

1. Overview

- 1.1. This assignment is to be undertaken by each student individually.
- 1.2. The assignment is a computer based exercise and report. The exercise will involve implementing and testing a computer program that you have developed. The report will contain details of analysis , design, implementation and testing. Both the exercise and the report need to be undertaken to secure a pass.
- 1.3. You should take the analysis and design part seriously. Your report should show evidence of your understanding of the problem and how you designed and tested your solution.
- 1.4. An **incomplete** Class Diagram has been included (see section 3) to help get you started.

2. Deliverables

- 2.1. You need to hand in the following:

- 2.1.1. A 3.5 inch diskette containing your programs.

- 2.1.2. A report that contains:

- 2.1.2.1. A contents page.

- 2.1.2.2. A Use Case Diagram.

- 2.1.2.3. A Use Case description for each system task.

- 2.1.2.4. Class Diagram(s) with attributes, methods and associations.

- 2.1.2.5. Sequence Diagram for each Use Case.

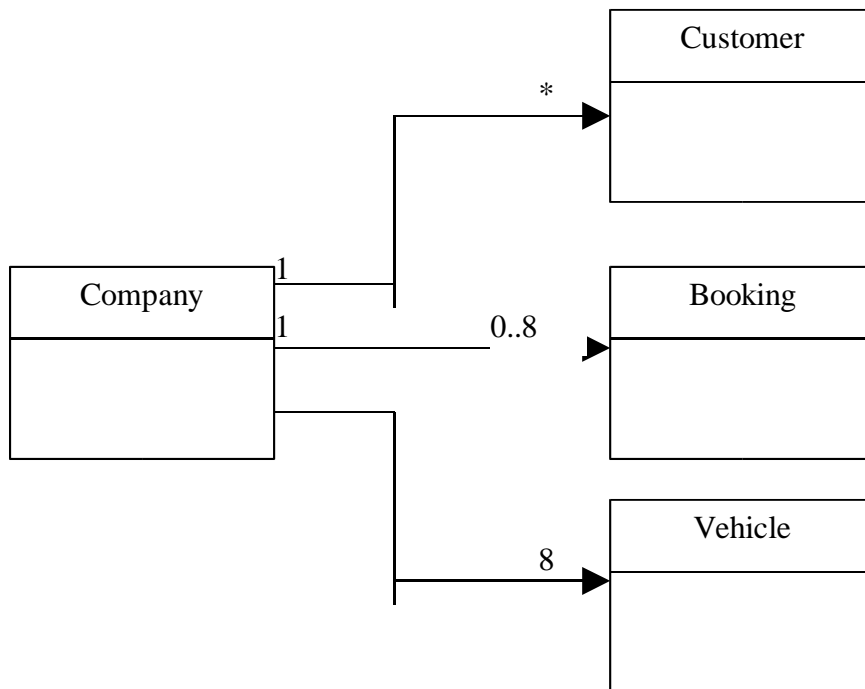
- 2.1.2.6. Table showing the order of development of the Use Cases.

- 2.1.2.7. Test plans and results.

- 2.1.2.8. Java source code listings.

3. Problem Outline

- 3.1. A company hires out vehicles to its customers for short trips on a daily basis and requires a computer program to automate its booking system. (See Figure 3.1).
- 3.2. Customers phone today to book a vehicle for tomorrow.
- 3.3. When a customer phones in, the company displays a list of the vehicles available for hire.
- 3.4. If a customer wishes to choose a vehicle, the company takes the customer's details (name and phone number), and notes the vehicle the customer wants to hire.
- 3.5. The company creates a booking that refers to the vehicle.
- 3.6. Customers sometimes phone the company and cancel their bookings. If a booking is cancelled, the vehicle is made available for hire.
- 3.7. The company can display a list of bookings showing the details of the customers and the vehicles they have hired.
- 3.8. The company's vehicles consist of cars and vans.
- 3.9. All vehicles have a registration number, a model name and an engine size.
- 3.10. Cars are either cabriolet or saloon styles, and have either two or four doors.
- 3.11. Vans have a maximum weight load that they can carry.
- 3.12. The company has five cars and three vans.



**Please note: this Class Diagram is incomplete.
You will need to copy this into your report then add
other classes, methods, attributes etc., as you consider appropriate.**

Figure 3.1 – Incomplete Class Diagram of Vehicle Booking System

4. Tasks

4.1 Study the problem outlined above. Identify the Use Cases and draw a Use Case Diagram (see section 2.1.2.2). Write a Use Case description for each Use Case (see section 2.1.2.3).

(Marks 10%)

4.2 Identify all the classes for your booking system. Extend the Class Diagram in 3.1 to include any additional classes and show the relationships between all the classes (see section 2.1.2.4).

(Marks 15%)

4.3 Draw a Sequence Diagram for each Use Case (see section 2.1.2.5). Add methods and attributes to your class diagram as you complete this task.

(Marks 20%)

4.4 Decide on the order in which you will develop and implement you Use Cases. Write this order in a table format in your report and explain the reasons for your decisions (see section 2.1.2.6).

(Marks 5%)

4.5 For each Use Case identified:

4.5.1 Write the classes, methods and attributes necessary to implement the Use Case. Include the Java listing for your classes involved in this Use Case in your report (see section 2.1.2.8). Copy your source code to your 3.5-inch diskette (see section 2.1.1).

(Marks 40%)

4.5.2 Develop the Test Plans for the Use Case and include it, and its results, in your report (see section 2.1.2.7).

(Marks 10%)

