**Mid-Term Exam**

**Windows Application Programming using .Net**

**Time Allowed: 2 Hours**

Student Name: \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ Student Id# \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

**Instructions:**

1. Make a windows application in Visual Studio and name it as **MTFirstnameLastname**.
2. Design and implement a Windows Forms Application that looks something like the one below:
   * The form layout does not have to be the same.
   * Use your own imagination to create the layout of the forms, and use of background colors.

|  |  |
| --- | --- |
|  | *Fig. 1: Form 1 – Initial form load, populate the combo-box with customer names and set the first item as “Select Customer”. Disable “Select Payment” button.* |
| *Fig. 2: Form 2 – If Credit card radio button is selected, enable all the fields, and you can’t go back to Form 1 until Form 2 isn’t closed.* | *Fig. 3: Form 1 – Display the values from Form 2 in the label on Form 1.* |
| *Fig. 4: Form 2 – If Bill customer radio button is selected, disable all the fields, and you can’t go back to Form 1 until Form 2 isn’t closed.* | *Fig. 5: Form 1 – Display “Bill customer” in the label on Form 1, if Bill customer radio button is selected on Form 2.* |

1. On the initial form load, populate the combo-box with customer names from a collection, such as a **List< >**.
   * Maintain the names of customers in a suitable collection.
   * After populating the combo-box, set the first item as **“Select customer”**, and should be default selected.
   * The combo-box should be read-only, and **Select Payment** button should be disabled.
2. On selecting a customer from the combo-box, enable the **Select Payment** button.
   * If **“Select customer”** is selected again in the combo-box, disable the **Select Payment** button.
3. On clicking the **Select Payment** button, display **Form 2**.
   * As long as **Form 2** is displayed, user can’t go back to **Form 1** without closing it.
4. On **Form 2**, if **Credit card** radio button is selected, enable all the fields.
   * For **Credit card type** – populate the **ListBox** with three credit card names.
   * For **Card number** – user can type in the card number in the **TextBox**.
   * For **Expiration date** – populate the month combo-box with months of the year. Populate the year combo-box with years starting from current year to five years in future. Do not   
     hard-code the current year.
   * For the check box – user can select it to set this as the default billing method.
5. On clicking **OK**, **Form 2** closes and the information filled in this form is passed back to **Form 1**, which displays it in the label.
   * The information displayed in the label should match the exact format and layout as displayed in *Fig. 3* on Page 1.
6. On **Form 2**, if **Bill customer** radio button is selected, disable all the fields.
7. On clicking **OK**, **Form 2** closes and displays **“Bill customer”** in the label in **Form 1**.
8. **Save** button on **Form 1** isn’t implemented.
9. **Exit** button on **Form 1** closes the application.
   * Display a confirmation message before the application is closed.
   * The confirmation message should be displayed by either clicking on the ***Exit*** button, or clicking on the cross (X) button in the title bar of the **Form 1**.
   * The confirmation message should only be displayed while closing **Form 1**, not while closing **Form 2**.
10. This application must be implemented using OOP approach.
11. Ensure all names follow the naming conventions used in the class (for classes, methods, properties, variables, controls etc.).
12. Ensure both forms are displayed centered on the screen.
    * Set the title text on both the forms.
    * Disable the Maximize button.
    * Set the border style to fixed, so that it can’t be resized by dragging its borders.
13. Ensure the tab order of all controls is set correctly.
14. Once you are finished, ZIP the project, and upload it into *Drop box on Blackboard*.