1st SIT COURSEWORK QUESTION PAPER	Year Long 2022/2023

Module Code:	**CS6004NI**
Module Title:	**Application Development**
Module Leader:	**Mr. Samyush Maharjan** (Islington College)

Coursework Type:	**Individual**
Coursework Weight:	This coursework accounts for **30%** of your total module grades.
Submission Date:	**Week 12 (5th January 2022)**
When Coursework is given out:	**Week 8 (8th December 2022)**
Submission Instructions:	Submit the following to the Islington College RTE department before the due date: ● The software application to be developed in C# ● The documentation in MS Word compatible or PDF format
Warning:	London Metropolitan University and Islington College take Plagiarism seriously. Offenders will be dealt with sternly.

Plagiarism Notice

You are reminded that there exist regulations concerning plagiarism.

Extracts from University Regulations on Cheating, Plagiarism and Collusion

Section 2.3: “The following broad types of offence can be identified and are provided as indicative examples …..

(i) Cheating: including copying coursework.
(ii) Falsifying data in experimental results.
(iii) Personation, where a substitute takes an examination or test on behalf of the candidate. Both candidate and substitute may be guilty of an offence under these Regulations.
(iv) Bribery or attempted bribery of a person is thought to have some influence on the candidate’s assessment.
(v) Collusion to present joint work as the work solely of one individual.
(vi) Plagiarism, where the work or ideas of another are presented as the candidate’s own.
(vii) Other conduct calculated to secure an advantage on assessment.
(viii) Assisting in any of the above.

Some notes on what this means for students:

(i) Copying another student's work is an offence, whether from a copy on paper or a computer file and in whatever form the intellectual property being copied takes, including text, mathematical notation and computer programs.

(ii) Taking extracts from published sources without attribution is an offence. To quote ideas, sometimes using extracts, is generally to be encouraged. Quoting ideas is achieved by stating an author's argument and attributing it, perhaps by quoting, immediately in the text, his or her name and year of publication, e.g. " e = mc2 (Einstein 1905)". A reference section at the end of your work should then list all such references in alphabetical order of authors' surnames. (There are variations on this referencing system that your tutors may prefer you to use.) If you wish to quote a paragraph or so from published work then indent the quotation on both left and right margins, using an italic font where practicable, and introduce the quotation with attribution.

Further information concerning the existing London Metropolitan University regulations concerning plagiarism can be obtained from http://www.londonmet.ac.uk/academic-regulations

Introduction

This individual coursework requires developing and documenting an application in .NET using C# programming language using an object-oriented approach. Your software artefact must be submitted as a Visual Studio project.

The coursework carries 30% of the module mark.

Submission Deadlines: Week 12 (5th January 2022)

Coursework Submission in-class Demo:

This individual coursework has 2 parts, both of which are to be submitted to RTE.

(1) The software application to be developed in C# .NET using C# programming language

(2) The documentation in MS Word compatible or PDF format

NB– Anyone not meeting the deadline must submit the work to the Undergraduate Registry with a completed *mitigating circumstances form*. It will only be marked if the mitigating circumstances are accepted. All parts of the late submission must be handed into the Undergraduate Registry with the form. You must ensure that you have a receipt from the Registry for your work.

Please note the rules on plagiarism

The application should be implemented individually. This is not a group/team effort. Any material which is a direct copy from someone else (student or another source) or a close paraphrase/code must be indicated where it is quoted i.e. it must be made clear what material is a quotation or close paraphrase e.g. by showing the text in italics or quotation marks. It is not sufficient to show the source in a list of references or bibliography. If you are unclear, please discuss your examples with your seminar tutor or the module leader. Plagiarism is a serious offence and conviction for plagiarism may lead to suspension from the University, even for a first offence (please see the section on Academic Misconduct in the Student Handbook).

Software Development Task

Design and implement a C# desktop application (not a web-based or database application) that helps a bike service centre to manage their stock/ inventory records. The bike service centre allows their staff to get or take items out from inventory only between 9 am to 4 pm, 5 days a week (from Monday to Friday). Staff are allowed to only get tools or other items out of inventory only after being approved by an admin. However, the system should allow you to add items to inventory at any time.

Currently the service centre is using a ***paper-based manual system*** to keep records of stocks.

The application is to keep track of each stock item detail with the staff name details of "approved by and taken by". The stock/ inventory should be only accessible to the admin who has the "ONLY" power to approve the items to be added or deducted from inventory.

The application should implement the following functions:

1. To create a Login System for the admin of the system and to store the details of the admin login in a json/csv file and system should not have more than 2 admin. To create a system for the admin to approve the removal of items from the inventory. (This can only be done between 9 am to 4 pm, 5 days a week (from Monday to Friday).)

2. To create a system for the admin to add items to the inventory. (This does not have a time or date restriction)

3. Save and retrieve the addition and removal details of each item and also the inventory of the service centre in a csv or a json file.

4. Create a table for the records of the items that have been taken from the inventory for every month. The table should include the columns for the Item, Quantity taken out, Approved by (this should be done by the admin), Taken by (this can be done by anyone) and the date it was taken out on.

Items taken from inventory Data Grid View

Inventory Log for the month of January

Items	**Quantity**	**Approved By**	**Taken By**	**Date Taken Out**
Wheel	5	Ram	Shyam	01/01/2022
Arrow Screw	40	Ram	Hari	05/01/2022
Brake Pads	…..	……	……	
Engine oil	……	……	……	
Item X	……	…….	…..	
Item Y	……	……	……	

Note*: You may add extra features - both data and functionality to the application if you wish.*

5. Create a Data Grid View for the display of all the items in the inventory (in stock currently). The table should include records for all the items currently available in the inventory and should include columns for the Name of the item, Quantity in the inventory and the date it was last taken on.

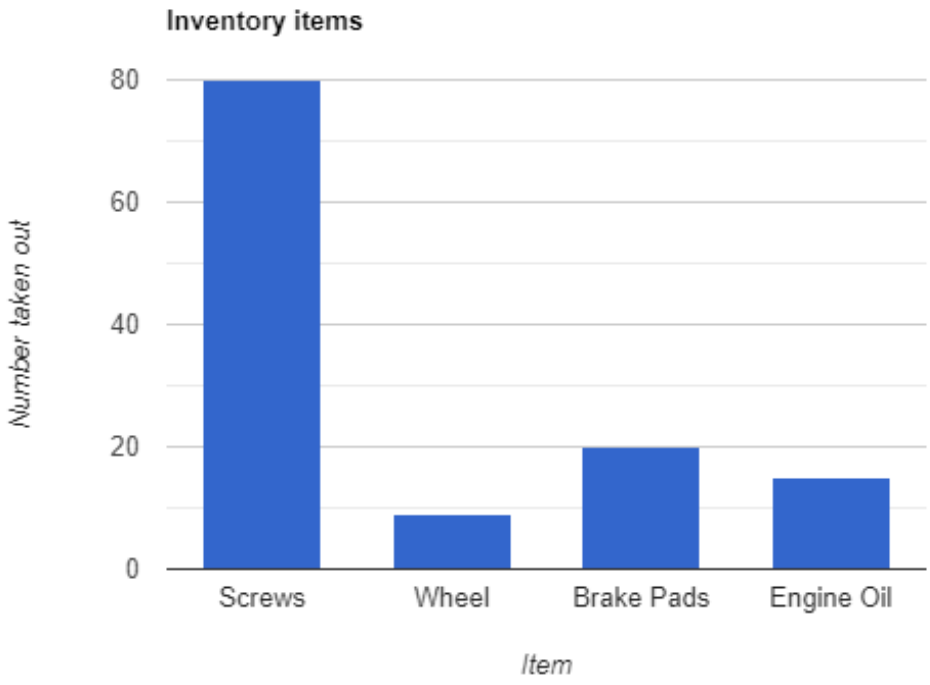
Inventory detail table

Item	**Quantity in inventory**	**Last taken out on**
Wheel	80	01/05/2022
Screws	5000	05/01/2022
Brake pads	…..	……

Engine oil	……	……
Group of 15	……	…….
Group of ……	……	……

6. To create a Bar Graph to display the items based on the number of that item that have been taken out. (The chart should clearly show the relation of the Item and the number that has been taken out.)

Inventory Items take out chart



Your software implementation should demonstrate the following features

1. Use of appropriate data types (built-in and programmer-defined) to handle the application data
2. Use of appropriate data structures e.g. arrays/ list/ dictionary for the required programming scenario
3. Use suitable algorithms e.g. sorting
4. Define and use your class or classes
5. Provide a desktop application build as MAUI (Multi-Platform App Ui with Blazor)
6. Save and retrieve the object's state using serialisation.

Deliverables

Your submission should include the software project and a reflective essay as described below.

1. Your software artefact is in the form of a Visual Studio 2022 project, which should include the program's source code, compiled classes, the executable file and data file (if any).

2. A reflective essay (1000 words), which concisely documents:

    a. detailed instructions to run the program

    b. the architecture of your software in terms of software classes, clearly indicating which classes to be of your own work and which classes from other sources (e.g. from textbooks, online sources such as MSDN etc.).

    c. a detailed description of the classes' properties and methods

    d. description of your algorithm to build the user record/ revenue collection chart in the form of a flowchart and/or decision table.

    e. which data structures and which algorithms you have used, in which part of your program, and why.

    f. your reflection of your own experience of using C# and Visual Studio for the development task, which features you like and why, what issues you experienced and your solution to overcome it.

Section	Topic	Full Marks
A	Implementation of Application	**60**
1.	Creating the Login System for the admin and to store the details of the admin login in a json file .	10
2.	Creating a system for the admin to approve the removal of items from the inventory.	5
3.	To create a system for the admin to add items to the inventory.	5
4.	Save and retrieve the addition and removal details of each item and also the inventory of the service centre in a csv or a json file.	10
5.	Creating a table for the records of the items that have been taken from the inventory for every month.	10
6.	Creating a table to display all the items in the inventory (in stock currently).	10

7.	To create a Bar Graph to display the items based on the number of that item that have been taken out.	10
B	**Documentation**	**10**
1.	Detailed instruction to run the program	2
2.	Concise description of your logical solution to each of the implemented functions of the application.	2
3.	The software architecture, i.e., class and their purpose	2
4.	Detailed description of the classes' properties and method	2
5.	Individual member's own reflection of own experience	2
C	**Programming Style**	**30**
1.	Clarity of code, Proper Naming conventions & comments	5
2.	Sensible naming of programmer-defined variables, classes, properties, and methods	3
3.	Useful comments in code	2
4.	Data validation and exception handling	10

5.	Interface design and usability of the system	10
	Total Marks	**100**