1. INTRODUCTION

VB.NET stands for Visual Basic.NET, and it is a computer programming language developed by Microsoft. It was first released in 2002 to replace Visual Basic 6. VB.NET is an object-oriented programming language. This means that it supports the features of object-oriented programming which include encapsulation, polymorphism, abstraction, and inheritance. Visual Basic .ASP NET runs on the .NET framework, which means that it has full access to the .NET libraries. It is a very productive tool for rapid creation of a wide range of Web, Windows, Office, and Mobile applications that have been built on the .NET framework.

The language was designed in such a way that it is easy to understand to both novice and advanced programmers. Since VB.NET relies on the .NET framework, programs written in the language run with much reliability and scalability. With VB.NET, you can create applications that are fully object-oriented, similar to the ones created in other languages like C++, Java, or C#. Programs written in VB.NET can also interoperate well with programs written in Visual C++, Visual C#, and Visual J#. VB.NET treats everything as an object.It is true that VB.NET is an evolved version of Visual Basic 6, but it's not compatible with it. If you write your code in Visual Basic 6, you cannot compile it under VB.NET.

- VB.NET was developed by Microsoft.
- It is an object-oriented language.
- The language is not case sensitive.
- VB.NET programs run on the .NET framework.
- In VB.NET, the garbage collection process has been automated.
- The language provides windows forms from which you can inherit your own forms.
- VB.NET allows you to enjoy the drag and drop feature when creating a user interface.

1.1 VB.NET Features:

VB.NET comes loaded with numerous features that have made it a popular programming language amongst programmers worldwide. These features include the following:

- VB.NET is not case sensitive like other languages such as C++ and Java.
- It is an object-oriented programming language. It treats everything as an object.
- Automatic code formatting, XML designer, improved object browser etc.
- Garbage collection is automated.
- Support for Boolean conditions for decision making.
- Simple multithreading, allowing your apps to deal with multiple tasks simultaneously.
- Simple generics.
- A standard library.
- Events management.

2. Design Process

1) Windows Form

It is a graphical API to display data and manage user interactions with easier deployment and better security in client applications. Windows Forms offers an extensive client library providing interface to access native Windows graphical interface elements and graphics from managed code. Visual Basic Form is the container for all the controls that make up the user interface. Every window you see in a running visual basic application is a form, thus the terms form and window describe the same entity. Visual Studio creates a default form for you when you create a Windows Forms Application.

2) Label

Labels are one of the most frequently used Visual Basic control. A Label control lets you place descriptive text. Where the text does not need to be changed by the user. If you want to change the display text of the Label, you have to set a new text to the Text property of Label. Windows Forms Label controls are used to display text or images that cannot be edited by the user. They are used to identify objects on a form to provide a description of what a certain control will do if clicked, for example, or to display information in response to a run-time event or process in your application.

3) Text Box

A Text Box control is used to display, or accept as input, a single line of text. VB.Net programmers make extensive use of the Text Box control to let the user view or enter large amount of text. A text box object is used to display text on a form or to get user input while a VB.Net program is running. Text box controls allow entering text on a form at runtime. By default, it takes a single line of text, however, you can make it accept multiple texts and even add scroll bars to it. Let's create a text box by dragging a Text Box control from the Toolbox and dropping it on the form.

4) Error Provider

Error Provider Control in VB.Net. ErrorProvider allows us to set an error message for any control on the form when the input is not valid. When an error message is set, an icon indicating the error will appear next to the control and the error message is displayed as Tool Tip when the mouse is over the control.

Error Provider Control is used to set Error Message for particular control on the Form. You can also also get Error Message associated with particular control on the form using Error Provider Control.

5) Button

VB.Net - Button Control. The Button control represents a standard Windows button. It is generally used to generate a Click event by providing a handler for the Click event. Let's create a label by dragging a Button control from the Toolbox ad dropping it on the form. Button in VB.NET. Button class in Windows Forms represents a Button control. A Button control is a child control placed on a Form and used to process click event and can be clicked by a mouse click or by pressing ENTER or ESC keys.

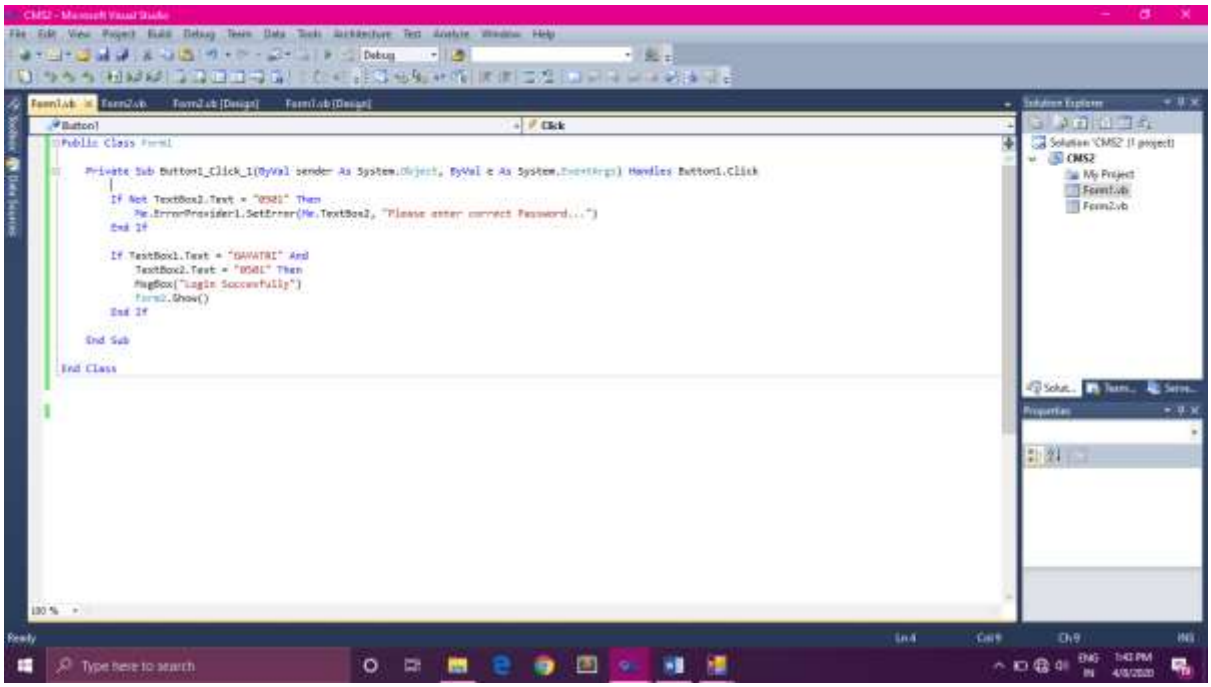
VB.Net - Button Control. The Button control represents a standard Windows button. It is generally used to generate a Click event by providing a handler for the Click event. Let's create a label by dragging a Button control from the Toolbox ad dropping it on the form.

6) Combo Box

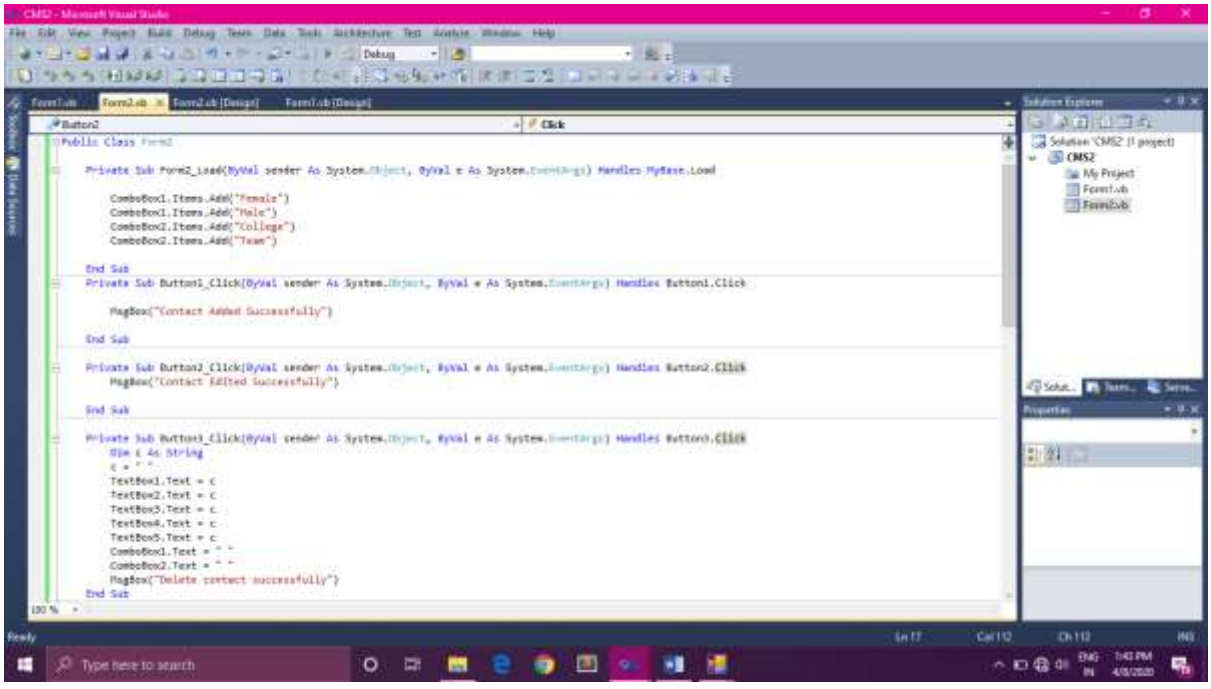
Combo Box Control. The Combo Box control is used to display a drop-down list of various items. It is a combination of a text box in which the user enters an item and a drop-down list from which the user selects an item. See it as a combination of a textbox in which a user enters text and a dropdown list from which a user selects an item. Note that the combo box shows one item at a time.

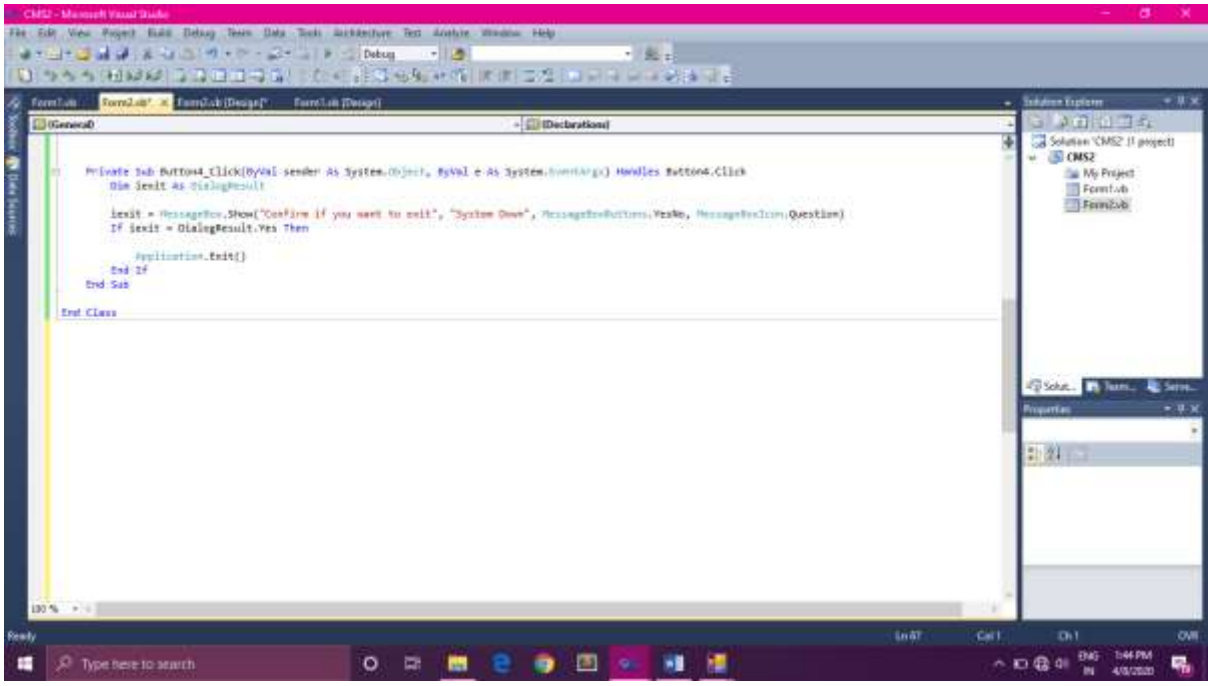
3. IMPLEMENTATION

3.1 Form 1 coding



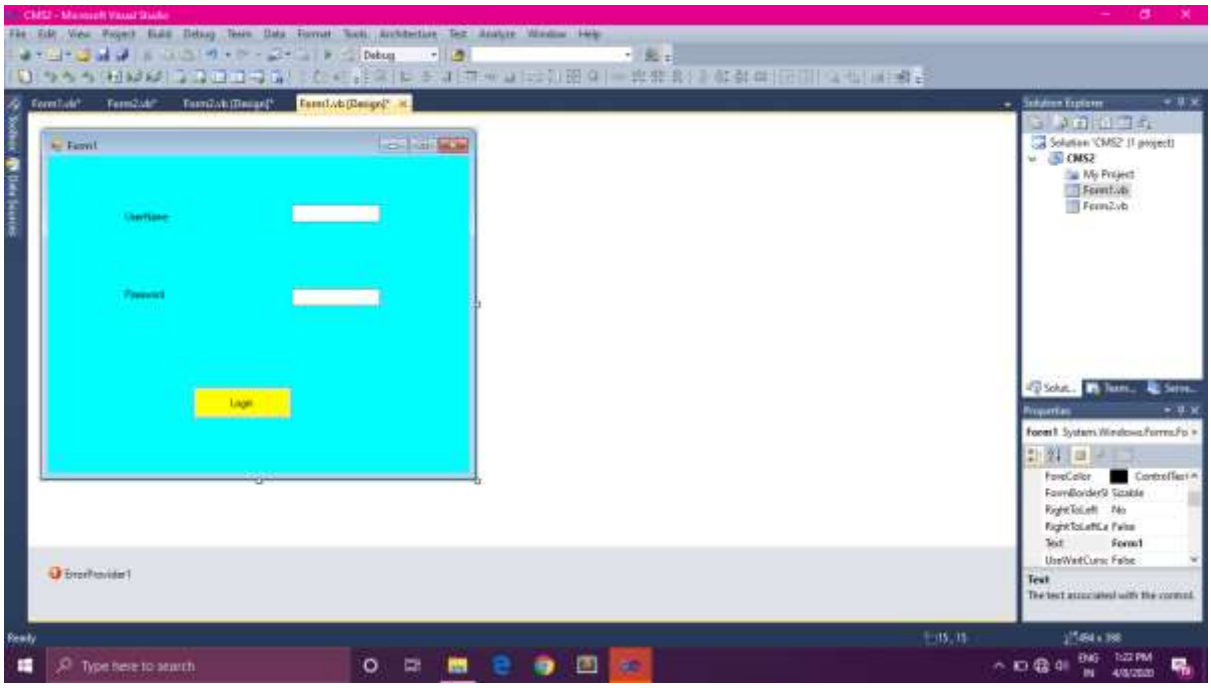
3.2 Form 2 Coding



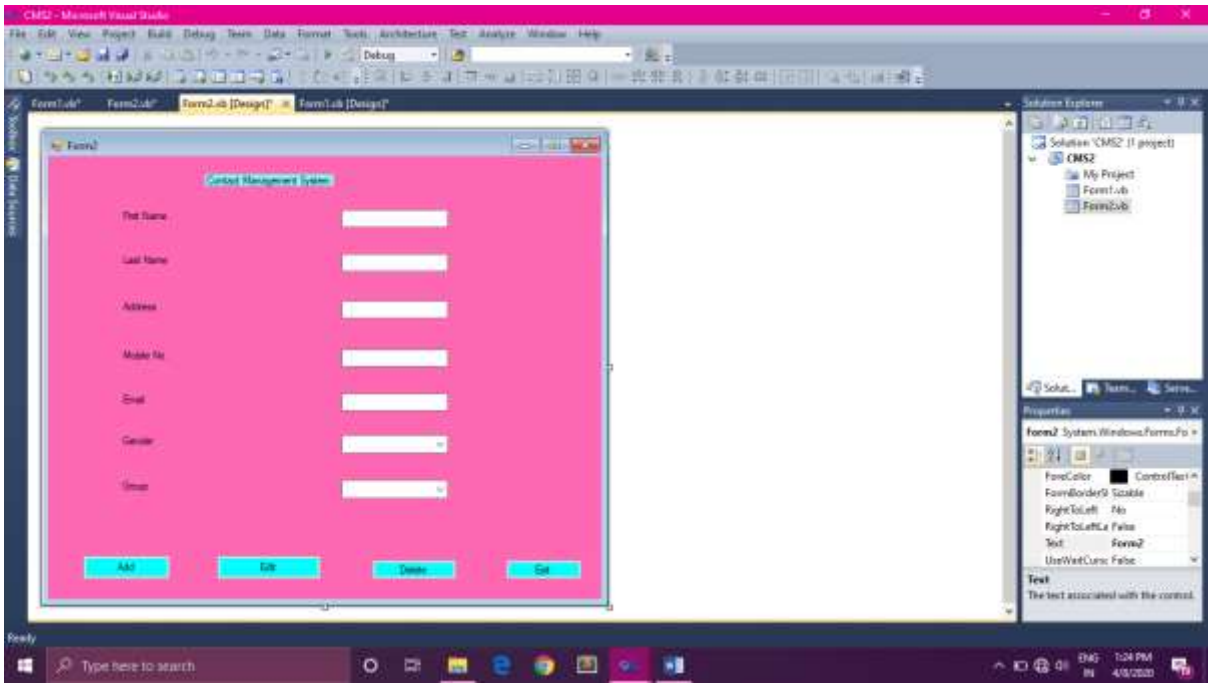

Private Sub Button4_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles Button4.Click
Dim iexit As DialogResult
iexit = MessageBox.Show("Confirm if you want to exit", "System Close", MessageBoxButtons.YesNo, MessageBoxIcon.Question)
If iexit = DialogResult.Yes Then
Application.Exit()
End If
End Sub
End Class

4. SCREENSHOT:

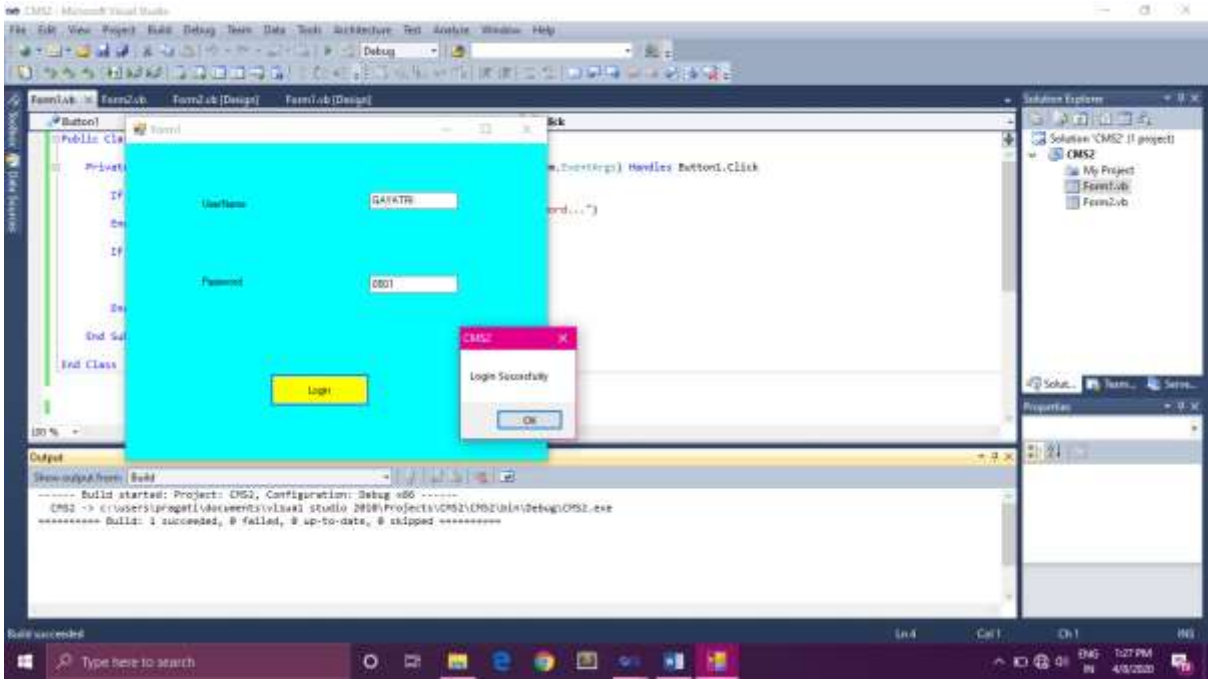
4.1 Form 1

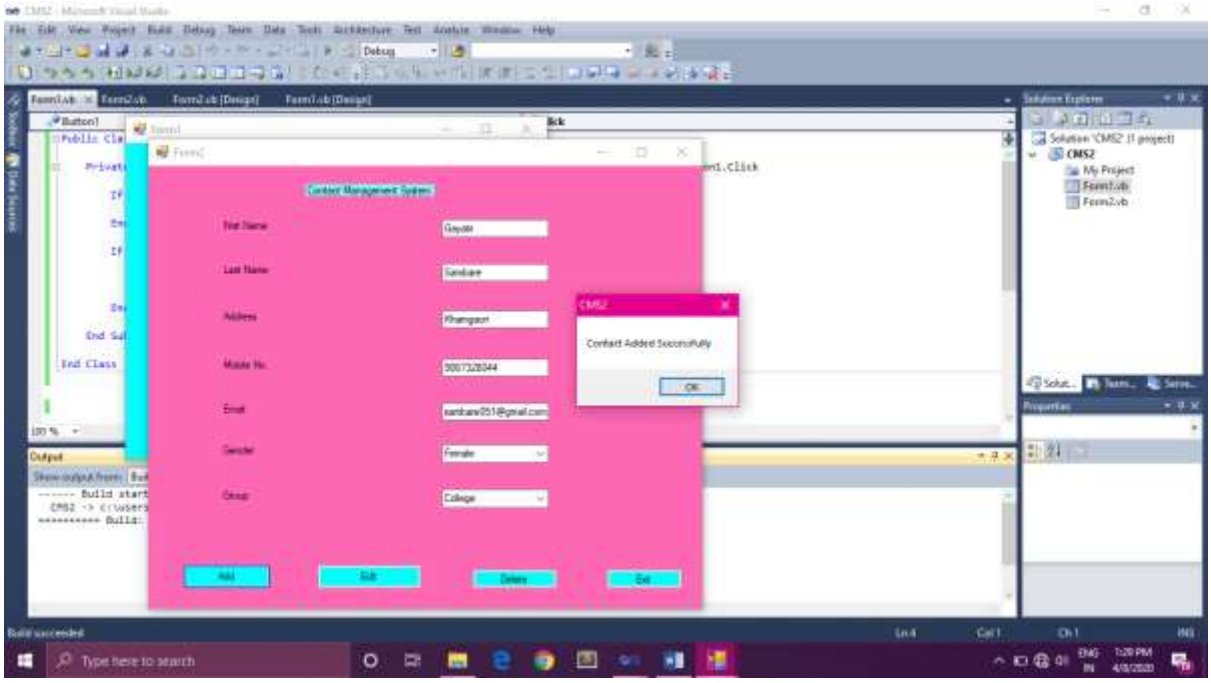


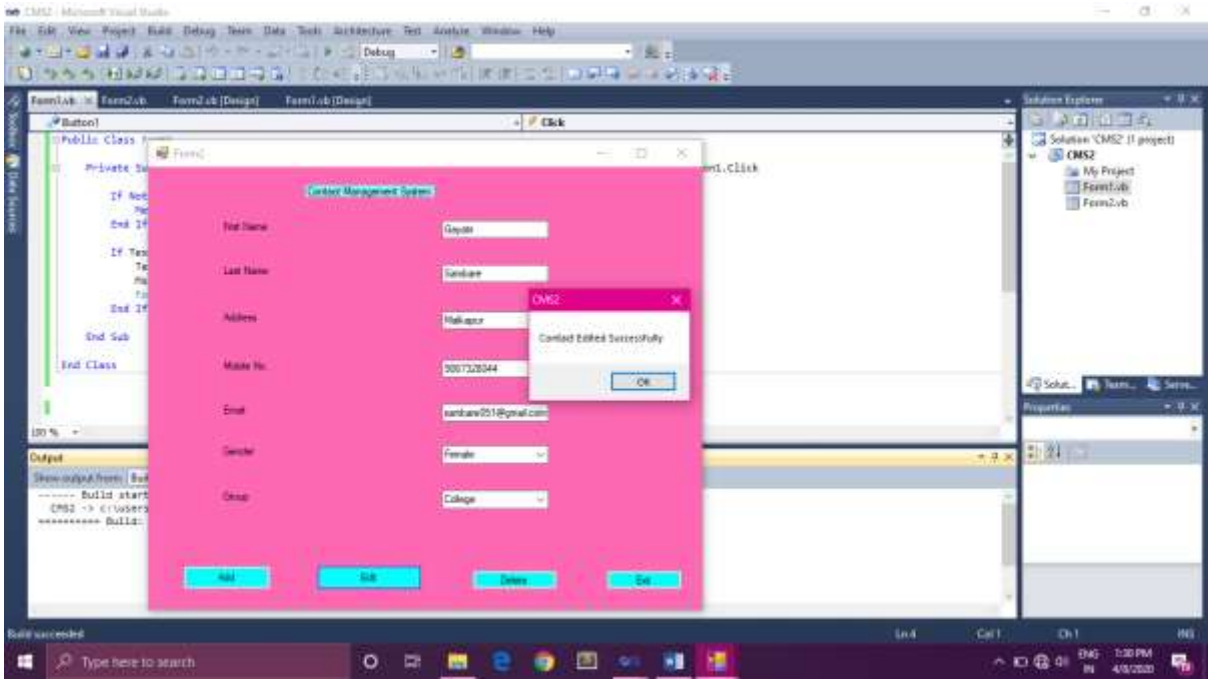
4.2 Form 2

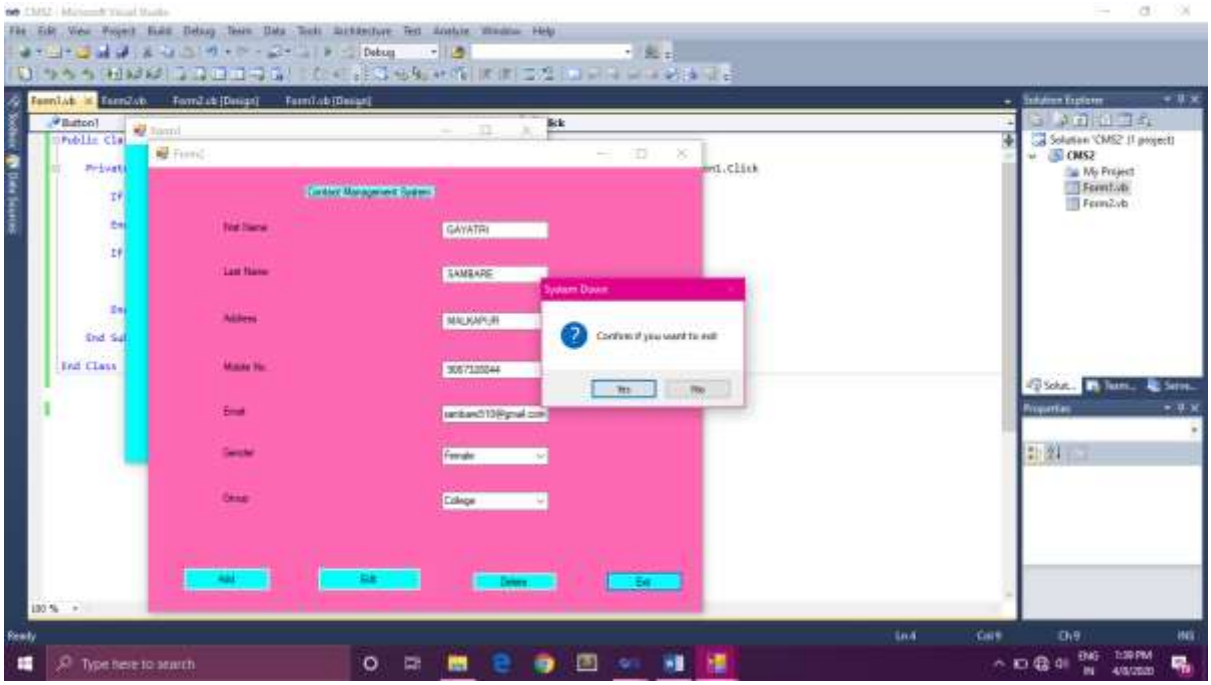


4.3 Code Execution






Contact Management System
CMS2
Contact Edited Successfully
OK


Contact Management System
GAYATRI
SAMBARE
MALKAPUR
Confirm if you want to exit
Yes
No

5. CONCLUSION

In this way, we create our "Graphical User Interface Application Using VB.Net" project "Contact Management System" with partial fulfillment Submitted in partial fulfillment of the requirement for the diploma of computer engineering. We studied that Graphical User Interface Application Using VB.Net is a special way of organizing and storing data in a computer so that it can be used efficiently. In this project we present the use of different windows forms. The main aim to present this project is increasing the basic general knowledge of students. We create it by using 'Microsoft Visual Basic Studio' software.

We studied in this project more about the subject "Graphical User Interface Application Using VB.Net". We understand how to link one form to another form each other.

6. REFERENCE

1. https://www.guru99.com/vb-net-introduction-features.html
2. https://www.freeprojectz.com/vbnet-projects-projects/vbnet-project-contact-management-system
3. https://www.tutorialspoint.com/vb.net/vb.net_button.htm
4. https://www.tutorialspoint.com/vb.net/vb.net_forms.htm
5. http://howtostartprogramming.com/vb-net/vb-net-tutorial-11-linking-forms/