

Final Report

1. **Please list out changes in the directions of your project if the final project is different from your original proposal (based on your stage 1 proposal submission).**

A change that we made from our original proposal is the Delay Predictor functionality. Originally, we thought that we would implement a function that takes in the data we collected and display graphs to identify the chance of a specific flight being delayed based on past history. However, in our final application, we did not implement this predictor functionality and did not produce any graphs. The users can still filter and look at past data, but we have not yet implemented the ability to predict whether future flights would be delayed or not.

2. **Discuss what you think your application achieved or failed to achieve regarding its usefulness.**

Our application is indeed very useful to the users because it allows them to filter the data in terms of the details of their flight, and provides a list of previous flights that have been delayed. It provides a good gauge of whether or not the flight they are planning to take will be delayed. Our application failed to achieve the exact prediction ability of whether or not a flight will be delayed. Our original approach was to use machine learning to predict the percentage of a flight being delayed and displaying graphs to illustrate it. However, our application did not have this functionality.

3. **Discuss if you change the schema or source of the data for your application.**

In our project, we modified the schema of the airport by adding an additional attribute, NumberOfDelayedFlight, which records the number of delayed flights for each airport. This change was made to reduce the cost of searching for the number of delayed flights, and was implemented through a trigger in the system that automatically and regularly updates the attribute based on any changes made to the flight table.

4. **Discuss what you change to your ER diagram and/or your table implementations. What are some differences between the original design and the final design? Why? What do you think is a more suitable design?**

We did not change our ER diagram or table implementations because we thought that our original design was most suitable for the approach we were going for in this project.

5. Discuss what functionalities you added or removed. Why?

- a. Data visualization, **not added**:
 - i. The project's scope was limited to providing answers to text-based questions, not data visualization. To add data visualization, it requires more data and more time.
 - ii. Implementing data visualization can be complex and require specialized expertise in UI/UX design, data analysis, and front-end development, which we did not have in our scope
- b. Delay prediction based on past data, **not added**:
 - i. The CSV files did not have access to a significant amount of past data to base predictions on.
 - ii. Prediction models are not always correct, and in the scope of this project it may seem safe to create, but in a real world application, it is not safe to calculate delays off a CSV file.
- c. Authentication and authorization, **added**:
 - i. While we did add different accounts to log in, Implementing user authentication and authorization in different instances without the need for logging in can be complex and require significant additional resources and time.
 - ii. Adding user authentication and authorization would require additional security measures to protect user data and prevent unauthorized access.

6. Explain how you think your advanced database programs complement your application.

SQL queries helped us a lot. It was a good way to see if our database was able to generate the data we wanted. The implementation of SQL queries allowed the team to test and refine the database in the early stages of the project to ensure that it could generate the required data sets.

As the project progressed, the team was able to leverage this knowledge and apply it to the application itself, integrating the results of the SQL queries into the application's user interface. This allowed users to easily access the information they needed and quickly retrieve the relevant data from the database.

In essence, the use of advanced database programs like SQL complemented the application by providing a reliable and efficient means of generating and accessing the data that the application relied on. By leveraging the power of these programs, the team was able to create a more robust and user-friendly application that met the needs of its users.

7. Each team member should describe one technical challenge that the team encountered. This should be sufficiently detailed such that another future team could use this as helpful advice if they were to start a similar project or where to maintain your project.

- a. Acchindra Thev: Linking the Frontend to the backend. While it took effort to plan the backend and front end separately, it was a bit difficult when trying to hook them up together. One advice is to make sure that when designing the backend and frontend, do them simultaneously instead of independently.
- b. Rahul Kasibhatla: Cleaning the dataset. It took us especially long to clean up the two datasets we were using. We used two separate datasets, one was the Flights dataset, and the other was the Cities dataset. These two datasets were hard to clean and merge together because there were some airports that did not have a corresponding city in the US, and some cities were not present in the flights dataset. I advise to select datasets with similar entries so that cleaning the datasets is not a tedious task.
- c. James Lu: Writing code synchronously. It was tough when all members of the team were working on the same part. As one member of the team made changes, it was hard for the rest of the team to follow up. One of the suggestions I would give is to divide up the code into different parts with each team member.
- d. Yiteng Zhang: The stored procedure was hard to debug, despite using MySQL workbench to implement it. It seems that the mysql compiler cannot locate the bug accurately. Sometimes, the message it alerts is not the real reason that causes the bug. Plus, we were asked to implement a complicated function in the project with stored procedures in which it's easier to encounter bugs.

8. Are there other things that changed comparing the final application with the original proposal?

- a. Mainly the data visualization tools, the security of the application with the login protocols requiring the user to log in on a different platform and delay prediction models. Alternatively, we did the extra credit but these are the biggest differences from our final application and the original proposal.

9. Describe future work that you think, other than the interface, that the application can improve on.

Though users can easily look up flight data through our application, we can further improve the application by generating personalized flight recommendations based on the user's past experiences. We could also integrate datasets for nearby hotels, car rentals, food, etc to provide the user with a better all around experience. We can also promote the connections between travelers, and help them to make friends along the way.

10. Describe the final division of labor and how well you managed teamwork.

In the designing stages, we discussed together about how the system should be designed and which schemas we can extract from the database.

In the development stages, the workload was divided into front end and back end. 3 team members took charge of the design and development of the front end, while 1 team member took charge of the design and development of the back end, as well as the integration and deployment.

In the final demo stage, everyone participated in the demo and wrote part of the report.

Below is the detailed contribution of each stage:

Stage	Member	Task
1 ~ 3	Yiteng Zhang	Design the system based on the database; Draw the ER diagram; Write reports for each stage;
	Rahul Kasibhatla	Design the system based on the database; Write reports for each stage;
	James Lu	<i>Haven't joined our group then</i>
	Acchindra Uday Thev	Design the system based on the database; Write reports for each stage;

4 ~ 5	Yiteng Zhang	Clean dataset and import it into the database; Design and implement Triggers and Stored Procedure; Design and implement the back end; Integrate front end and back end; Deploy database and server on GCP;
	Rahul Kasibhatla	Clean dataset and import it into the database; Design and implement part of the front end; Help deploy database on GCP;
	James Lu	Design and implement part of the front end;
	Acchindra Uday Thev	-
6	Yiteng Zhang	Write report; Make demo video;
	Rahul Kasibhatla	Write report;
	James Lu	Write report;
	Acchindra Uday Thev	Write report;