

Module: COMP1551 Application Development	Coursework
Contribution: 100% of the grade	Coursework submission: a single PDF document submitted on Moodle
Module leader: Konstantin Kapinchev	Due date: 19th August 2024
Approximate time to complete the coursework: 50 hours	
Learning outcomes: 1 Use UML to design object-oriented, interactive, data-driven, applications 2 Use a modern IDE with an object-oriented programming language to create interactive, data-driven applications 3 Demonstrate theoretical and practical skill in the design, implementation and testing of applications making use of object-oriented approaches such as classes, message passing, overloading, data connectivity, inheritance, threads and patterns	

Plagiarism is presenting somebody else's work as one's own. It includes copying information directly from online resources or books without referencing the material, submitting joint coursework as an individual effort; copying another student's coursework, purchasing coursework from someone else and submitting it as own work. Suspected plagiarism will be investigated and if found to have occurred will be dealt with according to the procedures set by the University. All material copied or amended from any source must be referenced correctly according to specific reference style. Courseworks will be submitted for electronic plagiarism checks.

Coursework Submission Requirements:

- An electronic copy of the coursework should be uploaded by midnight on the deadline date
- The last uploaded version will be the one that is marked
- The format of the coursework is a single PDF document containing the solutions and answers to all tasks

In order to be marked, the source code in the document needs to meet the following requirements:

1. To be selectable as text
 2. To be error and warning free
- The limit of the file size is 100 MB
 - The PDF document should be virus-free, not protected by a password or corrupted, otherwise it will be treated as not submitted
 - Feedback on the coursework will be available on Moodle
 - The grade will be made available in the portal
 - A paper copy cannot be submitted for this coursework

The University website has details of the current Coursework Regulations, including details of penalties for late submission, procedures for Extenuating Circumstances, and penalties for Assessment Offences. See <https://www.gre.ac.uk/student-services/regulations-and-policies> for details.

Coursework Specification:

An education centre is moving from paper-based book-keeping to a *desktop information system*. The *desktop information system* is expected to process information about the following groups of users:

1. Teaching Staff
2. Administration
3. Students

The system is expected to process the following **data** about each user from all groups:

1. Name
2. Telephone
4. Email
5. Role (teacher/admin/student)

The system is expected to process the following group specific **data**:

Teaching Staff:

1. Salary
2. Names of 2 subjects

Administration:

1. Salary
2. Full-time/part-time
3. Working hours

Students:

1. Names of 2 current subjects
2. Names of 2 previously studied subjects

The system is expected to provide the following **functionality**:

1. Add new data
2. View all existing data
3. View existing data by user group
4. Edit existing data
5. Delete existing data

Task 1

Based on the provided outline, provide a description of the **Desktop Information System**, which the education centre is planning to use. This description expresses the user's point of view about the functionality of the system, the data it is expected to process and the required platforms, including operating system (Microsoft Windows) and hardware. Maximum word count for the system description is 300 words.

Based on the provided description, write a **software requirements specification** of maximum 500 words for the **Desktop Information System**. The **software requirements specification** is expected to have the following content:

1. Introduction
 - Purpose
 - Project scope
2. Overall Description
 - Product
 - Users
 - Operational Environment
3. System Features
 - Description
 - Functional Requirements
4. User Interface Requirements
5. Platform Requirements
6. Quality Attributes
 - Performance
 - Security
 - Safety

Maximum word count for the **software requirements specification** is **400 words**.

*For more information about writing software requirements specification, refer to:
Wiegiers, Beatty, "Software Requirements", Third Edition, Microsoft:
Chapter 10 Documenting the Requirements, page 181
Chapter 11 Writing Excellent Requirements, page 203*

Task 2

Provide the following UML diagrams, which represent the design of the **Desktop Information System**:

- Class Diagram
- Use Case Diagram

For more information about UML, refer to Patrick Grassle "UML 2.0 in Action".

Task 3

Develop the *Desktop Information System*, as a C# Console Application with **data** stored in **appropriate data structures**.

Implement a base class named “*Person*”. By utilising the principles of encapsulation, inheritance and polymorphism, implement the classes “*Teacher*”, “*Admin*” and “*Student*” as derived classes from the base class “*Person*”. The derived classes are expected to handle the **data** and the **functionality** for the respective groups of users.

Store the objects into appropriate data structures. The data structures are expected to be able to store unknown number of objects.

The *Desktop Information System* is expected to provide a text-based menu, which provides access to each of the aforementioned functionalities of adding, viewing, viewing by role, editing and deleting records.

Add comments describing the functionality of the project. The comments are expected to be approximately **30%** of the source code.

Add the entire source code of the C# program into the coursework document as a **selectable text**.

Grading criteria

70-100% All requirements completed to an excellent standard.

60-69% All requirements completed. However, there are a number of minor deficiencies in significant areas.

50-59% All requirements completed. However, significant improvements could be made in many areas.

40-49% All requirements completed. However, significant improvements could be made in all areas.

30-39% All requirements attempted but the overall level of understanding and performance is poor.

0-29% There are requirements missing or completed to a very inadequate standard, which indicates a very poor or non-existent level of understanding.

Grading Components

Task	Achieved well	Partly achieved	Poorly achieved or not achieved
Task 1 (20%)			
Description of the Desktop Information System is provided	5-4	3-2	1-0
Content of Software Requirements Specification is provided as described	5-4	3-2	1-0
Every item in the Software Requirements Specification is successfully addressed	5-4	3-2	1-0
Accurate academic language is utilised	5-4	3-2	1-0
Task 2 (10%)			
UML Class Diagram describing the structure of the classes of the program is included	5-4	3-2	1-0
UML Use Case Diagram describing the expected behaviour of the program is included	5-4	3-2	1-0
Task 3 (70%)			
Object-Oriented Features			
Classes and objects are accurately utilised	10-8	7-3	2-0
Encapsulation, inheritance and polymorphism are accurately implemented	10-8	7-3	2-0
An appropriate data structure containing all objects is successfully implemented	10-8	7-3	2-0
Functionality of the Application			
Main Menu	5-4	3-2	1-0
Adding new record	5-4	3-2	1-0
Listing all records	5-4	3-2	1-0
Listing all records by role	5-4	3-2	1-0
Editing existing records	10-8	7-3	2-0
Deleting existing records	5-4	3-2	1-0
Comments accurately describing the functionality of the program	5-4	3-2	1-0