CS6004 –Application Development

Individual Coursework – 2020-21

Introduction

This individual coursework requires developing and documenting a Windows form .NET framework desktop application in C# using an object-oriented approach and Visual Studio. Your software artefact must be submitted as a complete Visual Studio project. It will be marked using Visual Studio 2019 and any features not working in the standard installation of Visual Studio 2019 will not be assessed.

The coursework carries 30% of the module mark.

Set: Friday 13/11/2020

Submission Deadlines: 3pm, Monday 11/1/2021

Mandatory Software Demo: TBA

This individual coursework has 2 parts, both of which are to be submitted via WebLearn before 3pm, Monday 11/1/2021.

1. **A complete Visual Studio project of the software application in C#**
2. **A reflective essay in PDF or MS Word compatible format**

NB– Anyone not meeting the deadline must submit the work to Weblearn with a completed mitigating circumstances form. It will only be marked if the mitigating circumstances are accepted. You must ensure that you have receipt from the Registry if your work is submitted to them.

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 other 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 in 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 client-server or database application) that assigns seats on an airplane. Assume the airplane has 20 seats in first class (5 rows of 4 seats each, separated by an aisle), and 180 seats in economy class (30 rows of 6 seats each, separated by an aisle).

The application should provide the following functionality:

1. Assign seat(s) to passenger(s). When assigning seats, ask for the class (first or economy), and the number of passengers travelling together (1 or 2 in first class, 1 to 3 in economy). Then try to find match and assign the seats. If no match exists, display a message.
2. Display seating status (assigned & available) on the computer screen, allowing the user to select from at least two different sorting orders, e.g. by seat numbers or by passenger names.
3. Save and retrieve the airplane's seating state between program runs.

4. Quit the application.

Passenger's details are not required to be maintained by the system; however, you may add extra features - both data and functionality to the application, if you wish.

Your software implementation should demonstrate/provide 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 for the required programming scenario
3. Use suitable algorithms e.g. sorting
4. Define and use your own class or classes
5. Provide either console-based or window-based user interface for your application
6. Save and retrieve the objects state using serialization.

Deliverables

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

1. Your software artefact in the form of a Visual Studio 2019 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, e.g. MSDN etc.).
    c. detailed description of the classes' properties and methods
    d. a description of your algorithm to assign the airplane seats 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. A reflection of your experience of using C# and Visual Studio for the development task, which feature you like and why, what issues you experienced and your solution to overcome it.

Marking Scheme for the CS6004 individual coursework

This individual coursework counts for 30% of the module mark. Please see the table below for the marking criteria and its weighting.

	Item	Weighting %
SOFTWARE DEMO [Design and Implementation]		**60**
1	The application user interface	7.5
2	Assign seats	15
3	Display seating status	15
4	Save and retrieve the airplane's seating state between program runs.	10
5	Use of data types	2.5
6	Use of algorithm	5
7	Class design & implementation	5
REFLECTIVE ESSAY		**27.5**
a	Detailed instructions to run the program	2.5
b	The software architecture (e.g. using a class diagram)	5
c	Description of your algorithm to assign seats	5
d	Detailed description of your software classes' properties and methods	5
e	Data structures and standard sorting/searching algorithms used in the project	5
f	A reflection of your own experience doing the project	5
Programming Quality and Style		**12.5**
1	Clarity of code which shows the underlying algorithm	2.5
2	Sensible naming of programmer-defined variables, classes, properties, and methods	2.5
3	Useful comments in code	2.5
4	Data validation and exception handling	2.5
5	Interface design and usability of the system	2.5
	Total mark	**100**