|  |  |  |
| --- | --- | --- |
| **ASSESSMENT BRIEF** | | |
| **Class Group:** | | 062CS |
| **Assessor:** | | Bernie Farrell |
| **Component Title and Code:** | | 6N0736 - Graphical User Interface Programming |
| **Assessment Technique:** | | Skills Demonstration |
| **Weighting:** | | 30% |
| **Title:** | | Assessment 2 |
| **Issue Date:** | | 13th March 2018 |
| **Submission Date:** | | 12th April 2018 |
| **Learning Outcomes Assessed:** | | 1, 2, 4, 5, 6, 7, 9 and 10 |
| **Guidelines:**  Carry out the skills demonstration in accordance with the guidance provided in this document. | | |
| **Assessment Criteria:**  Program documentation to include algorithms, prototyping and rapid application development.  (6 marks)  Application interface to include at least its look, ease of navigation, ease of use, simplicity, consistency, feedback, and a help subsystem. (8 marks)  Quality of application to include at least adherence to user specifications, correctness, user error trapping and feedback, robustness, and adherence to best practice. (12 marks)  Testing of application to include at least suitable test data, expected results and actual results.  (4 marks) | | |
| **Learner Name:** |  | |
| I confirm that:   1. I have been provided with information about Cork ETB’s assessment and appeals procedures and my responsibilities with regard to assessment 2. The assessment work produced by me is all my own original work | | |
| Learner Signature: | | |
| Date: | | |

|  |
| --- |
| **Note to Learners:**   * Assessments will not be accepted without this coversheet. * Plagiarism is the presentation of someone else’s ideas, arguments, concepts or work as your own by failing to reference or acknowledge it properly. All such work must be acknowledged. Any learner, who presents another’s work as their own, will be investigated in line with Cork ETB Assessment Malpractice procedures and may be awarded a zero grade. * Learners should keep copies of all assessment submitted, where applicable. |

## introduction

The Cork College of Commerce has many computer rooms. Each computer room has several computers. From time to time computers are unusable because a component fails, software needs to be installed or reinstalled etc. The failure is logged, repaired and marked as *resolved*.

As a starting point, you have been given the details held by the technician on the computer rooms. This includes:

* The computer room names and the number of computers in each room.
* The serial number, name (room number and computer number), model and manufacturer of each computer.

You are required to design, code and test a VB.NET application that will act as a digital fault logbook, subject to the following considerations.

## THE TEACHER

Teachers are tasked with filling in the fault book when they come across a fault. They are to be able to login to the system using their user name and password. The user names and passwords are held in an underlying database table.

The user can change their password on the system. They must first provide their old password, enter their new password and confirm their new password. If the old password is correct and the new password matches the confirmation password, the user's password is changed.

## THE logbook

Any faults found are to be logged with the following detail: The date the fault was logged, the room number, and the number of the computer, the fault with the computer and the name of the user who logged the fault.

* All fields are mandatory.
* The date should be taken from the system clock.
* The room number entered must be a valid computer room number.
* The computer number must be a valid computer number.
* The fault is a brief description of the fault with the computer.
* The name of the person logging the fault is the user who logged in to log the fault.

All users must be able to:

* View all entries. If a fault has been logged already, it should not be logged again.
* View all entries that they have logged and that are not resolved.
* View all entries that they have logged and that are resolved (see next section).

## THE TECHNICIAN

Repairs are carried out by the technician. He/She must be able to:

* View all faults logged.
* View faults by room.
* View individual faults (e.g. Computer 7 in room 116). In this case, the serial number, manufacturer and model are displayed.
* Indicate the status of the repair if the computer is not repaired e.g. “RAM ordered.”
* Indicate that a repair has been made. In this case, the fault details are transferred from the *logbook* table to the *resolved* table.
* View the number of faults and the detail on all faults that have been inactive for five days or more.
* At the end of every academic year, the principal considers replacing a computer room for next year's students and wants be told which room has had the most faults per computer. The technician makes this available to management.

## The Administrator

The administrator must be able to:

* Modify any details in the *Computer Room* and *Computer* tables. They should be able to remove a room, add a room, modify room details, remove a computer, add a computer and modify computer details.
* Add a user or remove a user.

## Brief

You are required to design, implement (using VB.NET) and test a program that will interface with a database to fulfill the requirements detailed above. The application is to:

* Be developed using prototyping and rapid application development, with each part of your development documented.
* Be error free, robust and reliable.
* Adhere to best practice in terms of the user interface and underlying coding.
* Be prepared for deployment.

## Submit

**1. The design of your project.**

This should include all the preliminary work for your project i.e. all the work that you prepared using algorithms, prototyping and rapid application development, to produce the final product.

**2. The screen layouts printed.**

**3. The source code printed.**

This should include the code for all the forms that you have used in your project and any

additional code.

**4. How you tested your program.**

The testing that you carried out on each implemented prototype should be shown. This should include testing of the functionality of the program and of any other tests e.g. data validation, key checking etc. that you carried out to ensure that your program is robust.