Kanchan Krishna
Nisha Ramanathan
Rishit Chatterjee
Anish Pokharkar

Stage 1 Project Description

What are the worst times to fly in order to avoid flight delays or cancellations?

1. Describe what data is stored in the database. (Where is the data from, and what attributes and information would be stored?)

The dataset is called "2015 Flight Delays and Cancellations" from the Kaggle website. There are 3 CSVs in the dataset: airlines.csv, airports.csv, and flights.csv. Each of these files has several columns. Airlines.csv contains IATA_Code and Airline. Airports.csv contains the IATA_Code, Airport, City, State, Country, Latitude, and Longitude. Flights.csv contains Year, Month, Day, Day_Of_Week, Airline, Flight_Number, Tail_Number, Origin_Airport, Destination_Airport, Scheduled_Departure, Departure_Delay, Taxi_Off, Wheels_Off, Scheduled_Time, Elapsed_Time, Air_Time, Distance, Wheels_On, Taxi_In, Scheduled_Arrival, Arrival_Time, Arrival_Delay, Diverted, Cancelled, and Cancellation_Reason. The attributes and information that will mainly be used for our queries include IATA_Code, Airline, Flight_Number, Year, Month, Day, Day_Of_Week, Arrival_Delay, and Cancelled.

The tables will be linked using columns IATA_CODE from airlines.csv and AIRLINE from flights.csv along with IATA_CODE from airports.csv and ORIGIN_AIRPORT from flights.csv.

2. What are the basic functions of your web application? (What can users of this website do? Which simple and complex features are there?)

The basic function of our web application is to allow the user to create flight reservations by being able to identify the worst times to fly in order to avoid flight delays or cancellations based on day (day number - for example, the 5th), month, or day of the week (for example, Tuesday). The simple features include allowing a user to query what times have the most flights delayed or cancelled based on a certain day, month, or day of the week. The complex features allows a user to insert their own flight delay or cancellation into the database in flights.csv. Users can also add a rating to an airline in airlines.csv to display their own review of their flying experience, therefore benefiting future passengers.

3. What would be a good creative component (function) that can improve the functionality of your application? (What is something cool that you want to include? How are you planning to achieve it?)

A creative component to improve the functionality of our application is allowing users to add their own flight delay or cancellation into the flights.csv table of the database. In order for the user to do so, a series of questions, including airline, flight number, year, month, day, and day of the week, will be presented. As the user enters the information, a new row will be created and inserted into the table. Also, the user will be able to add a rating for each airline into airlines.csv. This will be accomplished by allowing a user to select an option to add a rating. If they choose to add an airline rating, they will be presented with an option from 1 - 5 stars. The rating will be inserted (and averaged with all previous ratings for that airline) and the airline rating will be updated. We can take ratings as either integers or strings to allow a better user experience. Note that we can convert between the two; for example, if a user enters “Excellent experience,” this string is equivalent to a 5 on the numeric scale. Hence, users will have the option to either enter an integer or string description of their experience from a dropdown, and we will populate the row accordingly by converting the string description to an integer. We will do this by mapping the string answer (options provided to the user in a dropdown) to an integer as mentioned in the example.

4. Project Title

Worst Times to Fly in Terms of Day, Month, Day of Week

5. Project Summary: It should be a 1-2 paragraph description of what your project is.

Our project allows users to identify the best and worst times to fly in order to avoid flight delays or cancellations. We are building a web application that allows users to query flight data based on specific criteria such as day, month, or day of the week. The data comes from a reliable source, the U.S. Department of Transportation's (DOT) Bureau of Transportation Statistics, ensuring that users get the most accurate information; therefore, users can identify patterns or trends in flight delays and cancellations to make more informed travel decisions in regards to what timings they should or should not be traveling. Additionally, users can add to the database by adding their own flight delay or cancellation experiences and rating airlines based on their flying experiences.

6. Description of an application of your choice. State as clearly as possible what you want to do. What problem do you want to solve, etc.?

Our project is a flight tracker application that checks flight data across the provided dataset and provides to the user the lists of times (dates, months, day of flight, etc.) that have the highest reported flight delays and cancellations in order to know what times to avoid in order to avoid flight cancellation. The problem we're aiming to solve is avoiding flight delays and/or cancellations, and we aim to allow a user to plan their journey and bookings around days that are reported to have the highest amounts of delays and cancellations.

7. Usefulness. Explain as clearly as possible why your chosen application is useful. Make sure to answer the following questions: Are there any similar websites/applications out there? If so, what are they, and how is yours different?

Our chosen application is useful because it allows a user to leverage real flight cancellation and delay data in order to allow a user to choose the statistical best possible flight dates in order to avoid any disruptions in their travel plans (due to their flights). FlightAware is a similar website to our purpose, but they only show a user live cancellation data. Further Our website collates that data in order to summarize to the user what dates/flights a user should avoid to avoid delays and cancellations, and also provides users the option of rating a particular airline for possible reliability for future users.

Another website that performs a similar function is FlightView, but they face the same irrelevance to collated data by only providing data on specific tracked flights. While it makes use of airport delay data, and shows that across the US, it differs from our application in that we collate data on flight delays and cancellations and provide to the user a list of possible dates and times to avoid at particular airports, rather than showing live possible delays by flight as FlightView does.

8. Realness. Describe what your data is and where you will get it.

Our data is a collection of flight data from 2015 that we received from Kaggle, that describes the airlines, flights, airports, as well as the exact date (by month, day and date) and their related delay and cancellation data. (the link to our data: https://www.kaggle.com/datasets/usdot/flight-delays?resource=download&select=flights.csv)

Further, we also plan on accepting a rating from users on rating particular airlines based on reliability, which is a column of data we will insert into our dataset.

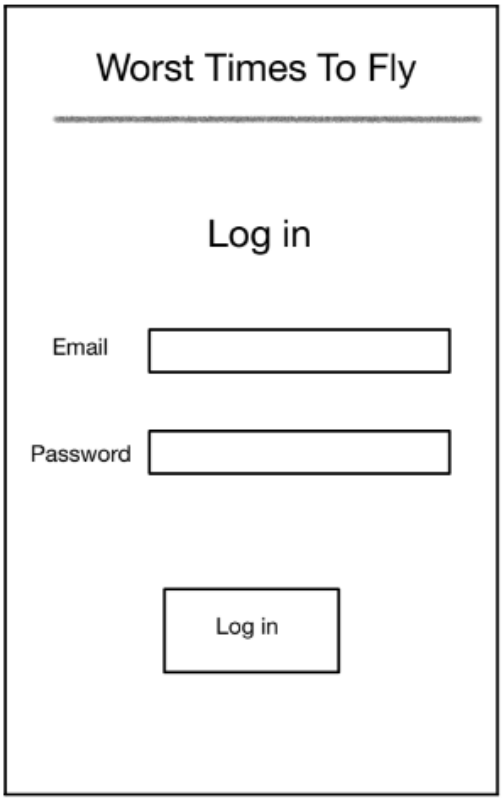
9. Description of the functionality that your website offers. This is where you talk about what the website delivers. Talk about how a user would interact with the application (i.e., things that one could create, delete, update, or search for). Read the requirements for stage 4 to see what other functionalities you want to provide to the users. You should include:

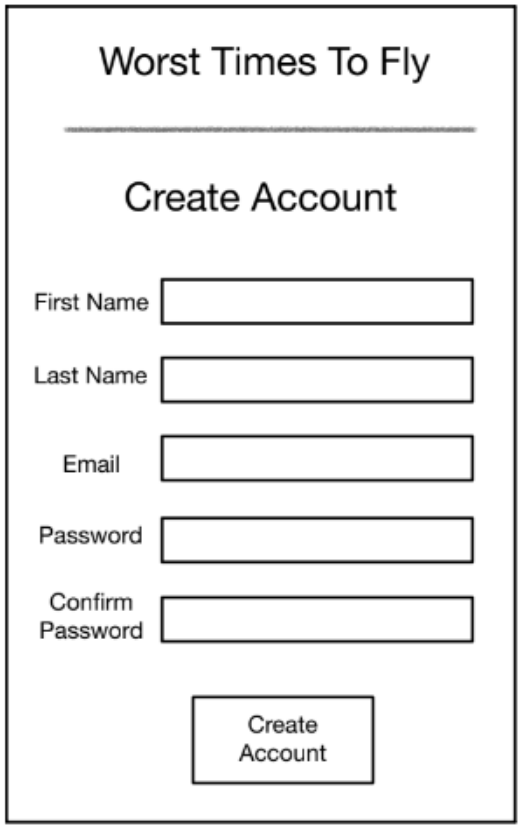
First, the user will be able to **create** an account, log in, log out, and delete their account. This step is necessary to the rest of the processes in the application because the user must be able to save their reservations under their name. Then, the user will be prompted to **create** a flight reservation by selecting their airport, city, state, country, and the dates on which they want to travel for the round trip. For the dates, the user will select a month, day, and year for their first flight & their second/last flight. It is important to note that we are assuming they are going to have 2 flights for their trip, one arriving at their destination and one taking them back home. Also, creating a flight reservation will give them a unique reservation ID.

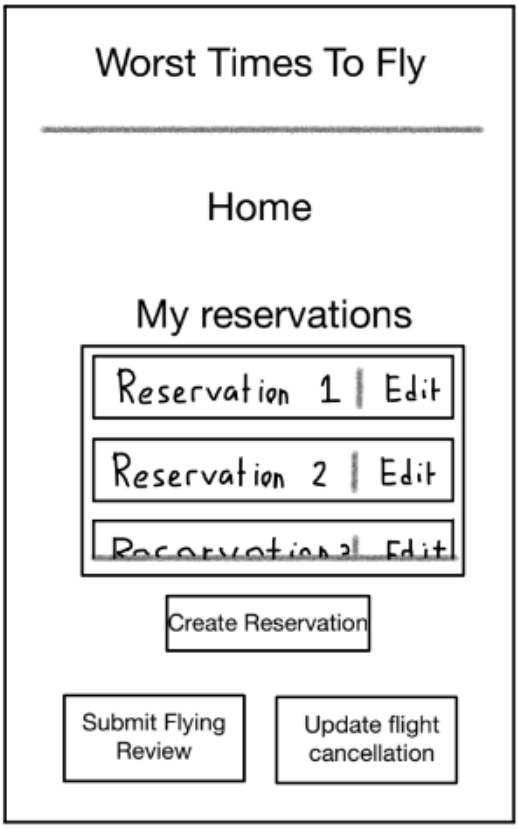
Furthermore, users can **create** a review of their flying experience of the airline and will be able to **update** this as a new column in airlines.csv. This is important because this data will help other users determine on which airline to book their flight. Additionally, users can **update** their flight reservation if they made a mistake or their plans have changed by **searching** for their reservation and updating the fields they want. They would also be able to **update** our own database of flight cancellations by inputting when their cancellation occurred and the additional, necessary fields in flights.csv. In regards to the searching capabilities, users can **search** for their flight reservation by their reservation ID and information regarding the best and worst times to travel after **querying** for the day, month, or week. Lastly, they will be able to **delete** their reservation by entering in their reservation ID and the entire record will be deleted from the relation.

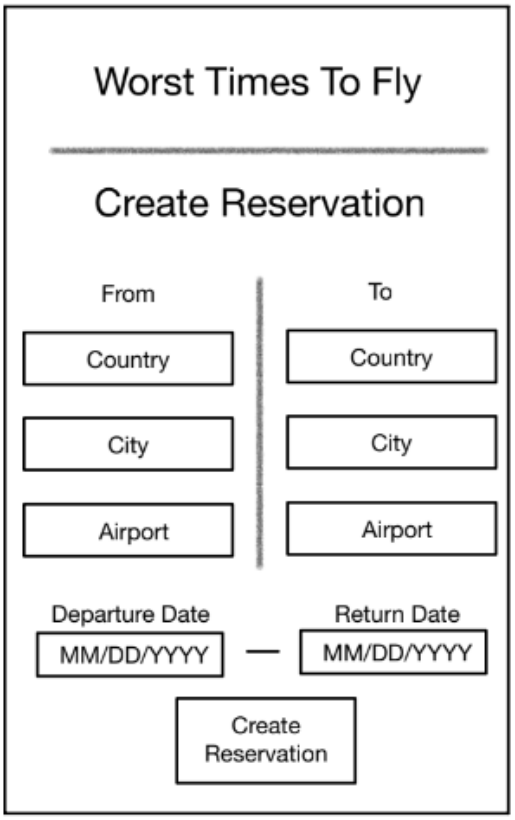
10. A low-fidelity UI mockup: What do you imagine your final application's interface might look like? A PowerPoint slide or a pencil sketch on a piece of paper works!

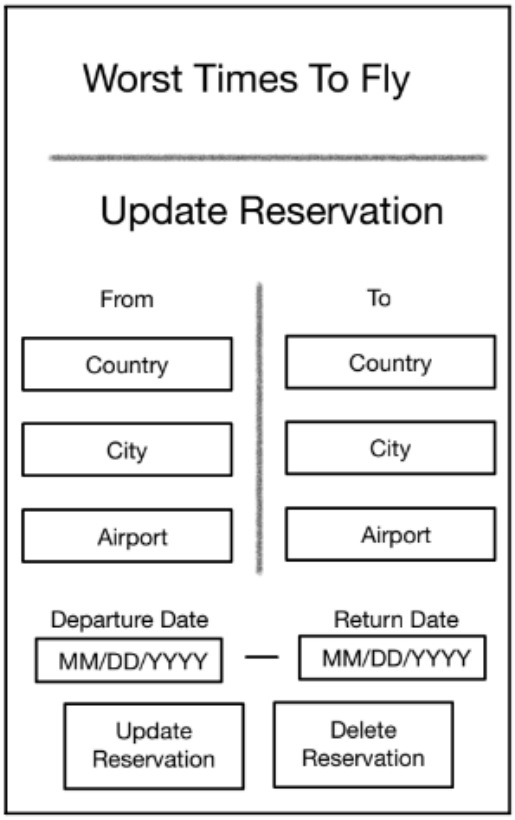

Worst Times To Fly
Welcome
Log in
Create Account


Worst Times To Fly
Log in
Email
Password
Log in


Worst Times To Fly
Create Account
First Name
Last Name
Email
Password
Confirm Password
Create Account


Worst Times To Fly
Home
My reservations
Reservation 1 | Edit
Reservation 2 | Edit
Create Reservation
Submit Flying Review
Update flight cancellation


Worst Times To Fly
Create Reservation
From
To
Country
Country
City
City
Airport
Airport
Departure Date
Return Date
MM/DD/YYYY
—
MM/DD/YYYY
Create Reservation


Worst Times To Fly
Update Reservation
From
To
Country
Country
City
City
Airport
Airport
Departure Date
Return Date
MM/DD/YYYY
—
MM/DD/YYYY
Update Reservation
Delete Reservation

11. Project work distribution: Who would be responsible for each of the tasks or subtasks? List of the person responsible for which exact functionalities in section 6. Explain how backend systems will be distributed across members. Be as specific as possible as this could be part of the final peer evaluation metrics.

a. Frontend (Javascript/HTML/CSS): Maintain the same theme of each page for consistency
   i. Create mockups of each pictorial state of our application **Anish**
   ii. **Page 1:** Welcome/home screen containing buttons to either log in or create account **Kanchan**
   iii. **Page 2:** Login page (button on the welcome home screen that leads to this) **Kanchan**
   iv. **Page 3:** Create account page (button on the welcome home screen that leads to this) -> once successfully done -> leads to log in page **Kanchan**
   v. **Page 4:** Second home page containing buttons to create reservation, update reservation, delete/cancel reservation, create flying experience reviews button, update flight cancellation page, logout button **Kanchan**
      1. **Page 5:** Create reservation page (can come here once you have successfully logged in) **Nisha**

2. **Page 6:** Update reservation page **Nisha**
3. **Page 7:** Delete reservation page **Nisha**

vi. **Page 7:** Create review of flying reviews **Nisha**

vii. **Page 8:** Update flight cancellation page **Anish**

b. Backend (Python, Flask, SQL):

i. Create an account, log in, log out, and delete their account **Kanchan**

ii. Create two different tables

1. User information: username, email, password, phone number, unique user ID **Kanchan**
2. Flight reservations: unique user ID, reservation ID, source airport, destination airport, start date 1 (month, day, year), end date 1 (month, day, year), start date 2 (month, day, year), end date 2 (month, day, year) **Rishit**
3. These two tables are connected via the primary key of unique user ID of user information and the foreign key of unique user ID of flight reservations (no need to assign)

iii. Functionalities:

1. Allow the user to **create** a flight reservation by selecting their airport, city, state, country, and the dates on which they want to travel for the round trip. **Kanchan**
    a. For the dates, the user will select a month, day, and year for their first flight & their second/last flight.
    b. Assign reservation ID to the user's flight reservation
2. Allow users to **create** a review of their flying experience of the airline and will be able to **update** this as a new column in airlines.csv. **Rishit**
3. Allow users to **update** their flight reservation if they made a mistake or their plans have changed by **searching** for their reservation and updating the fields they want. **Rishit**
4. Allow users to **update** our own database of flight cancellations by inputting when their cancellation occurred and the additional, necessary fields in flights.csv. **Rishit**
5. Allow users to **search** for their flight reservation by their reservation ID and information regarding the best and worst times to travel after **querying** for the day, month, or week **Anish**
6. Allow users to **delete** their reservation by entering in their reservation ID and the entire record will be deleted from the relation. **Anish**