

Application Development Project – Airplane

Asmer Bracho – 2016328

Group B

Lecturer: Mark Morrissey

March 31, 2018

Table of Contents

Assignment of Tasks	3
Role within the Project	3
Creation of Interfaces and packages	3
Staff Class	4
Pilot Class	4
Airplane Class	5

Assignment of Tasks

Classes	Staff	Pilot	AirPlane	Flight	Menu	CCTAir	DataGenerator
Adelo					X	X	
Asmer	X	X	X			X	
MiguelAntonio				X		X	X

Division of developing of classes between the three of us.

Role within the Project

Creation of Interfaces and packages

In the List of Roles it can be seems one of my tasks was the set up of the interfaces as first point to start the assignation of tasks and to follow the development of the different part of the system.

The InterfaceFlight push two methods with the same name but different signature.

The InterfaceAirPlane contain the getters for the variables given as well as an assignPilot method which takes an object pilot as parameter. Finally a licenseRequired method which intends to check the type of license a Pilot has to have in order to flight the Air Plane.

The Interface Menu The designing of the interface was with the purpose of following the way the program was going to behave, so in this interface contain methods that when apply will bring the user to a set of submenus to interact with.

The purpose of creating these interfaces as well as the different packages presented in the project, was the independence an availability of working in group being able to communicate each other in the same language so eventually by following this rules when putting all the part together there are not mismatch and time lost.

Staff Class

Part of the package cctair.staff.

Variables : staffId, name;

Constructor: public Staff (String name);

Methods: getStaffId(); getName();

This is a very simple Class that will handle the employees (staff) of the business. The reason of creating this class is that it can be a parent class that contain an IdStaff and a name that can be common for every staff in the business, so far the program handles only pilots, but thinking in a efficient way of developing, this could growth and get another type of staff such a flight attendant.

Pilot Class

Part of the Package cctair.staff.

Variables: license;

Constructors: public Pilot();

 public Pilot(String name, String license);

Methods: getLicense(); toString(); setLicense();

This class extends the staff class taking Id and name as variable of the super class an having as new variable a license. (This will allow checking against the Air Planes if the pilot is qualified to fly specific Air Planes Models) This Class has two different constructors. The first one and default will be set whit no parameters at all, (used to create a Co-pilot for each plane (No Assigned but instantiated)). A second constructor will take two parameters (name and license) used to create pilots. So that means that every plane will be giving 2 objects Pilot (one Pilot and one Co-Pilot).

This Class contain get and set method for license and inherit the methods for the staff class. The toString method is override to give a specific format in the retrieve of the variables contain in it.

Airplane Class

Part of the package cctair.plane.

Variables: planeId, make, model, capacity, pilot, coPilot, LicenseRequired;

Constructors: public AirPlane(String make, int model, int capacity, Pilot pilot);

 public AirPlane(String make, int model, int capacity, Pilot pilot, Pilot coPilot);

Methods: getPlaneId(); getMake(); getModel(); getPilot(); getCoPilot(); capacity;
 assignPilot(Pilot pilot); assignCoPilot(Pilot coPilot);

This class implements the InterfaceAirPlane, taking the variables listed above. Two constructor are given in this class the reason being is that an Air Plane can be created by ether assigning a Pilot and Co-Pilot or only a Pilot. The logic behind this complementary constructor and creation of instances is that when assigning a Pilot to an Air Plane there might be no reason to include a Co-Pilot (no availability or not needed for the flight for instance) so in such case there is no need to give a second Pilot objet to assign a co Pilot to the flight (second constructor take the two Pilot objet though).

Getters are given as method in this class as well as a toString method. Finally a licenseRequired method is assigning the license required to fly that specific airplane (base in a criteria subjectively chosen by us according to the complexity of the planes).

CCTAir Class

This class is the Main Class, and as a result and in the development of the project it was the point to collide the 3 of us together and put the individual work to run, so we decided to do this class as part of team being everyone involve in this specific one.