**Team project report**

* **Task**

Create a manager for an airline company  
Implement the following classes ('-' indicates a field):

----------------  
PLANE  
-Place []  
-Board  
----------------  
AIRPORT

-name  
-city  
----------------  
PASSENGER

-name  
-name  
------------------  
FLIGHT  
-place departure  
The place of arrival  
- Flight responsible for the flight  
-time departure  
-arrival time  
-----------------  
PLACE  
-number

The user should be able to:  
Create Airplane, Airport, Passenger, Flight  
view flight information, airplane  
create and cancel order look:  
Passenger  
Flight  
Place  
Price  
see income for all time  
  
Terms of implementation:  
save data in files  
console implementation and implemented help  
create documentation  
create readme

Write a report

* **Way of implementation, features**

Our idea was to build a chain of classes:

**Customer->place->plane->flight**

Human(customer) is in field of place

Place is in plane

Plane is in Flight

One of the features is similarity to playlist manager

* Results
* Problems encountered in the project

There are couple of problems related to merging all parts of code , written by our team+ Input and Output from streams+interaction of classes+connection between classes+construction the scheme of the project......... -\_-

But we fixed most of bags and hope evrything works well

* Team cotributions

1. Mykola Krainyk

Realisation of main functions +merging all parts+testing program

1. Orest Gorak

**Creating a project+Announce header files+Report**

1. Victor Hanyk

**Creating Read me file + class passanger**

1. Vadym Pyshnyak

**Class Airport**

1. Kiral Lukiana

auxiliary functions that helped in the process of execution



**COUT<< «THANK YOU FOR YOUR ATTENTION «<<ENDL;**

**HOPE YOU LIKED OUR PRODUCT))**



**What is our DREAM\_TEAM**