# A Web Application for Airline and Airport Delay Statistics

by

Rakesh Adloori

A PROJECT

Presented to the faculty of

The Graduate College at University of Nebraska

In Partial Fulfillment of Requirements

For the Degree of Master of Science

Major: Computer Science

Under the Supervision of Professor Vinodchandran Variyam

Lincoln, Nebraska

August, 2016

# A Web Application for Airline and Airport Delay Statistics

Rakesh Adloori, M.S

University of Nebraska, 2016

Advisor: Vinodchandran Variyam

Air transportation is one of the most widely used mode of transportation in US. It is also the fastest and safest way for transporting goods, individuals, etc. A major problem with air transportation is flight delays. It costs billions of dollars for airlines to manage flight delays. Flight delays also negatively impact passengers' travel experience severely. Analyzing the statistics of airlines and airports mainly related to delays is very important to cut the costs of airlines and for the benefit of the passengers. We developed a web application through which statistics related to airports and delays of airlines within US can be visualized. With these visualizations and statistics, one can know how the delays of an airline is affected by various factors such as weather delay, carrier delay, delays between origin and destination airports, and more. In addition, we also allow users to rate and review the airlines based on their performance and services provided by them.

# ACKNOWLEDGMENTS

I would like to thank Professor Vinodchandran Variyam for his guidance, support and encouragement throughout this project. I would also like to thank him for his suggestions for improving the project. I would like to thank Professor Stephen Scott and Professor Hongfeng Yu for being my committee members.

## Table of Contents

## List of Figures

# Chapter 1

## Introduction

Air transportation is a form of transportation which is done in vehicles such as flights, helicopters, blimps, etc. It has been growing at a great speed in the recent decades, it also requires less time to travel between two places in a less time compared to many other forms of transportation. The project that we are going to talk about in this paper mainly depends on the dataset. The dataset used in this project is a publicly available dataset obtained from Bureau of Transportation Statistics. There are two main goals of this project, which include learning PHP framework called laravel and the other is developing a web application for displaying flight and airport statistics related to delays. The web application designed for this project helps to get useful information related to flights and airports from the dataset which is stored in a local MySQL database. Some of the useful information include delay trends of a particular airline for different years, average departure delays of an airline and more.

### 1.1 Motivation

Bureau of Transportation Statistics have many public datasets which are related to Flights, Airports, Airline traffic data and more. It not only has these datasets, but also gives us statistics according to user input. It consists of statistics which is related to Airline On-Time Statistics which is the main motivation for web application project. The project discussed in this paper consists of more statistics with user friendly and advanced visualizations.

## 1.2. Tools

### 1.2.1. Data set

The dataset used for this project is a public dataset from Bureau of Transportation Statistics. The dataset includes data about flights travelling within US in the years of 2006, 2007and 2008. The dataset is about 2.5 GB and is in comma separated values (CSV) format. There are nearly 20 million records in this dataset. The dataset includes the following information.

| | | | |
|---|---|---|---|
| • | Year | - | Year |
| • | Month | - | Month |
| • | DayofMonth | - | Day of month |
| • | DayofWeek | - | Day of week |
| • | DepTime | - | Actual departure time (local time: hhmm) |
| • | CRSDepTime | - | CRS departure time (local time: hhmm) |
| • | ArrTime | - | Actual arrival time (local time: hhmm) |
| • | CRSArrTime | - | CRS arrival time (local time: hhmm) |
| • | UniqueCarrier | - | Unique carrier (flight) code |
| • | FlightNumber | - | Flight number |
| • | TailNumber | - | Tail number |
| • | ActualElapsedTime | - | Elapsed time of flight, in minutes |
| • | CRSElapsedTime | - | CRS elapsed time of flight, in minutes |
| • | AirTime | - | Flight time, in minutes |

- ArrDelay - Time difference in minutes between scheduled and actual arrival time
- DepDelay - Time difference in minutes between scheduled and actual arrival time
- Origin - Origin airport
- Dest - Destination airport
- Distance - Distance between airports (miles)
- TaxiIn - Taxi in time, in minutes
- TaxiOut - Taxi out time, in minutes
- Cancelled - Cancelled flight indicator (1 = cancelled)
- Diverted - Diverted flight indicator (1 = diverted)
- Delays - Cause of delay which include Carrier delay, Weather delay, NAS delay, Security delay and Late aircraft delay

There are other datasets used in this project some of which are obtained using the above dataset and other datasets include Airport dataset (contains airport code, airport name, latitude and longitude), Airline dataset (contains airline code, airline name).

### 1.2.2. MySQL

MySQL is a relational database management system (RDBMS) which is an open source system and is the most widely used RDBMS. It was developed by Oracle Corporation and its initial release was on 23 may, 1995. It is written in C, C++ and works with various operating systems which include Windows, Linux, OS X and more. Many high-profile companies such as Google, Facebook, YouTube use MySQL for storing their data. It is a fast and stable SQL database server.

### 1.2.3. WampServer

WampServer is a software for Windows operating system, which was created by Romain Bourdon. WampServer 2.65 is the stable release which was made on 19 November, 2014. It consists of Apache web server, OpenSSL which support PHP programming and MySQL database integration. WampServer was used in this project to work with MySQL database. Some of the other servers are listed below.

- MAMP - For Mac operating system
- LAMP - For Linux operating system
- XAMPP - A cross-platform web server
- WISA - For Windows operating system
- SAMP - For Solaris operating system

### 1.2.4. BootStrap

Designing front end for a web application from scratch is very time taking process. Most of the developers use frameworks for designing the front end for their web application. Bootstrap is a front end web framework used to design web pages, web application and web sites. It contains all the required CSS for styling and it also contains various components which are styled by default which include buttons, forms, icons and more. It is the most popular and most widely used open source web framework for developing web applications. Bootstrap is used to design interactive and responsive applications and is compatible with browsers like Google Chrome, Safari, Internet Explorer, Mozilla Firefox and more. Bootstrap was used in developing this project.

### 1.2.5. HTML

HTML is short for Hypertext Markup Language used to create web pages. HTML is generally used in combination with CSS to create great user interface and with JavaScript to make the web application more interactive. Tags are the building blocks of HTML pages; contents of HTML pages are enclosed within tags. Some of the basic tags of HTML pages are <HTML> </HTML>, <BODY></BODY>, <FORM></FORM>. JavaScript is encoded in HTML page using <SCRIPT></SCRIPT> tag.

### 1.2.6. CSS

Cascading Style Sheets is commonly abbreviated as CSS, it is used to style or design the web pages. There are three ways of adding styling to web pages, they are through

- External Style Sheet – The CSS file are written separately in .css files and are linked to HTML pages using <link> tag.
- Internal Style Sheet – The CSS code is written directly in the HTML pages using <Style> </Style> tag.
- Inline Style Sheet – The inline styling of web pages is used for styling individual components of HTML pages. It is done using the style attribute.

  Example: <h1 style=” color: green”>Green colored text</h1>

### 1.2.7. D3.js

D3 is short for Data-Driven-Documents which is a JavaScript library used for data visualization. It is used to design interactive and dynamic data visualizations in web applications. It makes use of CSS, HTML5 and SVG. Its initial release was made on 18th February, 2011 which is written in JavaScript. This project consists mainly of data visualizations, some of which use D3.js.

### 1.2.8. ChartJS

ChartJS is a JavaScript framework used for designing charts such as histograms, pie charts, scatter plots and more. It takes data in the form of Arrays, Series, JSON and projects the data as various charts based on programmer’s decision. It also makes use of HTML 5, CSS as D3.js and it is responsive i.e., the charts scales dynamically based on window size. It is an open source application and is maintained by a community.

# Chapter 2

## Background

### 2.1. Web Applications

A web application also known as web app is a software application which is used by users or clients. A web application generally runs on client's browser and use server to do the backend process. Web application have become very popular since its wide spread into different sectors like social networking, health care, education, etc. Today, there are many options for a developer in designing web applications. Some of the popular languages for developing web applications are ASP, JSP/Java, Node.js, PHP, Python, Ruby. Some of these programming languages also have their own popular frameworks, using which web applications can be developed in less time and more securely. Another advantage of developing a web application is that it is often developed using open source software's and frameworks. Most of web applications developed today use MVC architecture.

## 2.2. MVC Architecture

Model-View-Controller (MVC) is most popular architecture used for developing web applications. This architecture divides the code three basic blocks which are model, view and controller. Model is the central component of this architecture which manages data and performs logical operations on the data. A controller is another block which gets data from views and passes to model and vice versa. Views are responsible for the user interface i.e. the front end part of the web applications is maintained by views.

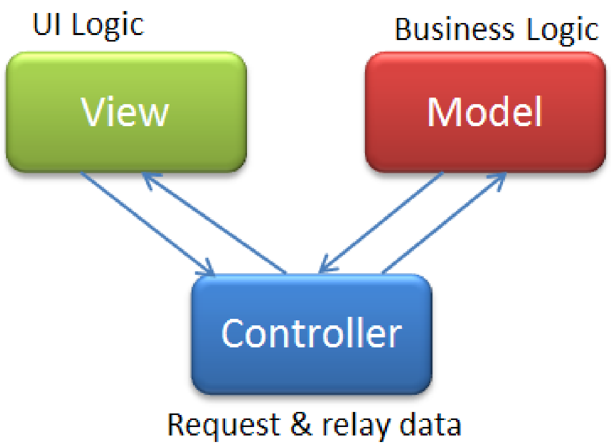


Figure 2.1: MVC architecture 1

## 2.3. Database

A database is a collection of data in an organized manner. Data is very important and has to be stored in an organized way such that it can be useful in the future. Today the size of data is growing exponentially, so we require databases which store large amount of data. Data in a database is stored using software's known as Database Management Systems (DBMS). Some of the DBMS software's include MySQL, Oracle, Microsoft SQL Server, IBM DB2, etc. The data in a database is stored in relational tables. Data in a database is managed using data manipulation statements often called as DML statements. The following are some DML statements

- Insert – Data is inserted into a database using insert command
- Syntax: INSERT into table_name values (data1, data2, ….);
- Update – Data in database can be updated using update command
- Syntax: UPDATE table_name set data1 = new_data where data column_name = some_value;
- Delete – Delete command is used to delete data from database
- Syntax: DELETE from table_name;
- Another useful command which is used is retrieve data from database is SELECT
- Syntax: SELECT column1, column2, column3, …. From table_name;

## 2.4. PHP

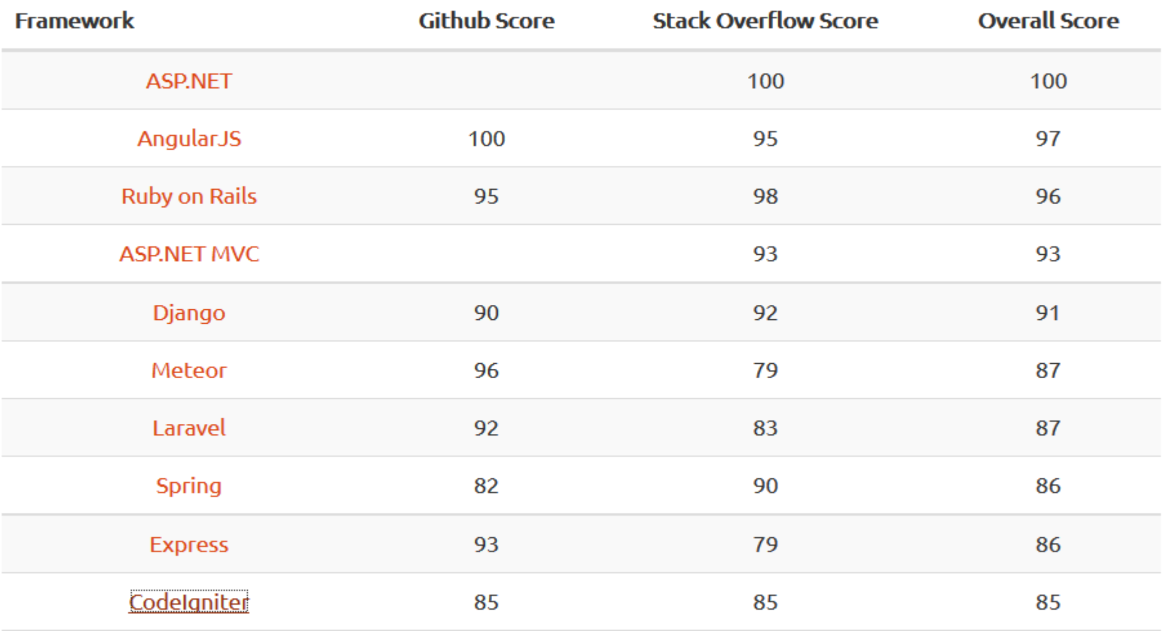
PHP is short for Hypertext Preprocessor. Its development was started in 1995 by Rasmus Lerdorf and was implemented in "C" primarily. Its latest stable release PHP 7.1.0 was on June 23, 2016. PHP is a server side scripting language mainly developed for web development.

## 2.5. Framework

A framework also known as web framework (WF) is a tool to design web applications, web services in a very easy and efficient way. Frameworks provide libraries, security, session management and many more. So the programmer need not worry about them while designing a web application. Some of the advantages of using framework are given below

- Rapid project development
- Open source
- Design pattern
- Built in classes, interfaces and functions
- Security
- Integration

There are different frameworks which depends on the language the web application is implemented.



| Framework | Github Score | Stack Overflow Score | Overall Score |
|---|---|---|---|
| ASP.NET | | 100 | 100 |
| AngularJS | 100 | 95 | 97 |
| Ruby on Rails | 95 | 98 | 96 |
| ASP.NET MVC | | 93 | 93 |
| Django | 90 | 92 | 91 |
| Meteor | 96 | 79 | 87 |
| Laravel | 92 | 83 | 87 |
| Spring | 82 | 90 | 86 |
| Express | 93 | 79 | 86 |
| CodeIgniter | 85 | 85 | 85 |

Figure 2.2: Top ten frameworks 1

PHP laravel framework was used in developing this project. Laravel is the most popular and most used framework in PHP. It is an open source framework designed by Taylor Otwell and its first beta version was made available on June $9^{th}$ 2011. It uses Model View Controller (MVC) architecture for developing web applications. MVC as the name indicates, the code is divided into three parts which are Model, View and Controller. Model consists of the code which deals with backend (database), View consists of code which deals with the user interface (front end pages) and Controller acts as medium for retrieving data from backend (Model) and passing it to frontend (View) and vice versa.

### 2.6. Ajax

Asynchronous JavaScript and XML in short is known as AJAX. It is used in create dynamic web pages to communicate with server side scripts. It sends and receives data to and from server in an asynchronous way. Data received by AJAX can be in various formats like JSON, HTML, XML, text files and more. The most common format is JSON format. AJAX is mainly used for updating web pages without the need of reloading the page. Some applications are listed below

- Validate or check usernames as user types
- Updating contents in applications like Stocks, Google maps

### 2.7. Indexing

Data at present is very large even in small industries, so accessing such a large data from database is a very tedious and time taking process. Indexing is a concept of speeding up the process of retrieving data from database. Indexing is done on some columns to search the date quickly without iterating through the entire rows of the table.

Syntax: CREATE INDEX index_name on table_name (column_1, column_2, …);

### 2.8. MySQL engine

Selecting an engine before creating a database is very important. Different engines have different purpose, and the default engine that is selected if we don't specify any engine is InnoDB. There are many engines some of which are listed below

- InnoDB – It is the default storage engine for MySQL versions 5.7 and above. It is ACID compliant (transaction safe) storage engine of MySQL.
- MyISAM – It is often used in applications which require fast read of data
- CSV – This type of engine allows programmers to directly dump a CSV file into tables.
- Memory – It is used for fast access of data, which is usually stored in memory.

### 2.9. JQuery

JQuery is a JavaScript library which simplifies the scripting code for designing web applications. It main purpose is to manipulate Document Object Model (DOM). It is the most popular and most widely used library of JavaScript. Writing code for handling events such as click, mouse over, mouse move, etc., is very simple and easy to write. JQuery is an open source plug in which can be downloaded and linked to HTML pages or we can directly use URL called as CDN which includes JQuery plugin into out page.

# Chapter 3

## Setup

Designing a web application requires some software's based on the application that is being developed. This project is developed in PHP using laravel framework. MySQL workbench software is used for creating database and tables for storing the data. WampServer software is used to make the database connection. SQL queries are written in laravel controllers which executes the queries and forwards the result to front end (views). Laravel is installed using composer, composer is a package manager software which is used to manage dependencies and packages. The below syntax is entered into command prompt to install laravel using composer.

```
composer create-project --prefer-dist laravel/laravel blog
```

The above command creates all the required files in the specified directory including environmental file, configuration file, etc. All the information required for database connection is specified in config/database.php file. The information for database connection includes host name, user name, password and database name. laravel also has an inbuilt server which can be started by the command below php artisan serve (This command starts the serve at address http://localhost:8000)

### 3.1. Basic SQL Queries

SQL queries in laravel looks little different from original SQL queries

```
DB::table('airports')->select('place','lat','lon')->where('place',$name)->get(); (Select statement)

DB::table('reviewandrating')->insert([‘name’ => $name, ‘airline’ => $airline); (Insert statement)
```

### 3.2. Implementation

In this project, a web application is developed using PHP. This web application gives user’s statistical information related to airlines and airports. First, the data which is in comma separated value format is stored in a relation database. The missing data is removed from the file before storing it in the database. The data in the CSV files can be directly loaded into the tables using the syntax below

```
LOAD DATA LOCAL INFILE 'PATH OF THE FILE'

INTO TABLE TABLE_NAME

FIELDS TERMINATED BY ','

ENCLOSED BY '"'

LINES TERMINATED BY '\n'

IGNORE 1 LINES
```

All the pages designed in this web application use almost the same procedure of storing and retrieving the data. First the data required for showing information to the users is analyzed. Based on the analysis different tables are created from the main airline data table, if the data cannot be accessed directly. After creating the required data tables, a function is written which takes user’s input and based on the user’s input queries are executed and appropriate results are gathered. User’s input is taken from the HTML pages (front end) which are known as views. Each view consists of a submit button which call an ajax function. The ajax function posts the user’s input to specified URL. There is a routes file which checks for a particular URL and redirects to its corresponding function in the controllers. In controllers, we first store all the users input, then a query is executed and the results are formatted in JSON format which are passed to the same HTML page from where the user inputs are taken. The results are passed by the controller to an HTML page, which shows users the information in an appropriate format. The different formats for showing information to the user include maps, donut charts, tables, lines charts, etc. Passing and retrieving data between HTML pages is done using AJAX, hence the information is displayed on the same page where the user gives input. A sample ajax function is given below

```
$.ajax({
    url: '/some_url’,
    type: ‘post’,       //method post or get
    data: { 'Variable1': Data1 ,'Variable2': Data2, ….. },
    success: function(data){
      //code
    }
});
```

The final web application includes eight web pages which are discussed below

### 3.3. Home page

The home page consists of the US map with all the airports represented using red circles. The information related to airports such as airport code, airport name, latitude and longitude is from Bureau of Transportation Statistics public dataset which is in CSV format. This data is stored in database and then it is used to represent airport locations on the US map. D3.js was used to show the US map and the airports. When the home page is selected, the application takes data from the database and then passes to D3.js plugin which then show results on the web page. Airport information such as airport code, airport name and location is displayed which consists of a submit button. By clicking on submit button user is displayed with selected airport and its connecting airports known as destination airports. Additionally, user is also given information about the average delays between origin airport and connecting airports.

### 3.4. Average delays page

In this page user has to select the origin airport, destination airport and month. Based on the user selection, a SQL query is executed which gives information consisting of origin, destination, airline, airtime and average departure delay in a table format. This information helps users to know which airline has less average departure delay. The record which has less average departure delay will be shown on top and the table which has high average departure delay will be shown at the bottom of the table. There is one more option in this page where user can select carrier along with other three options which are origin, destination, month.  With this option user gets information for the month that is selected.

### 3.5. Delays page

The information related to delays of an airline is presented in this page. User can select any airline and information related to delays is shown to them. The delays are based on five factors which are weather delay, carrier delay, security delay, NAS delay and late aircraft delay. In this page user is given information which tells user how much percentage of the total delay of an airline depends on these five factors. There are two other options in this web page where user can also get the results based on month and airport.

### 3.6. Flights and delays page

This page gives users the information related to number of flights travelled and number of flights which were delayed more than 25 mins. Users are allowed to select a year, since the database consists of data of airlines for three years 2006, 2007 and 2008; user can select any of the three years or all the years. Based on the year selected by the user the information about how many flights of a particular airline were flown in that year and how many flights were delayed out of the total is displayed.

### 3.7. Delay trends page

This page allows user to select any two airlines and based on the user selection, information related to delay percentage of the selected airlines is displayed. When user selects two airlines, the delay percentage of those two airlines for 2006, 2007 and 2008 years are displayed. When the user selects two airlines, then a SQL query is executed with percentage of delays as output. This output is converted to JSON format and sent to the web page and is represented using line chart.

### 3.8. Connecting airports page

In this page user can know how he can travel from one airport to another. User can know if there is a direct flight between the origin and destination airport. If there are no direct flights between the two selected airports (origin and destination), the users are given alternative options of travelling. If there is no direct flight, one can travel to another airport and from there they can reach the destination airport. This page also gives information about how long it will take to reach a selected airport.

### 3.9. Top airports and airlines page

This page lists top ten airports and airlines based on percentage of departure delay. Here the user is allowed to select year to know top airports and airlines for that particular year. The top airlines are listed based on the percentage of departure delay, the airlines with less percentage are listed on top and the airlines with high percentage of delay are listed at the end. To get the top airports, the percentage of delays of all the airlines of that particular airport is calculated. For top airports, airports with more than fifty thousand of flights travelling are considered.

### 3.10. Reviews and ratings page

This page allows users to rate and review different airlines. All the user ratings and reviews for a particular airline is averaged and displayed. All the user’s ratings and reviews are stored in the database and retrieved from the database based on user’s selection. Users can give ratings for different parameter in this page such as value for me, service, food, entertainment, space and overall rating.

### 3.11. Some laravel CLI statements

php artisan serve – used to start the inbuilt server

# Chapter 4

# Results

## 4.1. Home page

Figure 4.1 is a screen shot of the home page of the developed web application. This page shows all the airports within the US represented in red circles. By hovering on these red circles we get the airports code (IATA code).

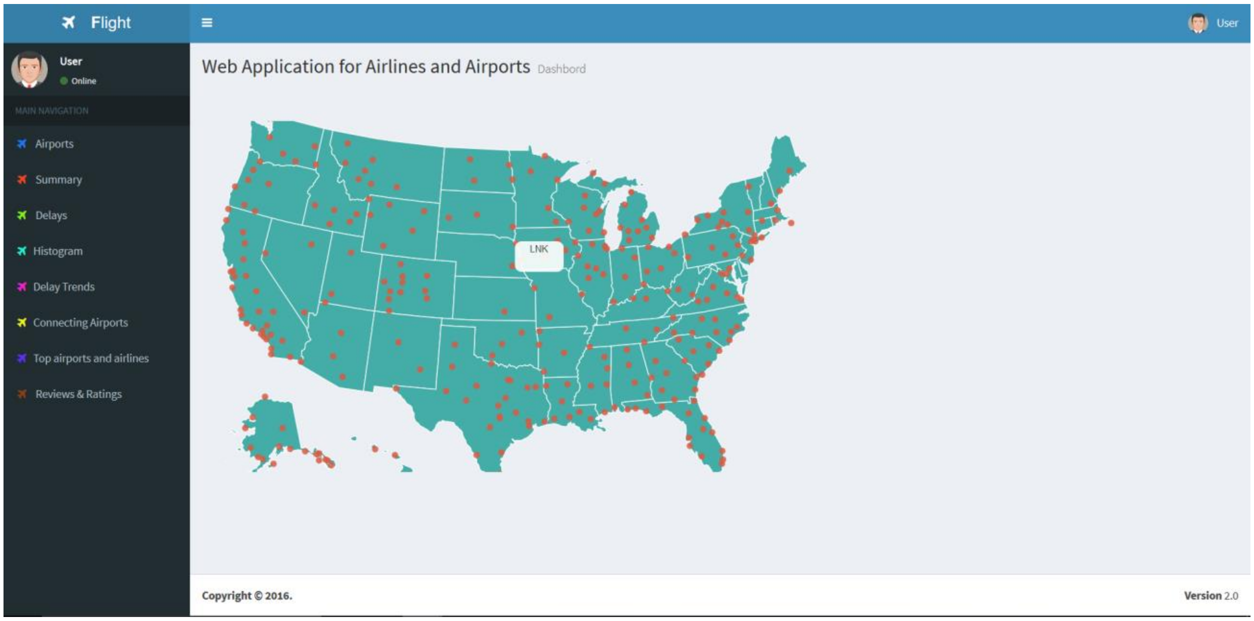
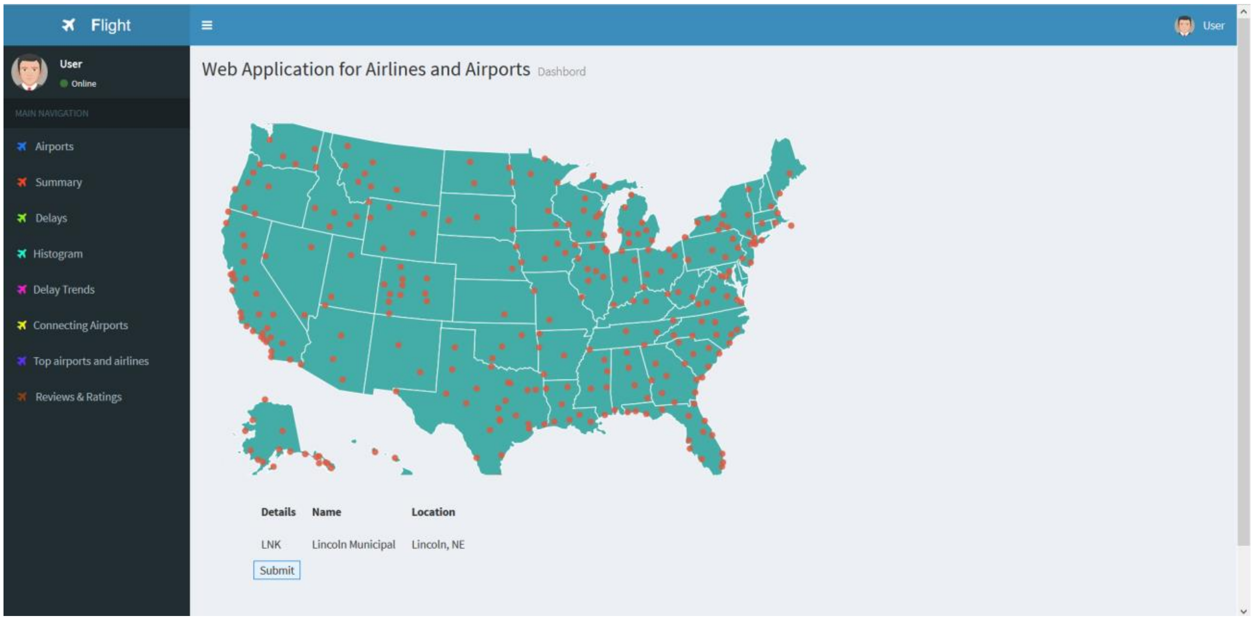


Figure 4.1: Home page with hovered LNK airport 1

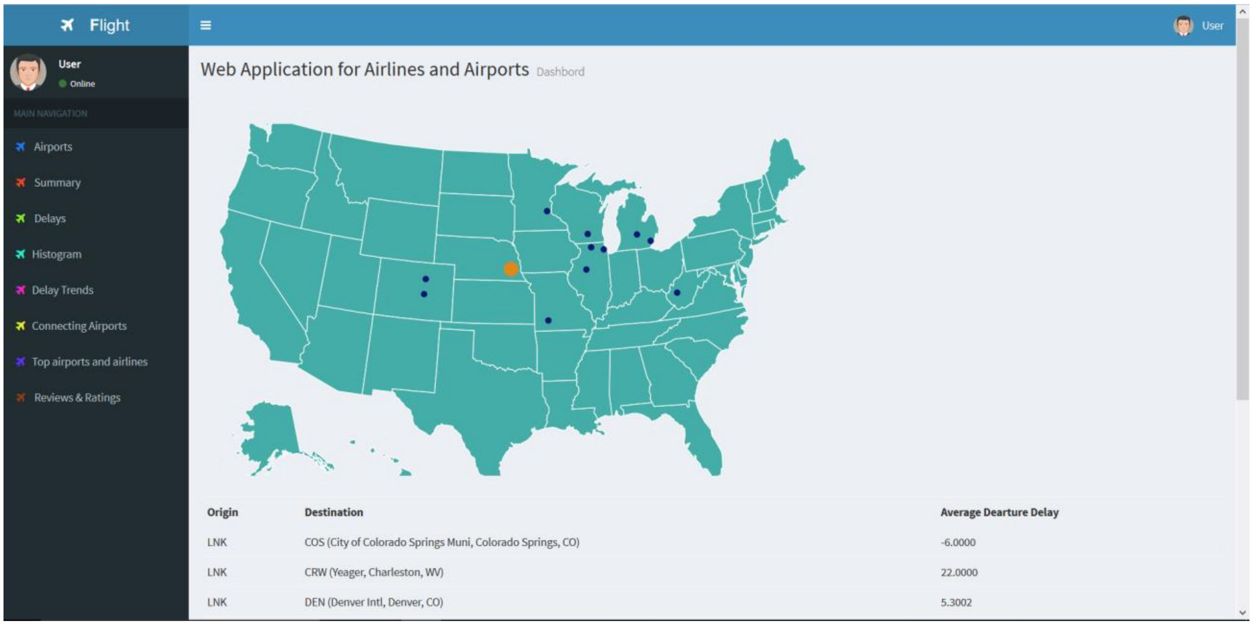
By clicking any of the circle, the information about that particular airport is displayed. The information that is displayed to the user include airport IATA code, name of the airport and location of the airport.

After that when the user clicks on submit button below, the user will be displayed with another US map containing origin airport (user selected airport) and all the connection airports (destination airports).  It also contains information about average departure delays between the origin airport and its connecting airports.



| Details | Name | Location |
|---|---|---|
| LNK | Lincoln Municipal | Lincoln, NE |

Figure 4.2: More information about LNK airport 1



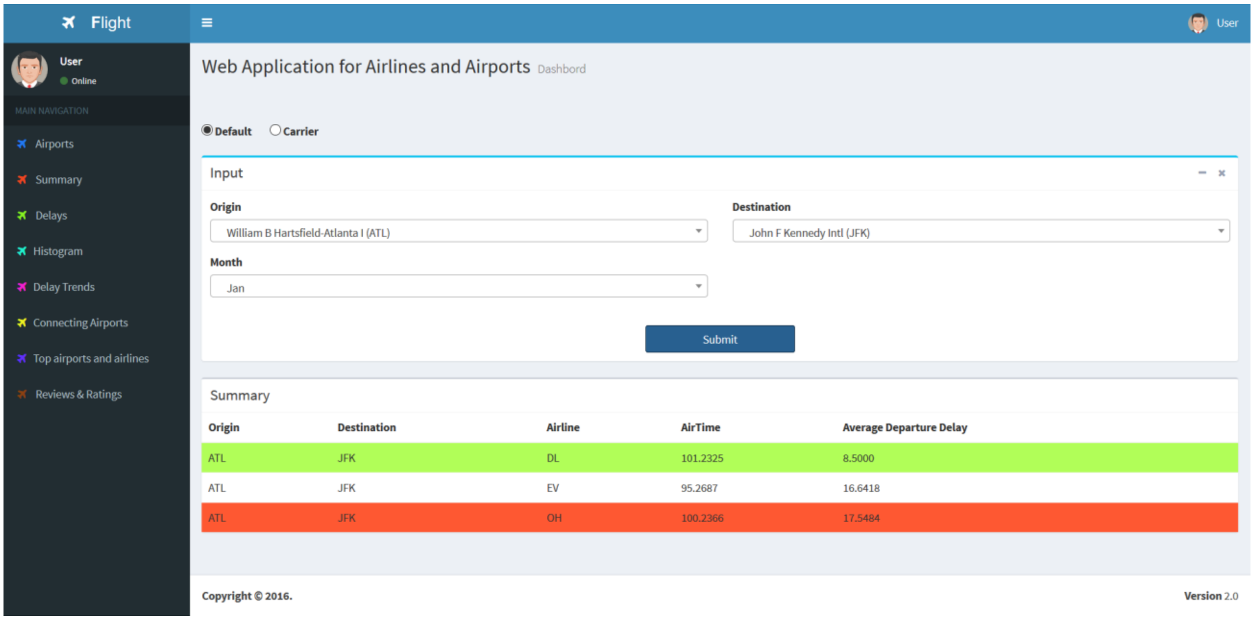
Web Application for Airlines and Airports Dashbord

| Origin | Destination | Average Dearture Delay |
|---|---|---|
| LNK | COS (City of Colorado Springs Muni, Colorado Springs, CO) | -6.0000 |
| LNK | CRW (Yeager, Charleston, WV) | 22.0000 |
| LNK | DEN (Denver Intl, Denver, CO) | 5.3002 |

Figure 4.3: LNK airport (orange) and its connecting airports (blue) 1

## 4.2. Average delays page

The figure 4.4 below is how the average delays page looks. Here user has two options which are default and carrier. Default option allows users to select origin, destination and month. By clicking on submit button user will be displayed with results. In the figure 4.4 we can see that user has selected origin airport ATL, destination airport JFK and month January. We can interpret the result as, for airline DL the average departure delay is 8.50 mins and airtime is 101.2325 mins for selected origin, destination and month.



Web Application for Airlines and Airports Dashbord

● Default ○ Carrier

Input

Origin: William B Hartsfield-Atlanta I (ATL)

Destination: John F Kennedy Intl (JFK)

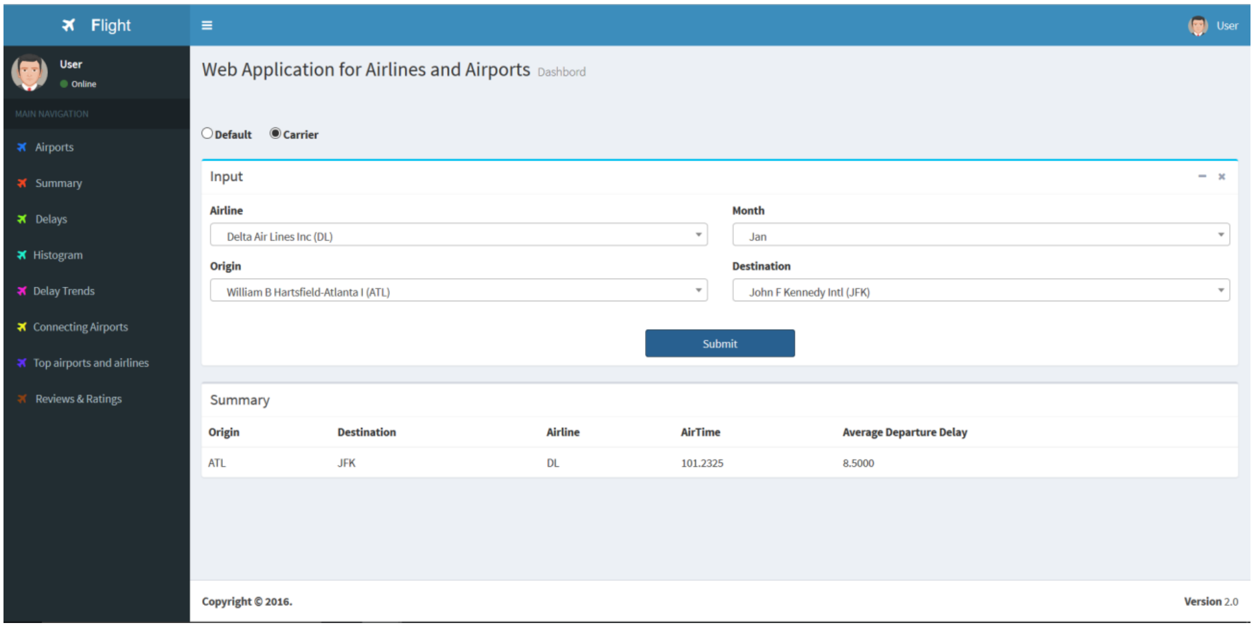
Month: Jan

Summary

| Origin | Destination | Airline | AirTime | Average Departure Delay |
|---|---|---|---|---|
| ATL | JFK | DL | 101.2325 | 8.5000 |
| ATL | JFK | EV | 95.2687 | 16.6418 |
| ATL | JFK | OH | 100.2366 | 17.5484 |

Figure 4.4: Average departure delays 1

In the carrier option (second option) there is another option for selecting the airlines, where user will be able to get the information for a particular airline. In the figure 4.5, user has selected DL as airline, January as month, ATL as origin and JFK as destination and the results can be interpreted as, there is an average departure delay of 8.50 mins and it takes airtime of 101.232 mins to reach the destination for DL airline in the month of January between ATL and JFK airports.



Web Application for Airlines and Airports Dashbord

○ Default ● Carrier

Input

Airline: Delta Air Lines Inc (DL)

Month: Jan

Origin: William B Hartsfield-Atlanta I (ATL)

Destination: John F Kennedy Intl (JFK)

Summary

| Origin | Destination | Airline | AirTime | Average Departure Delay |
|---|---|---|---|---|
| ATL | JFK | DL | 101.2325 | 8.5000 |

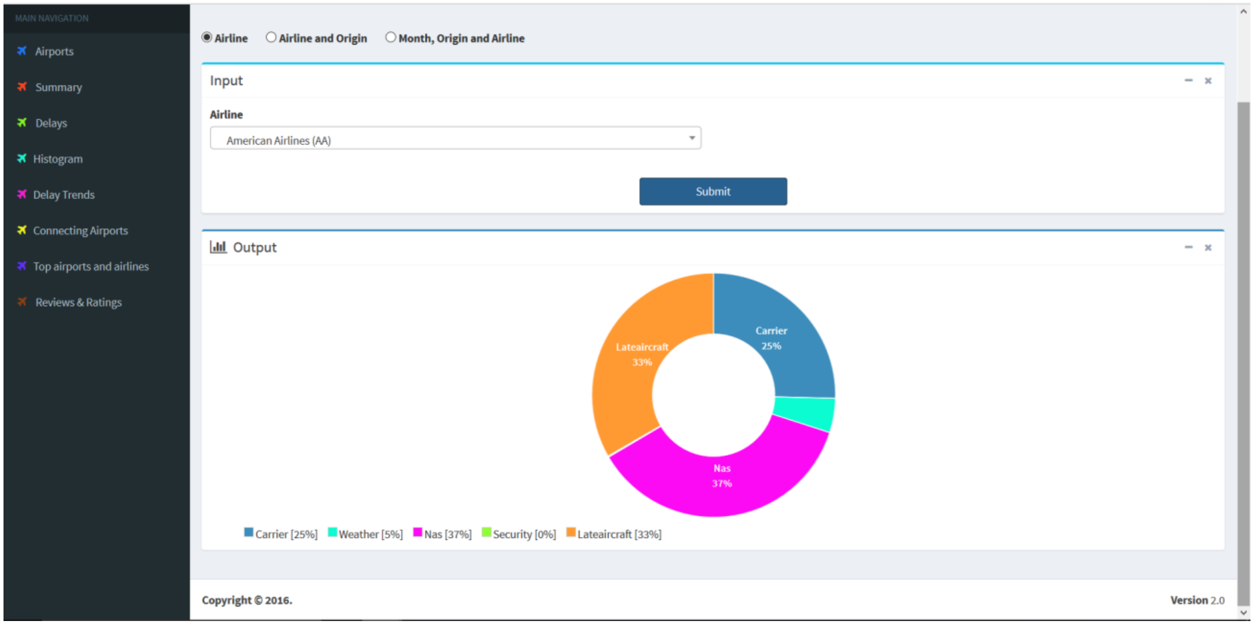
Copyright © 2016.

Version 2.0

Figure 4.5: Average departure delays for DL airline 1

## 4.3. Delays page

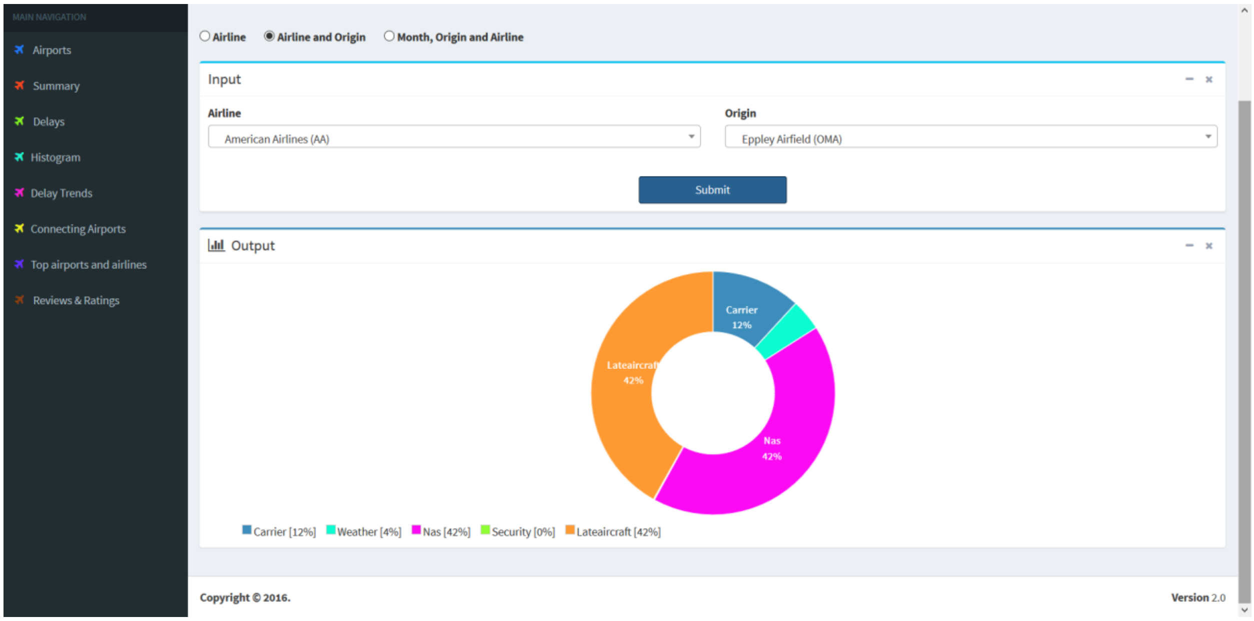
As we have discussed previously, there are five main factors which cause delay for a flight which are carrier delay, weather delay, NAS delay, security delay and late aircraft delay. Here there are three options for the user, in the first option user can select any airline and can know how much percentage of total delay is due to the above five factors. In the figure 4.6, user has selected American airlines (AA), and the results can be interpreted as, 25 percent of the total delays of AA is due to carrier delays, 5 percent is due to weather delay, 37 percent is due to NAS delay and 33 percent is due to late aircraft delay. We can also observe that there is no impact of security delays for AA. Similarly, user can also get the results for any airlines for any particular origin airport using option two (figure 4.6). And in option three (figure 4.6), user can select any month besides other two selections (airlines, origin) to get the same results for any particular month.



● Airline ○ Airline and Origin ○ Month, Origin and Airline

Airline: American Airlines (AA)

Figure 4.6: Delay factors (1) 1



○ Airline ● Airline and Origin ○ Month, Origin and Airline

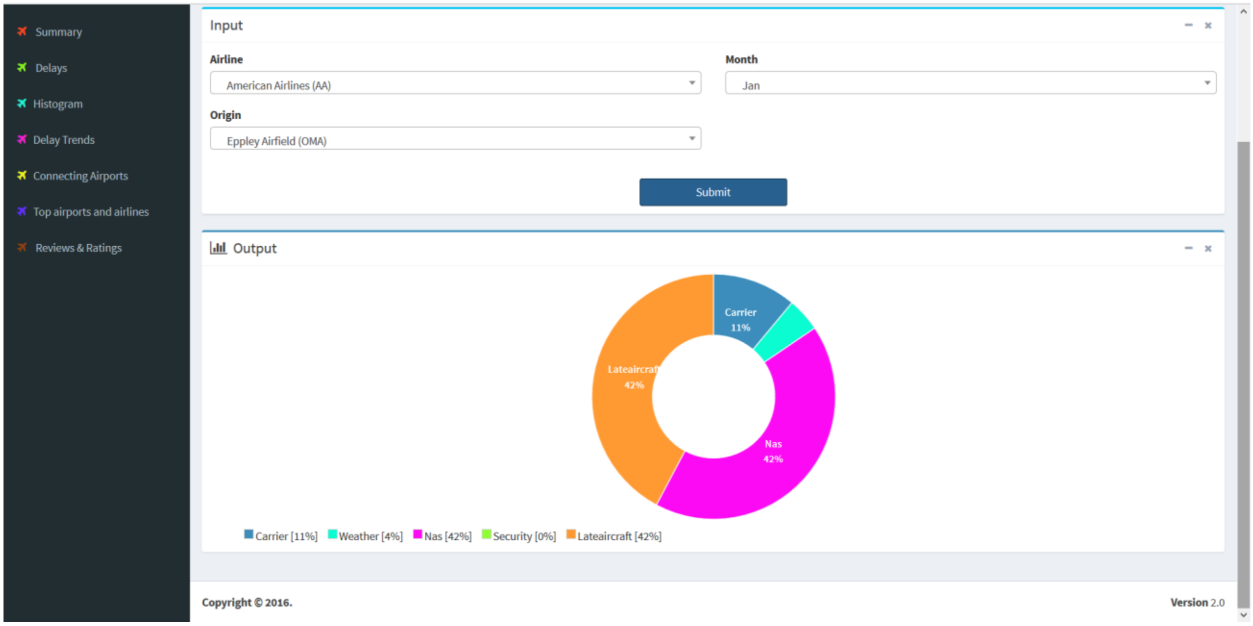
Airline: American Airlines (AA)

Origin: Eppley Airfield (OMA)

Figure 4.7: Delay factors (2) 1

In the figure 4.7 user has selected airline as AA and origin as OMA. The result can be interpreted as the total delay of AA in OMA airport is 12 percent due to carrier delay, 4 percent due to weather delay, 42 percent due to NAS delay, and 42 percent due to late aircraft delay.

In figure 4.8, user has selected airline as AA, origin as OMA and month as January, the result can be interpreted as the total delay of AA in OMA airport for January month is 11 percent due to carrier delay, 4 percent due to weather delay, 42 percent due to NAS delay and 42 percent due to late aircraft delay.



Input

Airline: American Airlines (AA)

Month: Jan

Origin: Eppley Airfield (OMA)

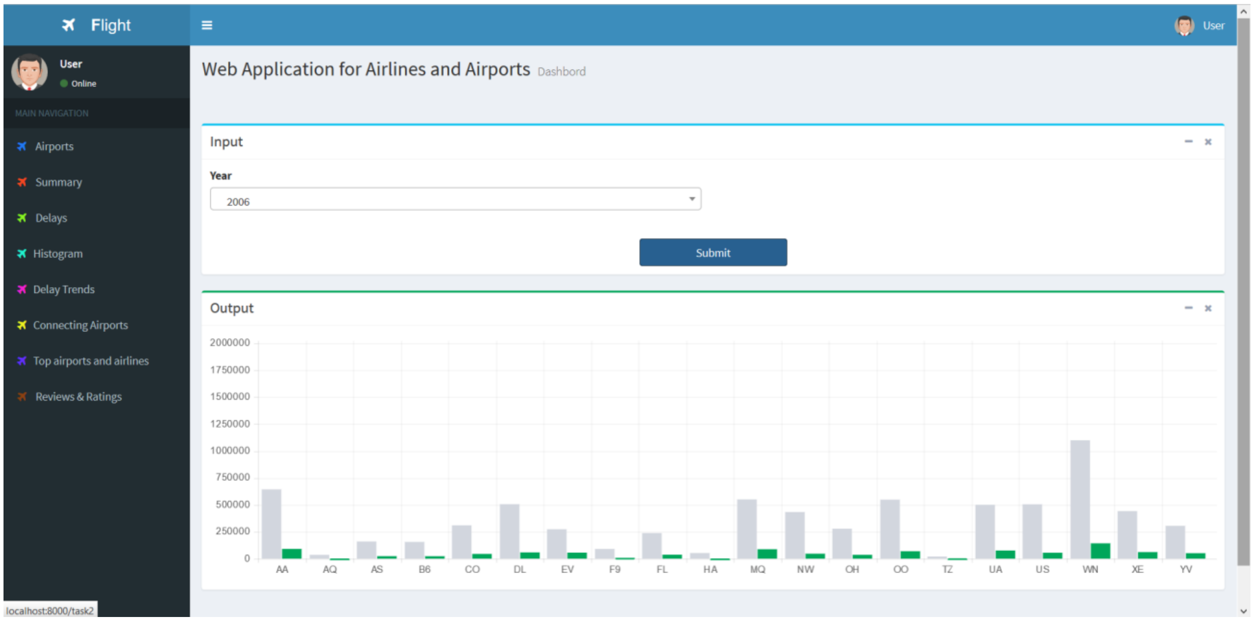
Copyright © 2016.

Version 2.0

Figure 4.8: Delay factors (3) 1

### 4.4. Flights and delays page

The below figure 4.9 is how the flights and delays page looks. In this page user is displayed with histograms. Here there is an option for the user called year, by selecting a year the user is displayed with histograms which tells, for a selected year the number of flights of an airline travelled and number of flights delayed. Here the user has selected year 2006, the x-axis represents all the different airlines. In the figure 4.9, we can see that for airline WN the number of flights travelled are around 1.1 million and the number of flights delayed are around 0.2 million.



Web Application for Airlines and Airports Dashbord

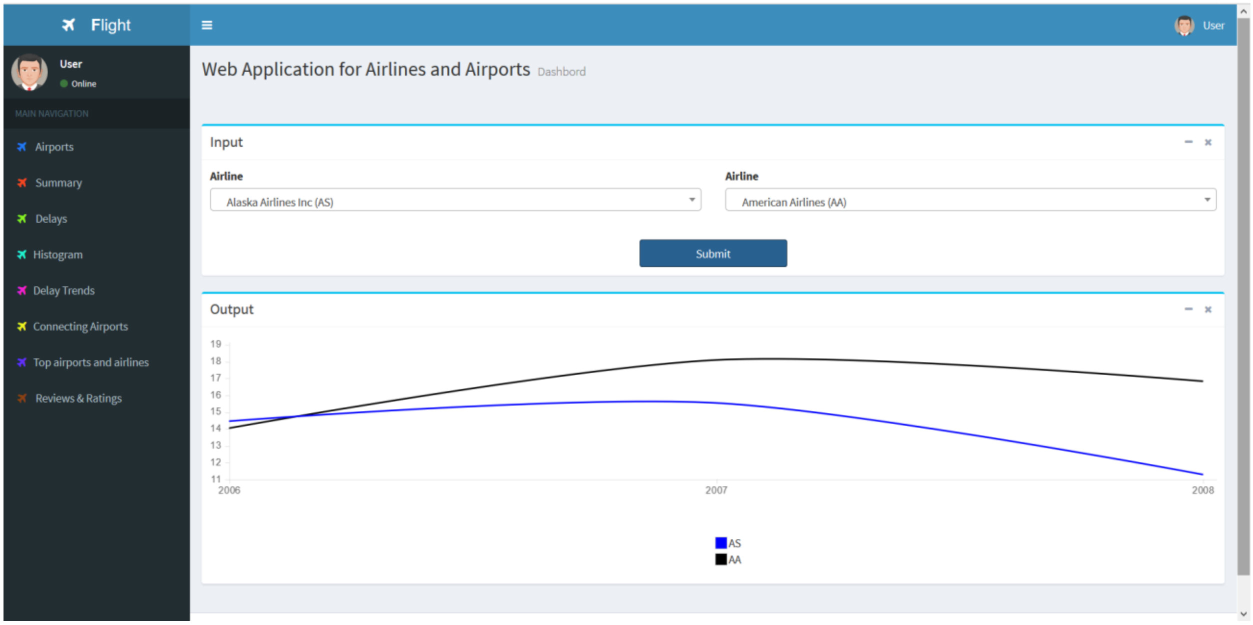
Input

Year: 2006

Figure 4.9: Number of flights vs delayed flights 1

## 4.5. Delay trends page

In this page user can compare the delay trends of two different airlines for three years (2006, 2007 and 2008). In the figure 4.10 the user has selected two airlines which are AS and AA, the below line chart represents the delay trends for these two airlines for three different years. The black line represents AA and blue line represents AS. We can see how the delays vary for both the airlines from 2006 to 2008. For AA, the percentage of departure delays was around 14, 18 and 17 for 2006, 2007 and 2008 years respectively and for AS, the percentage of departure delays was around 14.5, 15.5 and 11 for 2006, 2007 and 2008 years respectively.



Web Application for Airlines and Airports Dashbord

Input

Airline: Alaska Airlines Inc (AS)

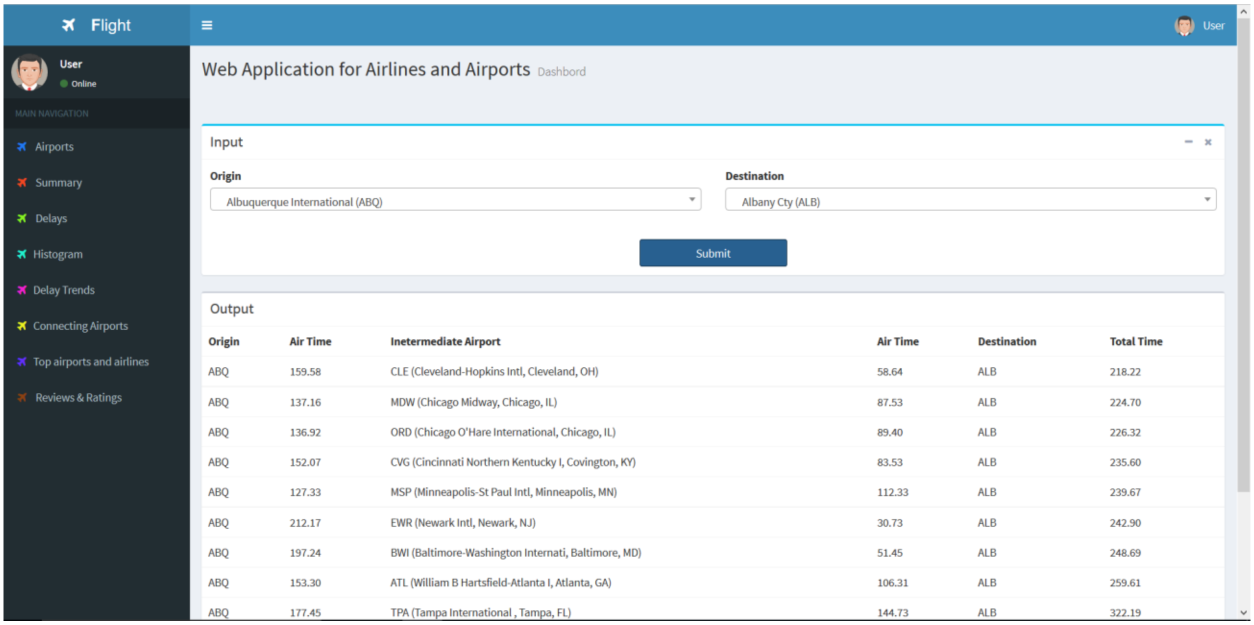
Airline: American Airlines (AA)

Output

Figure 4.10: Delay trends for AA and AS 1

### 4.6. Connecting airports page

In this page, user can check all the possible ways of travelling from one airport to another. For a selected origin and destination, user can know if there is a direct flight between the selected airports and the user can also know if there is another way of travelling like a one hop travel, where the flight first goes to an intermediate airport and from that airport user can reach the destination airport. Users will also be presented with the total time taken for the travel. This information will be helpful if the users want to know how they can travel from origin to destination if there is no direct flight. In figure 4.11, user has selected origin airport as ABQ and destination airport as ALB, and the first record in results can be interpreted as for travelling from ABQ to ALB user has to first travel to CLE airport which takes 159.58 minutes to travel. Then the user can travel from CLE to ALB airport with travel time of 58.64 minutes. The total time for the user travel is 218.22 minutes



Web Application for Airlines and Airports Dashbord

Input

Origin: Albuquerque International (ABQ)

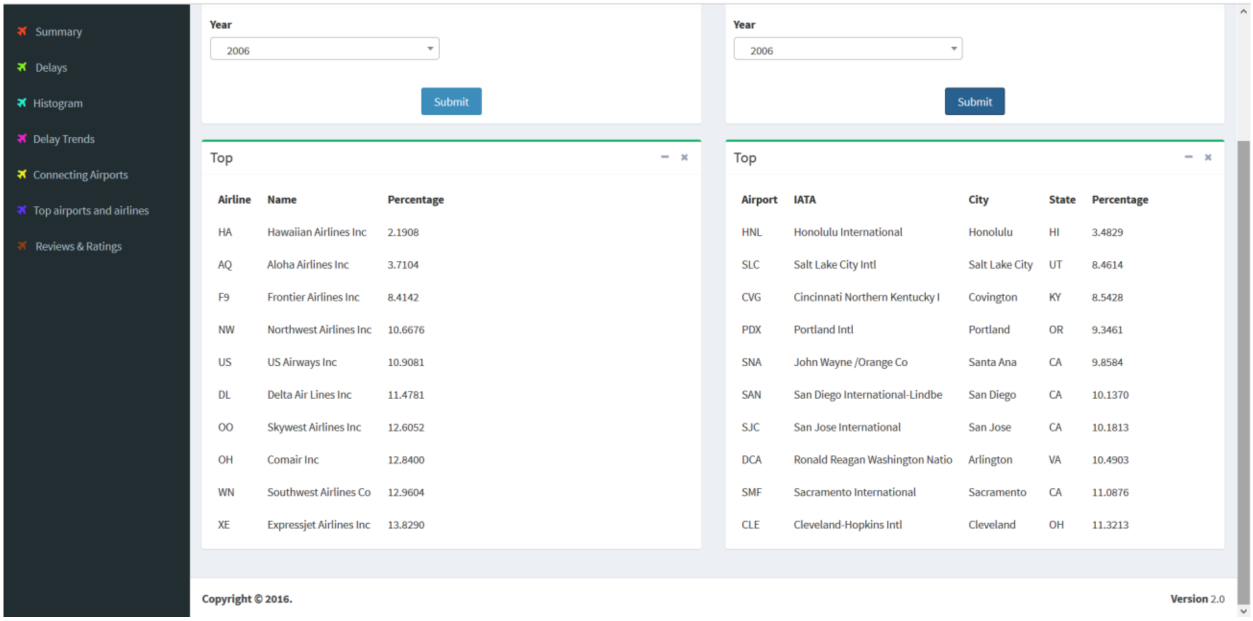
Destination: Albany Cty (ALB)

Output

| Origin | Air Time | Inetermediate Airport | Air Time | Destination | Total Time |
|---|---|---|---|---|---|
| ABQ | 159.58 | CLE (Cleveland-Hopkins Intl, Cleveland, OH) | 58.64 | ALB | 218.22 |
| ABQ | 137.16 | MDW (Chicago Midway, Chicago, IL) | 87.53 | ALB | 224.70 |
| ABQ | 136.92 | ORD (Chicago O'Hare International, Chicago, IL) | 89.40 | ALB | 226.32 |
| ABQ | 152.07 | CVG (Cincinnati Northern Kentucky I, Covington, KY) | 83.53 | ALB | 235.60 |
| ABQ | 127.33 | MSP (Minneapolis-St Paul Intl, Minneapolis, MN) | 112.33 | ALB | 239.67 |
| ABQ | 212.17 | EWR (Newark Intl, Newark, NJ) | 30.73 | ALB | 242.90 |
| ABQ | 197.24 | BWI (Baltimore-Washington Internati, Baltimore, MD) | 51.45 | ALB | 248.69 |
| ABQ | 153.30 | ATL (William B Hartsfield-Atlanta I, Atlanta, GA) | 106.31 | ALB | 259.61 |
| ABQ | 177.45 | TPA (Tampa International , Tampa, FL) | 144.73 | ALB | 322.19 |

Figure 4.11: Path from ABQ to ALB 1

## 4.7. Top airlines and airports page



Year: 2006

Year: 2006

Top

| Airline | Name | Percentage |
|---|---|---|
| HA | Hawaiian Airlines Inc | 2.1908 |
| AQ | Aloha Airlines Inc | 3.7104 |
| F9 | Frontier Airlines Inc | 8.4142 |
| NW | Northwest Airlines Inc | 10.6676 |
| US | US Airways Inc | 10.9081 |
| DL | Delta Air Lines Inc | 11.4781 |
| OO | Skywest Airlines Inc | 12.6052 |
| OH | Comair Inc | 12.8400 |
| WN | Southwest Airlines Co | 12.9604 |
| XE | Expressjet Airlines Inc | 13.8290 |

Top

| Airport | IATA | City | State | Percentage |
|---|---|---|---|---|
| HNL | Honolulu International | Honolulu | HI | 3.4829 |
| SLC | Salt Lake City Intl | Salt Lake City | UT | 8.4614 |
| CVG | Cincinnati Northern Kentucky I | Covington | KY | 8.5428 |
| PDX | Portland Intl | Portland | OR | 9.3461 |
| SNA | John Wayne /Orange Co | Santa Ana | CA | 9.8584 |
| SAN | San Diego International-Lindbe | San Diego | CA | 10.1370 |
| SJC | San Jose International | San Jose | CA | 10.1813 |
| DCA | Ronald Reagan Washington Natio | Arlington | VA | 10.4903 |
| SMF | Sacramento International | Sacramento | CA | 11.0876 |
| CLE | Cleveland-Hopkins Intl | Cleveland | OH | 11.3213 |

Copyright © 2016.

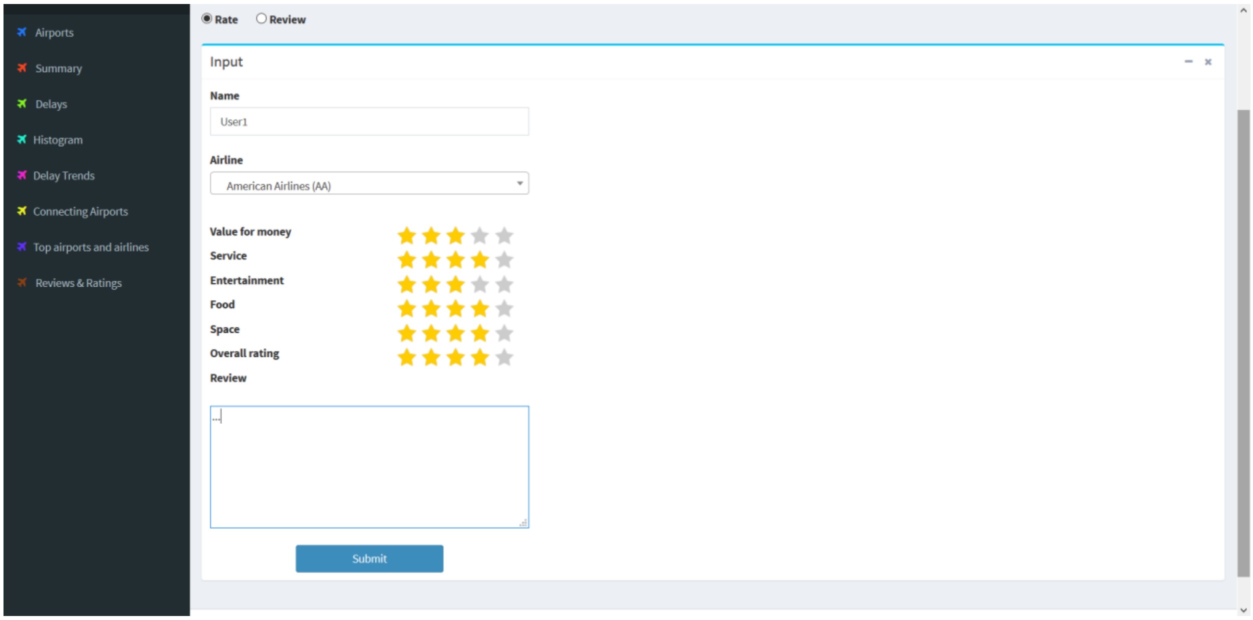
Version 2.0

Figure 4.12: Top 10 airlines and airports 1

In this page user is presented with list of top ten airlines and airports for any particular year. Figure 4.12 is how this page looks. Here, the user has selected 2006 for both airlines and airports. So, the top ten airlines and airports for 2006 are listed in a tabular form.

## 4.8. Reviews and ratings page

Figure 4.13 represents the reviews and ratings page, where user can rate and write a review for an airline. User has to give his name, airline for which he wants to rate and review, ratings for different services on a scale of 1 to 5 and the user can also write a review about his experience. By clicking the submit button, all the user inputs will be stored in the database.



● Rate ○ Review

Input

Name: User1

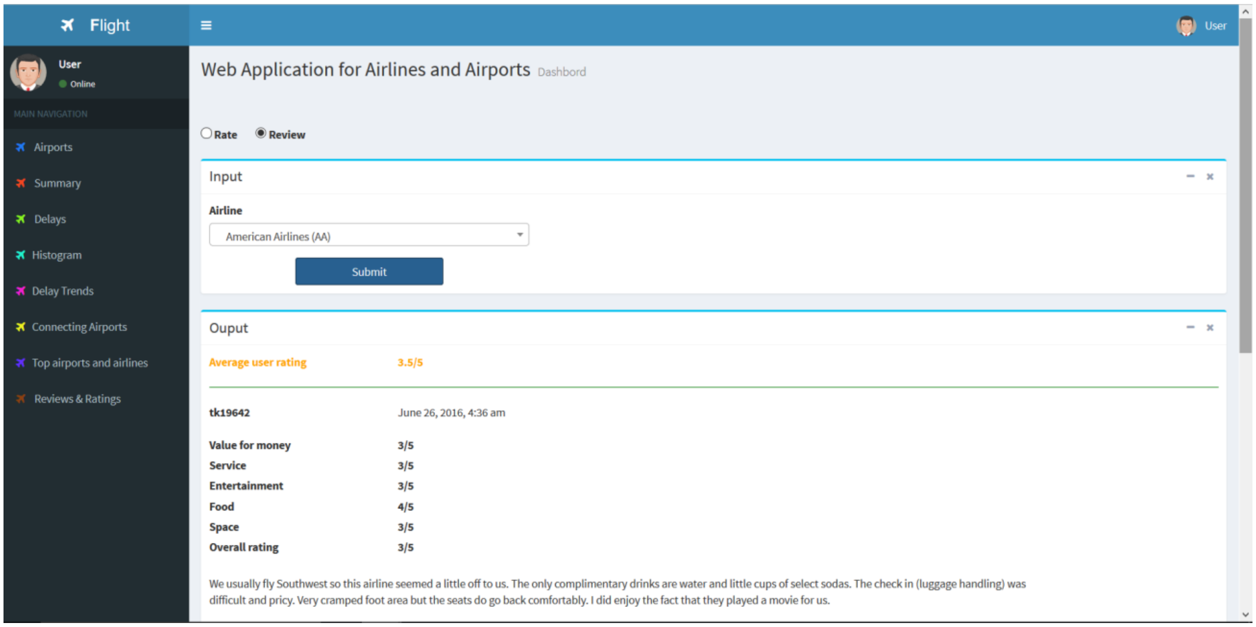
Airline: American Airlines (AA)

| | |
|---|---|
| Value for money | ★★★☆☆ |
| Service | ★★★★☆ |
| Entertainment | ★★★☆☆ |
| Food | ★★★★☆ |
| Space | ★★★★☆ |
| Overall rating | ★★★★☆ |

Review

Figure 4.13: User ratings form 1

Users can also check others ratings and reviews for any airline. Here in the figure 4.14, user has selected AA and the average user ratings for AA is displayed along with all the other user ratings and reviews for that particular airline.



Web Application for Airlines and Airports Dashbord

○ Rate ● Review

Input

Airline

American Airlines (AA)

Ouput

| | |
|---|---|
| Average user rating | 3.5/5 |
| tk19642 | June 26, 2016, 4:36 am |
| Value for money | 3/5 |
| Service | 3/5 |
| Entertainment | 3/5 |
| Food | 4/5 |
| Space | 3/5 |
| Overall rating | 3/5 |

We usually fly Southwest so this airline seemed a little off to us. The only complimentary drinks are water and little cups of select sodas. The check in (luggage handling) was difficult and pricy. Very cramped foot area but the seats do go back comfortably. I did enjoy the fact that they played a movie for us.

Figure 4.14: User review form 1

# Chapter 5

# Summary and Future Work

To sum up, in this paper we have discussed about the tools and technologies required for developing an interactive web application using PHP with laravel framework. The dataset from Bureau of Transportation Statistics is the main factor on which the whole application works. We have developed a web application which can be used by common users as well as airline and airport operators. We have discussed the implementation and the final results of the web application.

## 5.1 Future Direction

Adding day to day data or latest data to the database may be more useful. Storing all the data may be a tedious process and cannot be done on a local system, because of its increasing size. Now the database for this application contains only data related to the US, we can extend to various countries as well by adding data of those countries.

## Bibliography

[1] http://www.codeproject.com/Articles/552846/Why-s-How-s-of-Asp-Net-MVC-Part

[2]