

Project of Programming Fundamentals II



UCLM

UNIVERSIDAD DE CASTILLA-LA MANCHA

PROGRAMMING FUNDAMENTALS II,
INFORMATIC ENGINEERING.



Autores:

- Alejandro García Peña [30%]

- Enrique Rodríguez Villafranca [70%]

April 30th, 2018

Index

Analysis of the requirements 2

Design..... 3

 UML 3

 User’s manual..... 4

Analysis of the requirements

Our project is to create a concessionaire to rent or hire cars, that simple, but we have many requirements on the project. First of all, on our project's guideline there was many problems introduce to us. For example, we have many different types of cars which are: hybrid cars, sports cars, vans and standard cars.

Furthermore, we have many differences between this type of cars which has the identifiers of: car registration, mileages, price per day, type of vehicle stored all by the system. As another requirement we have to set the identifier of the manual gears of the hybrid car or if its automatic; and for the cars left we have to know if they have air conditioning.

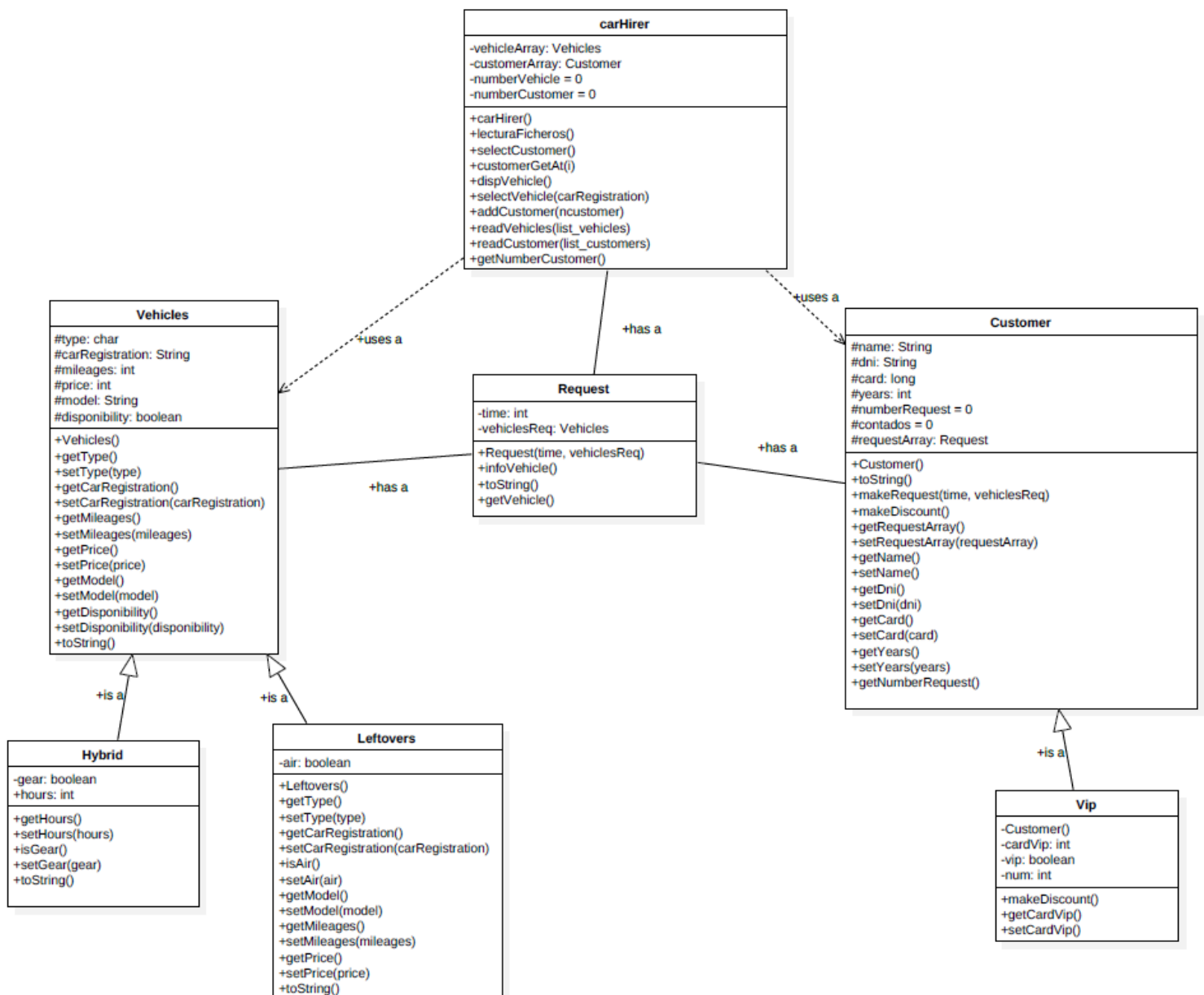
On one hand, we have the cars and all their identifiers who change because of the type of car which these are. On the other hand, we have the customers, which has to indicate their personal data ID or name, credit card number, the identifiers of the vehicle they are interested to hire, the time they want to hire the car, and for every car we have to create a request. And, like for the cars, we have some differences types of clients or customers. We have the standard type of client and a special type of client which is the VIP type. These types of client will obtain a 20% discount if they want to hire a car, and to this discount we would apply also the discount of the customers who has their car licenses more than 10 years that it's a 5% of the original price, so the VIP clients will have a discount of 25%, when the standard customers will have a discount of 5% if they have their car license for more than 10 years, in both cases.

And like we said before, the customers will be able to hire the cars, but there are other options which are required into the guideline of the project, so the clients should be able to: Look for vehicles to hire (these vehicles are all save in a file that was given to us), make a request to rent a car (if the client is not register we should offer him or her that option), we should allow the clients to see how many vehicles each one hired and at last we have to show the types of discount a specific customer has.

All these requirements will be carried by an option menu where the customer will interact with the system, and whatever the information a client decided to introduce, the program or system will be not case sensitive between lower and upper case.

Also, the program will be able to catch at least a few exceptions which are: whether or not the files that contain the data of the customers and the vehicles exist, whether or not the options selected on the menu are in the range of valid options (if this option is not in the range, the program will ask the customer to enter another option without aborting the program), whether the customer's ID or DNI is entered and it doesn't consist of eight numbers and one letter at the end and whether when awaiting of numeric data the client introduce non-numeric data.

Design: UML

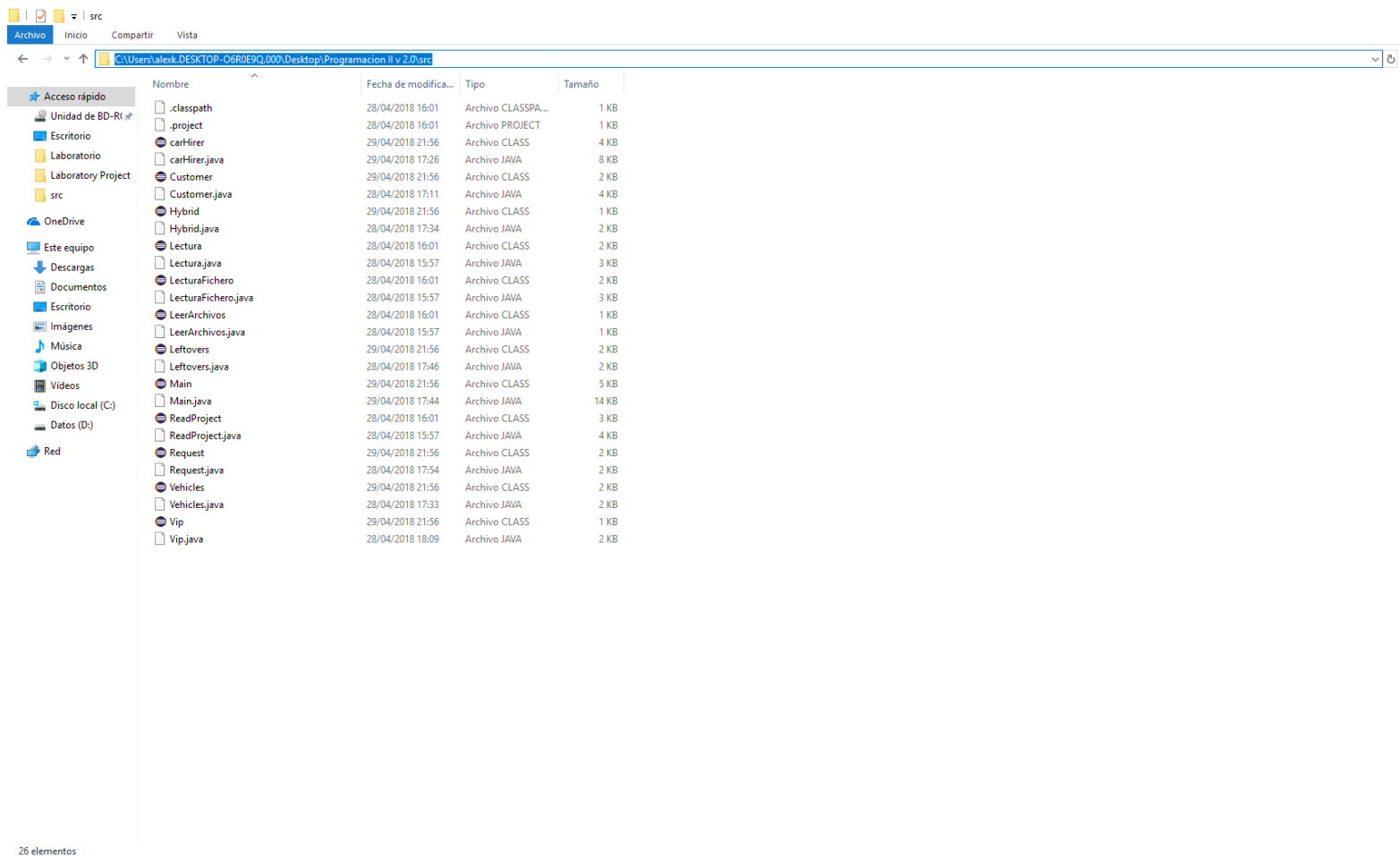


User's manual

If we want to open our project on the console of the system (cmd) we can follow the next steps:

First of all we have to open the cmd. To do that we should press **(Windows + R)** and write on the new execution screen that will appears in front of us (cmd). Now we have to write the directory of the file, but before we have to write the command: **cd "Here we will put the directory"**. The directory we have it's the one highlighted in blue.

We should write this, in my case: **cd C:\Users\alexx.DESKTOP-06R0E9Q.000\Desktop\Programacion II v 2.0\src**



Now we have to call the program "Main.java". To do that we must write: **javac Main.java**

Finally we have to call the file "Main.class". To do that we should write: **java Main**. In this case without the extension's name.