| CLIENT | multinational automotive company |
| --- | --- |
| USER | Concessionaire |
| FUNCTIONAL REQUIREMENTS | R1:  add a vehicule in the concessionaire  R2:  Calculate the total sold price of a vehicule  R3:  Print the information of the vehicules in the concessionaire  R4:  based on the id of a vehicle print the status of the document and a list of the codes of each document  R5: Created and print the parking lot of the concessionaire with its occupation  R6:  Generate parking lot occupancy reports according to the following criteria:   * A: List of vehicles (and their information) given a range of years. * B. Data of the oldest and newest vehicle. * C. Percentage of occupation of the parking lot. |
| PROBLEM | A multinational automotive company wants to open a dealership in Cali and they handle various types of vehicles. So they need a program that allows them to register the different types of vehicles that enter the dealership for sale and along with these vehicles must go the registration of the documentation for each one of them. The dealer needs to be able to consult vehicle information, register vehicles, have a parking lot record and together with this the information of the cars in the between a range of years and the oldest car in terms of model |

| NAME | R1:  add a vehicule in the concessionaire | | |
| --- | --- | --- | --- |
| SUMMARY | The user is asked to register the data of the vehicle to be added to the dealer and it is saved in an arrayList | | |
| INPUTS | Name | Type | Selection condition |
| id | int | ALL THE INPUTS NEED TO BE INITIALIZED |
| basePrice | double |
| mark | String |
| model | int |
| cylinderCapacity | double |
| milage | double |
| type | TYPEVEHICULE |
| plate | String |
| propertyC | Document |
| soat  l | Soat |
| typeAutomobiles | TYPEAUTOMOBILE |
| typeGasoline | TYPEGASOLINE |
| consumptionGasoline | double |
| typeCharger | TYPECHARGER |
| duration | double |
| consumptionElectric | double |
|  | mechanical | MechanicalReview |
| RESULT | the vehicle is added to the arrayList | | |
| OUTPUTS | Name | Type | Selection condition |
| Only continue and display the menu options | | |

| NAME | R2:  Calculate the total sold price of a vehicule | | |
| --- | --- | --- | --- |
| SUMMARY | With the inputs register before, the concessionaire automatically calculates the total selling price | | |
| INPUTS | Name | Type | Selection condition |
| soat | Soat | ALL THE INPUTS NEED TO BE INITIALIZED |
| mechanical | MechanicalReview |
| basePrice | double |
| RESULT | the vehicle is added to the arrayList | | |
| OUTPUTS | Name | Type | Selection condition |
| soldPrice | double | BE INITIALIZED |

| NAME | R3:  Print the information of the vehicules in the concessionaire | | |
| --- | --- | --- | --- |
| SUMMARY | The user is asked about the type of vehicle and its type of gasoline (depending on his first choice) to print the information of all vehicles with these characteristics | | |
| INPUTS | Name | Type | Selection condition |
| typeVehicule | int | ALL THE INPUTS NEED TO BE INITIALIZED |
| typeCombustible | int |
| RESULT | the vehicle is added to the arrayList | | |
| OUTPUTS | Name | Type | Selection condition |
| with a System.out.println  the program print all the information | | |

| NAME | R4:  based on the id of a vehicle print the status of the document and a list of the codes of each document | | |
| --- | --- | --- | --- |
| SUMMARY | The user is asked about the id of the vehicule of which you want to see the status of the documents and their codes | | |
| INPUTS | Name | Type | Selection condition |
| id | int | INPUTS NEED TO BE INITIALIZED |
| RESULT | Show the state of the documents and their codes | | |
| OUTPUTS | Name | Type | Selection condition |
| documentsCode | String | !=null |

| NAME | R5: Created and print the parking lot of the concessionaire with its occupation | | |
| --- | --- | --- | --- |
| SUMMARY | the parking lot is created with columns for each corresponding year and they are filled with used cars that have a model less than 2015 | | |
| INPUTS | Name | Type | Selection condition |
| does not receive any input, it only uses the data registered in requirement 1 and these are the type of vehicle, the type of gasoline and its model | | |
| RESULT | Show the parking | | |
| OUTPUTS | Name | Type | Selection condition |
| parking | String | !=null |

| NAME | R6:  Generate parking lot occupancy reports according to the following criteria:   * A: List of vehicles (and their information) given a range of years | | |
| --- | --- | --- | --- |
| SUMMARY | the user is asked for a range of years and according to this range, he searches inside the parking lot if there are vehicles in it that correspond to that age and prints his information | | |
| INPUTS | Name | Type | Selection condition |
| year1 | int | !=null |
| year2 | int |
| RESULT | Show the information of the cars in the range of years | | |
| OUTPUTS | Name | Type | Selection condition |
| with a System.out.println  the program print all the information | | |

| NAME | R6:  Generate parking lot occupancy reports according to the following criteria:   * B. Data of the oldest and newest vehicle. | | |
| --- | --- | --- | --- |
| SUMMARY | the system search for the car with the oldest model in the parking and print his information | | |
| INPUTS | Name | Type | Selection condition |
| does not receive any input, it only uses the data registered in requirement 1 and these are the type of vehicle, the type of gasoline and its model | | |
| RESULT | Show the information of the oldest car in the parking | | |
| OUTPUTS | Name | Type | Selection condition |
| printer | String |  |

| NAME | R6:  Generate parking lot occupancy reports according to the following criteria:   * C. Percentage of occupation of the parking lot. | | |
| --- | --- | --- | --- |
| SUMMARY | the system count the num of cars in the parking and in base of the capacity of the parking calculate the occupation percentage | | |
| INPUTS | Name | Type | Selection condition |
| does not receive any input, in the request to print the parking lot, the number of vehicles in it is counted, so the calculation is done in the same method | | |
| RESULT | Show the occupation percentage of the parking | | |
| OUTPUTS | Name | Type | Selection condition |
| ocupationPercentage | double | BE INITIALIZED |