

# Project Report

## Changes from Original Proposal

Our final application stayed mostly consistent with our project proposal. We aimed to display information about flights, their delays, and predict future delays based on airline, airport, and weather data. One notable change, however, was that we did not end up creating interactive visualizations for the delays, which we had originally planned to include. Instead, we focused more on building out a strong search functionality and user alert system.

## Achievements and Limitations of Usefulness

We believe our application achieved several important goals:

- **Search functionality:** Users can search for flights and check if they were delayed.
- **Average delays:** We calculated average delays for airlines and airports and categorized them for easier understanding.
- **User alerts:** Through CRUD operations, users can create, update, and delete alerts based on flight information.
- **Weather impact analysis:** Our advanced queries assess how weather conditions influence flight delays and airline performance.
- **Bar graph:** We added a graph which displayed the average delay time of all airlines.

However, we were not able to implement personalized emails and SMS alerts for users, which would have made the alert system more proactive and user-friendly.

## Changes in Data Schema or Source

We stuck with the original data sources and schema outlined in our project proposal. No major changes were made in this regard.

## Changes to ER Diagram and Table Implementations

One adjustment we made to our original ER diagram and table design was removing the state column from the Airport table. We decided to do this because the datasets we pulled from had conflicting information about states. Instead, each airport is now clearly identified by its city and unique airport code. This made the information cleaner and more reliable for users.

Overall, the final design is more streamlined and better suited for our application's needs.

## Added or Removed Functionalities

### Added:

- A search functionality that allows users to search for flights, airports, and airlines.
- CRUD operations that enable users to set, update, and delete alerts.

### Removed:

- Interactive visualizations.

We focused on building core features that made the app functional and helpful.

## How Our Advanced Database Programs Complement the Application

Our advanced queries significantly enhance the application's capabilities. They allow users to:

- Analyze the effectiveness of their user alerts.
- View how flight delays are affected by different weather conditions and times of day.
- Understand how different airlines perform under various weather scenarios.

This information provides users with more insight into potential flight delays beyond just historical averages.

## Technical Challenges Faced by Each Team Member

- **Anitej:** Loading large datasets into the workbench was time-consuming and required careful mapping of dataset columns to database columns. We also had to manage our GCP credits efficiently by not leaving resources running unnecessarily.
- **Tanmay:** Writing advanced SQL queries that were both functional and relevant to our application goals was challenging. It required multiple iterations to meet both the rubric's advanced query requirements and the real-world needs of our app.
- **Naina:** Implementing CRUD operations on the website was difficult because actions like inserting, updating, and deleting worked in the workbench but not always on the site. Understanding how to use React and proper GET/POST requests helped solve these issues.
- **Bhavika:** Connecting the website backend to the database was tough, especially figuring out GCP access. Managing GCP instance uptime to save credits added another layer of complexity.

## Other Changes Compared to Original Proposal

Besides the removal of interactive visualizations and the addition of the search functionality, most aspects of the final application matched our initial plans. The focus shifted slightly more towards building a practical and informative user experience.

## Future Work

There are several ways the application could be improved in the future:

- **Flight cancellations:** If a flight gets canceled, the app could suggest alternative routes, including connecting flights or later departures.
- **Better alerts:** Implement personalized email or SMS notifications to alert users when their flights are delayed or canceled.

- **More detailed weather analysis:** Provide real-time weather updates related to specific flights.

## Final Division of Labor and Teamwork Reflection

- **Anitej:** Gathered and loaded datasets, created two advanced queries related to weather impacts, and worked on transactions, triggers, and stored procedures.
- **Tanmay:** Designed the initial database schema and tables, created two advanced queries, and helped connect database queries and CRUD operations to the frontend.
- **Naina:** Wrote the project proposal and database design document, worked on connecting the database to the frontend, and developed the search functionality.
- **Bhavika:** Focused on linking the backend with the frontend, ensuring users could input and search flight-related queries easily.

Overall, our team worked really well together. We divided tasks based on our strengths and consistently communicated to stay on track. Each member contributed significantly, and we were able to adapt quickly whenever we ran into obstacles.