Faculty of Science, Engineering and Technology

Object Oriented Programming

Topic 5: UML Class and Sequence Diagrams

Resources

The following resources can help you with this topic:

- ***Concept map depicting the principles of OOP presented during this week's lecture***
- Wikipedia article on Concept Maps
- UML Class Diagram basics

Topic Tasks

Before starting to work on the tasks, first read through this entire document to get a sense of the direction in which you are heading. Complete the following tasks and submit your work through the Canvas before the deadline .

**Pass Task 13 - Conceptual Modelling and Implementation [Industry Case]

Credit Task 1 - System Modelling

Remember to submit your progress, even if you haven't finished everything.

After you have **discussed** your work with your tutor and **corrected any issues**, it will be signed off as complete.

Pass Task 13: Conceptual Modelling and Implementation

This pass task is based upon a real industry case. It is to be done individually. You are required to understand the requirement specifications given, design the conceptual model and implement the program to solve the business problem.

Introduction

Total Oil Malaysia Sdn Bhd was introduced in Malaysia in the year 1985 through the ELF brand. Its products are distributed throughout the country and include lubricants, glass coatings, industrial oil and specialty products. Total Oil Malaysia, result of the merger of Total Lubricant Malaysia and Elf Lubricant Malaysia, started its business in 2004. Total Oil Malaysia now reaches a market share of approximately 5% and its turnover exceeds MYR 80 million per year. Further information can be obtained at www.total.com.my

In Sarawak, the headquarters in Kuching is responsible for distributing oil to its customers in various cities; this requires the company to have several sales advisers in its several branches within the state, who visit customers and record their orders. After that, those orders are dealt with by the city branch if stock is available. Once products of any order are ready, a delivery team is responsible about delivering the products of specific order either in one trip or in many trips (depends on the availability and quantity of the ordered products. Customers are required to make payment for each order they placed either in full or in installments.

In order to stay competitive in today's marketplace, the headquarters would like to automate the process of order placement by the customers. Instead of customers wait for the sales advisors' visit, they could just login to the company's website, sign in and place their order and make the payment.

Functional Requirements

Table below shows the listing of requirements: -

Features	Requirements
Login Module	This module allows user to log into the system with correct username and password.
Main Menu	This module allows selection of different module with the correct credentials entered.
Users Management Module	This module able to perform the followings: • Able to create new user account • Able to edit user and reset password • Able to delete user
Stock Management Module	This module needs to cater the stock management for different type of vehicles such as cars, motorcycles, truck & bus, and able to perform the followings: • Able to add new stock

	• Able to edit stock details • Able to delete the stock(s)
Stock Availability Module	This module allows user to check the stock availability based on the stock name entered.
Order Placement Module	This module allows customer to place an order and check the order status.
Sales Management Module	This module able to perform the followings: • Able to view list of customer orders • Able to update the order status based on stock availability • Able to print the sales details based on the type of stock
Delivery Management Module	This module allows the delivery man to update the order delivery status.

** Please have a detailed read on the different type of products available on the company's web site.

Task 1 – System Modelling [5 marks]

Develop a UML Class Diagram based on the project functional requirements given for Total Oil Malaysia Sdn Bhd. You are expected to capture and model all relevant information provided into the class diagram. You are required to use any Computer-Aided Software Engineering (CASE) tool i.e. MS Visio, StarUML or draw.io to produce the UML Class Diagram.

The UML Class Diagram must fulfil the following criteria:

- All requirements stated in the industry case and relevant information are precisely captured.
- Appropriately demonstrate the use of OOP principles.
- Appropriately demonstrate the use of UML Class Diagram notations.

Task 2 – Class Implementation [5 marks]

Implement the Class Diagram you have developed in Task 1 in C#. You are expected to implement the following:

- Code the classes and methods identified into a C# program. (Note: Your class file must reflect the attributes and methods given in the class diagram)
- Create at least 3 unit tests for any two classes.
- The code must follow a self-consistent coding convention and be well-documented.

Task 3 – Create a Console Application [3 marks]

Create a console application to demonstrate the functional requirements using the classes defined in Task 2:

The console application should cover the functional requirements as a single complete application with main menu and proper designed program flow.

Task 4 – Presentation [2 marks]

Applicable only to student who has completed Task 1 to Task 3 with a total weightage of more than 10 marks for the three tasks. Presentation date and venue are to be confirmed in Week 10.

Pass Task 13 - Assessment Criteria

Make sure that your task has the following in your submission:

- Task 1 to Task 3 are implemented.
- Complete UML Class Diagram (Softcopy, not hand-drawn).
- The code must compile, show the tests passing and main program running.
- Classes must have XML documentation.
- Code must follow the C# coding convention used in the unit.

Credit Task 1: System Modelling

In this task, you are required to develop a UML Class Diagram based on the case study given below. You are expected to capture and model relevant information provided into the class diagram. You may use any Computer-aided software engineering (CASE) tool i.e. MS. Visio, StarUML or draw.io to produce the UML Class Diagram.

Case Study: Console-based Multiple Choice Questions (MCQ) Generator

> This application allows users to create, manage and attempt multiple choice questions. The user can be either teacher or student. A teacher can create a pool of questions and develop question sets by adding questions from the pool. The student can then attempt question set and the outcome i.e. answer as well as result will be captured for review later.

Complete ALL the tasks below:

Task 1: Create class diagram to represent the relationships (e.g. association, aggregation or generalization) among objects in the Console-based MCQ Generator. If the relationship is association or aggregation, specify the cardinality (multiplicity).

Task 2: For each class in Task 1, insert attributes, which specify its visibility, name, type and initial value (if any).

Task 3: For each class in Task 1, insert operations, which specify its visibility, name, arguments (if any), and return type.

The UML Class Diagram must fulfill the following criteria:

1. All requirements stated in the case study and relevant information are precisely captured.
2. Appropriately demonstrate the use of 4 OOP principles (Abstraction, Encapsulation, Inheritance & Polymorphism).
3. Appropriately demonstrate the use of UML Class Diagram notations.

This task aims to help you think through the various relationships between the object oriented programming concepts and the associated programming artefacts. You may make any assumptions relevant to the case study given and please state down your assumptions clearly in your submission document.

Credit Task 1 - Assessment Criteria

Make sure that your task has the following in your submission:

- Complete UML Class Diagram (Softcopy, not hand-drawn).
- The work is at its best quality as the number of attempt/submission allowed are limited.