INFORMATICS INSTITUTE OF TECHNOLOGY

UNIVERSITY OF WESTMINSTER

Informatics Institute of Technology

Department of Computing

(B.Sc.) in Computer Science

Module: 5COSC007C.1 Object Oriented Programming

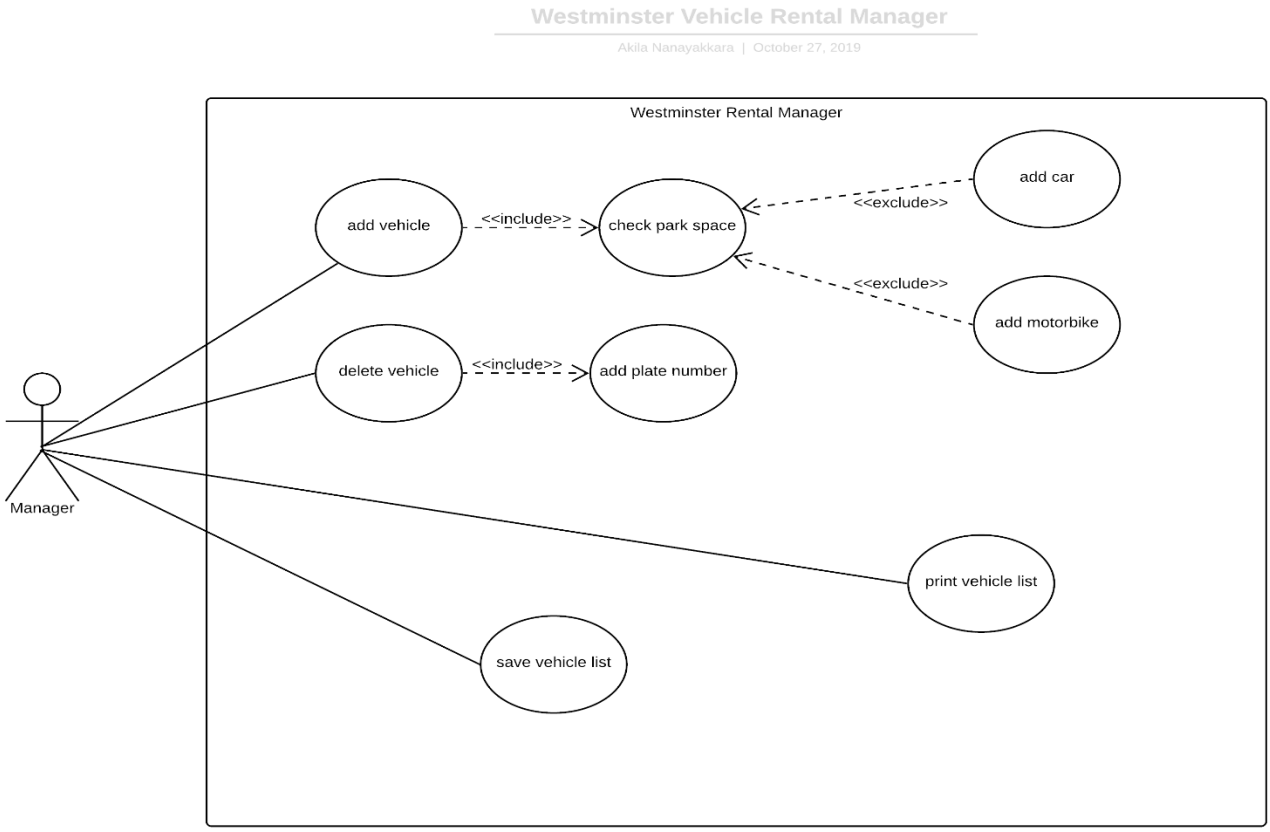
Coursework 1

Phase 1

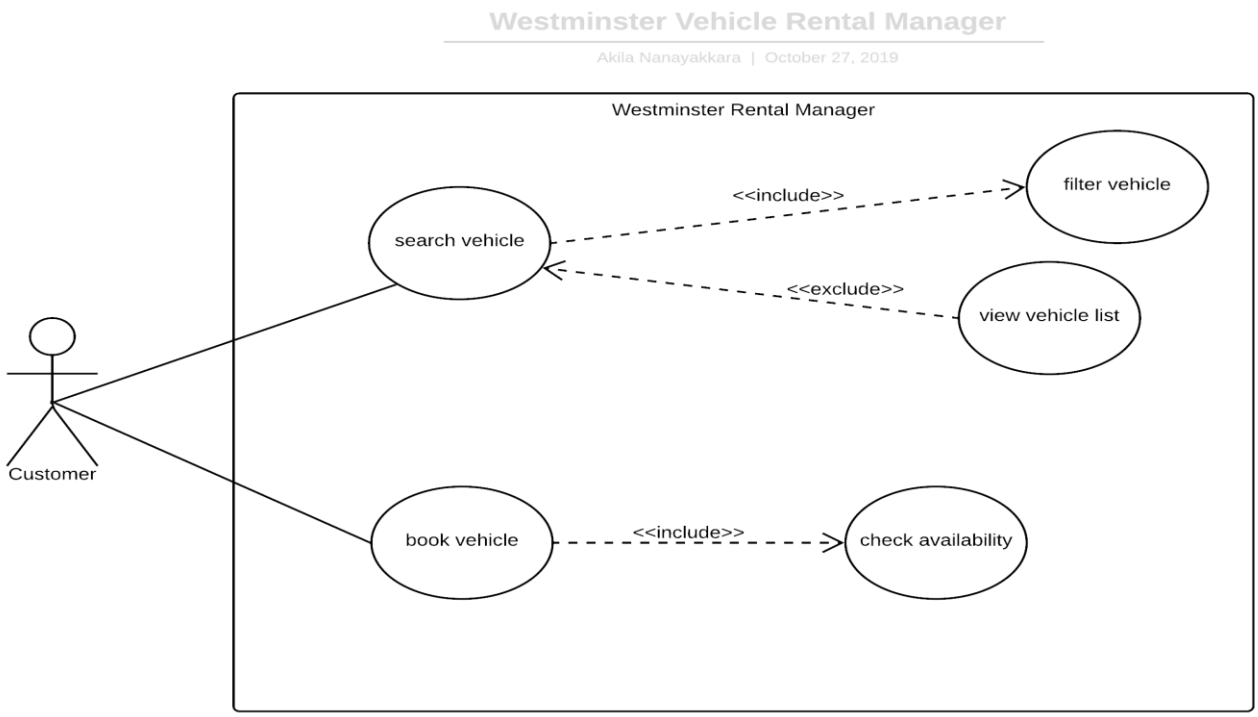
Date	: 27/10/2019
Student ID	: 2018400
Student UoW ID	: w1742308
Student First Name	: Akila
Student Surname	: Nanayakakra

Table of Contents

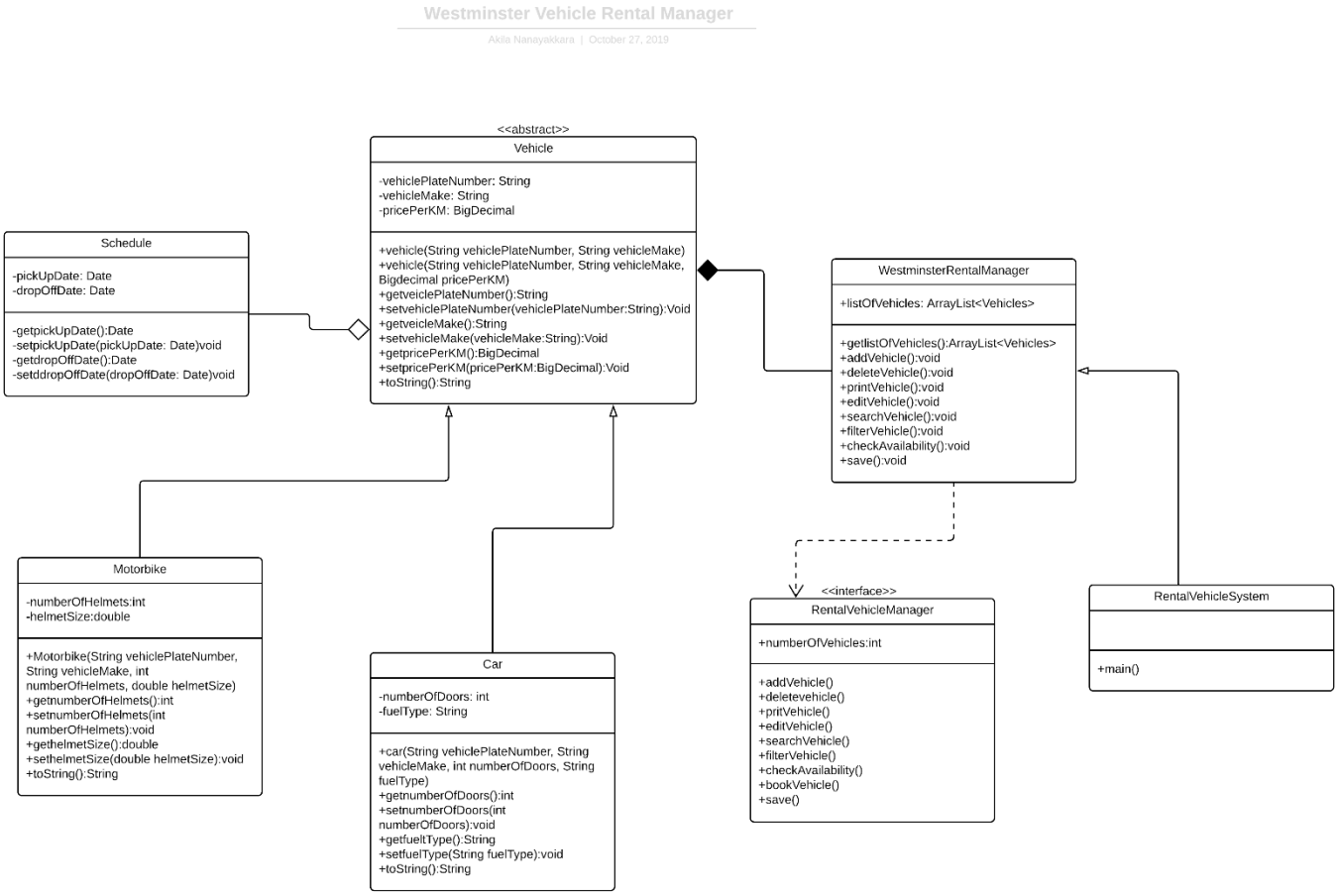
Use case diagram for the console application



Use case diagram for the GUI application



Class diagram



Code

Vehicle Class

```
import java.math.BigDecimal;

public abstract class Vehicle {

    private String vehiclePlateNumber;
    private String vehicleMake;
    private BigDecimal pricePerKM;

    public Vehicle(String vehiclePlateNumber, String vehicleMake) {
        this.vehiclePlateNumber = vehiclePlateNumber;
        this.vehicleMake = vehicleMake;
    }

    public Vehicle(String vehiclePlateNumber, String vehicleMake, BigDecimal pricePerKM) {
        this.vehiclePlateNumber = vehiclePlateNumber;
        this.vehicleMake = vehicleMake;
        this.pricePerKM = pricePerKM;
    }

    public String getVehiclePlateNumber() {
        return vehiclePlateNumber;
    }

    public void setVehiclePlateNumber(String vehiclePlateNumber) {
        this.vehiclePlateNumber = vehiclePlateNumber;
    }

    public String getVehicleMake() {
        return vehicleMake;
    }

    public void setVehicleMake(String vehicleMake) {
        this.vehicleMake = vehicleMake;
    }

    public BigDecimal getPricePerKM() {
        return pricePerKM;
    }

    public void setPricePerKM(BigDecimal pricePerKM) {
        this.pricePerKM = pricePerKM;
    }

    public abstract String vehicleType();

    @Override
    public String toString(){
        return "Plate number: " + this.vehiclePlateNumber + "Vehicle Make: " +
this.vehicleMake;
    }
}
```

Car class

```
public class Car extends Vehicle {

    private int numberOfDoors;
    private String fuelType;

    public Car(String vehiclePlateNumber, String vehicleMake, int numberOfDoors, String
fuelType) {
        super(vehiclePlateNumber, vehicleMake);
        this.numberOfDoors = numberOfDoors;
        this.fuelType = fuelType;
    }

    public int getNumberOfDoors() {
        return numberOfDoors;
    }

    public void setNumberOfDoors(int numberOfDoors) {
        this.numberOfDoors = numberOfDoors;
    }

    public String getFuelType() {
        return fuelType;
    }

    public void setFuelType(String fuelType) {
        this.fuelType = fuelType;
    }

    public String vehicleType(){
        return "Car";
    }

    @Override
    public String toString(){
        return super.toString()+ "Number of Doors: " + this.numberOfDoors + "\n" +
                "Fuel type: " + this.fuelType;
    }
}
```

Motorbike class

```
public class Motorbike extends Vehicle {

    private int numberOfHelmets;
    private double helmetSize;

    public Motorbike(String vehiclePlateNumber, String vehicleMake, int numberOfHelmets,
double helmetSize) {
        super(vehiclePlateNumber, vehicleMake);
        this.numberOfHelmets = numberOfHelmets;
        this.helmetSize = helmetSize;
    }

    public int getNumberOfHelmets() {
        return numberOfHelmets;
    }

    public void setNumberOfHelmets(int numberOfHelmets) {
        this.numberOfHelmets = numberOfHelmets;
    }

    public double getHelmetSize() {
        return helmetSize;
    }

    public void setHelmetSize(double helmetSize) {
        this.helmetSize = helmetSize;
    }

    public String vehicleType(){
        return "Bike";
    }

    @Override
    public String toString(){
        return super.toString()+"Number of helmets: " + this.numberOfHelmets+ ", Helmet
Size: " + this.helmetSize;
    }


}
```

Schedule class

```
import java.util.Date;

public class Schedule {
    private Date pickUpDate;
    private Date DropOffDate;

    public Date getPickUpDate() {
        return pickUpDate;
    }

    public void setPickUpDate(Date pickUpDate) {
        this.pickUpDate = pickUpDate;
    }

    public Date getDropOffDate() {
        return DropOffDate;
    }

    public void setDropOffDate(Date dropOffDate) {
        DropOffDate = dropOffDate;
    }


    @Override
    public String toString(){
        return super.toString()+"Pickup Date: " + this.pickUpDate+ ", Drop off date: " +
this.DropOffDate;
    }
}
```

WestminsterRentalManager class

```
import java.util.ArrayList;
import java.util.Scanner;

public class WestminsterRentalManager implements RentalVehicleManager{

    public ArrayList<Vehicle> listOfVehicles;

    public WestminsterRentalManager(){
        listOfVehicles = new ArrayList<Vehicle>();
    }

    public void addVehicle (Vehicle vehicle){
        if(listOfVehicles.size()<numberOfVehicles){
            listOfVehicles.add(vehicle);
        }else {
            System.out.println("No More Parking!!");
        }
    }

    //adding vehicle method
    @Override
    public void addVehicle(){

        Scanner scanner = new Scanner(System.in);

        //Checking if the space if available
        System.out.print("\n" +
                "Select vehicle type" +
               "\n" +
                "1. Car \n" +
                "2. Motorbike \n" +
                "Choose: ");
        while (!scanner.hasNextInt()){
            System.out.println("Invalid Data Type!!!");
            scanner.next();
            System.out.print("Select vehicle type" +
                    "\n" +
                    "1. Car \n" +
                    "2. Motorbike \n"+
                    "Choose: ");
        }
        int optionVehicle = scanner.nextInt();

        if (optionVehicle==1){

            Scanner carOptionScanner = new Scanner(System.in);

            //Input car plate number
            System.out.print("Please enter the plate number: ");
            String carPlateNumber = carOptionScanner.nextLine();

            //Input car make
            System.out.print("Please enter the make: ");
            String carMake = carOptionScanner.nextLine();

            //Input number of car doors
```

```
            System.out.print("Please Enter the number of doors: ");
            while (!carOptionScanner.hasNextInt()){
                System.out.println("Invalid Data Type !!!");
                carOptionScanner.nextLine();
                System.out.print("Please enter the number of doors: ");
            }
            int numberOfCarDoors = carOptionScanner.nextInt();
            carOptionScanner.nextLine();

            //Input the fuel type
            System.out.print("Please enter the fuel type: ");
//            while(!(carOptionScanner.equals("95") || carOptionScanner.equals("92"))){
//                System.out.println("Insert the correct fuel type!!!");
//                carOptionScanner.next();
//                System.out.print("Please enter the fuel type: ");
//            }
            String carFuelType = carOptionScanner.nextLine();

            Car car = new Car(carPlateNumber, carMake, numberOfCarDoors, carFuelType);
            addVehicle(car);
        }

        else if(optionVehicle==2){
            Scanner motorbikeOptionScanner = new Scanner(System.in);

            //Input bike number plate
            System.out.print("Please enter plate number: ");
            String motorbikeNumber = motorbikeOptionScanner.nextLine();

            //Input bike make
            System.out.print("Please enter make: ");
            String motorbikeMake = motorbikeOptionScanner.nextLine();

            //Input the number of helmets
            System.out.print("Please enter the number of helmets: ");
            while (!motorbikeOptionScanner.hasNextInt()){
                System.out.println("Invalid Data type");
                motorbikeOptionScanner.next();
                System.out.print("Please enter the number of helmets: ");
            }
            int bikeNumberOfHelmets = motorbikeOptionScanner.nextInt();

            //Input the helmet size
            System.out.print("Please enter the helmet size: ");
            while (!motorbikeOptionScanner.hasNextDouble()){
                System.out.println("Invalid Data type");
                motorbikeOptionScanner.next();
                System.out.print("Please enter the helmet size: ");
            }
            double bikeHelmetSize = motorbikeOptionScanner.nextDouble();

            Motorbike motorbike = new Motorbike(motorbikeNumber, motorbikeMake,
bikeNumberOfHelmets, bikeHelmetSize);
            addVehicle(motorbike);
        }
        else {
            System.out.println("Invalid vehicle option!!! ");
        }
```

```
    }

    @Override
    public void deleteVehicle() {
        Scanner deleteVehicleScanner = new Scanner(System.in);
        System.out.print("Please enter the plate number of the vehicle that you want to
remove: ");
        String deleteplateNumber = deleteVehicleScanner.nextLine();

        for(Vehicle vehicle : listOfVehicles){

        if(vehicle.getVehiclePlateNumber().equals(deleteplateNumber)){
            String vehicleType = vehicle.vehicleType();
            listOfVehicles.remove(vehicle);
            System.out.println(vehicleType+" deleted.");
            System.out.println(listOfVehicles.size()+ " spaces left in the park.");
            break;
        }else{
            System.out.println("Plate number does not exist!!");
            }
        }

    }

    @Override
    public void printVehicle() {
        System.out.format("_%1$-20s_%2$-20s_%3$-
20s\n","____________________","____________________","____________________");


        System.out.format("|%1$-20s|%2$-20s|%3$-20s|\n","       TYPE","    PLATE NUMBER","
MAKE ");

        System.out.format("_%1$-20s_%2$-20s_%3$-
20s\n","____________________","____________________","____________________");
        System.out.format("%1$-20s%2$-20s%3$-20s\n","","","");

        for(int i=0; i<listOfVehicles.size(); i++){
            if(listOfVehicles.get(i).vehicleType().equals("Car")){
                System.out.format("|%1$-20s|%2$-20s|%3$-20s|\n","       Car         ","
"+listOfVehicles.get(i).getVehiclePlateNumber(),"
"+listOfVehicles.get(i).getVehicleMake());
                System.out.format("_%1$-20s_%2$-20s_%3$-
20s\n","____________________","____________________","____________________");
            }else {
                System.out.format("|%1$-20s|%2$-20s|%3$-20s|\n","       MotorBike ","
"+listOfVehicles.get(i).getVehiclePlateNumber(),"
"+listOfVehicles.get(i).getVehicleMake());
                System.out.format("_%1$-20s_%2$-20s_%3$-
20s\n","____________________","____________________","____________________");
            }
        }

    }

    @Override
    public void editVehicle(){
    }
```

```
    @Override
    public void searchVehicle(){
    }

    @Override
    public void filterVehicle(){
    }

    @Override
    public void checkAvailability(){
    }

    @Override
    public void bookVehicle(){
    }

    @Override
    public void save() {
    }
}
```

RentalVehicleManager class

```
public interface RentalVehicleManager {

    int numberOfVehicles = 3;
    void addVehicle();
    void deleteVehicle();
    void printVehicle();
    void editVehicle();
    void searchVehicle();
    void filterVehicle();
    void checkAvailability();
    void bookVehicle();
    void save();
}
```

RentalVehicleSystem class

```
import java.util.Scanner;

public class RentalVehicleSystem {

    public static void main(String[] args) {
        WestminsterRentalManager rent = new WestminsterRentalManager();

        Scanner mainMenuScanner = new Scanner(System.in);

        int menuOption = 0;

        while (true){
            System.out.println(" \n" +
                    "      ____________________________________________________ \n" +
                    "      Welcome to the Westminster Rental Vehicle Manager \n" +
                    "      ---------------------------------------------- \n" +
                    "\n" +
                    "1. Add Vehicle \n" +
                    "2. Delete Vehicle \n" +
                    "3. Print Vehicle List \n" +
                    "4. Edit Vehicle List \n" +
                    "5. Open the Console \n"+
                    "6. Exit the programme \n" +
                    "\n");
                System.out.print("Choose an option: ");
                while (!mainMenuScanner.hasNextInt()){
                    String wrongdatatype = mainMenuScanner.next();
                    System.out.println( wrongdatatype + " is an invalid data type!!");
                    System.out.print("Choose an option: ");
                }
                menuOption = mainMenuScanner.nextInt();
                switch (menuOption){
                    case 1:
                        rent.addVehicle();
                        break;
                    case 2:
                        rent.deleteVehicle();
                        break;
                    case 3:
                        rent.printVehicle();
                        break;
                    case 4:
                        //edit vehicle list
                        break;
                    case 5:
                        //open the console
                        break;
                    case 6:
                        System.out.println("------->> Program End <<--------");
                        System.exit(0);
                    default:
                        System.out.println("Please the choose the correct option!!");
                }
        }
    }
}
```