

GitHub Link: <https://github.com/MaryamAlwars/Assignment-1-Software-Modelling->

Draw.io Link: https://app.diagrams.net/#G1c2W28QFcXIspHZshtwSK_iEU-UMYW-IN

Classes, attributes and functions

Customer

- firstName: String
- lastName: String
- phoneNumber: int
- email: String
- address: String

- + getFirstName(): String
- + setFirstName(firstName : String)
- + getLastName(): String
- + setLastName(lastName : String)
- + getPhoneNumber(): Integer
- + setPhoneNumber(phoneNumber : Integer)
- + getEmail(): String
- + setEmail(email : String)
- + getAddress(): String
- + setAddress(address : String)

Vehicle

- make: ENUM
- type: ENUM
- color: ENUM
- ID: String
- licensePlate: String

- + getMake(): ENUM
- + setMake(make : Make)
- + getType(): ENUM
- + setType(type : Type)
- + getColor(): ENUM
- + setColor(color : Color)
- + getID(): String
- + setID(ID : String):
- + getLicensePlate(): String
- + setLicensePlate(licensePlate: String):

Car (it inherits from Vehicle class)

- seaters: ENUM 2,4,5,7,8

- + setSeaters(seaters: Seaters)
- + getSeaters(): ENUM

Service

- serviceName: String
- description: String
- partsCost: Float
- laborCost: Float
- totalPrice: Float

- + getServiceName(): String
- + setServiceName(serviceName : String)
- + getDescription(): String
- + setDescription(description : String)
- + getPartsCost(): Float
- + setPartsCost(partsCost : Float)
- + getLaborCost(): Float
- + setLaborCost(laborCost : Float)
- + getTotalPrice(): Float
- + setTotalPrice(totalPrice : Float)

Price

- taxes: Float
- totalPriceWithoutDiscount: Float
- discount: Float
- profit: Float ←(modify)
- finalAmount: Float

- + getTaxes(): Float
- + setTaxes(taxes : Float)
- + getTotalPriceWithoutDiscount(): Float
- + setTotalPriceWithoutDiscount(totalPriceWithoutDiscount : Float)
- + getDiscount(): Float
- + setDiscount(discount : Float)
- + getProfit(): Float
- + setProfit(profit : Float)
- + getFinalAmount(): Float
- + setFinalAmount(finalAmount : Float)

Use case diagram:

Actors:

- Appointment and Billing team
- Serviceman
- Customer

Use cases:

- Create a customer database
- Manage appointments
- Diagnose the vehicle
- Repair the vehicle
- Generate invoices