**ASSESSMENT BRIEF**



|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Title and Code of Minor Award: Graphical User Interface Programming 6N0736** | | |  | | | |
| **Class: ACS61 & ACS62** | | |  | | | |
| **Title: Skills Demo 2** | | |  | | | |
| **Assessment No2. and Technique: Assignment** | | |  | **Weighting (30%):** | | | |  | |
| **Issue Date: 26/02/2018** | | |  | **Submission Date: 23/03/2018** | | | |  | |
| **Feedback Date: 22/04/2018** | **Teachers** | Sonya Fay & Sean Ridge: | | |  | | | |
| **Learning Outcomes: 8,11,12** | | | | | |

|  |  |
| --- | --- |
| **Learner Name (Print):** |  |
| **I confirm that I have kept a copy of my work and that this is my own original work.** | |
| **Signature:** |  |

|  |  |  |
| --- | --- | --- |
| **Assessment Criteria** | **Max. Mark** | **Learner Mark** |
| Program documentation to include evidence of prototyping and rapid application development and a deployment plan. | 6 |  |
| Application interface to include at least its look, ease of navigation, ease of use, simplicity, consistency, feedback and a help subsystem. | 8 |  |
| Quality of application to include at least adherence to user specifications, correctness, user error trapping and feedback, robustness, adherence to best practice and development of a runtime installation package. | 12 |  |
| Testing of application to include at least suitable test data, expected results and actual results and result of deployment of the application | 4 |  |
| **Subtotal** | **30** |  |
| ***Marks deducted for Late Submission:*** | |  |
| **Total Mark:** Mark is **provisional** and subject to change by the external authenticator. | |  |

|  |
| --- |
| **Teacher Feedback: It is the learner’s responsibility to be in class when feedback is being given.** |
| **Strengths:** |
| **Areas for improvement:** |

Rathmines College, College of Further Education

**GUI Programming Assignment 2 (30%)**

You are required to develop a quiz and basic calculator application.

Design, implement and test a solution to that problem using prototyping and rapid applications development. Draw up a documented deployment plan for your application. (LO 11)

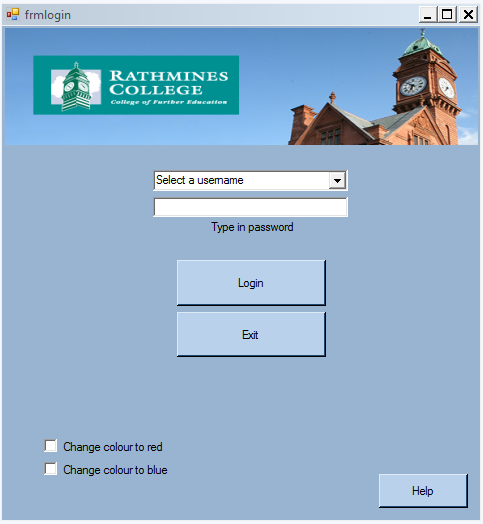
This VB 2008 Express application will have a number of **FORMS** as follows; a login screen, a menu options screen, multiple choice questionnaire (MCQ) forms with an associated results form, and a realistic calculator.

1. The **login screen** must provide password access to two users, a **tester** with administrative rights and a **student** user account.

The user names **tester** and **student** are to be selected from a **drop down combobox**. The **password** for **tester** and **student** users is **password** and must be entered as a fixed text property in the **password** text box and must be compared to the valid passwords for both users that you have saved in a text file named **validpasswords.dat**. The user should be able to proceed to the options screen.by **clicking on a “Login on” command control.**

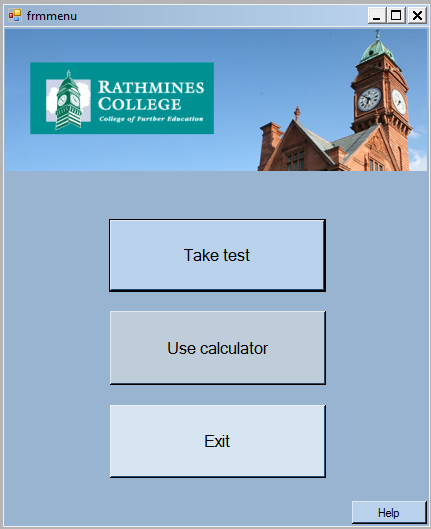
Demonstrate the use of the following controls on the login screen.

A **check box** to change the form colour to **red** and **option button** to change the form colour to **blue**.



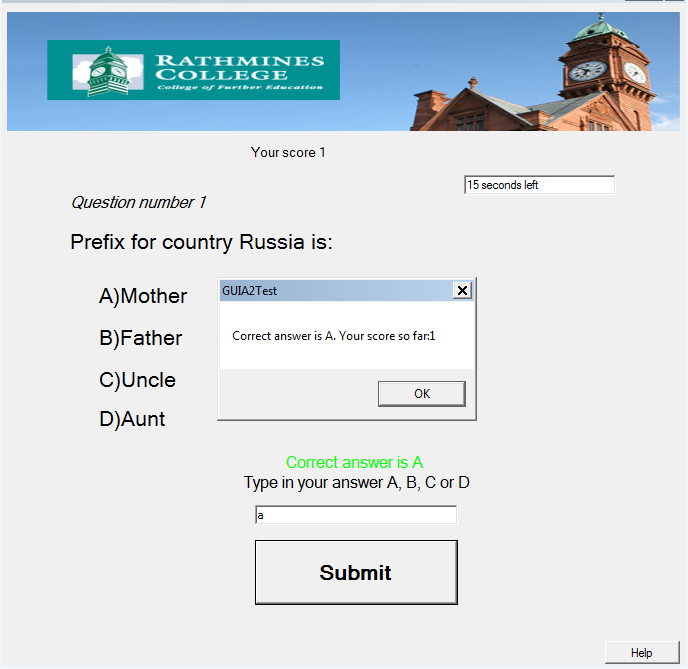
**FORM 1**

2. The valid user will be presented with the following menu options.

****

**FORM 2**

**3. The following is a test screen for question 1.**

******

Only visible for the tester

Label for time elapsed feedback

**FORM 3**

The MCQ option consists of an educational quiz, with **ten** multiple-choice questions each with four possible answers. The questions can be drawn from the sample questions in Appendix A or you can make up your own questions. The **Tester** user only is presented with correct answer field at runtime. The tester and student users will proceed to next question by clicking on the next button**.** Users have **20 seconds to answer** each question.Program control will pass automatically to the next question or results form when 25 seconds elapse. **Remind users to hurry up when there are six seconds left.** The questions, possible answers, correct answer will be stored in one dimensional arrays in a Module.

Initialize textboxes and labels to blank as each question is presented.

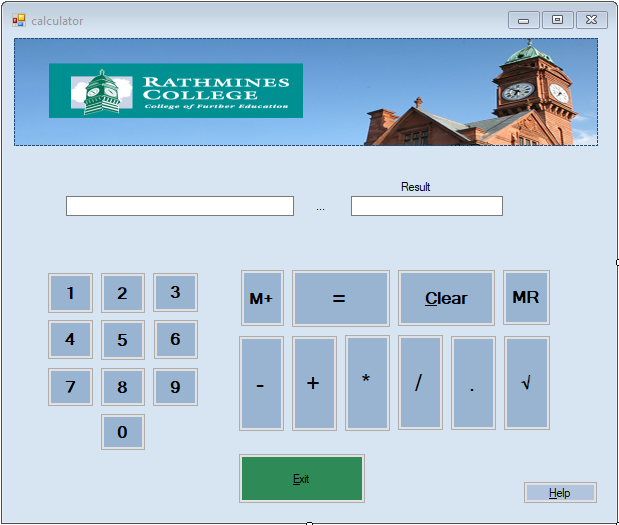
**Include two instructions for users on all MCQ forms.** Use “On error resume next” **on all command buttons** to ensure continuity at runtime. The program will count the number of correct answers and communicate progress with the user after each answer.

Display the score out of 10 and an appropriate image on a results form **(see example below).** If the user score is 6 or more correct answers, display a happy image and display an equally appropriate image if the score is less than 4. This picture box should be invisible until the student completes the test. The user should also be given the option on the result’s screen to exit the program using a VByes/no option. The time taken to complete the test must be indicated on this form also.



**FORM 4**

4. **The Calculator**



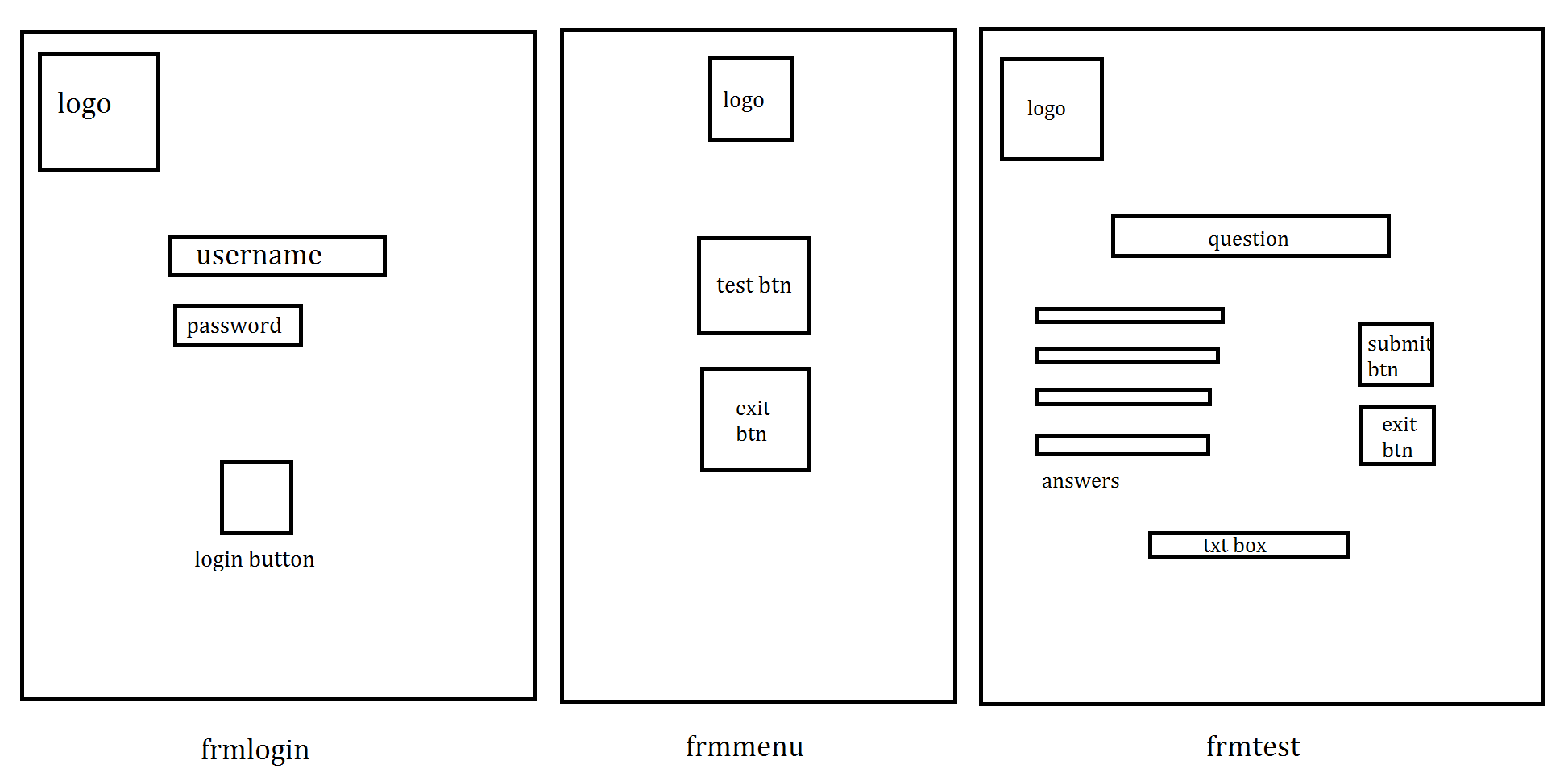
**FORM 5**

**The calculator** screen will provide the **student** with access to a realistic calculator with buttons for **(0-9, & decimal)** that will enable the **student** to add, subtract, divide and multiply numeric data. The calculator will also have a  button to return the square root of values requested by the **user**, a **memory plus** button  to accumulate values as required, a **memory recall** button **MR** and **clear button** that will clear all the appropriate controls on this screen. **Ensure that there is no repetition of decimal point as a number is added to the text box when the user clicks on the decimal command button.** This calculator will respond to click events only.

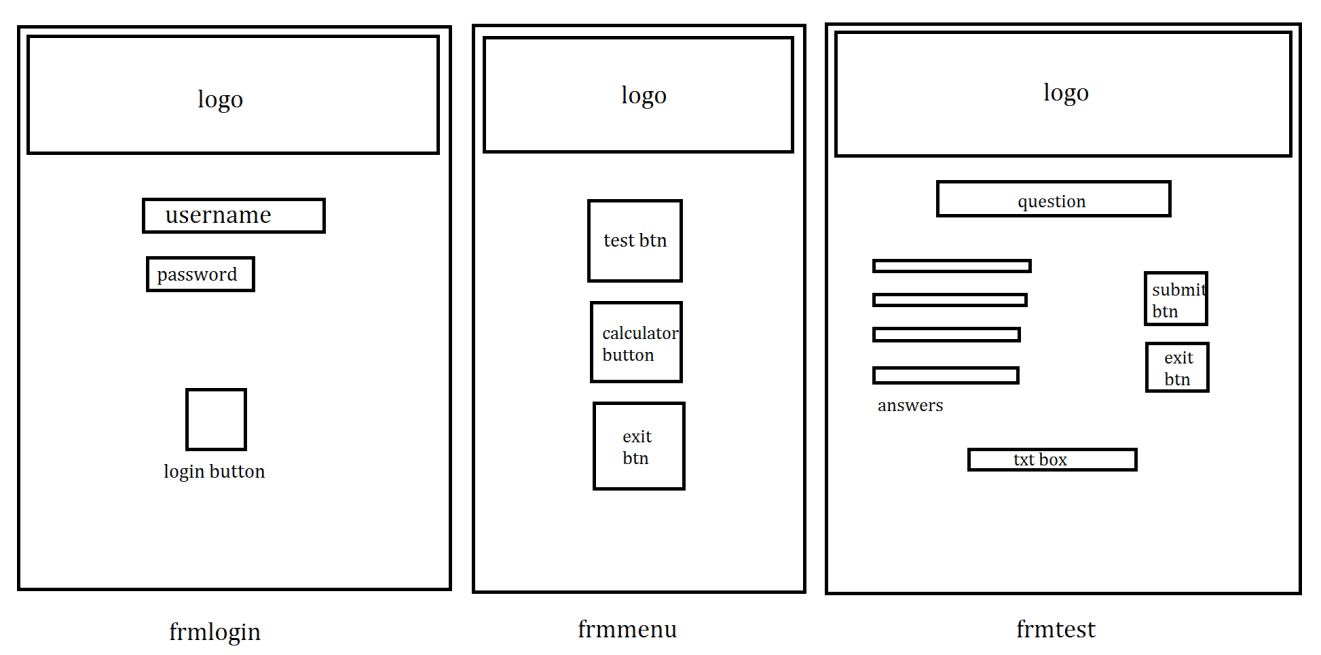
5. Please submit a short report as follows;

(i) Illustrate your use of prototyping and rapid applications development during the design process by providing three versions of the forms making up your application during development. Indicate the 2nd and 3rd prototype represents an improvement on its predecessor.

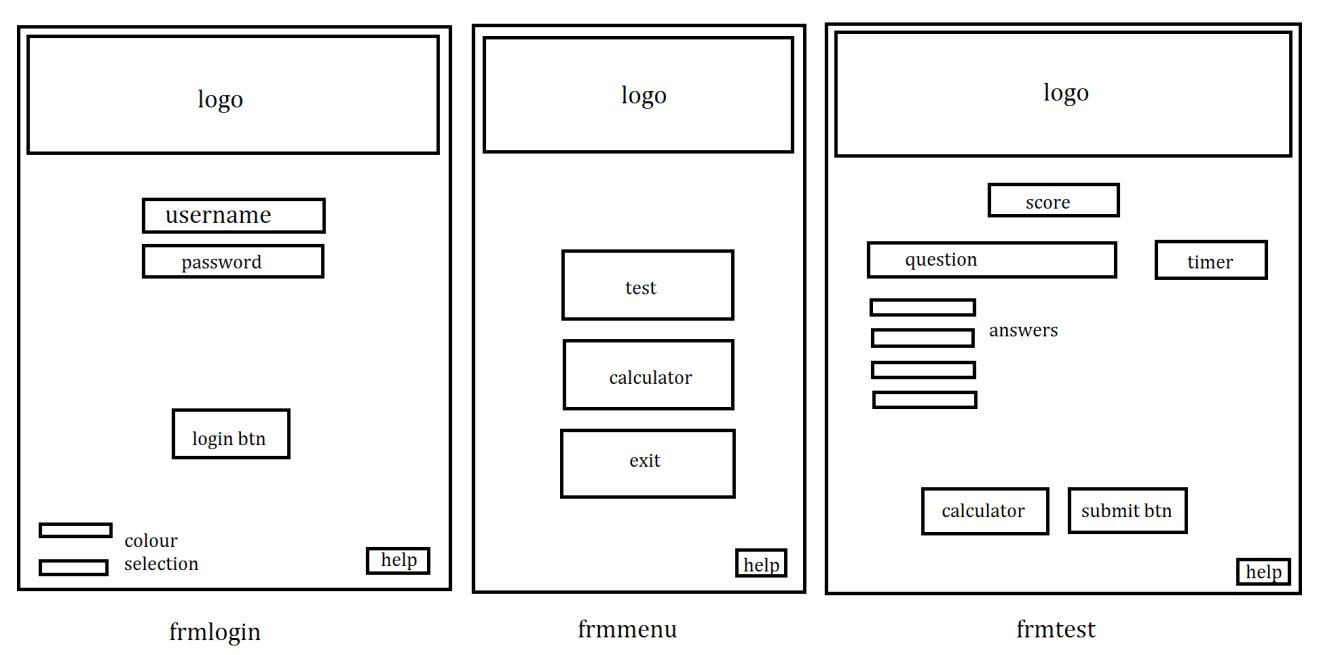
The first version was the most basic version including only relevant fields as shown in a prototype drawing below. Form for login screen had a text field where user had to type username and a text box below where user had to type password as well. Upon successful login, he was presented with menu screen where he had an option to take the test or exit. The test form was consisting of label for question, labels for answers and two buttons for submitting and exit. The answer had to be typed in to the text box.



Second version of the program brought some changes. The most obvious change was the visual look of the logo as it was expanded from simple square box to a rectangle that covers most of the upper part of the form. The calculator button was also introduced, but all the other functions were left almost the same.



The last version was the final version as it implemented some useful changes. In this version, user didn’t have to type in his username but could simply select it from the combo box. An option to change the background colour to red or blue was also introduced. Another new feature was added called “help” where user can open the window with instructions on each form. Regarding the question forms, the score bar, timer and calculator button were added.

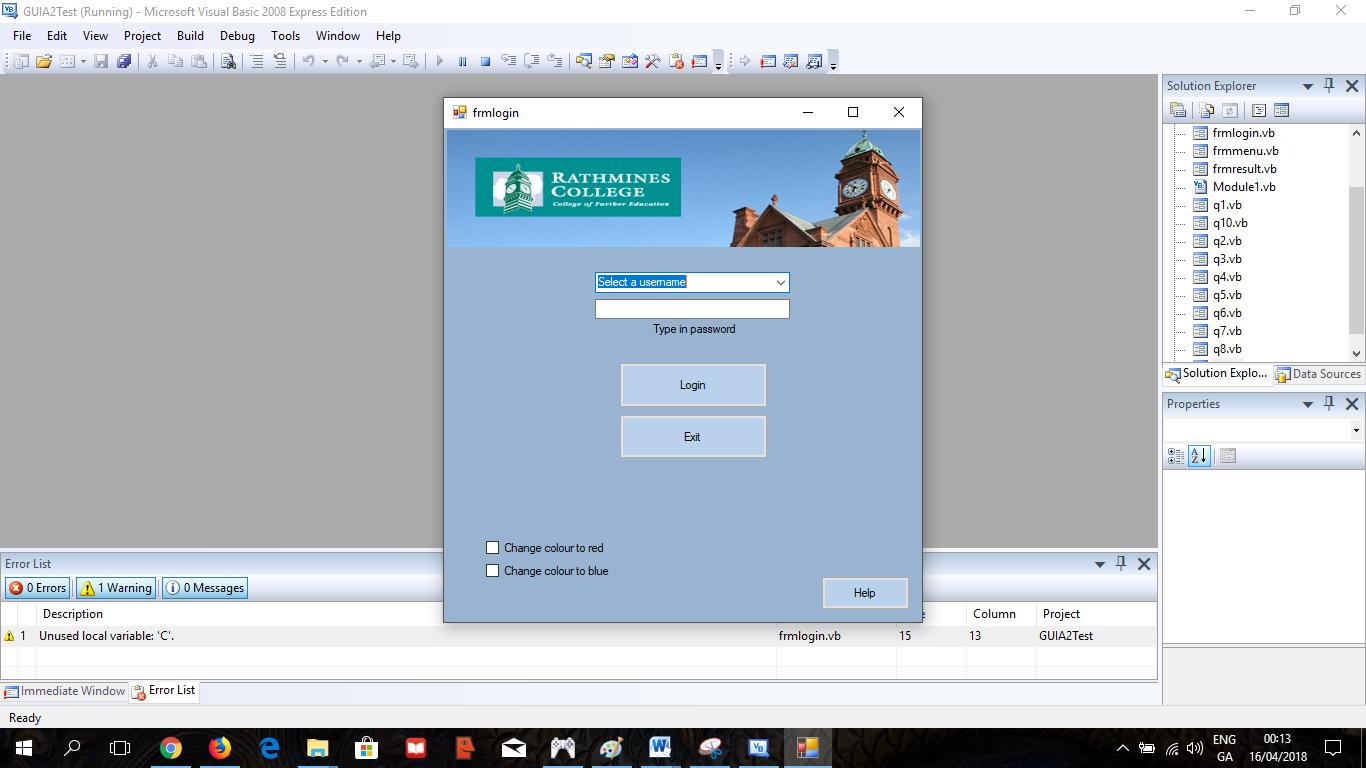


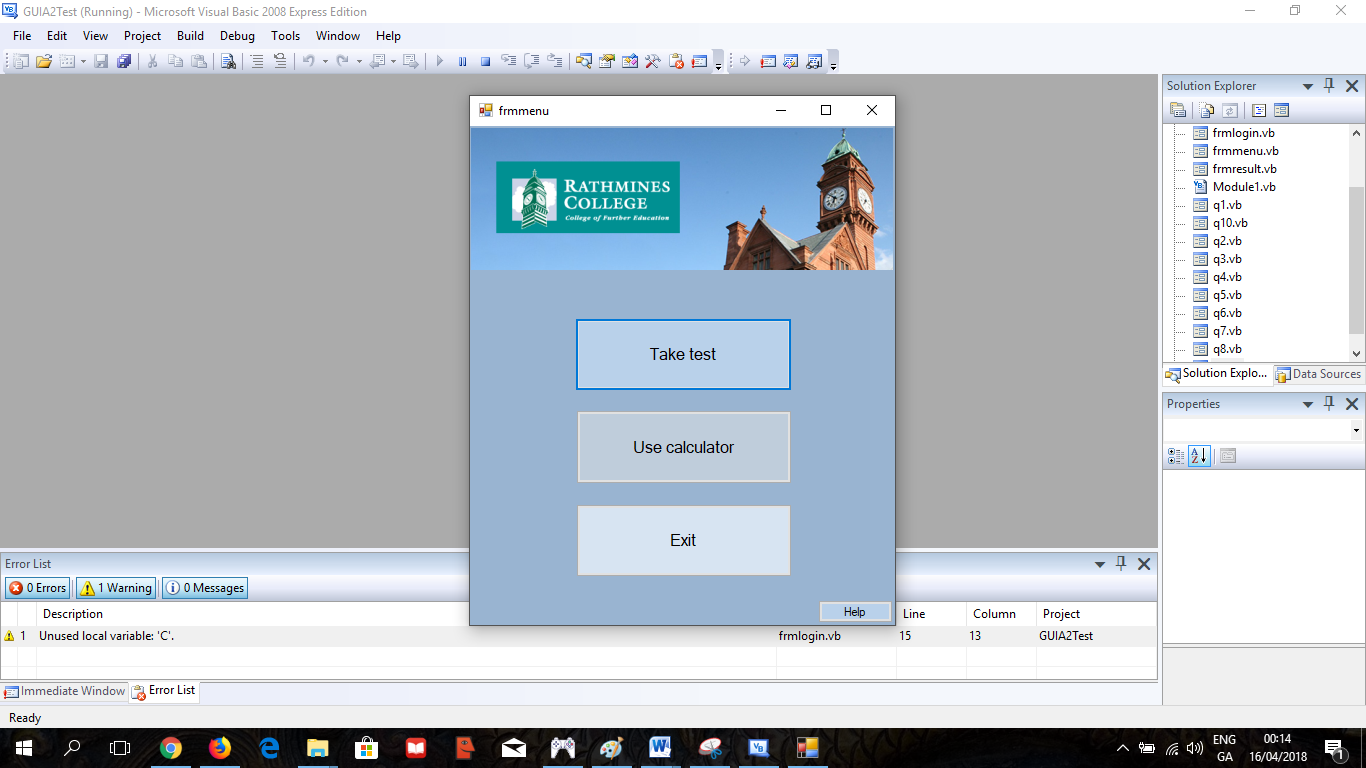
(ii)The report must include an outline of a typical deployment plan and an appropriate deployment plan for your application . (LO12).

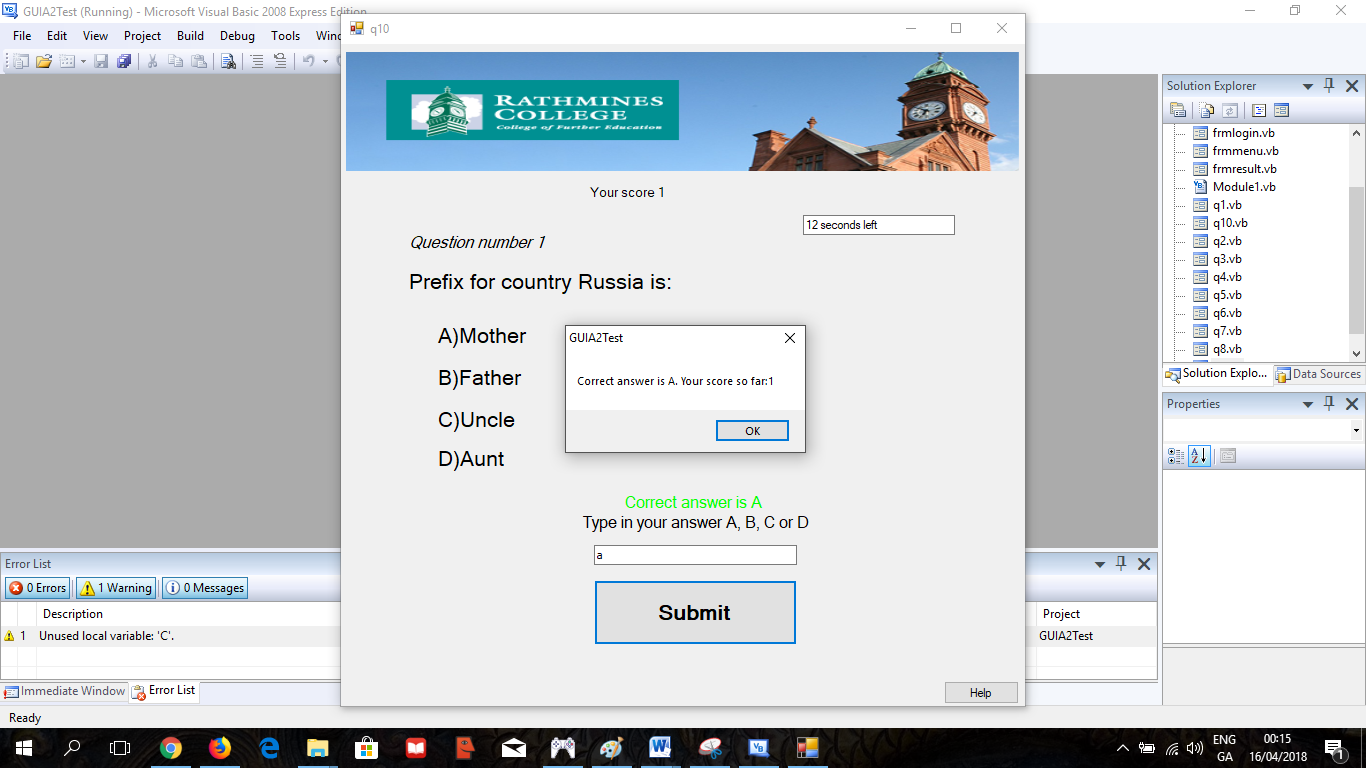
In order to use this program, user will need to have a working computer with keyboard, mouse and a monitor. The computer will need to have at least 1GB of RAM, standard keyboard with English language input and numeric input enabled. This program was tested on Windows 7 and Windows 10. On Windows 7, as well as Windows 10, the program works as expected without any crashes. Of course, more powerful processor and more rams will result in program running faster.

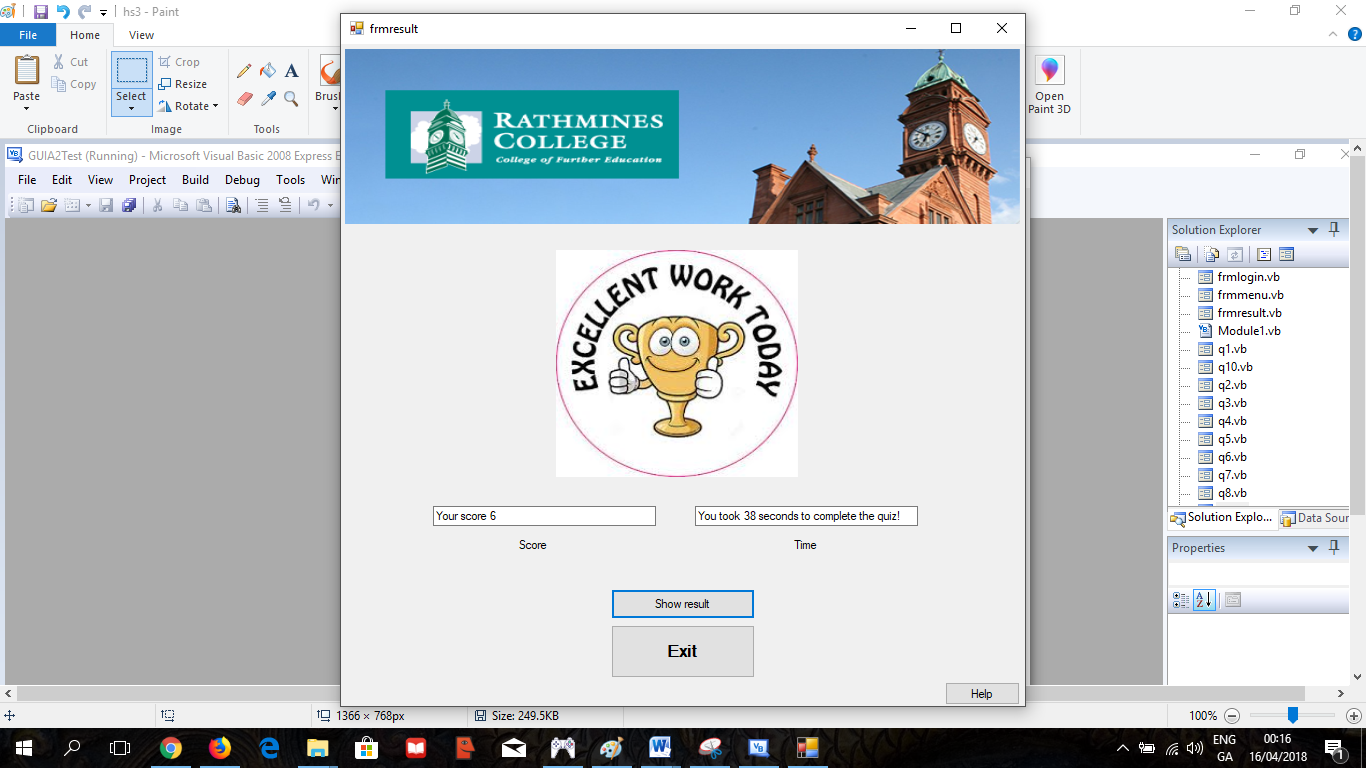
(iii) Include evidence (including screen grabs) of independent testing of your application (on a system that has not been used during development process to ensure that the completed and installed package works in an external environment e.g. your own laptop/PC). The evidence will include suitable test data, expected results and actual results resulting from the deployment of the application.

The following screen grabs were taken as I tested this program on my personal PC (Windows 10 64 bit, 8GB RAM, AMD A8 2.20 GHz)



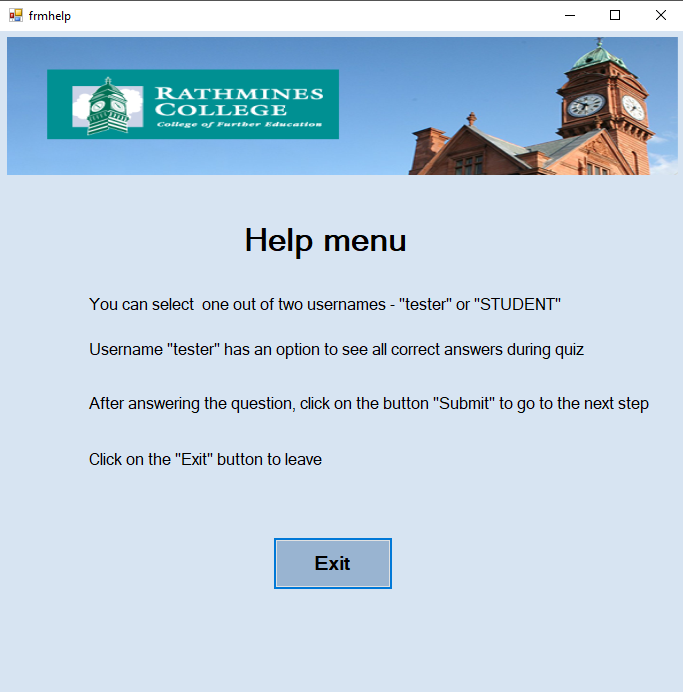






6.

Your application must have a help subsystem consisting of a command button that opens a help screen accessible from all forms with user information regarding features and facilities of the current form e.g. login, calculator and MCQ and results screens.



Marks are allocated for simplicity and consistency of design. Include **three elements of design enhancement** to each form using **consistent background colour**, **text colour,** text **formatting**, **text size** etc. Include two instructions on each form e.g.indicating the presence of help button for the help system, explaining user rights or describing what the current form enables the user to do. Each form must provide appropriate validation for error trapping and context sensitive error messages e.g. when users encounters an invalid password or invalid data. The program should be documented appropriately with four comments on each form.

Follow the naming conventions for forms and controls.

Start-up position for each form should be CentreScreen.

All forms should be the same size.

Define a keyboard access key for each Command control.

.

7. Save the program in a zipped folder called your name GUI A2 2018

This folder should be submitted on MOODLE or emailed to [seanridge@rathminescollege.ie](mailto:seanrodge@rathminescollege.ie) Friday 23rd March 2018