**Douglas College**

**CSIS 3275-002**

**Project Report**

**Prepared for:**

**Nelson Eng**

**Guillermo Martinez 300250579**

**Kevin Ruan 300245417**

**Monther Balghonaim 300183437**

**Kim Cyrus Esquibel 300186553**

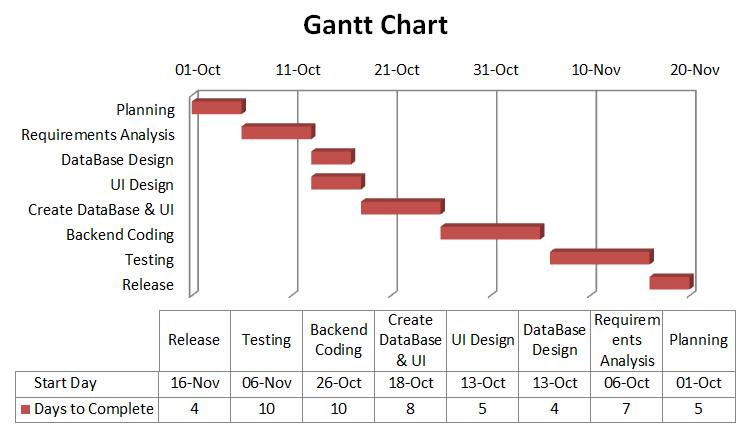
## **Statement of Work**

For our project, we created a web application that tracks and assists in booking flights. The inspiration for this project came from the group’s shared interest in traveling and frustration in how booking flights can be an unnecessary hassle. With this software we aim to create an intuitive, easy to use app that will prevent said hassles. We will not be delivering this software to a real client as this project was created based on our interests alone and, since this is the case, we will be acting as the traveling agency this software was meant for.

        The goal of the application is to create a simple way to view/book flights and allow the best experience possible for users. The app will guide the user through the booking process, asking for input for each step up until they decide on and pay for a flight before the app verifies and stores the information in a database and forwards it to the airline of user’s choice.

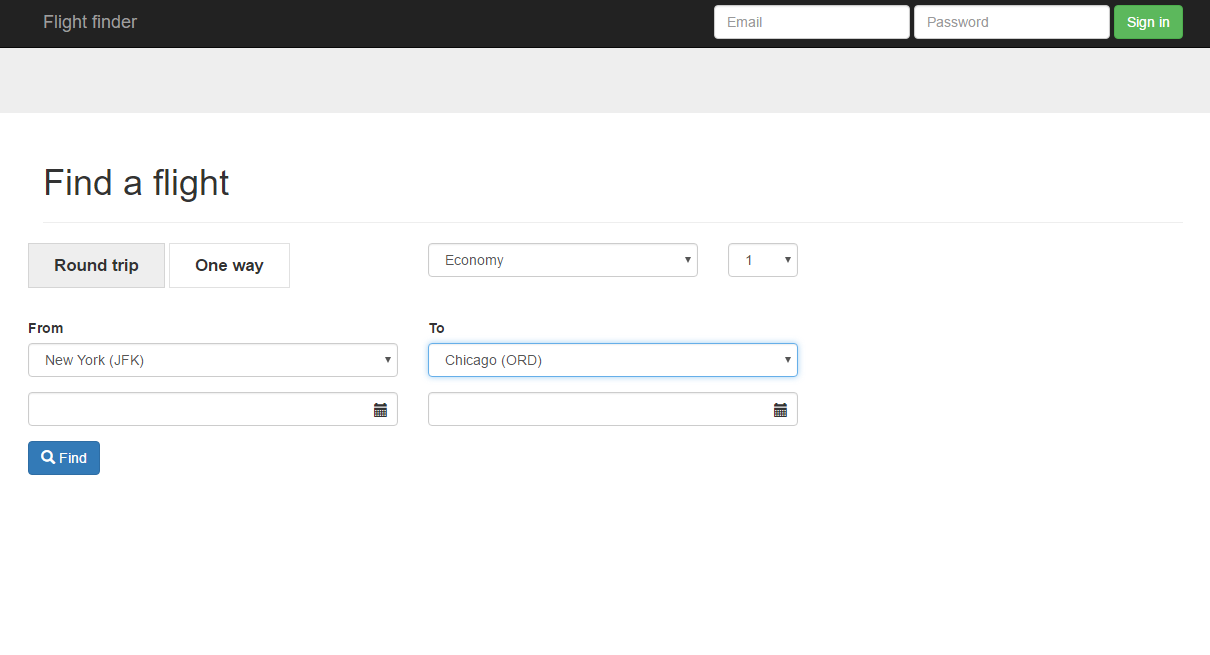
The system is built using PHP+MySQL for the backend and data management, and HTML+JavaScript for the frontend. As the application we built is only for demonstration, we will not include a payment gateway, forward the data to an actual airline, nor live flight data.

Shown below is the schedule we followed for this project:



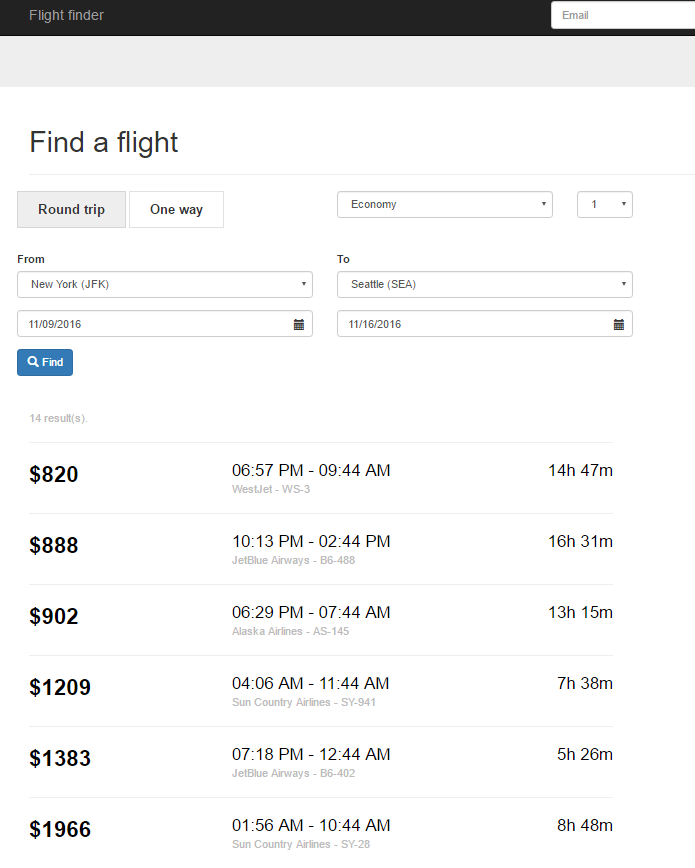
## **Input & Output of System**

For the input of our system, users will see the “Find a Flight” screen.

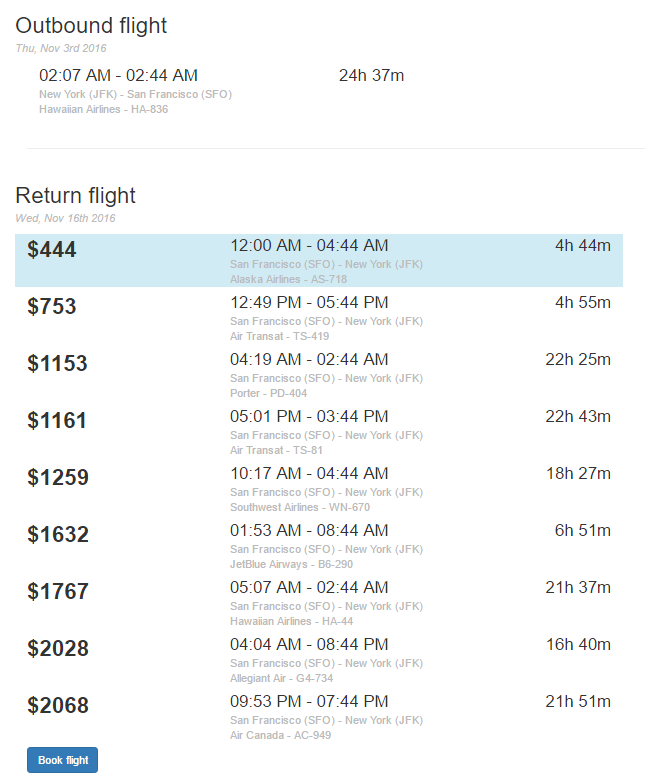


Users will select a departure and the destination point of their trip including the seat class and the number of passengers booking the flight. If the user selects the roundtrip option for their flight, they will also have to select the option of a returning date. After the user enters the data and click the Find button, the data entered will be processed.

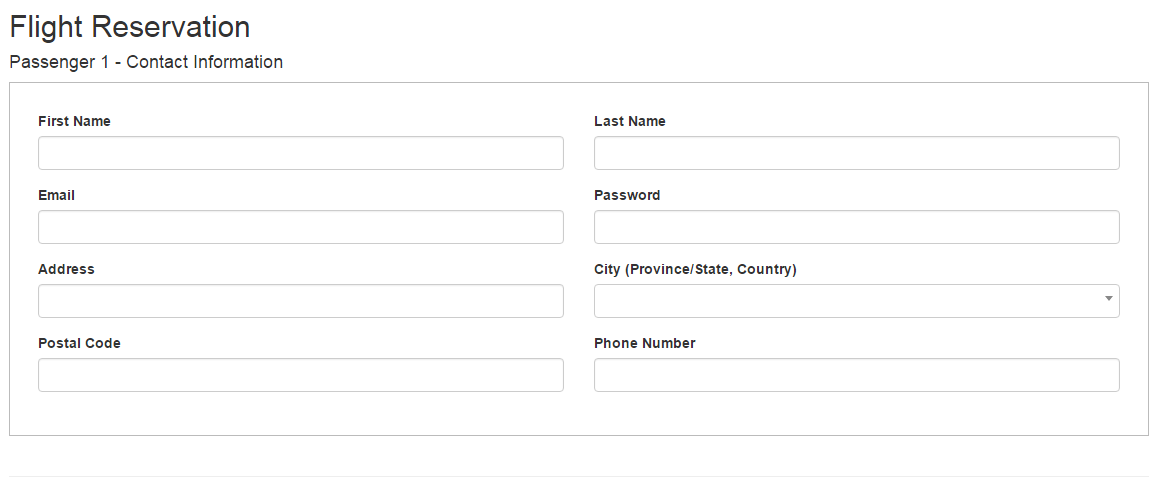
For the output of our system, the user will see a result page of the flights that match the user’s input including the price and the time of the flight.

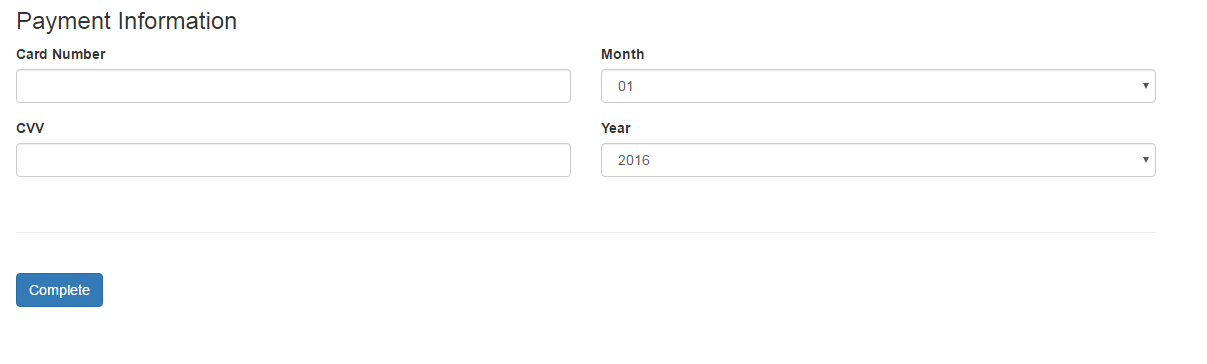


If the user selected the roundtrip option, they will see another result page where they can select a returning flight and proceed with the booking.



After selecting the round trip, the user may move on to the next page where contact information such as first name, last name, email, account password, address, city ,postal code and phone number are taken as input for the creation of an account.

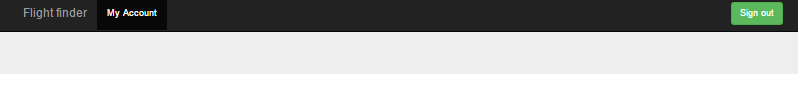


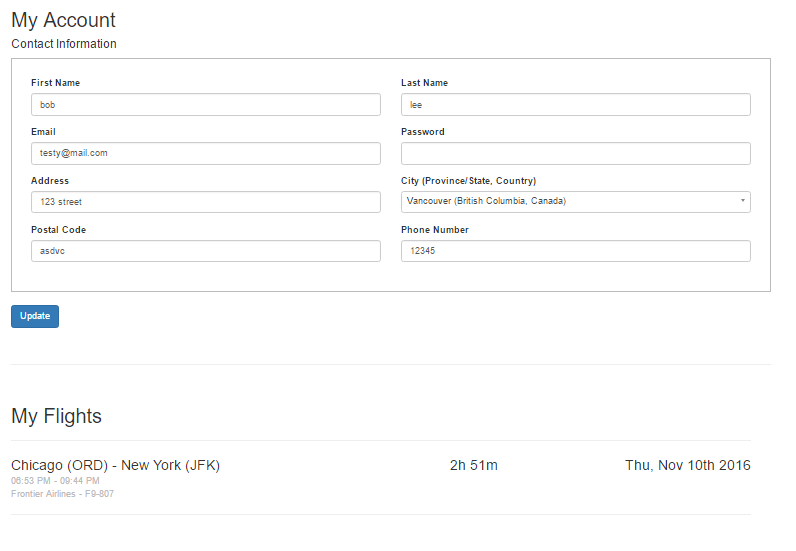
Furthermore, visa or credit card information are also taken as input in order to process the payment. 

## After confirming the reservation and payment a message is presented to the user to thank him for his business and confirm his reservation. The user may log in.



Email and password that were setup previously are inputted, and then the user is transferred to his account where he may update his account information and check his reservation.



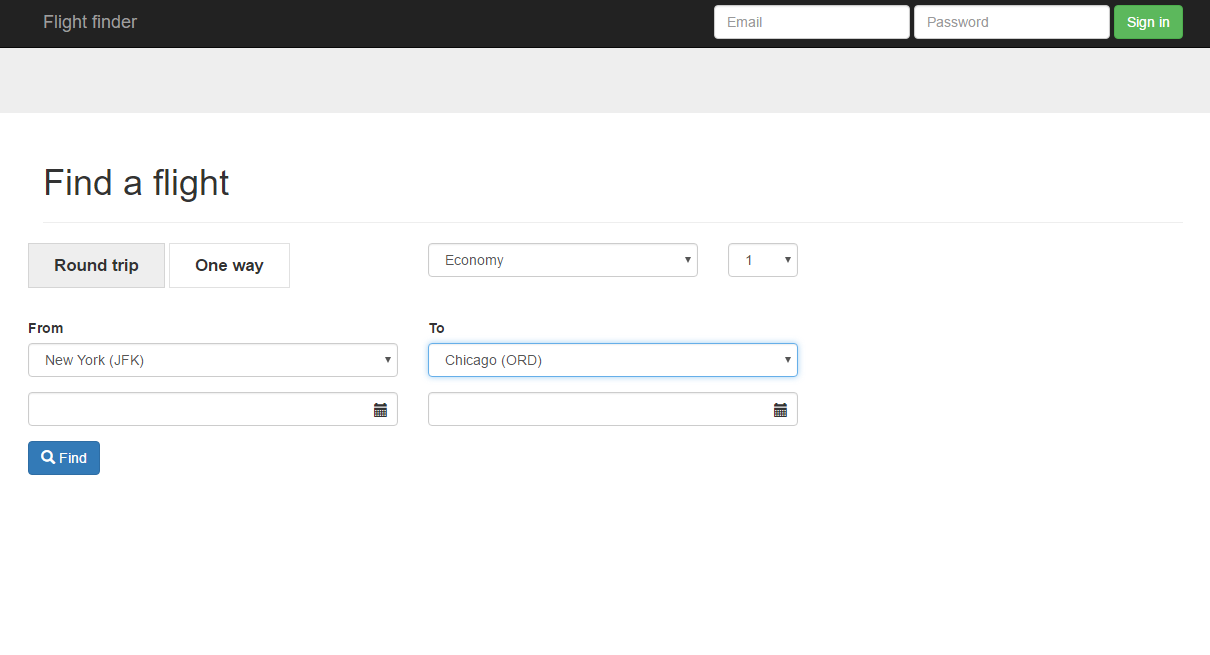


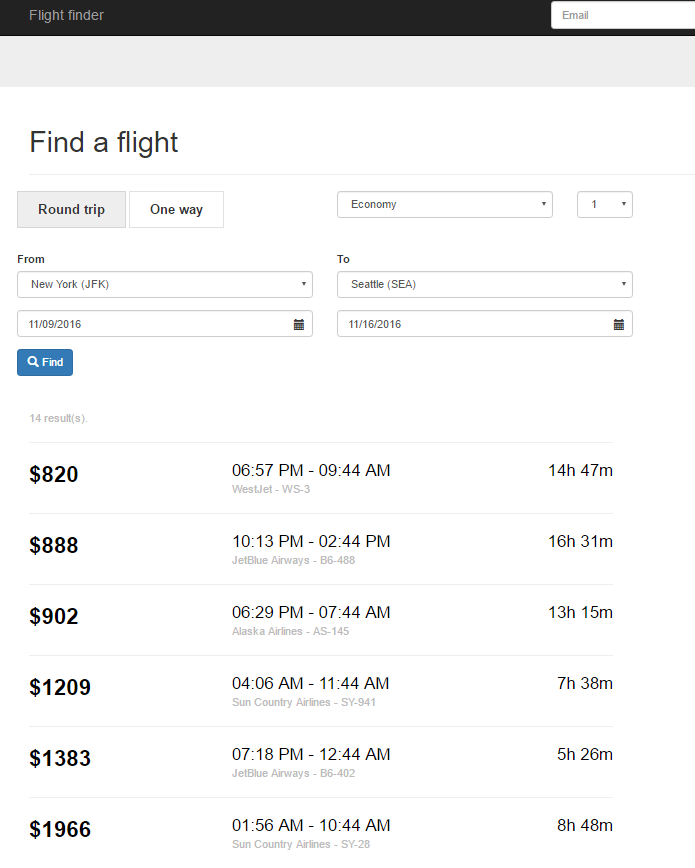
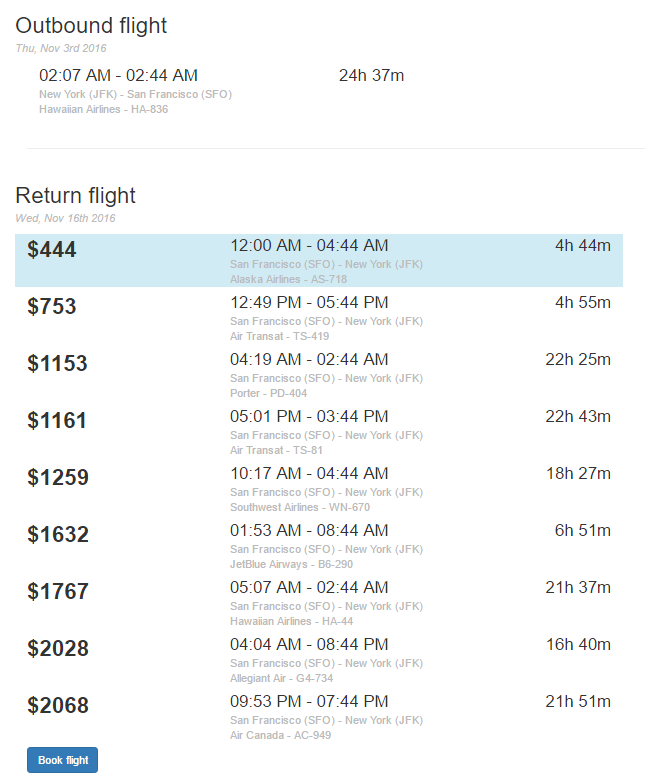
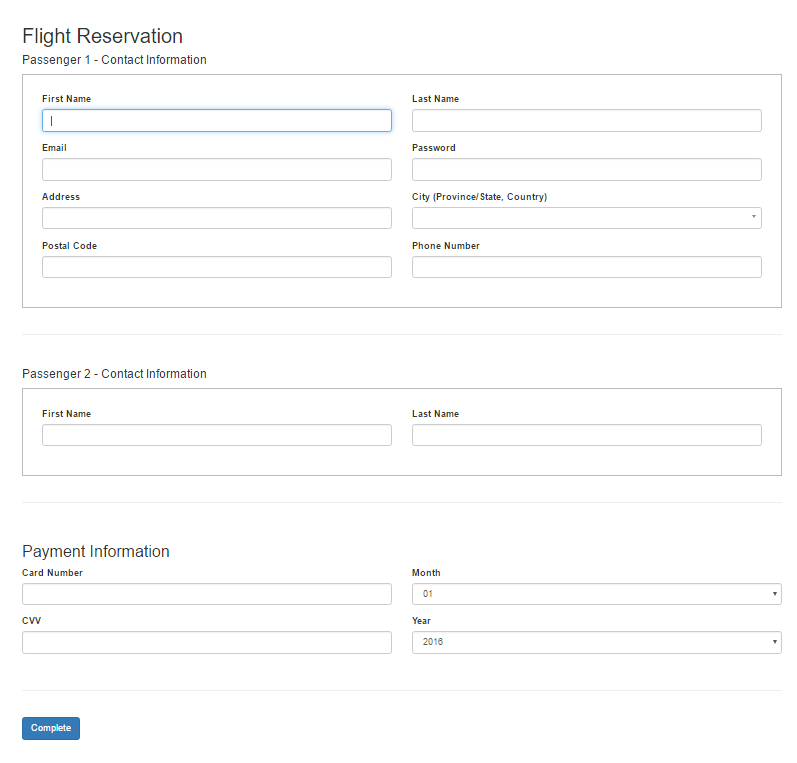
## **Implementation & Backup Plans**

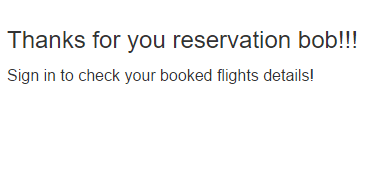
Currently, our application is uploaded on a free hosting site so from time to time there may be problems or the site may even go down. In the event that this may occur, we have all the files required to run the application on GitHub as well as a copy on an external hard drive. Due to not having real users, we do not periodically back up our database, but in the future, ideally we would sync the database with a cloud service periodically. Preferably a period of two weeks would be the optimum amount of time between each update.

## 

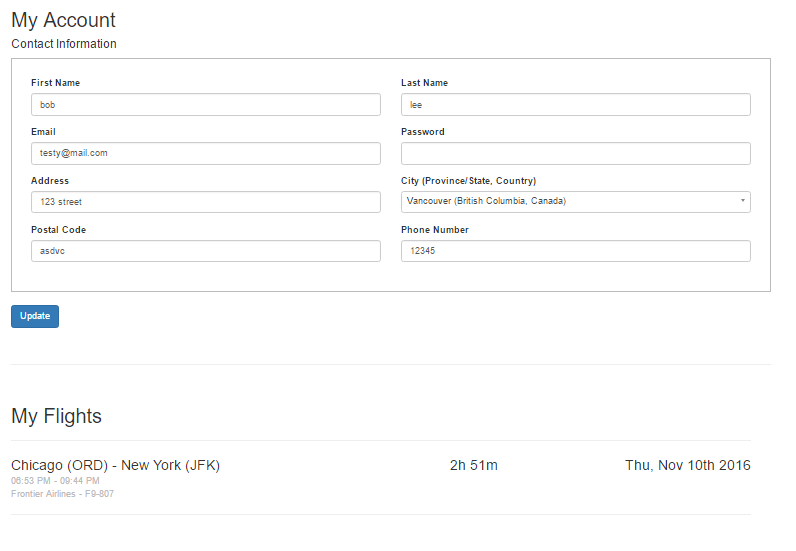
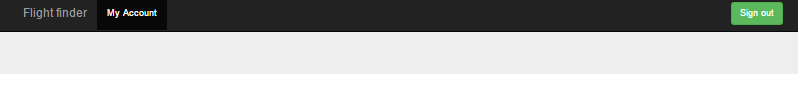
## **Functionalities of Proposed System**

Upon visiting the application ([flightfinder.co.nf](https://www.google.com/url?q=http://flightfinder.co.nf&sa=D&ust=1480313663182000&usg=AFQjCNH4eoQdr5sARmPfSU7s2skHKqye1A)), the user will see this page, where the user inputs their departure and destination point and a departure date, along with how many seats they’d like and number of seats. If user selects round trip, they will also be required to input a second date for the return trip. When the user presses the “find” button, a PHP script validates the input fields to ensure that everything is entered correctly before searching for flights based on the date(s) that the user has chosen.

This is the page that displays the user’s search results. A PHP script searches the database for flights that match the user input before printing the results.The price for a ticket is listed on the left. In the center we have the estimated time period that the flight will take as well as the airline and airplane number. Finally, on the right is the duration of the flight based on the time period of which the flight occurs. If the user selected the “One-Way” option, at the bottom of this page they will see a “Book Flight” button that will bring them to the “customer information” page.If the user has selected the “Round Trip” option, they will be brought to this page that displays information on return flights. After choosing an option, users may continue via the “Book Flight” button at the bottom of the page. The “Book Flight” page brings us to this page where they are required to input their own information. Depending on how many seats the user specified on page one, additional forms will be created according the desired seat numbers asking for names of the passengers. At the bottom we ask users to input their payment information and once they are done they are to hit the “Complete” button, and their inputs are stored into the “customer” table in the database.



If the user fills the form and the php validates the fields, the user whom in this case is “bob”, will be shown this message and can now sign in using their own email and password.



Upon signing in, there is a “My Account” tab at the top of the page, which will lead to a page where they can update their current information or check up on their booked flights. If they wish to cancel a flight, they will have to call the travel agency in order to cancel their flight.

## **Reflections and Suggestions**

**Issues**

The only issue that our team had encountered was communication. We had different schedules and as such is the case, we had many conflicts in time and were unable to meet up as a group. To combat this issue, we decided to use GitHub, emails and Google Docs to communicate and forward work to other members of the team.

**Improvements**

The project can be improved in a multitude of ways, listed below are ones we found would be important.

Encryption: We would need to encrypt the information of our users when we forward it to airlines during the booking process. This is very sensitive information and as such we must ensure the privacy and security of our client’s information.

Pulling live flight data: For the purpose of demonstration, our application currently uses mock data. Pulling real time flight data would be a key component in the functionality of our application. In order to provide accurately provide flight times and calculate expenses.

Forwarding booking data: Our current application stores the user’s reservation into a database, but by not forwarding that information to airlines, our application would simply be a flight search application, so being able to push that information and confirm the reservation with the airline(s) would be a must for the application.

Payment gateway: A payment gateway must be implemented to authorize transactions to ensure that things such deactivated credit cards can’t be used.

Different payment methods: from a business perspective offering different payment methods such as Paypal and KashU may increase our pool of clients.

Administration and reporting system: An administration and reporting system would need to be built in order to manage the application and handle issues. From the administration point of interest, access to databases, flight cancelations and account security classifications would be the main managerial duties. As for the reporting system, business activity tracking such as trends, business flow, and balance sheets might be needed to help in making certain decisions regarding future prospects of business development.

Layout: If we were to sell this application, the layout should be adjusted to fit the buyer’s needs, for example, adding their logo and changing the theme.

User Feedback: Currently, the application is built based on what our team wanted, so to improve user experience, we could create a form for feedback and input from users.

Additional Design options: The buyer may desire to add a promotional section where group coupons or limited time sales for certain flights may be offered.

Customer support line: addition of this feature is a must for the buyer of the application, so that their clients may contact them to cancel or adjust reservations. Which falls under the jurisdiction of the system administrator.

Data Backup: As we have no real users, we do not back up our database, but for future development, we would periodically sync the database with a cloud service and keep an updated copy on an external drive as well.