## UNIVERSITY OF TECHNOLOGY, JAMAICA

## School of Computing and Information Technology

## Object-Oriented Programming (CIT2004) Project

T. Edwards / C. Panther /

Group Assignment (3 - 4 **registered** students to each group)

**Given Week of February 21, 2022** **Due Week of April 8, 2022**

Cars4Rent is a successful vehicle rental company in the parish of Kingston. The company is expecting an increase in vehicle rental during the summer period, due to expected relaxation of COVID-19 restrictions, and the owner would like an automated system through which persons can both view and rent vehicles. You have been asked to implement the proposed system.

There are three types of vehicles, these are Car, Truck, and Bike. For each vehicle, the system records the license plate number, brand, model, year, color, engine size, fuel type (Gas or Diesel), transmission type (A, M or CVT), mileage, the seating capacity, rental status, and rate per day. For cars and trucks, the system stores data on their interior type (Leather or Fabric). For bikes, the number of helmets provided (1 or 2) is stored and for trucks the system tracks the towing capacity in kg.

The four main functions of the application are:

1. View Vehicles
2. Search Vehicles
3. My Rentals
4. Return Vehicle

- **View Vehicles**

The user should be able to view all the vehicles available (see excel sheet for dataset) and select the vehicle they want to rent. All the relevant vehicle information must be displayed in a readable format (e.g. table or grid). After the vehicle has been selected the system should capture and store the following rental information to a text file:

- License plate number
- Customer name
- Home address
- Phone number
- Date rented
- Expected return date
- Deposit paid

- **Search Cars**

The user should be able to search for a vehicle using the following attributes:

- License plate #
- Brand
- Model
- Year
- Interior

When the user selects the vehicle from the search results, the system should store the rental information to a text file. The search should be performed for **one attribute** at a time.

- **My Rental**

The My Rental screen displays all the vehicles you have rented. The user should not be able to rent more than 3 vehicles.

- **Return Vehicle**

The return vehicle screen should allow the user to specify which vehicle they are returning using the license plate number. The system should capture the following information when a vehicle is being returned:

- License plate #
- Actual return date
- Current mileage

A receipt should be displayed to the user, showing the cost breakdown. The deposit amount is calculated by multiplying the rate per day by the number of days the vehicle will be rented. If the actual return date exceeds the expected return date, the user should be charged an additional $2500 for each day over the limit. The formulas are shown below:

- Deposit = Number of rental days (Date Borrowed - Expected Return Date) * Rate per day

- Final Cost = Deposit + Late Fee (If any)

- Late Fee = IF Actual Return Date > Expected Return Date
  THEN (Expected Return Date - Actual Return Date) * 2500

After the vehicle has been returned, it should be removed from My Rentals, and its rental status and mileage changed/updated.

**Required:**

Perform an object-oriented analysis on the proposed system described above, and then design the system using the Unified Modelling Language (UML). Utilize composition and inheritance in your design to increase reusability and reduce system complexity.

**Grading Scheme (100 marks): General Mark Breakdown**

- ***Documentation (15 marks)***
  - Group Report (outlining contribution(s) of each member) **[2 marks]**
  - Object-Oriented Analysis and Design of system **[10 marks]**
  - User Manual **[3 marks]**
  - Declaration of Authorship for each group member - **CUMPULSORY**
- ***Source Code (35 marks)***
  - Comments **[3 marks]**
    - **Each file should have details for the student(s) who wrote the file**
    - Practice use of self-commenting files (i.e. proper variable and method naming)
    - Proper use of inline and method comments where necessary
  - Naming Convention **[2 marks]**
    - Pascal Case should be used for naming classes
    - Camel Case should be used for variable and method naming
    - Ensure class files are named appropriately
  - Object-Oriented Programming Techniques **[20 marks]**
    - Inheritance, Polymorphism, Composition
    - Method overriding and overloading
  - Use of Files **[10 marks]**
    - Proper implementation of appropriate file management
- ***Functionality (50 marks)***
  - Robustness **[10 marks]**
    - User Input validation
    - Error / Exception Handling
    - Program Navigation (i.e. Menu System)
  - User Interface
    - Ease of User Interaction **[6 marks]**
    - Appropriate Notifications (i.e. error and information messages) **[4 marks]**
  - System Functionality Implemented
    - View Vehicles **[5 marks]**
    - Search Vehicles **[10 marks]**
    - My Rentals **[5 marks]**
    - Return Vehicle **[10 marks]**

**Extra Marks (10 marks):**

Aproject that satisfies the program's functional requirements **can** gain additional 10 marks:

- + 10 marks – Awarded for use of colour and graphics to enhance the look and feel of the Program.

**Submission:**

Zip all your documents and upload to the link below on or before the due date of April 1, 2022.

Your zip file should contain all Documentation (User Manual, OOA & OOD, and Declaration of Authorship for each member), Source Files and Executable. The name of the zip file should be in this format:

**LASTNAME_OF_EACH_GROUP_MEMBER.zip e.g., BROWN_CLARKE_MILLS.zip**

Each group will have to do a 20 minutes interview with their tutor to receive a grade. Any group member who misses the grading interview will automatically attract a zero (0).

**Late Submission:**

Any project submitted after the due date will be late and will have a penalty applied for each day late. Additionally late projects will **not** be considered for extra marks. Please see table below for a breakdown of penalty for late submissions:

| **Time of Submission after Deadline** | **Penalty (%)** |
|---|---|
| One Day | 10 |
| Two Days | 20 |
| Three Days | 50 |
| Beyond Three Days | 100 |

**Group members:**

No group should exceed four members, any group exceeding the limit will receive a 20% deduction in their overall grade. Similarly, any group with less than three members will receive a 20% deduction in their overall grade (unless the group size was expressly approved by the lab tutor).