FlyNow

Team #40
Armando Benavidez
Juan Carlos Sandoval
Nithil Sethupathi

Project Description

- The aim of this project is to design a Database Application that can hold flight data and passenger information and act as a hub for managing flights. The goal is to give users the ability to book, modify, and cancel flights, as well as viewing their current flights and previous records. Admins will be able to view users' data and modify them as needed. This application will have a database that has tables with account information for users and admins, and also tables for flight information. This information will be available to those with permission to view it and it will be secured so that those without permission can't see it.
- The stakeholders would be mainly airline executives but also possibly aviation colleges. It is important to have the airline executives as stakeholders because they want a product that is tailored for their specific needs so their input is important in designing the project. Another stakeholder could also be an aviation college, where they could study passenger and flight data, additionally a government entity could be a stakeholder such as the Federal Aviation Administration or Department of Transportation, where they could also analyze passenger and flight data it would be important for them to be stakeholders because if they analyze the data they could suggest improvements for a better flying experience.

System Environment
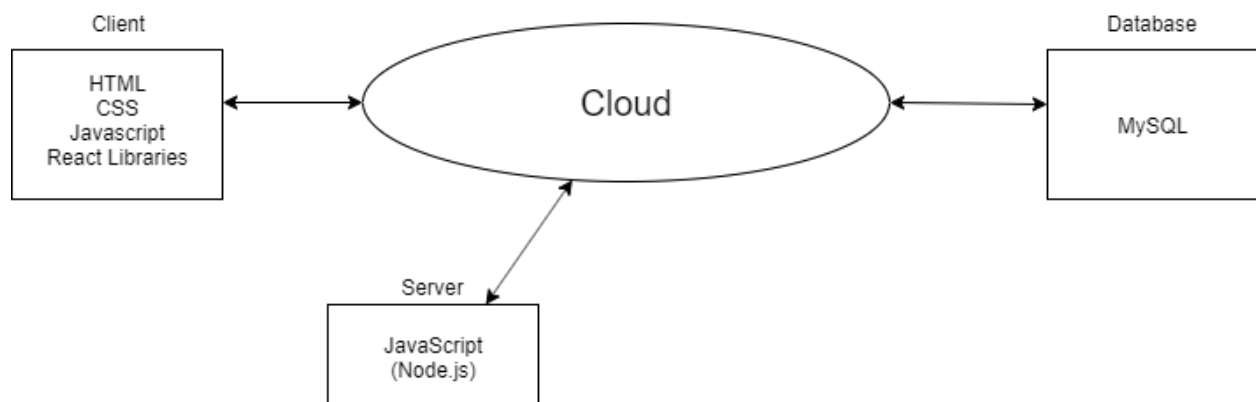
<u>Hardware and Software Used</u>
- MacBook Pro
- macOS
- Visual Studio Text Editor
- Apache

<u>RDBMS</u>
- MySQL 8.0.15

<u>Application Languages</u>
- HTML, CSS, JavaScript, React.js, Node.js, Passport.js (authentication), SQL

Functional Requirements

The web application is designed to be accessed by multiple sets of user types: Consumers, Admins, and partners. All three types of users will have a different set of views and functionalities from each other. Since consumers are the core target, and we need more consumers to get partners, we will focus more on the consumer side of things initially. The difference between consumers and admins/partners will mainly differ in the UI and access to data. Admins and partners will have access to the data so that they can study the taste of consumers, whereas consumers are the data providers. So, the whole application will be developed by prioritizing the need of consumers.

- Sign-up
  - Consumers
    - Consumers can signup using their Email address, first name, last name, and password. This will give them an account of which they will be able to keep track of their booking, save flights, and view previous records.
  - Admins and partners
    - Partners will be given a special login credentials from our website. These types of users will have the ability to create and manage flights for their airlines.
    - Admin will be also able to sign up with email, name, password. This will give them an account that allows them to view flight data, and some user data, with their admin privileges.
- Login
  - Consumers
    - Consumers can log in using their Email address and password that they signed up with. They get access to all of the user functionality,

such as searching for, booking, modifying and canceling flights. They also get access to view their current flights and previous booking history.

- ○ Admin and partners
  - ■ Admin gets to access the complete data and statistics of the website. They are able to see things like flight data, users (but not their passwords), and statistics.
  - ■ Partners can log in to the web app to look into the statistics. Different airline partners can look into the performance of their airline in terms of booking, and where they stand.
- ● Search for flights
  - ○ To search for flights the user will need to specify whether it is a one-way trip or round trip. An origin and destination must be provided with a selected date(s). If an invalid origin or destination is inputted the user will be provided with a message stating that it is not a valid choice. Searching properly will result in a list of available/potential flights for the user.
- ● Book a flight
  - ○ In order for the user to book a flight, the name of the flier(s) will need to be entered, date of birth will need to be provided, along with a valid email address to send out reservation details. If the reservation is successful the user will see a reservation confirmation page with an on-screen message saying "Your reservation is confirmed" or "Reservation was successful", and will send a unique reservation via email. In the event the reservation is not successful there will be a message saying "Reservation Unsuccessful, please try again later" or some sort of error message.
- ● Popular Flights/Destinations
  - ○ Destination popularity will be determined by the average number of times that the destination has been booked per day. An increase in booking average will increase the popularity of the destination.
- ● Save/Delete a flight
  - ○ If the user desires, they can save a flight to their portal, but must first log into their account by selecting save flight when searching for flights. If they wish to book it in the future it will be on their user portal. If the user no longer wants that flight on their portal then they can remove it.
- ● Change/Cancel a flight
  - ○ Upon entering their booking information, the user will be prompted with two choices one is to cancel their flight, second to change the flight(s) by searching for another flight upon entering the new origin/destination or

date/time. This is seen in the User's portal where they can manage existing flights.
- ○ Admins will also have the ability to change a user's flight or cancel it for any valid reason.
- Current/previous bookings
  - ○ The user can log in to their account and go into their records. From here, they can see their currently booked flights or look at their records for past flights.
  - ○ Admins may also be able to see users' currently booked flights or past flights.
- Share your flight details
  - ○ Let your friends or family know your coming. The user who has booked a reservation will be able to share their flight details if they wish to do so by providing an email address for the recipient they wish to send this information to.

Non-functional Issues

For the Graphical User Interface (GUI)
- Likely be using the React library for Javascript to simplify the process of creating a clean-looking and cohesive UI. We may even use a framework such as the Material-UI framework. Material UI is based on google's material design and it will be used to give our application a professional and coherent look.
- The first screen the user is greeted upon would be the main page which would be presented with options represented as buttons which would be to search for a flight, book a flight, change/cancel a flight, my account, and popular destinations. To keep the options distinct each button will have its own color. It is important to have these items on the main screen/landing page because the user is likely wanting to choose one of those options.
- If the user clicks the search for the flight they would be directed to another portal that will ask for various inputs to process the request.
- If the user clicks the book a flight option it will ask the user to enter the passenger information and then to choose the desired flight with date(s).
- If the user clicks change/cancel flight it will prompt for unique code and passenger information and then options to change or cancel the flight(s).
- Selecting my account button it will direct the user to log in page which will require a username and password associated with the account. In my account you will be able to view your current and past trips.

- If the user clicks on the popular destinations button a list of top 3 destinations will appear.

Security and Access

- For security, we will use passport.js authentication for login/signup authentication which will protect the privacy/security of the users.
    - The API is available online for public use and is free of cost.
- We will use different portals for consumers, admin, and partners. The information will be limited depending on the user type. Information provided will be different based on the user type.
- Levels of Access
    - User's portals will be completely distinct with different levels of permissions and will be color-coded.
    - Admin: background will be red, and will have complete access to all records with no restrictions. This account type will also have a protection mechanism after 3 attempts account will become locked for 4 hours and will not require a password reset.
    - User: background will be blue, and will have access to only their own portal/account. If they choose to share their flight details with friends they will not be able to modify their flight only the person flying will be able to, because of the unique code given at confirmation of the successful reservation. For security, users will have only 3 attempts to log in, after all, 3 have been exhausted account will be locked and will require a reset of password.