

Dhruv
Charlie
Ebuka

1.

Compared to our stage 1 proposal submission, our final project was not drastically different. We did make some small changes however. We decided not to implement a function for any type of cancellation as we did not think it would be very relevant to the goal we were trying to achieve. We also did not see a strong way to get the data relevant for calculating cancellation data. We also did not implement a complete sign in feature, but rather had a functionality for username and a user log which could later be converted to have an authentication system. The overall direction of our project still stayed within the realm of flight data and giving relevant information to users.

2.

The users of the application are indeed able to see flights along with adding certain airports to their favorites. From there they can mark whether or not they have visited the airport, which provides additional functionality. In terms of usefulness, we were also able to provide users with other information such as popular flights during the holiday season or simply which flights flew an average more than others. We also provided an easy to use front end interface that anyone can use. We did also fail to meet a few things, such as using authentication for users logging in or having a cancellation rate table for users, but both these things can be worked on and added in easily in the future.

3.

We stuck with the same source of data with our flight data. We did however originally have our SQL database on a gcp connection, but then had to move the entire database to a different gcp connection as our original one got banned. We also added a few tables that we did not originally anticipate. We already had the idea of the user table, but we added the user log table as well to keep track of how many times a user logged in. We were not originally sure about which front end to use, but we did end up using react.

4.

In our ER diagram we decided to implement user sign-on, and therefore the user table without a user password. We decided to use this design instead given the time frame of the project. We only wanted to add user passwords if we could have OAuth authentication, which we didn't see as a worthwhile add on given the time frame of the project, but is definitely something we would incorporate in the future. We also decided that there was no need for a month table in our final design. We had this in the ER diagram as we originally thought it would be useful to users to know the most traveled months of the year, but later decided it didn't add as much valuable information to the application as we originally anticipated. We decided to include a popular states table

though, which was not in the ER diagram. We decided this table provided a lot of valuable information to users as it could give popular destination information which may give users travel suggestions. Finally, we decided to add a user log table into the final design. This wasn't in the original design as we didn't come up with the idea until about half way through the project. This table provides both users, and the application's owners valuable user activity information.

5.

We decided to not have a home feature given the time frame of the project and the limited features currently available for the application. As more features are implemented and the application becomes more complex, we believe a home option would provide useful navigation functionality that would not be as helpful in the early stages of development. We decided to not have the functionality of viewing specific flights in the application. This was because we decided seeing previous flights to various destinations provided no additional value to users, when trying to seek future destinations to travel. We added a visited column to our favorite airports table, allowing users the functionality of tracking which destinations on their favorites table they have visited. We decided this would be helpful to users so they can track where they've been, and where they have left to visit. Also, as previously mentioned in prompt 4, we decided to add users the functionality of viewing sign-on data pertaining to how many times each user has logged into their account. This is useful to see, as we want to move this application to a more social media inspired platform, and this could provide users fun information regarding how active their friends are on the app.

6.

Our trigger complements our program very well. It allows us to document how many times a user has logged into their account, which provides useful information to users, the user's friends, and the business. I also believe that our procedure provides great information. Our procedure creates a table of the most popular destination states. This is helpful to users as it can provide travel suggestions on the most popular tourist destinations.

7.

Ebuka: Setting up the front end was a big challenge. Learning react native and creating a good UI was a challenge that involved learning a new framework, quickly learning its quirks and learning asynchronous programming in this framework. I had worked with javascript before so that made life easier but picking up ideas like state and use effect was new skills i was able to pick up.

Charlie: When trying to implement the stored procedure and trigger, we could not run the code in the gcp cloud console. It took a long time to figure out but it was a quick fix. I just had to change the delimiter in the console and the entire create procedure function was run. Another challenge was connecting the backend to the gcp database. The app would connect to the database in one location, but not another. We realized that this was due to our IP addresses changing depending on our location, and us not including these different IP addresses in the allowables for the GCP instance.

Dhruv: One big technical challenge we had was maintaining our GCP server. This was due to the fact that our first instance actually got banned, as mentioned earlier. This meant that we essentially had to recreate our SQL database from scratch on GCP, which did end up taking some time, especially because we had to redo work which was already done. One big piece of advice regarding this technical challenge would be to simply stay updated on the status of what you are working on, as it could definitely save some time in the future and help maintain progress. We also had to put in some time to learn how to fully connect our front end and back end as that took some time. Advice for that would be to attentively read relevant documentation and watch up to date videos.

8.

Most things did not change. In the end we decided to not add full authentication since this was a trial and error application. We also added one or two more tables. One to log user activity and one that is created to show popular airports with a procedure call

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

One thing we want to add is a full user auth using oath or maybe storing our own passwords. The risk of storing your own passwords is high. The next thing is to add some UI functionality for the popular states.

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

Ebuka worked on the front end while Charlie worked on the backend and Dhruv worked with the database. We managed teamwork very well as we would meet as a group

weekly and see what had to be done and how we could work together to speed up the process as a whole.