Engineer Method

Problem identification

1.1 Context

The air transport industry is a big and competitive market around the world. Currently in USA the market is dominated by four airlines, American Airlines Inc., Jetblue Airways, Hawaiian Airlines Inc., and SkyWest Airlines Inc.

Delta Airline Inc. has incursed recently in the air transport industry in the united states, their goal is to be recognized but, they have an issue, because they just hit the market they don't much information about how well are they doing compared to their competitors, moreover, they are unsure about how well did they distributed their sources and if they are covering the market properly.

1.2 Needs List

- Find Information about how is Delta Airline Inc. distributed around united states.
- Find Information about how are Delta Airline's competitors distributed around united states.
- Find useful information to compare Delta Airline perform against their competitors.
- Use appropriated tools to compare big data sets.
- Display correctly all the information the company needs to know.

1.3 Problem definition

Delta Airline Inc. wants to know how are they covering the united states and how are they performing against their competitors.

1.4 Requirement List

R: Requirement

R1	
Name	Show flights information on screen for the user.
Description	The program needs a friendly interface so the user can compare the results and navigate through them.
Entry	User selection of a company.
Exit	In the interface a map with the information required by the user would be shown.

R2	
Name	User can filter information.
Description	The user can choose between given filters information he wants to display.

Entry	A filter chosen.
Exit	The maps with a selected company will update their data.

R2.1	
Name	User can filter information by an origin city.
Description	This filter allows the user the see from where the flights left.
Entry	Value that indicates this action.
Exit	The maps with a selected company will update their data showing the information related to this topic and a total of cities would be shown as well.

R2.2	
Name	User can filter information by positive delay.
Description	This filter allows the user the see which flights took more time than the scheduled to leave the city.
Entry	Value that indicates this action.
Exit	The maps with a selected company will update their data showing the information related to this topic and a total of cities would be shown as well.

R2.3	
Name	User can filter information by negative delay.
Description	This filter allows the user the see which flights took less time than the scheduled to leave the city.
Entry	Value that indicates this action.
Exit	The maps with a selected company will update their data showing the information related to this topic and a total of cities would be shown as well.

Information search

2.1 Theoretical framework

2.1.1 Graph

"Graphs are mathematical structures that represent pairwise relationships between objects. A graph is a flow structure that represents the relationship between various objects. It can be visualized by using the following two basic components:

Nodes: These are the most important components in any graph. Nodes are entities whose relationships are expressed using edges. If a graph comprises 2 nodes A and B and an undirected edge between them, then it expresses a bi-directional relationship between the nodes and edge.

Edges: Edges are the components that are used to represent the relationships between various nodes in a graph. An edge between two nodes expresses a one-way or two-way relationship between the nodes.”.[4]

This is a handy tool in the programming world that helps to build almost everything you can need.

2.1.2 Query

Query means a specific question or concern. In the informatic world Query is used to ask a data base about a specific information.

This action could give you a data set or a specific data.

2.2 State of the art

2.2.1 Bureau of transportation statistics

The Bureau of transportation statistics gather historical information about all the transport modes, air, ship, truck and rail. [2]

They have tons of records sorted in tables, in order to search for something specific or analyze information you have to find a way to so. They only offer limited summaries, which have to be accessed from a source different of the table.

All the historical information they have in their records are free to access and is very complete.

2.2.2 Airport Authority Performance Summary

[1] “The Airport Authority Performance Dashboard provides a quick snapshot of all relevant data with a user-centric view to offer a detailed analysis”.

This analysis is based on arrival delays and departure delays using that companies have in the airports. This tool shows that information with a comparative table and a map where the user can see origin and destiny of the flights.

The Airport Authority Performance Dashboard also has a tab which provides information about traveler's satisfaction and feedback.

Creative Solutions Research

3.1 Information filtering techniques for C#

Filtering refers to the operation of restricting the result set to contain only those elements that satisfy a specified condition.

The following are the standard query operator methods that perform filtering:

1. OfType. Selects values, depending on their ability to be cast to a specified type. Methods:
    a. *Enumerable.OfType<TResult>(IEnumerable)*. Filters the elements of an IEnumerable based on a specified type.

    b. *Queryable.OfType<TResult>(IQueryable)*. Filters the number of an IQueryable based on a specified type.

2. Where. Selects values that are based on a predicated function. Methods:
    a. *Enumerable.Where.* Filters a sequence of values based on a predicate.
        I. *Where<TSource>(IEnumerable<TSource>, Func<TSource,Boolean>)*. Example:

```
C#

List<string> fruits =
    new List<string> { "apple", "passionfruit", "banana", "mango",
                   "orange", "blueberry", "grape", "strawberry" };

IEnumerable<string> query = fruits.Where(fruit => fruit.Length < 6);

foreach (string fruit in query)
{
    Console.WriteLine(fruit);
}
/*
 This code produces the following output:

 apple
 mango
 grape
*/
```

        II. *Where<TSource>(IEnumerable<TSource>,Func<TSource>,Int32,Boolean>).* Example:

```
C#

int[] numbers = { 0, 30, 20, 15, 90, 85, 40, 75 };

IEnumerable<int> query =
    numbers.Where((number, index) => number <= index * 10);

foreach (int number in query)
{
    Console.WriteLine(number);
}
/*
 This code produces the following output:

 0
 20
 15
 40
*/
```

    b. *Queryable.Where*. Filters a sequence of values based on a predicate.

I. *Where<TSource>(IQueryable<TSource>, Expression<Func<TSource,Boolean>>)*.
Example:

```
C#

List<string> fruits =
    new List<string> { "apple", "passionfruit", "banana", "mango",
                       "orange", "blueberry", "grape", "strawberry" };

// Get all strings whose length is less than 6.
IEnumerable<string> query =
    fruits.AsQueryable().Where(fruit => fruit.Length < 6);

foreach (string fruit in query)
    Console.WriteLine(fruit);

/*
    This code produces the following output:

    apple
    mango
    grape
*/
```

II. *Where<TSource>(IQueryable<TSource>,Expression<Func<TSource,Int32,Boolean>>)*.
Example:

```
C#

int[] numbers = { 0, 30, 20, 15, 90, 85, 40, 75 };

// Get all the numbers that are less than or equal to
// the product of their index in the array and 10.
IEnumerable<int> query =
    numbers.AsQueryable()
    .Where((number, index) => number <= index * 10);

foreach (int number in query)
    Console.WriteLine(number);

/*
    This code produces the following output:

    0
    20
    15
    40
*/
```

3.2 Visualization of relevant information using GMap.

In order to implement a properly way to show the information the graphs concept is a good option to model the visualization feature.

Following these ideas:

1. Each city of departure and arrival could be a vertex.
2. Each airport used for the companies could be the vertex.
3. Each State could be used as a vertex.

4. Each flight would be an edge.

5. Air-traffic intensity could be represented as an edge, this means many flights would be merged in one edge.

6. Use a unique screen in order to show all the info of the company and its competitors on the same map.
7. Use a split-screen interface to show the flights of the company and competitors in different maps.

By using these ideas, the user would be able to see clearly on a map the air traffic of its company as well as the competitors air traffic.

Preliminary Designs Transition

It is crucial for the project the selection of a filtering method that could works efficiently and easy to implement. About the selection of the best solution for the method to filter the information, it would be treated in the next step, where some criteria would be defined by the team.

Otherwise, taking a look on what the view of these high-volume data representation would be for the pc and even for the user, we decided to decline the option of represent every flight as an edge. For the same reasons, it is ineffective to use only one map to show the flights, if one of the objectives is compare the traffic of our company against competitors' market.

In terms of performance of the program, we can infer that the fact of show every airport and every route that have been made would cause a slower response of the GUI's functionalities like the zoom or move around the map.

Due to the above, we've decided to left the GUI's aspect to options that haven't been ruled out.

Evaluation and selection of the best choice

Criteria:

- High-volume management.

  Based on how fast the method can handle each flight's data, being 1 the slowest speed and 3 the fastest.

- Learning curve.

  Based on how much time have to be spend to learn about the method and implement it efficiently, being 1 a high spend or cost of time and 4 the minimum, as we already know about it.

- Clarity in Data Visualization

Based on easy is to see the data in the interface and to compare one flight with another being intuitive, so for it each idea will be ordered to choose the first in terms of personal opinion of the team.

Data Filtering Techniques

	Criteria 1	Criteria 2	Total
OfType	2	1	3
Where	3	2	5

So the conclusion is to use the where method and also use only the INumerable sub method because the IQueryable use queries which are an important part of SQL tool, and we haven't see anything about Data Base.

Now listing the view options, we have the next result:

- Each city of departure and arrival could be a vertex.
- Air-traffic intensity could be represented as an edge, this means many flights would be merged in one edge.

For the last option we decided to show each airline traffic in the same screen but with different map views so there would be a problem like unnecessary quantity of screens in the monitor.

References

[1] Airport Authority Performance Summary, “Microsoft Power BI Community BTS”. [Online]. Available: https://community.powerbi.com/t5/Data-Stories-Gallery/Airport-Authority-Performance-Summary/td-p/628963 [Accessed: Feb 1, 2020].

[2] BTS, “BTS, Bureau of transportation statistics”. [Online]. Available: https://www.bts.gov/ [Accessed: Feb 1, 2020].

[3] Filtering Data (C#), “LINQ, Language Integrated Query”. [Online]. Available: https://docs.microsoft.com/en-us/dotnet/csharp/programming-guide/concepts/linq/filtering-data [Accessed: Feb 2, 2020].

[4] Graph Representation, “hackerearth”. [Online]. Available: https://www.hackerearth.com/practice/algorithms/graphs/graph-representation/tutorial/ [Accessed: Feb 3, 2020].