

FIGURE 10-34 Overall Design of the Vehicle Rental Agency Program

The Vehicle class has three subclasses—Car, Van, and Truck. It is responsible for maintaining a vehicle’s type, its VIN, and its reservation status. Method `getDescription` is provided in the Vehicle class to return the information common to all vehicles: miles per gallon, and a VIN. Each subclass builds on this inherited method to include the specific information for that vehicle type. The Car class stores the maximum number of passengers and number of doors, the Van class stores the maximum number of passengers, and the Truck class stores its length and the number of rooms of storage it can hold.

The VehicleCost class does not have any subclasses. Its `create (__init__)` method is passed six arguments: the daily/weekly/weekend rates, the number of free miles, the per mile charge, and the daily insurance rate to initialize the object with. The `getVehicleCost` method of the VehicleCosts aggregating class returns the cost of a specified vehicle type as a single string for display. The Reservation and corresponding Reservations aggregator class are designed in a similar manner.

Finally, a SystemInterface class provides all the methods that any user interface would need for interacting with the system. Such a set of methods is referred to as an API—*Application Programming Interface*. Thus, the SystemInterface object is created first. It then reads all the vehicle rental agency data from text files `VehiclesStock.txt` and `RentalCost.txt` and populates the corresponding objects. Then, an instance of the RentalAgencyUI is created and initialized with a reference to the system interface. The only public method of the

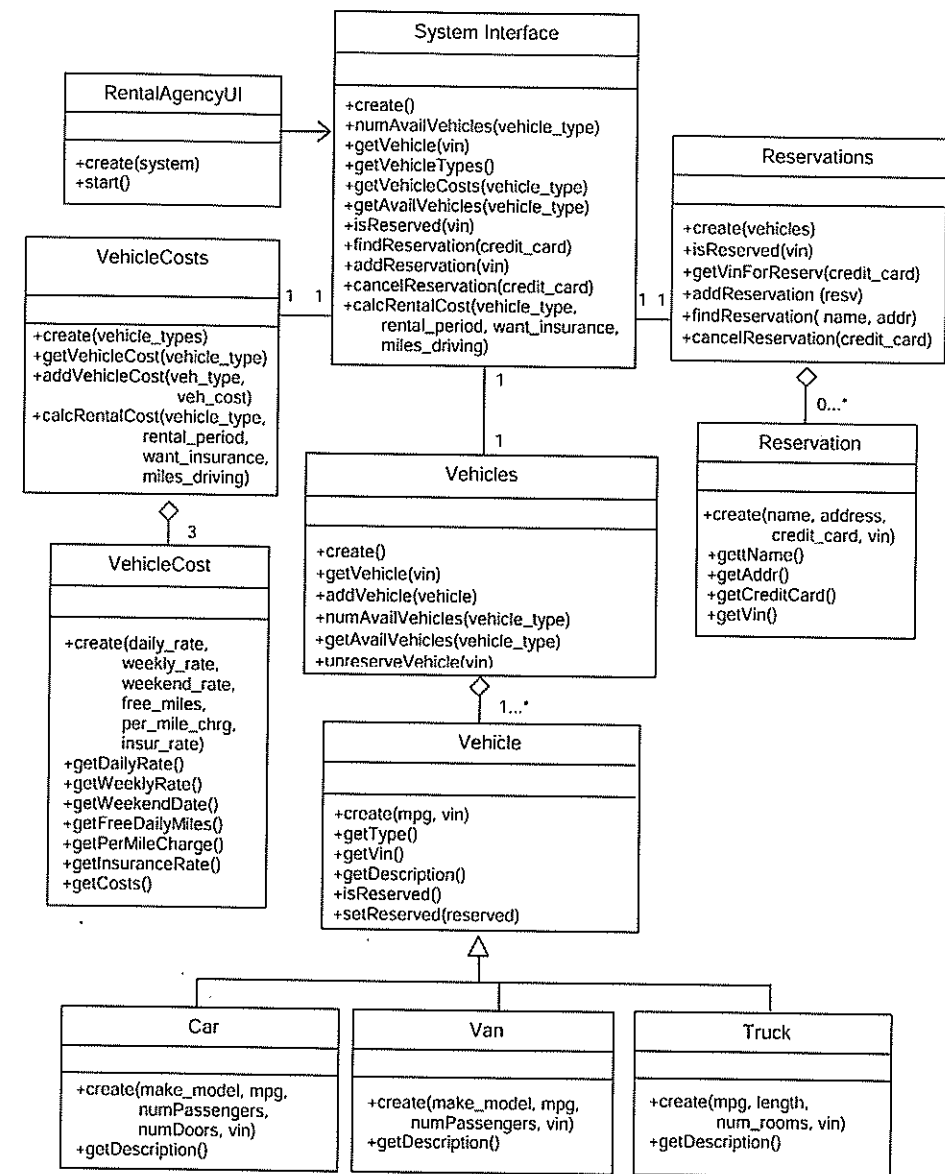


FIGURE 10-35 Class Diagram for Vehicle Rental Agency Program

RentalAgencyUI class, `start`, is called to start the console interaction. The main menu for the program is given in Figure 10-36.

Example use of the system is shown in Figure 10-37. (For the sake of space, the main menu is not repeatedly shown before each selection as in the actual program execution.)