# Luminus Technical University College - Assignment Brief (RQF)

## Higher National Diploma in computing

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Student Name** | | **Rashed Hassan.** | | | **Language of assessment** | | | **AR** | **EN** |
| **College ID:** | | |  | |
| **Pearson ID:** | | |  | |
| **Unit Number and Title** | |  | **20 Advanced Programming** | | | | | | |
| **Academic Year** | | **2022/2023** | | | | | | | |
| **Unit Tutor** | |  | | | | | | | |
| **Internal Verifier Name and Approval (Signature)** | |  | | | | **Approval Date:** | | | |
|  | | | |  | | | |
| **Assignment number and Title** | | **1** | **OOP Development** | | | | | | |
| **Issue Date (1St Submission)** | | **20/11/2022** | | **Submission Date (1st Submission)** | | | **10/12/2022** | | |
| **Issue Date (2nd Submission)** | | **20/12/2022** | | **Completion Date (2nd Submission)** | | | **23/12/2022** | | |
| **Submission Format** | | | | | | | | | |
| **The submission form is an individual written report, should be written in a concise formal business style using single spacing and font size 12. You are required to make use of headings paragraphs, and subsections as appropriate, your work must be referenced using Harvard or APA reference style.** | | | | | | | | | |
| **Unit Learning Outcomes** | | | | | | | | | |
| **LO1** | **Examine the key components related to the object-orientated programming paradigm, analysing design pattern types** | | | | | | | | |
| **LO2** | **Design a series of UML class diagrams** | | | | | | | | |
| **LO3** | **Implement code applying design patterns** | | | | | | | | |
| **LO4** | **Investigate scenarios with respect to design patterns** | | | | | | | | |
| Transferable skills and competencies developed | | | | | | | | | |
| Understanding how to translate a scenario into UML class diagrams and implementing it using OOP concepts | | | | | | | | | |
| **Vocational scenario:** | | | | | | | | | |
| You are a software developer at Space Dimensions Company. You have been assigned to work on a project to handle a factory system.  The factory has a name, address, invoice value and products. Each product consists of id, name and price.  The factory has several types of employees in the factory in which they all share a name, id, email, salary and phone. The manager has department in which he/she works in, office number and project name in which he/she is responsible for. The workers have a position, shift and a flag that defines if the worker is a team leader or not. The accountant has a level and a list of responsibilities.  A client has id, name, address and purchase order. The purchase order has client linked to it, id, invoice value and products. The purchase order must have an option to add products to the purchase which will increase the invoice value based on the products’ price.  Both the factory and client should have a pay option that will pay the invoice value. All the employees should have an option to get their salaries based on their profession as follow:   * Manager salary = base salary + 25% of the base salary * Worker salary = base salary + 5% of the base salary * Accountant salary = base salary + 10% of the base salary   **Note: Employee is too general.**  It has been decided to use an object oriented approach to design and implement the system. | | | | | | | | | |
| Assignment activity and guidance | | | | | | | | | |
| **Task 1**  Briefly describe the following Object oriented concepts. Provide diagrams and code snippets (**you must write code**) to supplement your explanations.   1. Static variables and methods 2. Default constructor and parametrise constructors 3. Access modifiers and encapsulation 4. Abstraction   **Task 2**   1. Create a detailed UML class diagram for the **factory system** using a suitable UML tool. The classes should include attributes and methods. 2. Explain one **inheritance** and one **composition** relationships existed in your UML.   **Task 3**   1. **Implement** the system using a suitable object oriented language. Provide **screen shots** (in addition to source code) as evidence of program execution.   **Task 4**   1. The factory system has a HR system and they want to integrate it with your new developed system. The main problem is that they do not have any documentation. You need to extract the UML class diagram from the already existed code to able to do the task. Investigate how we can create a tool to extract the UML automatically from the given code in details.   **Task 5**   1. There are three categories of design patterns which are creational, behavioural and structural. Given the list below that has design patterns examples. Sort them into the three categories and explain your choices.    1. Singleton Pattern    2. Facade Pattern    3. Observer Pattern    4. Decorator Pattern    5. Factory Pattern    6. Strategy Pattern | | | | | | | | | |
| **Recommended Resources**  **Please note that the resources listed are examples for you to use as a starting point in your research – the list is not definitive.**  **Textbooks** | | | | | | | | | |

**Learning Outcomes and Assessment Criteria**

|  |  |  |  |
| --- | --- | --- | --- |
| Pass | Merit | | Distinction |
| **LO1** Examine the key components related to the object orientated programming paradigm, analysing design  pattern types | | |  |
| **P1** Examine the  characteristics of the  object-orientated  paradigm as well as the  various class  relationships**.** | **M1** Determine a design  pattern from each of the  creational, structural and behavioural pattern types. | | **D1** Analyse the relationship between the object-orientated paradigm and design patterns.  **D2** Analyse how class  diagrams can be derived from a given code scenario using a  UML tool. |
| **LO2** Design a series of UML class diagrams | | |
| **P2** Design and build class diagrams using a UML tool. | **M2** Define class diagrams for specific design patterns using a UML tool. | |
| **LO3** Implement code applying design patterns | | |  |
| **P3** Build an application  derived from UML class  diagrams. | | **M3** Develop code that  implements a design  pattern for a given purpose. | **D3** Evaluate the use of design patterns for the given purpose specified in M3. |
|  | | |
|  | |  |