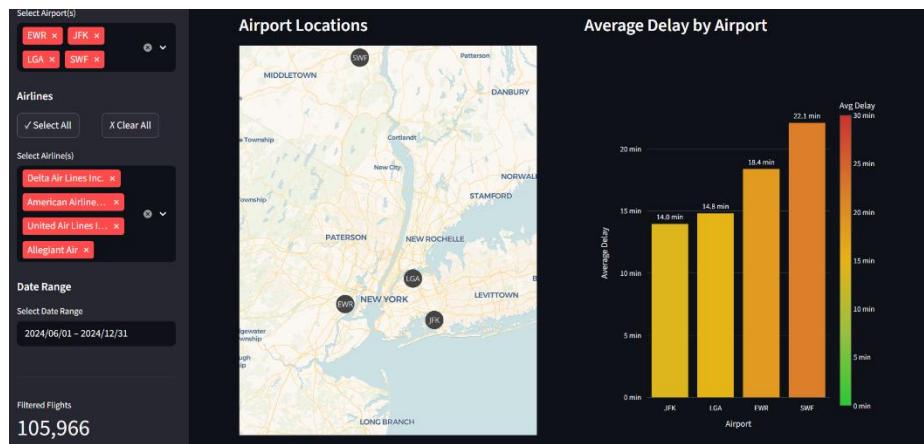


# Data Visualization – Mini Project: Documentation

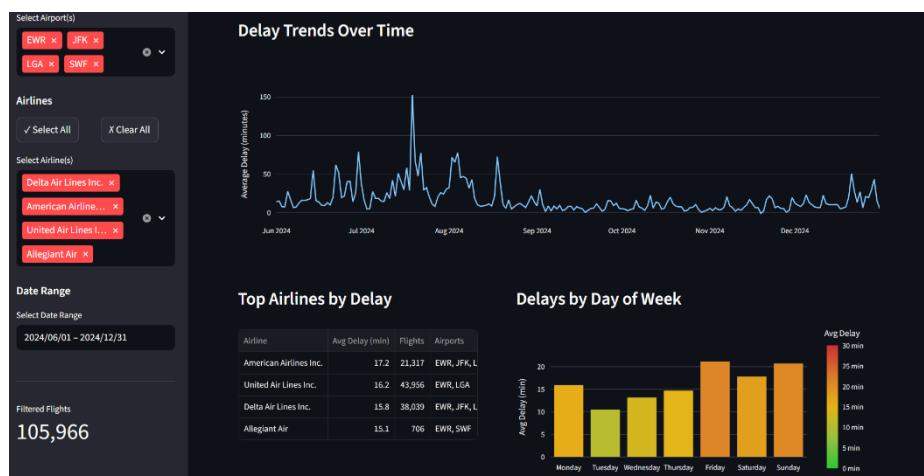
## 1. Important Choices

- Sidebar with filters to keep the main dashboard ‘clean’ but still control displayed airports, airlines and date ranges
- Excluding all cancelled flights to avoid skewed statistics
- Deciding on what to show in the plots and how (eg. I couldn’t decide on one bar chart, but I figured that, since it’s supposed to be a prototype, the members of the Port Authority could make a better decision after seeing more plots.)



## 2. Biggest Challenge

- Cummulative Area Chart: I had some trouble dividing the overall distribution into the parts belonging to the respective airports. I also tried a bunch of different variations, e.g. calculating the percentage for each airport independently and then stacking them, but I think this would have required an even more detailed description than the version I decided on.
- Map plot: My original idea was to display flight routes as lines connecting pairs of origin and destination, colour-coded according to the average delay time per route, with extra information available when the user hovers over the map. Unfortunately, this proved to be quite difficult to implement, and I didn’t have enough time to complete it. That’s why the map only shows the origin airports.



- Multi-select filters: I had some trouble implementing the page refresh after selecting different options in the multi-select filters, but I solved this issue using session states and rerun statements.

### 3. Streamlit

I started creating the dashboard in Tableau, but I struggled to produce even one simple visualisation. I had never worked with Tableau before, so it would probably have taken me some time to familiarise myself with the tools and learn how to prepare the data so that the plots worked instantly. However, since I was short on time, I decided to switch to Streamlit, which I had already used in the Cloud Computing course.

#### *Best Aspects*

- Familiar Python environment: Using Pandas for data manipulation and Plotly for visualisations saved me from having to learn how to use the tools.
- Built-in caching was helpful, as I only had to load the data once per session, which would have been annoying if it had happened every time I made a minor change.
- The dashboard automatically reruns when saved, which is also quite convenient.

#### *Worst Aspects*

- Limited layout control: Reaching the desired alignment required workarounds, such as adding columns solely for whitespace, which made creating a proper layout somewhat frustrating. That's why I stuck to some basic page splits.
- No built-in cross-filtering between visualisations, as in Tableau, which is definitely an appealing feature and one of the reasons why I initially wanted to create the dashboard in Tableau.
- Styling complexity: Even simple customisations sometimes required a lot of manual work and research to determine which Plotly parameter combinations would achieve the desired result.

### 4. Future Improvements

- Cross-filtering: As already mentioned, I would have liked to use the built-in cross-filtering tool between plots in Tableau. With enough time, that might be achievable in Streamlit as well.
- Map Plot Design: The map plot as it is right now is probably not valuable for members of the Port Authority, since they likely know where the airports are. Improving this plot by adding more information, as I already tried, is something I would do if I had more time.
- Migrating to Tableau: I would also consider migrating everything to Tableau to see what works better and what works worse there.