**1- \*\*UML Use-Case \*\*Diagram and Description.**

First, creating a file to the customer will help with another services and ensure the bill is created effectively, and efficiently. It is a use case that relates to the figure provided since it will store the information and ease the process.

|  |  |
| --- | --- |
| Use case: | Create a file |
| Trigger: | The customer wants to have a file to store his information before doing the services on the garage. |
| Precondition: | The customer is authenticated. |
| Main scenarios: |  |
| 1- | The customer reaches out the clerk. |
| 2- | The clerk provides the customer with a list to fill in. |
| 3- | The customer specifies their information. |
| 4- | The clerk enters the information into the system. |
| 5- | The system stores the information provided. |
| 6- | The system displays the information. |
| 7- | The clerk singles to the system that everything is correct to start creating a file |
| 8- | The system creates a file with the information provided. |
| Exceptions: |  |
| 7a | 1- The information is wrong or missing  2- The system notifies the cleck to adjust it with the customer. |

Second, checking the vehicle is an important use case to include since it shows the process the customer will go through. It relates to the figure as it shows the services the customer had at the garage and having that information will guide us with creating the bill later.

|  |  |
| --- | --- |
| Use case: | Check the vehicle |
| Trigger: | The customer needs his vehicle to be checked on the garage. |
| Precondition: | The customer is authenticated |
| Main scenarios: |  |
| 1- | The customer enters their personal details into the system. |
| 2- | The customer enters the services they want. |
| 3- | The system stores the information provided. |
| 4- | The system checks if the services are available. |
| 5- | The system checks who is the responsible mechanic for those services. |
| 6- | The system notifies the mechanic that they have a customer. |
| 7- | The mechanic does the services. |
| Exceptions: |  |
| 4a | 1-The services are not available.  2-The customer chooses a different service. |
| 5a | 1- The mechanic is not available.  2- The system looks for a substitute. |

Thirdly, the receipt or invoice will be created which is the most important use case that will help the car garage manage their billing system effectively. It ensures the information, and calculations are provided, and created appropriately.

|  |  |
| --- | --- |
| Use Case: | Create a receipt |
| Trigger: | The car garage wants to create a receipt to manage bills. |
| Precondition: | The customer is authenticated |
| Main scenarios: |  |
| 1- | The customer enters their personal details. |
| 2- | The clerk receives the customer’s information. |
| 3- | The clerk check who was the mechanic. |
| 4- | The clerk enters the mechanic’s name and date of the services. |
| 5- | The clerk ensures all the information are entered and appropriate. |
| 6- | The system stores the information. |
| 7- | The clerk enters the customer chosen services into the system. |
| 8- | The system displays the amount of each service. |
| 9- | The system calculates the total amount and taxes. |
| 10- | The system calculates discounts. |
| 11- | The system displays the final number of services. |
| 12- | The system creates the bill with all the information. |
| 13- | The clerk prints the bill and upload it to the system. |
| Exceptions: |  |
| 10a | The customer might not have a discount. |

**2- \*\*UML Class \*\*Diagram and Description.**

List of Classes and attributes:

-Customer: firstName, lastName, cellPhoneNumber, gender, dateOfBirth.

-Service: serviceName, servicePrice, mechanicName, date, numberOfServices.

-Vehicle: type, color, id, yearOfVehicle, make.

-Car: doors, type, color, id, yearOfVehicle, make.

-Price: taxes, total, discount, finalAmount, servicePrice.

—------------------------

|  |
| --- |
| **James: Customer** |
| -firstName=”James"  -lastName="W.Jones"  -cellPhoneNumber:"816-897-9862"  -gender=Gender.Male  -dateOfBirth=[1990-10-05] |

The class customer has 5 attributes with their type and their setter and getter function.

**Object1**

|  |
| --- |
| Customer |
| -firstName: String  -lastName: String  -cellPhoneNumber: String  -gender: ENUM  -dateOfBirth: Date |
| +setFirstName(firstName:String)  +getFirstName():String  +setLastName(lastName:String)  +getLastName():String  +setCellPhoneNumber(cellPhoneNumber:String)  +getCellPhoneNumber():String  +setGender(gender:Gender)  +getGender():ENUM  +setDateOfBirth(dateOfBirth:Date)  +getDateOfBirth():Date |

The class Service has 5 attributes with their type

and their setter and getter function are included.

**Object 1**

|  |
| --- |
| Service |
| -serviceName: String  -servicePrice: Float  -mechanicName: String  -dateOfService: Date  -numberOfService: Integer |
| +setServiceName(serviceName:String)  +getServiceName():String  +setServicePrice(servicePrice:Float)  +getServicePrice():Float  +setMechanicName(mechanicName:String)  +getMechanicName():String  +setDateOfService(dateOfService:Date)  +getDateOfService():Data  +setNumberOfService(numberOfService:Integer)  +getNumberOfService():Integer |

|  |
| --- |
| **Service1: Service** |
| -serviceName="Diagnostics"  -servicePrice=”15”  -mechanicName=”Hans K"  -dateOfService="March 13, 2022"  -numberOfService=”1” |

**Object 2**

|  |
| --- |
| **Service2: Service** |
| -serviceName="Oil Replacement"  -servicePrice=”120”  -mechanicName=”Hans K"  -dateOfService="March 13, 2022"  -numberOfService=”2” |

**Object 3**

|  |
| --- |
| **Service3: Service** |
| -serviceName="Oil Filter Parts"  -servicePrice=”35”  -mechanicName=”Hans K"  -dateOfService="March 13, 2022"  -numberOfService=”3” |

**Object 4**

|  |
| --- |
| **Service4: Service** |
| -serviceName="Tire Replacement (2)"  -servicePrice=”100”  -mechanicName=”Hans K"  -dateOfService="March 13, 2022"  -numberOfService=”4” |

**Object 5**

|  |
| --- |
| **Service5: Service** |
| -serviceName="Tire (2)"  -servicePrice=”160”  -mechanicName=”Hans K"  -dateOfService="March 13, 2022"  -numberOfService=”5” |

|  |
| --- |
| Vehicle |
| -model: ENUM  -color: ENUM  -id: String  -yearOfVehicle: Integer  -make: ENUM |
| +setModel(model:Model)  +getModel():ENUM  +setColor(color:Color)  +getColor():ENUM  +setID(id:String)  +getID():String  +setYearOfVehicle(yearOfVehicle:Integer)  +getYearOfVehicle():Integer  +setMake(make:Make)  +getMake():ENUM |

Table

Description automatically generated

Child

Parent

The class Car has 6 attributes with their type and their setter and getter function are included. It is the child class of the class Vehicle, and it inherits its attributes. One attribute “numberOfDoors” is related to Cars specifically since the vehicle can be motorcycles, boats, or airplanes, where the concept of doors may not apply. There is a “is a” relationship between them where “A car is a Vehicle”, and it is a Single inheritance. The class Vehicle has 5 attributes with their type and their setter and getter function are included. It is the parent class of the class car.

**Object 1 Object 1**

|  |
| --- |
| **Vehicle1: Vehicle** |
| -model=Model.Altima  -color=Color.Silver  -id=”AD-89034”  -yearOfVehicle=”2014”  -make=Make.Nissan |

|  |
| --- |
| **Car1: Car** |
| -model=Model.Altima  -color=Color.Silver  -id=”AD-89034”  -yearOfVehicle=”2014”  -make=Make.Nissan  -numberOfDoors=”4” |

The class Price has 5 attributes with their type and their setter and getter function are included.

|  |
| --- |
| Price |
| -taxes: Float  -total: Float  -discount: Float  -finalAmount: Float  -servicePrice: Float |
| +setTaxes(taxes:Float)  +getTaxes():Float  +setTotal(total:Float)  +getTotal():Float  +setDiscount(discount:Float)  +getDiscount():Float  +setFinalAmount(finalAmount:Float)  +getFinalAmount():Float  +setServicePrice(servicePrice:Float)  +getServicePrice():Float |

The other relationships between those classes might be as an association, but not inheritance, the only one is the vehicle and the car.

**Object 1 Object 2**

|  |
| --- |
| **Price2: Price** |
| -taxes=”21.5”  -total=”451.5”  -discount=”11.5”  -finalAmount=”440”  -servicePrice=”120” |

|  |
| --- |
| **Price1: Price** |
| -taxes=”21.5”  -total=”451.5”  -discount=”11.5”  -finalAmount=”440”  -servicePrice=15” |

**Object 3 Object 4**

|  |
| --- |
| **Price3: Price** |
| -taxes=”21.5”  -total=”451.5”  -discount=”11.5”  -finalAmount=”440”  -servicePrice=”35” |

|  |
| --- |
| **Price4: Price** |
| -taxes=”21.5”  -total=”451.5”  -discount=”11.5”  -finalAmount=”440”  -servicePrice=”100” |

**Object 5**

|  |
| --- |
| **Price5: Price** |
| -taxes=”21.5”  -total=”451.5”  -discount=”11.5”  -finalAmount=”440”  -servicePrice=”160” |