

Jonathan Benedict Sirait

001201900022

IT 3 2019

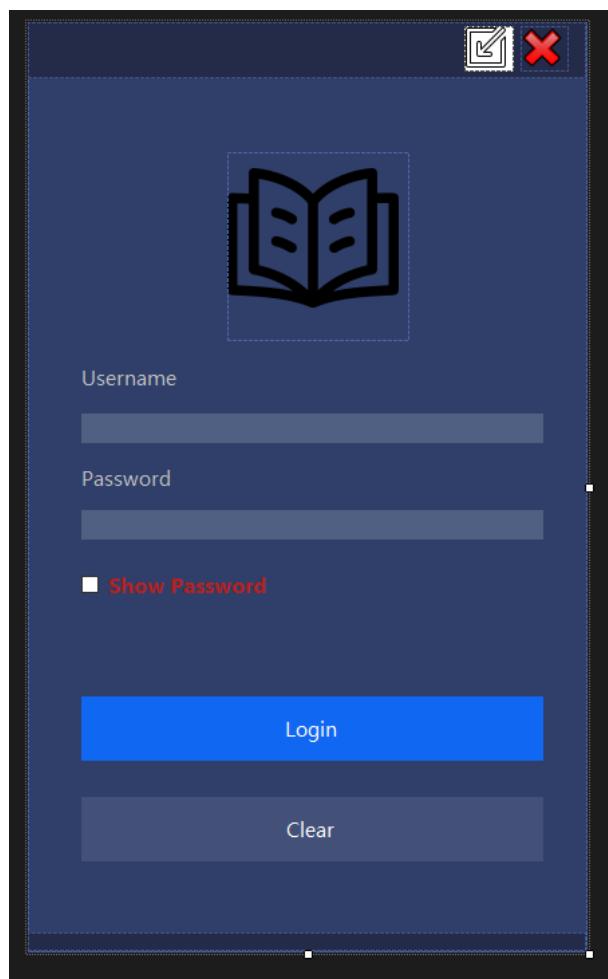
*(Since the code is many, I took a screenshot at large view which cause the text become smaller. Due to that condition, I would suggest to zoom in (**using hotkeys ctrl + scroll up**) to clarify the code or the design. I'm really sorry for the inconvenient.)

PU Library Management System (PU LMS)

1. Design & Code

1) Universal

a. Login Form



```

Imports System.Data.OleDb
Imports System.Runtime.InteropServices

Public Class LoginForm
    Private Sub btnLogin_Click(sender As Object, e As EventArgs) Handles btnLogin.Click
        Dim username, password As String
        username = txtUsername.Text
        password = txtPassword.Text

        If txtUsername.Text = "" Or txtPassword.Text = "" Then
            MsgBox("Please input the missing field", MessageBoxIcon.Warning)
        Else
            If username = "adminPU" And password = "PUadmin" Then
                MsgBox("Login successfully!")
                Me.Hide()
                Dim loading As New Loading()
                loading.Show()
                loading.Text = "Welcome, Admin."
            Else
                Call Connection()
                cmd = New OleDbCommand("SELECT * FROM Tbl_User WHERE [Email] = '" & username & "' AND [Password] = '" & password & "'", conn)
                dr = cmd.ExecuteReader()
                If (dr.Read()) = True Then
                    Me.Hide()
                    Dim loading As New Loading()
                    loading.Show()
                    loading.Text = "Welcome, " & dr("Name")
                    userForm.lblUsername.Text = dr("Name")
                    userForm.lblIDStud.Text = dr("Student_ID")
                    dr.Close()
                    conn.Close()
                Else
                    MsgBox("You've inputted wrong Username or Password! Please try again!", MessageBoxIcon.Warning)
                End If
            End If
        End Sub

        Private Sub btnLogin_Paint(sender As Object, e As PaintEventArgs) Handles btnLogin.Paint
            Dim buttonPath As Drawing2D.GraphicsPath = New Drawing2D.GraphicsPath()
            Dim myRectangle As Rectangle = btnLogin.ClientRectangle
            myRectangle.Inflate(0, 30)
            buttonPath.AddEllipse(myRectangle)
            btnLogin.Region = New Region(buttonPath)
        End Sub

        Private Sub btnClear_Paint(sender As Object, e As PaintEventArgs) Handles btnClear.Paint
            Dim buttonPath As Drawing2D.GraphicsPath = New Drawing2D.GraphicsPath()
            Dim myRectangle As Rectangle = btnClear.ClientRectangle
            myRectangle.Inflate(0, 30)
            buttonPath.AddEllipse(myRectangle)
            btnClear.Region = New Region(buttonPath)
        End Sub

        Private Sub btnClear_Click(sender As Object, e As EventArgs) Handles btnClear.Click
            txtUsername.Clear()
            txtPassword.Clear()
        End Sub

        Private Sub chkBoxShowPass_CheckedChanged(sender As Object, e As EventArgs) Handles chkBoxShowPass.CheckedChanged
            If chkBoxShowPass.CheckState = CheckState.Checked Then
                txtPassword.PasswordChar = ""
            Else
                txtPassword.PasswordChar = "*"
            End If
        End Sub
    End Class

```

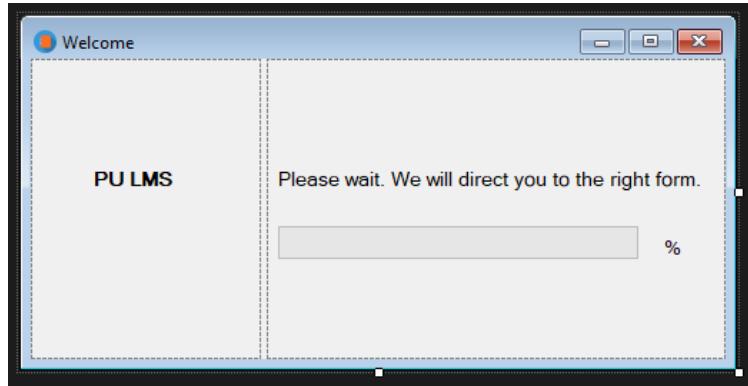
```

REFERENCES
    Private Sub pictExit_Click(sender As Object, e As EventArgs) Handles pictExit.Click
        If MessageBox.Show("Are you sure want to close the application?", "Warning",
                           MessageBoxButtons.YesNo, MessageBoxIcon.Warning) = DialogResult.Yes Then
            Application.Exit()
        End If
    End Sub
    0 references
    Private Sub pictMinimize_Click(sender As Object, e As EventArgs) Handles pictMinimize.Click
        Me.WindowState = FormWindowState.Minimized
    End Sub

<DllImport("user32.DLL", EntryPoint:="ReleaseCapture")>
2 references
    Private Shared Sub ReleaseCapture()
    End Sub
<DllImport("user32.DLL", EntryPoint:="SendMessage")>
2 references
    Private Shared Sub SendMessage(hWnd As IntPtr, wMsg As Integer, wParam As Integer, lParam As Integer)
    End Sub
    0 references
    Private Sub titleBar_MouseDown(sender As Object, e As MouseEventArgs) Handles titleBar.MouseDown
        ReleaseCapture()
        SendMessage(Me.Handle, &H112&, &HF012&, 0)
    End Sub
    0 references
    Private Sub FormLogin_MouseDown(sender As Object, e As MouseEventArgs) Handles MyBase.MouseDown
        ReleaseCapture()
        SendMessage(Me.Handle, &H112&, &HF012&, 0)
    End Sub

```

b. Loading Form



```

4 references
    Public Class Loading
    0 references
        Private Sub Loading_Load(sender As Object, e As EventArgs) Handles MyBase.Load
            loadingTimer.Start()
        End Sub
    0 references
        Private Sub loadingTimer_Tick(sender As Object, e As EventArgs) Handles loadingTimer.Tick
            If progBarLoading.Value < 100 Then
                progBarLoading.Value = progBarLoading.Value + 10
                lblPercent.Text = Math.Round((progBarLoading.Value / 100) * 100, 2) & "%"
            ElseIf progBarLoading.Value = 100 Then
                If LoginForm.txtUsername.Text = "adminPU" Then
                    Call LoadingAdmin()
                Else
                    Call LoadingUser()
                End If
            End If
        End Sub
    1 reference
        Private Sub LoadingAdmin()
            loadingTimer.Stop()
            Me.Hide()
            MainForm.Show()
            MsgBox("Loading completed. Welcome Admin!", MessageBoxIcon.Information)
        End Sub
    1 reference
        Private Sub LoadingUser()
            loadingTimer.Stop()
            Me.Hide()
            userForm.Show()
            MsgBox("Loading completed. Welcome User!", MessageBoxIcon.Information)
        End Sub
    End Class

```

c. LibraryFunction (Class)

```

1  Imports System.Data.OleDb
2  Public Class LibraryFunction
3      Shared Function ShowUser() As DataTable
4          Dim dt As New DataTable
5          Call Connection()
6          cmd = New OleDbCommand("SELECT * FROM Tbl_User ORDER BY Name", conn)
7          dr = cmd.ExecuteReader()
8          dt.Load(dr)
9          conn.Close()
10         dr.Close()
11         Return dt
12     End Function
13     Shared Sub AddUser(_id, _name, _major, _batch, _email, _pwd, _phoneNum, _gender, _actDate, _valdTill, _addr)
14         Call Connection()
15         cmd = New OleDbCommand("INSERT INTO Tbl_User VALUES ('" & _id & "','" & _name & "','" & _major & "','" & _batch & "'",
16         :           "' & _email & "','" & _pwd & "','" & _phoneNum & "','" & _gender & "','" & _actDate & "','" & _valdTill & "'",
17         :           "' & _addr & "')", conn)
18         If cmd.ExecuteNonQuery() Then MsgBox("New user has been inputted!", MessageBoxIcon.Information)
19         conn.Close()
20     End Sub
21     Shared Sub DeleteUser(_id)
22         Call Connection()
23         cmd = New OleDbCommand("DELETE * FROM Tbl_User WHERE Student_ID ='" & _id & "", conn)
24         If cmd.ExecuteNonQuery() Then MsgBox("Selected user has been deleted!", MessageBoxIcon.Information)
25         conn.Close()
26     End Sub
27     Shared Sub EditUser(_id, _email, _pwd, _major, _phoneNum, _addr)
28         Call Connection()
29         cmd = New OleDbCommand("UPDATE [Tbl_User] SET [Email] ='" & _email & "', [Password] ='" & _pwd & "', [Major] ='" & _major & "'",
30         :           [Phone_Num] ='" & _phoneNum & "', [Address] ='" & _addr & "' WHERE [Student_ID] ='" & _id & "'", conn)
31         da = New OleDbDataAdapter(cmd)
32         da.UpdateCommand = conn.CreateCommand()
33         da.UpdateCommand.CommandText = "UPDATE [Tbl_User] SET [Email] ='" & _email & "', [Password] ='" & _pwd & "', [Major] ='" & _major & "'",
34         :           [Phone_Num] ='" & _phoneNum & "', [Address] ='" & _addr & "' WHERE [Student_ID] ='" & _id & "'"
35         If da.UpdateCommand.ExecuteNonQuery() Then MsgBox("Data has been updated!", MessageBoxIcon.Information)
36         conn.Close()
37     End Sub
38
39     Shared Function ShowBook( ByVal opt As Integer) As DataTable
40         Dim dt As New DataTable
41         Call Connection()
42         If opt = 1 Then
43             cmd = New OleDbCommand("SELECT * FROM Tbl_Book WHERE [Book_Status] = 'Available' ORDER BY Book_Name", conn)
44         ElseIf opt = 2 Then
45             cmd = New OleDbCommand("SELECT Tbl_Book.[Book_ISBN], Tbl_Book.[Book_Name], Tbl_Book.[Book_Author], Tbl_Book.[Book_Genre],",
46             :           Tbl_Book.[Book_Price], Tbl_Book.[Book_Status]",
47             :           WHERE Tbl_Book.[Book_Status] = 'Rented' AND Tbl_Book.[Student_ID] = Tbl_Book.[Student_ID] ORDER BY Tbl_Book.[Book_Name]", conn)
48         Else
49             cmd = New OleDbCommand("SELECT * FROM Tbl_Book ORDER BY Book_Name", conn)
50         End If
51         da = New OleDbDataAdapter(cmd)
52         da.Fill(dt)
53         conn.Close()
54         Return dt
55     End Function
56
57     Shared Sub AddBook(_id, _name, _author, _genre, _publisher, _publishYear, _price, _desc, _status)
58         Call Connection()
59         cmd = New OleDbCommand("INSERT INTO Tbl_Book ([Book_ISBN], [Book_Name], [Book_Author], [Book_Genre], [Book_Publisher], [Book_Publish_Year],",
60         :           [Book_Price], [Book_Status])",
61         :           VALUES ('" & _id & "','" & _name & "','" & _author & "','" & _genre & "'",
62         :           "' & _publisher & "','" & _publishYear & "','" & _price & "','" & _desc & "','" & _status & "')", conn)
63         If cmd.ExecuteNonQuery() Then MsgBox("New book has been inputted!", MessageBoxIcon.Information)
64         conn.Close()
65     End Sub
66     Shared Sub DeleteBook(_id)
67         Call Connection()
68         cmd = New OleDbCommand("DELETE * FROM Tbl_Book WHERE Book_ISBN ='" & _id & "", conn)
69         If cmd.ExecuteNonQuery() Then MsgBox("Selected book has been deleted!", MessageBoxIcon.Information)
70         conn.Close()
71     End Sub
72     Shared Sub RentBook(book_id, student_id)
73         Call Connection()
74         cmd = New OleDbCommand("UPDATE Tbl_Book SET [Book_Status] = 'Rented', [Student_ID] = '" & student_id & "' WHERE Book_ISBN ='" & book_id & "'", conn)
75         If cmd.ExecuteNonQuery() Then MsgBox("You have successful rent a book!", MessageBoxIcon.Information)
76         conn.Close()
77     End Sub
78     Shared Sub ReturnBook(book_id)
79         Call Connection()
80         cmd = New OleDbCommand("UPDATE Tbl_Book SET [Book_Status] = 'Available', [Student_ID] = '' WHERE Book_ISBN ='" & book_id & "'", conn)
81         If cmd.ExecuteNonQuery() Then MsgBox("You have successful return a book!", MessageBoxIcon.Information)
82         conn.Close()
83     End Sub
84
85 End Class
86

```

d. ModuleConnection (Module)

```

1 Imports System.Data.OleDb
2 0 references
3 Module ModuleConnection
4     Public cmd As OleDbCommand
5     Public dr As OleDbDataReader
6     Public conn As OleDbConnection
7     Public da As OleDbDataAdapter
8     Public ds As DataSet
9
10    16 references
11    Sub Connection()
12        conn = New OleDbConnection("Provider=Microsoft.Jet.OLEDB.4.0;Data Source=PULibrary.mdb;Persist Security Info=False;")
13        If conn.State = ConnectionState.Closed Then
14            conn.Open()
15            MsgBox("Sukses")
16        End If
17    End Sub
18  End Module
19

```

e. Table Design in Access Database (PULibrary DB)

- Book Table (Tbl_Book)

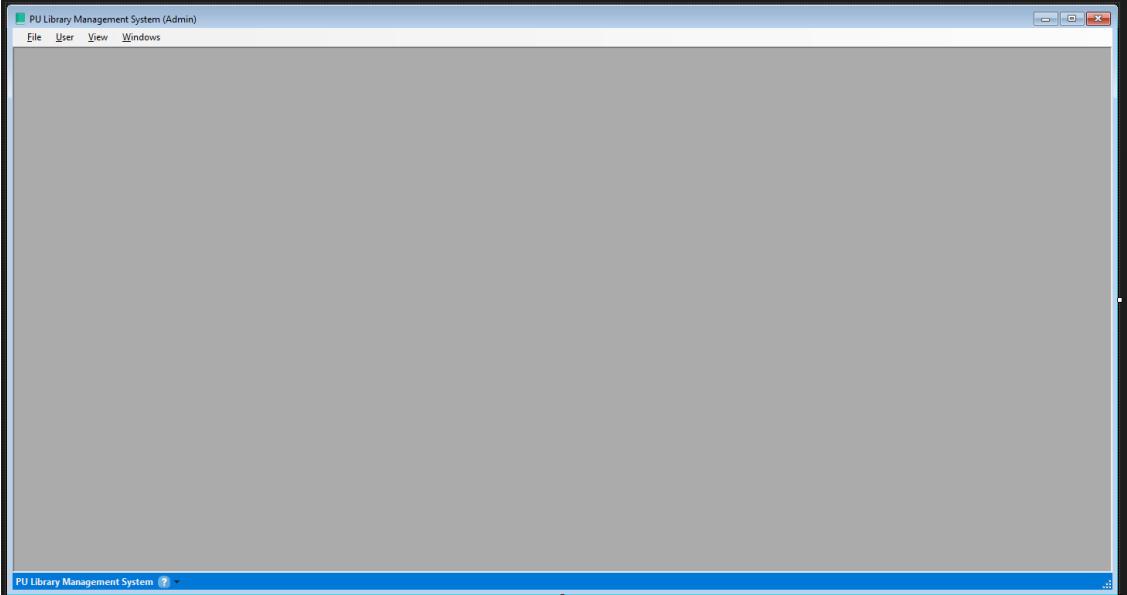
Field Name	Data Type
Book_ISBN	Short Text
Book_Name	Long Text
Book_Author	Short Text
Book_Genre	Short Text
Book_Publisher	Short Text
Book_Publish_Year	Short Text
Book_Price	Number
Book_Desc	Long Text
Book_Status	Short Text
Student_ID	Short Text

- User Table (Tbl_User)

Field Name	Data Type
Student_ID	Short Text
Name	Short Text
Major	Short Text
Batch	Number
Email	Short Text
Password	Short Text
Phone_Num	Short Text
Gender	Short Text
Activation_Date	Date/Time
Valid_Till	Date/Time
Address	Short Text

2) Admin Role

a. Main Form



```
2 references
1  Public Class MainForm
2      0 references
3          Private Sub AddBookToolStripMenuItem_Click(sender As Object, e As EventArgs) Handles AddBookToolStripMenuItem.Click
4              AddBook.MdiParent = Me
5              AddBook.Show()
6          End Sub
7          0 references
8              Private Sub ViewBooksToolStripMenuItem_Click(sender As Object, e As EventArgs) Handles ViewBooksToolStripMenuItem.Click
9                  ListBook.MdiParent = Me
10                 ListBook.Show()
11             End Sub
12             0 references
13                 Private Sub AddUserToolStripMenuItem_Click(sender As Object, e As EventArgs) Handles AddUserToolStripMenuItem.Click
14                     AddUser.MdiParent = Me
15                     AddUser.Show()
16                 End Sub
17                 0 references
18                     Private Sub ViewUserToolStripMenuItem_Click(sender As Object, e As EventArgs) Handles ViewUserToolStripMenuItem.Click
19                         UserDetail.MdiParent = Me
20                         UserDetail.Show()
21                     End Sub
22                     0 references
23                         Private Sub CloseAllToolStripMenuItem_Click(sender As Object, e As EventArgs) Handles CloseAllToolStripMenuItem.Click
24                             Dim child As Form
25                             For each child In MdiChildren
26                                 child.Close()
27                             Next
28                         End Sub
29                         0 references
30                             Private Sub ExitToolStripMenuItem1_Click(sender As Object, e As EventArgs) Handles ExitToolStripMenuItem1.Click
31                                 Me.Hide()
32                                 LoginForm.Show()
33                             End Sub
34                             0 references
35                                 Private Sub UploadAnImageToolStripMenuItem_Click(sender As Object, e As EventArgs) Handles UploadAnImageToolStripMenuItem.Click
36                                     With mainOpenFileDialog
37                                         .FileName = ""
38                                         .Filter = ".jpg|*.jpg|.png|*.png|.gif|*.gif|.bmp|*.bmp"
39                                     End With
40                                     If mainOpenFileDialog.ShowDialog = Windows.Forms.DialogResult.OK Then
41                                         UploadAnImageToolStripMenuItem.Checked = True
42                                         Dim BackFile As String
43                                         BackFile = mainOpenFileDialog.FileName
44                                         Me.BackgroundImage = Image.FromFile(BackFile)
45                                         My.Settings.MainImage = BackFile
46                                         My.Settings.Save()
47                                     End If
48                                 End Sub
49                                 0 references
50                                     Private Sub aboutToolStripSplitButton_ButtonClick(sender As Object, e As EventArgs) Handles aboutToolStripSplitButton.ButtonClick
51                                         MsgBox("This project is brought to you by" & vbCrLf & "Jonathan Benedict Sirait" & vbCrLf & "IT 3 2019", MessageBoxIcon.Information)
52                                 End Sub
53                             End Class
```

b. Add a Book Form

Add A Book

Your progress: | Fill the field below

Add a book here:

ISBN	978_____	Publisher	<input type="text"/>
Book Name	<input type="text"/>	Publishing Year	<input type="text"/>
Genre	<input type="text"/>	Price	<input type="text"/>
Author	<input type="text"/>		

Description

Control

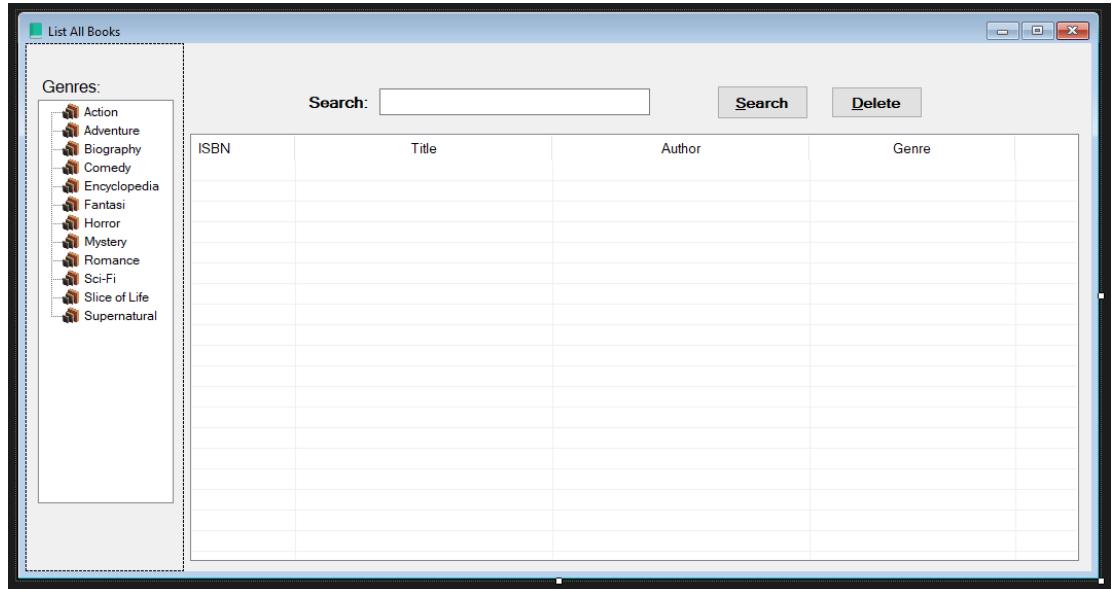
A screenshot of a Windows-style application window titled "Add A Book". The window has a blue header bar with the title and standard window controls (minimize, maximize, close). Below the header is a progress bar labeled "Your progress:" followed by a progress indicator and the text "Fill the field below". The main area is titled "Add a book here:" and contains a grid of input fields. The first row has "ISBN" with the value "978_____", "Publisher" with an empty input field, and two empty input fields for "Book Name" and "Publishing Year". The second row has "Genre" with an empty input field, "Price" with an empty input field, and two empty input fields for "Book Name" and "Publishing Year". The third row has "Author" with an empty input field, and two empty input fields for "Genre" and "Price". Below this is a large empty text area for "Description". At the bottom is a "Control" section with three buttons: "Add Book", "Clear", and "Cancel".

```

1 2 references
1 16 Public Class AddBook
2 0 references
3 17     Private Sub btnAdd_Click(sender As Object, e As EventArgs) Handles btnAdd.Click
4 18         If ValidateInput() Then
5 19             Dim status As String = "Available"
6 20             progBarBook.Value = progBarBook.Value + 4
7 21             LibraryFunction.AddBook(mskTxtISBN.Text, txtBookName.Text, txtAuthor.Text, txtGenre.Text, txtPublisher.Text,
8 22                             txtPublishYear.Text, txtPrice.Text, txtDesc.Text, status)
9 23         End If
10 24     End Sub
11 0 references
11 25     Private Sub btnClear_Click(sender As Object, e As EventArgs) Handles btnClear.Click
12 26         Call ClearField()
13 27     End Sub
14 0 references
14 28     Private Sub btnCancel_Click(sender As Object, e As EventArgs) Handles btnCancel.Click
15 29         Me.Hide()
16 30     End Sub
17 31
18 32
19 33
20 34     Private Function ValidateInput() As Boolean
21 35         If mskTxtISBN.Text = "" Then
22 36             MsgBox("Please input the ISBN!", MessageBoxIcon.Warning)
23 37         ElseIf txtBookName.Text = "" Then
24 38             MsgBox("Please input the Book Name!", MessageBoxIcon.Warning)
25 39         ElseIf txtAuthor.Text = "" Then
26 40             MsgBox("Please input the Author!", MessageBoxIcon.Warning)
27 41         ElseIf txtGenre.Text = "" Then
28 42             MsgBox("Please input the Genre!", MessageBoxIcon.Warning)
29 43         ElseIf txtPublisher.Text = "" Then
30 44             MsgBox("Please input the Publisher!", MessageBoxIcon.Warning)
31 45         ElseIf txtPublishYear.Text = "" Then
32 46             MsgBox("Please input the Publish Year!", MessageBoxIcon.Warning)
33 47         ElseIf txtPrice.Text = "" Then
34 48             MsgBox("Please input the Price!", MessageBoxIcon.Warning)
35 49         Else
36 50             Return True
37 51         End If
38 52         Return False
39 53     End Function
40 54     Private Sub ClearField()
41 55         txtDesc.Clear()
42 56         txtGenre.Clear()
43 57         txtPrice.Clear()
44 58         txtBookName.Clear()
45 59         txtPublisher.Clear()
46 60         mskTxtISBN.Clear()
47 61         txtAuthor.Clear()
48 62         txtPublishYear.Clear()
49 63         progBarBook.Value = 0
50 64     End Sub
51 65
52 66     Private Sub txtBookName_Leave(sender As Object, e As EventArgs) Handles txtBookName.Leave
53 67         progBarBook.Value = progBarBook.Value + 12
54 68     End Sub
55 69
56 70     Private Sub txtGenre_Leave(sender As Object, e As EventArgs) Handles txtGenre.Leave
57 71         progBarBook.Value = progBarBook.Value + 12
58 72     End Sub
59 73
60 74     Private Sub txtAuthor_Leave(sender As Object, e As EventArgs) Handles txtAuthor.Leave
61 75         progBarBook.Value = progBarBook.Value + 12
62 76     End Sub
63 77
64 78     Private Sub txtPublisher_Leave(sender As Object, e As EventArgs) Handles txtPublisher.Leave
65 79         progBarBook.Value = progBarBook.Value + 12
66 80     End Sub
67 81
68 82     Private Sub txtPrice_Leave(sender As Object, e As EventArgs) Handles txtPrice.Leave
69 83         progBarBook.Value = progBarBook.Value + 12
70 84     End Sub
71 85
72 86     Private Sub txtPublishYear_Leave(sender As Object, e As EventArgs) Handles txtPublishYear.Leave
73 87         progBarBook.Value = progBarBook.Value + 12
74 88     End Sub
75 89
76 90     Private Sub txtDesc_Leave_1(sender As Object, e As EventArgs) Handles txtDesc.Leave
77 91         progBarBook.Value = progBarBook.Value + 12
78 92     End Sub
79 93
80 94     Private Sub mskTxtISBN_Leave(sender As Object, e As EventArgs) Handles mskTxtISBN.Leave
81 95         progBarBook.Value = progBarBook.Value + 12
82 96     End Sub
83 97
84 98 End Class

```

c. List All Book Form



```

1 Imports System.Data.OleDb
2 2 references
3 3 Public Class ListBook
4 4     0 references
5 5     Private Sub ListBook_Load(sender As Object, e As EventArgs) Handles MyBase.Load
6 6         Call Retrieve()
7 7     End Sub
8 8     0 references
9 9     Private Sub btnSearch_Click(sender As Object, e As EventArgs) Handles btnSearch.Click
10 10        If txtSearch.Text = "" Then
11 11            listViewBook.Items.Clear()
12 12            Call Retrieve()
13 13        Else
14 14            Dim dt As New DataTable
15 15            listViewBook.Items.Clear()
16 16            dt = SearchBook(txtSearch.Text)
17 17            For Each row In dt.Rows
18 18                Call Populate(row(0), row(1), row(2), row(3), row(4), row(5), row(8))
19 19            Next
20 20            dt.Rows.Clear()
21 21            conn.Close()
22 22            dr.Close()
23 23        End If
24 24    End Sub
25 25    Dim id As String
26 26    0 references
27 27    Private Sub listViewBook_SelectedIndexChanged(sender As Object, e As EventArgs) Handles listViewBook.SelectedIndexChanged
28 28        For i As Integer = 0 To listViewBook.Items.Count - 1
29 29            If listViewBook.Items(i).Selected = True Then
30 30                id = listViewBook.Items(i).SubItems(0).Text
31 31            End If
32 32        Next
33 33    End Sub
34 34

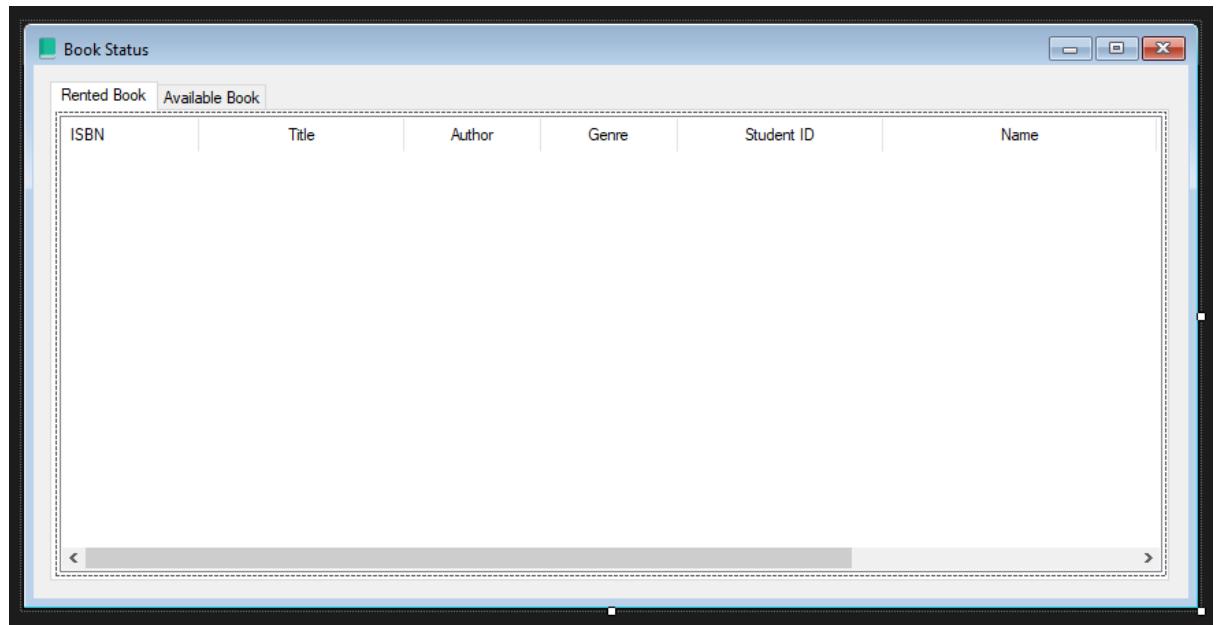
```

```

32      Private Sub btnDelete_Click(sender As Object, e As EventArgs) Handles btnDelete.Click
33          If MsgBox("Are you sure want to delete it?", MessageBoxButtons.YesNo, MessageBoxIcon.Question) = Windows.Forms.DialogResult.Yes Then
34              LibraryFunction.DeleteBook(id)
35              listViewBook.Items.Clear()
36              Call Retrieve()
37          End If
38      End Sub
39      Private Sub Retrieve()
40          Dim dt As New DataTable
41          dt = LibraryFunction.ShowBook(3)
42          For Each row In dt.Rows
43              Call Populate(row(0), row(1), row(2), row(3), row(4), row(5), row(8))
44          Next
45          dt.Rows.Clear()
46      End Sub
47      Private Sub Populate(isbn As String, name As String, author As String, genre As String, publisher As String,
48                           publishYear As Integer, stat As String)
49          'Row Array
50          Dim row As String() = New String() {isbn, name, author, genre, publisher, publishYear, stat}
51          Dim item As ListViewItem = New ListViewItem(row)
52
53          'Add rows to collection
54          listViewBook.Items.Add(item)
55      End Sub
56
57      Private Function SearchBook(ByVal input As String) As DataTable
58          Call Connection()
59          cmd = New OleDbCommand("Select * From Tbl_Book Where Book_Name Like '%" & txtSearch.Text & "%'
59          | OR Book_Author Like '%" & txtSearch.Text & "%' OR Book_Genre Like '%" & txtSearch.Text & "%'
59          | OR Book_Publish_Year Like '%" & txtSearch.Text & "%' Order By Book_Name", conn)
60
61          dr = cmd.ExecuteReader()
62          dr.Read()
63          If dr.HasRows Then
64              da = New OleDbDataAdapter("Select * From Tbl_Book Where Book_Name Like '%" & txtSearch.Text & "%'
64              | OR Book_Author Like '%" & txtSearch.Text & "%' OR Book_Genre Like '%" & txtSearch.Text & "%'
64              | OR Book_Publish_Year Like '%" & txtSearch.Text & "%' Order By Book_Name", conn)
65
66              ds = New DataSet
67              da.Fill(ds)
68              Return ds.Tables(0)
69          Else
70              MsgBox("Data was not found!", MessageBoxIcon.Error)
71              Return LibraryFunction.ShowUser()
72          End If
73      End Function
74
75  End Class

```

d. Book Status Form



```

1 Imports System.Data.OleDb
2 references
3 Public Class StatusBook
4     References
5     Private Sub StatusBook_Load(sender As Object, e As EventArgs) Handles MyBase.Load
6         Call Retrieve("Available")
7         Call Retrieve("Rented")
8     End Sub
9     References
10    Private Sub Retrieve(_status)
11        Dim dt As New DataTable
12        If _status = "Available" Then
13            dt = LibraryFunction.ShowBook(1)
14            For Each row In dt.Rows
15                Call PopulateAvailable(row(0), row(1), row(2), row(3), _status)
16            Next
17        Else
18            dt = LibraryFunction.ShowBook(2)
19            For Each row In dt.Rows
20                Call PopulateRented(row(0), row(1), row(2), row(3), row(4), row(5), row(6), row(7), _status)
21            Next
22        End If
23        dt.Rows.Clear()
24    End Sub
25    References
26    Private Sub PopulateAvailable(isbn As String, name As String, author As String, genre As String, status As String)
27        'Row Array
28        Dim row As String() = New String() {isbn, name, author, genre}
29        Dim item As ListViewItem = New ListViewItem(row)
30
31        'Add rows to collection
32        If status = "Available" Then
33            availableStatus.Items.Add(item)
34        Else
35            rentedStatus.Items.Add(item)
36        End If
37    End Sub
38
39 End Class

```

e. Add a User Form

The screenshot shows a Windows application window titled "Add A User". The window is divided into two main sections: "Your progress:" and "Fill the field below". The "Fill the field below" section is currently active. It contains the following fields:

- Student ID: Text input field
- Gender: Radio buttons for Male (selected) and Female
- Name: Text input field
- Address: Text input field with up/down arrows
- Email: Text input field
- Phone Number: Text input field
- Activation Date: Date picker showing "Friday , December 18, 2020"
- Valid Till: Date picker showing "Friday , December 18, 2020"
- Major: Drop-down menu
- Batch: Drop-down menu showing "2017"
- Checkboxes at the bottom:
 - I accept any terms and condition
 - I responsible if the book that I borrowed broke, then I must pay the fine
- Buttons at the bottom: "Register" and "Clear"

```

1  Public Class AddUser
2      Dim major(15) As String
3      Dim i As Integer
4      References
5      Private Sub AddUser_Load(sender As Object, e As EventArgs) Handles MyBase.Load
6          AddMajor(major)
7          For i = 0 To 15
8              cmbMajor.Items.Add(major(i))
9          Next
10         End Sub
11     References
12     Private Sub btnRegister_Click(sender As Object, e As EventArgs) Handles btnRegister.Click
13         If ValidateInput() Then
14             Dim gender As String
15             If rdGender1.Checked = True Then
16                 gender = "Male"
17                 progBarInput.Value = progBarInput.Value + 10
18             ElseIf rdGender2.Checked = True Then
19                 gender = "Female"
20                 progBarInput.Value = progBarInput.Value + 10
21             End If
22             LibraryFunction.AddUser(txtID.Text, txtName.Text, cmbMajor.SelectedItem, numUpDownBatch.Value, txtEmail.Text, txtPassword.Text,
23             maskedtxtPhoneNumber.Text, gender, dtpActvDate.Value.Date, dtpValidTill.Value.Date, txtAddress.Text)
24         End If
25     End Sub
26     References
27     Private Sub btnClear_Click(sender As Object, e As EventArgs) Handles btnClear.Click
28         Call ClearField()
29     End Sub
30     Reference
31     Private Sub ClearField()
32         txtID.Clear()
33         txtName.Clear()
34         txtEmail.Clear()
35         txtPassword.Clear()
36         cmbMajor.SelectedIndex = -1
37         txtAddress.Clear()
38         maskedtxtPhoneNumber.Clear()
39         dtpActvDate.ResetText()
40         dtpValidTill.ResetText()
41         For i As Integer = 0 To eulaCheckedListBox.Items.Count - 1
42             eulaCheckedListBox.SetItemChecked(i, False)
43         Next
44         progBarInput.Value = 0
45     End Sub
46     Reference
47     Function AddMajor(ByVal major As String()) As String()
48         major(0) = "Accounting"
49         major(1) = "Actuarial Science"
50         major(2) = "Business Administration"
51         major(3) = "Civil Engineering"
52         major(4) = "Communication"
53         major(5) = "Electrical Engineering"
54         major(6) = "Elementary School Teacher Education"
55         major(7) = "Environmental Engineering"
56         major(8) = "Industrial Engineering"
57         major(9) = "Information System"
58         major(10) = "Information Technology"
59         major(11) = "International Relations"
60         major(12) = "Law"
61         major(13) = "Management"
62         major(14) = "Mechanical Engineering"
63         major(15) = "Visual Communication Design"
64
65         Return major
66     End Function

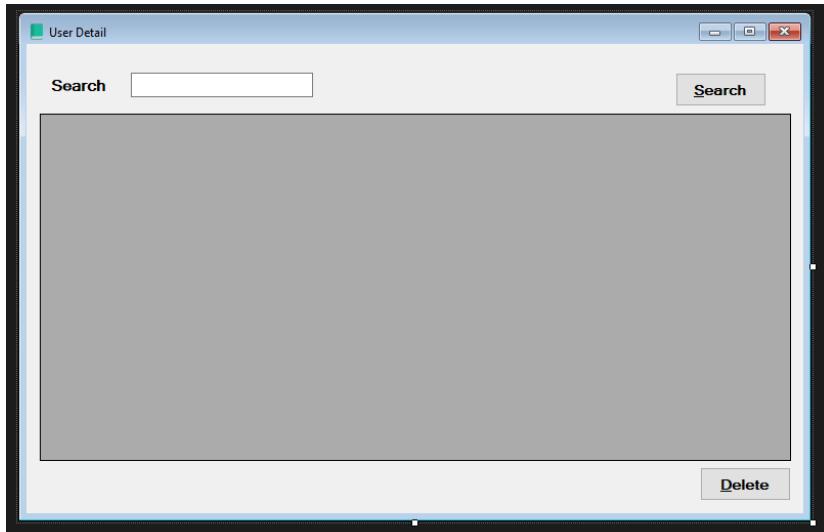
```

```

66  Private Function ValidateInput() As Boolean
67  :   If txtID.Text = "" Then
68  :       MsgBox("Please input the Student ID!", MessageBoxIcon.Warning)
69  :   ElseIf txtName.Text = "" Then
70  :       MsgBox("Please input the Name!", MessageBoxIcon.Warning)
71  :   ElseIf txtEmail.Text = "" Then
72  :       MsgBox("Please input the Email!", MessageBoxIcon.Warning)
73  :   ElseIf txtPassword.Text = "" Then
74  :       MsgBox("Please input the Password!", MessageBoxIcon.Warning)
75  :   ElseIf maskedtxtPhoneNumber.Text.Length < 12 Then
76  :       MsgBox("Please input the right format for Phone Number!", MessageBoxIcon.Warning)
77  :   ElseIf cmbMajor.SelectedIndex = -1 Then
78  :       MsgBox("Please select the Major!", MessageBoxIcon.Warning)
79  :   ElseIf txtAddress.Text = "" Then
80  :       MsgBox("Please input the Address!", MessageBoxIcon.Warning)
81  :   ElseIf dtpActvDate.Value = dtpValidTill.Value Then
82  :       MsgBox("The period of activation should not less than 4 year!", MessageBoxIcon.Warning)
83  :   ElseIf eulaCheckedListBox.CheckedItems.Count = 0 Then
84  :       MsgBox("Please check the EULA!", MessageBoxIcon.Warning)
85  :   Else
86  :       Return True
87  :   End If
88  :   Return False
89  End Function
90
91  'Increase the Progress Bar, when the control is leaved
92  References
93  Private Sub txtID_Leave(sender As Object, e As EventArgs) Handles txtID.Leave
94  :   progBarInput.Value = progBarInput.Value + 10
95  End Sub
96
97  References
98  Private Sub txtName_Leave(sender As Object, e As EventArgs) Handles txtName.Leave
99  :   progBarInput.Value = progBarInput.Value + 10
100 End Sub
101 References
102 Private Sub txtEmail_Leave(sender As Object, e As EventArgs) Handles txtEmail.Leave
103 :   progBarInput.Value = progBarInput.Value + 10
104 End Sub
105 References
106 Private Sub txtPassword_Leave(sender As Object, e As EventArgs) Handles txtPassword.Leave
107 :   progBarInput.Value = progBarInput.Value + 10
108 End Sub
109 References
110 Private Sub cmbMajor_Leave(sender As Object, e As EventArgs) Handles cmbMajor.Leave
111 :   progBarInput.Value = progBarInput.Value + 10
112 End Sub
113 References
114 Private Sub numUpDownBatch_Leave(sender As Object, e As EventArgs) Handles numUpDownBatch.Leave
115 :   progBarInput.Value = progBarInput.Value + 10
116 End Sub
117 References
118 Private Sub maskedtxtPhoneNumber_Leave(sender As Object, e As EventArgs) Handles maskedtxtPhoneNumber.Leave
119 :   progBarInput.Value = progBarInput.Value + 10
120 End Sub
121 References
122 Private Sub dtpActvDate_Leave(sender As Object, e As EventArgs) Handles dtpActvDate.Leave
123 :   progBarInput.Value = progBarInput.Value + 10
124 End Sub
125 End Sub
End Class

```

f. User Detail Form



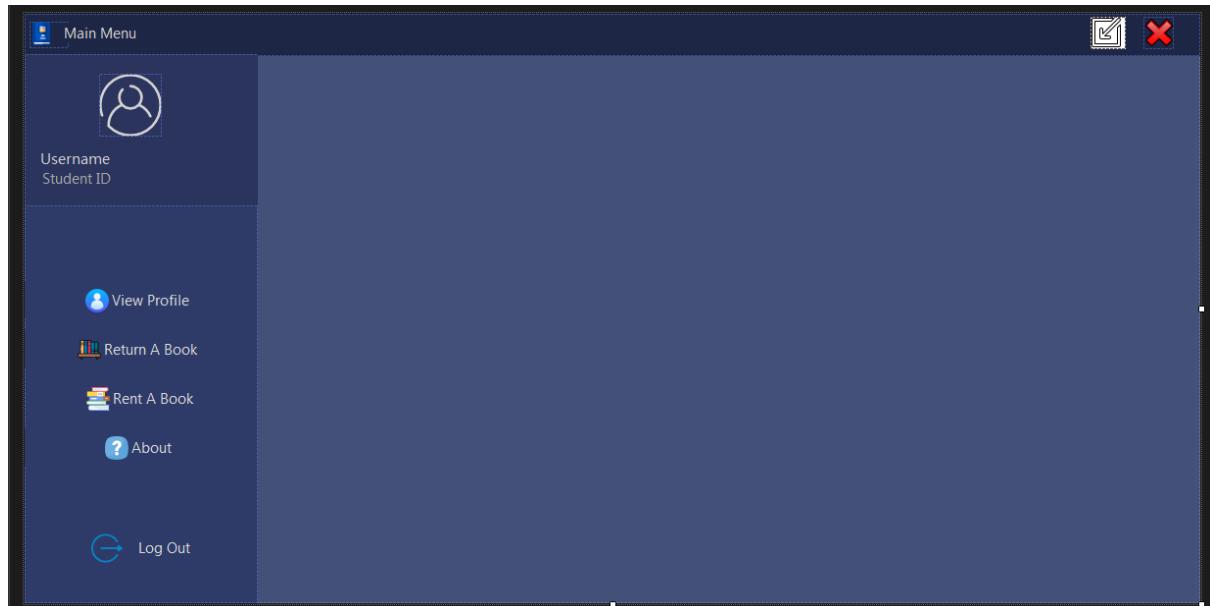
```

1 Imports System.Data.OleDb
2 2 references
3 Public Class UserDetail
4     Dim id As String
5     0 references
6     Private Sub UserDetail_Load(sender As Object, e As EventArgs) Handles MyBase.Load
7         dataGridUser.DataSource = LibraryFunction.ShowUser()
8     End Sub
9
10    0 references
11    Private Sub btnSearch_Click(sender As Object, e As EventArgs) Handles btnSearch.Click
12        If txtSearch.Text = "" Then
13            dataGridUser.DataSource = LibraryFunction.ShowUser()
14        Else
15            dataGridUser.DataSource = SearchUser(txtSearch.Text)
16            conn.Close()
17            dr.Close()
18        End If
19    End Sub
20
21    0 references
22    Private Sub btnDelete_Click(sender As Object, e As EventArgs) Handles btnDelete.Click
23        If MsgBox("Are you sure want to delete it?", MessageBoxButtons.YesNo, MessageBoxIcon.Question) = Windows.Forms.DialogResult.Yes Then
24            LibraryFunction.DeleteUser(id)
25            dataGridUser.DataSource = LibraryFunction.ShowUser()
26        End If
27    End Sub
28
29    0 references
30    Private Function SearchUser(ByVal input As String) As DataTable
31        Call Connection()
32        cmd = New OleDbCommand("Select * From Tbl_User Where Name Like '%" & input & "%' OR
33                                Major Like '%" & input & "%' OR Email Like '%" & input & "%' OR Batch Like '%" & input & "%' OR
34                                Gender Like '%" & input & "%'", conn)
35        dr = cmd.ExecuteReader()
36        dr.Read()
37        If dr.HasRows Then
38            da = New OleDbDataAdapter("Select * From Tbl_User Where Name Like '%" & input & "%' OR
39                                    Major Like '%" & input & "%' OR Email Like '%" & input & "%' OR Batch Like '%" & input & "%' OR
40                                    Gender Like '%" & input & "%'", conn)
41            ds = New DataSet
42            da.Fill(ds)
43            Return ds.Tables(0)
44        Else
45            MsgBox("Data was not found!", MessageBoxIcon.Error)
46            Return LibraryFunction.ShowUser()
47        End If
48    End Function
49
50 End Class

```

3) User Role

a. User Form



```

1  Imports System.Runtime.InteropServices
2  References
3  Public Class userForm
4  #Region "FUNCTIONALITIES OF THE FORM"
5  'RESIZE OF THE FORM- CHANGE SIZE
6  Dim cGrip As Integer = 10
7  Protected Overrides Sub WndProc(ByRef m As Message)
8  If (m.Msg = 132) Then
9      Dim pos As Point = New Point((m.LParam.ToInt32 And 65535), (m.LParam.ToInt32 + 16))
10     pos = Me.PointToClient(pos)
11     If ((pos.X _
12         >= (Me.ClientSize.Width - cGrip)) _ 
13         AndAlso (pos.Y _
14             >= (Me.ClientSize.Height - cGrip))) Then
15         m.Result = CType(17, IntPtr)
16         Return
17     End If
18     MyBase.WndProc(m)
19 End Sub
20 '-----DRAW RECTANGLE / EXCLUDE CORNER PANEL
21 Dim sizeGripRectangle As Rectangle
22 Dim tolerance As Integer = 15
23
24 References
25 Protected Overrides Sub OnSizeChanged(ByVal e As EventArgs)
26     MyBase.OnSizeChanged(e)
27     Dim region = New Region(New Rectangle(0, 0, Me.ClientRectangle.Width, Me.ClientRectangle.Height))
28     sizeGripRectangle = New Rectangle(Me.ClientRectangle.Width - tolerance, (Me.ClientRectangle.Height - tolerance), tolerance, tolerance)
29     region.Exclude(sizeGripRectangle)
30     Me.PanelContainer.Region = region
31     Me.Invalidate()
32 End Sub
33
34 '-----COLOR AND GRIP OF LOWER RECTANGLE/CORNER
35 References
36 Protected Overrides Sub OnPaint(ByVal e As PaintEventArgs)
37     Dim blueBrush As SolidBrush = New SolidBrush(Color.FromArgb(67, 81, 120))
38     e.Graphics.FillRectangle(blueBrush, sizeGripRectangle)
39     MyBase.OnPaint(e)
40     ControlPaint.DrawSizeGrip(e.Graphics, Color.Transparent, sizeGripRectangle)
41 End Sub
42 'DRAG / MOVE THE FORM
43 <DllImport("user32.DLL", EntryPoint:="ReleaseCapture")>
44 Private Shared Sub ReleaseCapture()
45 End Sub
46 <DllImport("user32.DLL", EntryPoint:="SendMessage")>
47 Private Shared Sub SendMessage(ByVal hWnd As System.IntPtr, ByVal wMsg As Integer, ByVal wParam As Integer, ByVal lParam As Integer)
48 End Sub
49 References
50 Private Sub titleBar_MouseMove(sender As Object, e As MouseEventArgs) Handles titleBar.MouseMove
51     ReleaseCapture()
52     SendMessage(Me.Handle, &H112&, &HF012&, 0)
53 End Sub
54 #End Region
55
56 #Region "OPEN FORM METHODS"
57 References
58 Private Sub openFormOnPanel(Of Myform As {Form, New})()
59     Dim Formula As Form
60     Formula = Panel3.Controls.OfType(Of Myform)().FirstOrDefault()
61     If Formula Is Nothing Then
62         Formula = New Myform()
63         Formula.TopLevel = False
64         Formula.FormBorderStyle = FormBorderStyle.None
65         Formula.Dock = DockStyle.None
66         Panel13.Controls.Add(Formula)
67         Panel13.Tag = Formula
68         AddHandler Formula.FormClosed, AddressOf Me.closedForm
69         Formula.BringToFront()
70         Formula.Show()
71     Else
72         Formula.BringToFront()
73     End If
74 End Sub
75 References
76 Private Sub closedForm(ByVal sender As Object, ByVal e As FormClosedEventArgs)
77     If (Application.OpenForms("UserProfile") Is Nothing) Then
78         btnViewProfile.BackColor = Color.FromArgb(46, 59, 104)
79     End If
80     If (Application.OpenForms("ReturnBook") Is Nothing) Then
81         btnReturn.BackColor = Color.FromArgb(46, 59, 104)
82     End If
83     If (Application.OpenForms("RentBook") Is Nothing) Then
84         btnRent.BackColor = Color.FromArgb(46, 59, 104)
85     End If
86     If (Application.OpenForms("AboutForm") Is Nothing) Then
87         btnAbout.BackColor = Color.FromArgb(46, 59, 104)
88     End If
89 End Sub
90

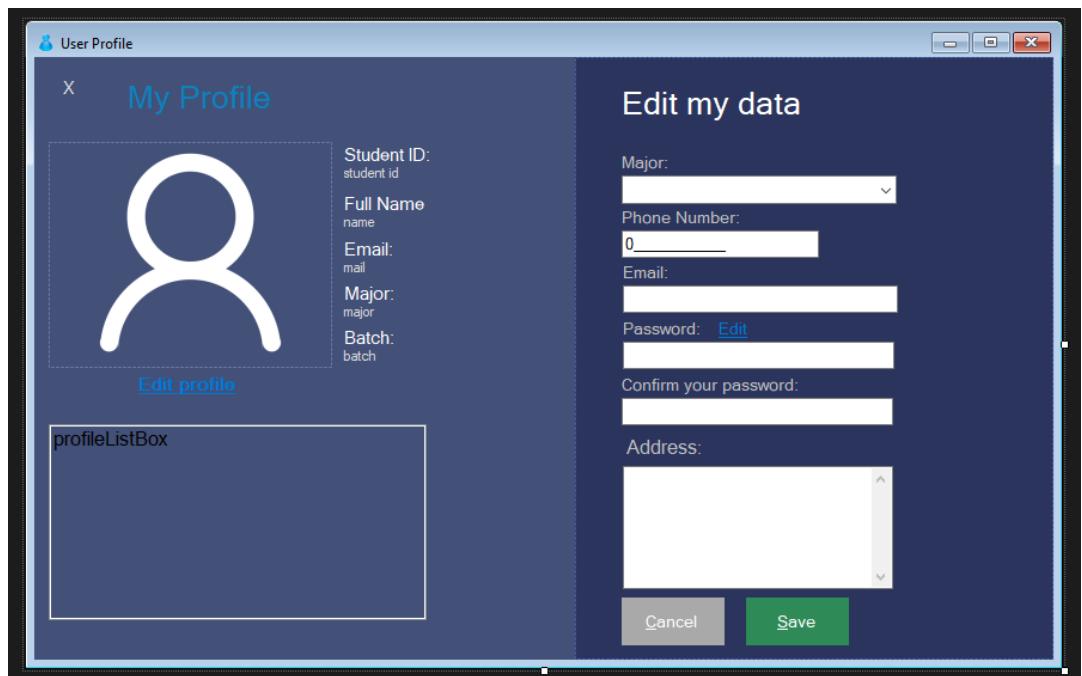
```

```

96      #Region "Close Form, Minimize Form, Logout Form"
97      Private Sub exitPictureBox_Click(sender As Object, e As EventArgs) Handles exitPictureBox.Click
98          If MessageBox.Show("Are you sure want to close the application?", "Warning",
99                         MessageBoxButtons.YesNo, MessageBoxIcon.Warning) = DialogResult.Yes Then
100             Application.Exit()
101         End If
102     End Sub
103
104     Private Sub minimizePictureBox_Click(sender As Object, e As EventArgs) Handles minimizePictureBox.Click
105         Me.WindowState = FormWindowState.Minimized
106     End Sub
107
108
109
110     Private Sub btnLogout_Click_1(sender As Object, e As EventArgs) Handles btnLogout.Click
111         If MessageBox.Show("Are you sure want to log out?", "Warning",
112                         MessageBoxButtons.YesNo, MessageBoxIcon.Warning) = DialogResult.Yes Then
113             Me.Hide()
114             LoginForm.Show()
115         End If
116     End Sub
117
118 #End Region
119
120
121
122
123
124
125
126
127
128
129
130
131
132
133

```

b. User Profile Form



```

1  Imports System.Data.OleDb
2  Public Class UserProfile
3      Dim major(15) As String
4      Dim currentPass As String
5
6      Private Sub UserProfile_Load(sender As Object, e As EventArgs) Handles MyBase.Load
7          AddMajor(major)
8          For i As Integer = 0 To 15
9              cmbMajorEdit.Items.Add(major(i))
10         Next
11         Call LoadUserData()
12         Call InitializeControlsPass()
13     End Sub
14
15     Private Sub btnSave_Click(sender As Object, e As EventArgs) Handles btnSave.Click
16         If ValidateInput() = True Then
17             LibraryFunction.EditUser(userForm.lblIDStud.Text, txtEmail.Text, txtPassword.Text,
18                                     cmbMajorEdit.Text, maskedtxtPhoneNumber.Text, txtAddress.Text)
19             ShowUpdate()
20             Call Reset()
21         End If
22     End Sub
23
24     Private Sub btnCancel_Click(sender As Object, e As EventArgs) Handles btnCancel.Click
25         Call Reset()
26     End Sub
27
28     References
29     Private Sub linkEditPass_LinkClicked(sender As Object, e As LinkLabelLinkClickedEventArgs) Handles linkEditPass.LinkClicked
30         If linkEditPass.Text = "Edit" Then
31             linkEditPass.Text = "Cancel"
32             lblConfirmPass.Visible = True
33             txtPassword.Enabled = True
34             txtPassword.Text = ""
35             txtConfirmPass.Text = ""
36             txtConfirmPass.Visible = True
37         ElseIf linkEditPass.Text = "Cancel" Then
38             'Restart only in edit password
39             linkEditPass.Text = "Edit"
40             lblConfirmPass.Visible = False
41             txtPassword.Text = currentPass
42             txtConfirmPass.Text = currentPass
43             txtPassword.Enabled = False
44             txtConfirmPass.Visible = False
45         End If
46     End Sub
47
48     References
49     Private Sub linkEdit_LinkClicked(sender As Object, e As LinkLabelLinkClickedEventArgs) Handles linkEdit.LinkClicked
50         Call LoadUserData()
51         Panel1.Width = 10
52         Panel1.Visible = True
53         Timer1.Start()
54     End Sub
55
56     References
57     Private Sub btnClose_Click(sender As Object, e As EventArgs) Handles btnClose.Click
58         Me.Close()
59     End Sub
60
61     References
62     Private Sub Reset()
63         Call InitializeControlsPass()
64         Call LoadUserData()
65     End Sub
66
67     References
68     Private Sub LoadUserData()
69         Dim dt As New DataTable
70         Call Connection()
71         cmd = New OleDbCommand("SELECT * FROM Tbl_User WHERE [Student_ID] = '" & userForm.lblIDStud.Text & "'", conn)
72         dr = cmd.ExecuteReader()
73         dt.Load(dr)
74         conn.Close()
75         dr.Close()
76         For Each row In dt.Rows
77             lblID.Text = row(0)
78             lblName.Text = row(1)
79             lblMajor.Text = row(2)
80             lblBatch.Text = row(3)
81             lblEmail.Text = row(4)
82             cmbMajorEdit.Text = row(2)
83             txtEmail.Text = row(4)
84             txtPassword.Text = row(5)
85             maskedtxtPhoneNumber.Text = row(6)
86             txtAddress.Text = row(10)
87             currentPass = row(5)
88         Next
89     End Sub
90
91     References
92     Private Sub InitializeControlsPass()
93         Panel1.Visible = False
94         linkEditPass.Text = "Edit"
95         lblConfirmPass.Visible = False
96         txtPassword.UseSystemPasswordChar = True
97         txtPassword.Enabled = False
98         txtConfirmPass.UseSystemPasswordChar = True
99         txtConfirmPass.Visible = False
100    End Sub

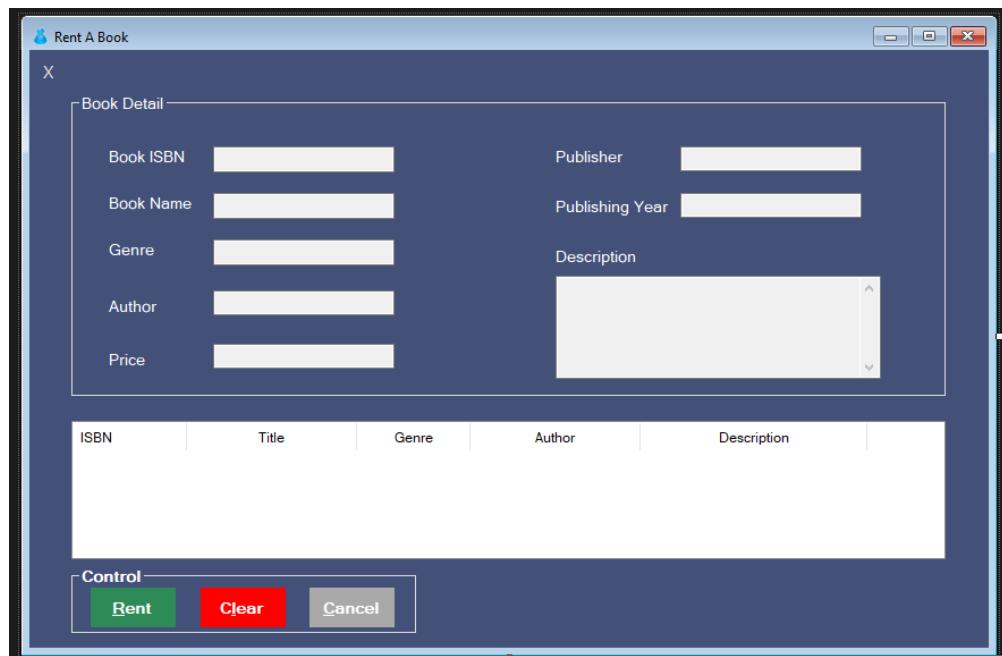
```

```

92     Function AddMajor(ByVal major As String()) As String()
93         major(0) = "Accounting"
94         major(1) = "Actuarial Science"
95         major(2) = "Business Administration"
96         major(3) = "Civil Engineering"
97         major(4) = "Communication"
98         major(5) = "Electrical Engineering"
99         major(6) = "Elementary School Teacher Education"
100        major(7) = "Environmental Engineering"
101        major(8) = "Industrial Engineering"
102        major(9) = "Information System"
103        major(10) = "Information Technology"
104        major(11) = "International Relations"
105        major(12) = "Law"
106        major(13) = "Management"
107        major(14) = "Mechanical Engineering"
108        major(15) = "Visual Communication Design"
109
110        Return major
111    End Function
112
113    Private Function ValidateInput() As Boolean
114        If cmbMajorEdit.SelectedIndex = -1 Then
115            MsgBox("Please input the Major!", MessageBoxIcon.Warning)
116        ElseIf maskedtxtPhoneNumber.Text.Length < 12 Then
117            MsgBox("Please input the right format for Phone Number!", MessageBoxIcon.Warning)
118        ElseIf txtEmail.Text = "" Then
119            MsgBox("Please input the Email!", MessageBoxIcon.Warning)
120        ElseIf txtPassword.Text = "" Then
121            MsgBox("Please input the Password!", MessageBoxIcon.Warning)
122        ElseIf txtConfirmPass.Text = "" Or Not txtPassword.Text.Length = txtConfirmPass.Text.Length Then
123            MsgBox("Please confirm the Password!", MessageBoxIcon.Warning)
124        ElseIf txtAddress.Text = "" Then
125            MsgBox("Please input the Address!", MessageBoxIcon.Warning)
126        Else
127            Return True
128        End If
129    Return False
130 End Function
131
132    References
133    Private Sub ClearField()
134        txtEmail.Clear()
135        txtPassword.Clear()
136        txtAddress.Clear()
137        maskedtxtPhoneNumber.Clear()
138        cmbMajorEdit.SelectedIndex = -1
139    End Sub
140
141    Reference
142    Private Sub ShowUpdate()
143        profileListBox.Items.Add("Data has been updated!")
144        profileListBox.Items.Add("List of updated data:")
145        profileListBox.Items.Add("Email: " & txtEmail.Text)
146        profileListBox.Items.Add("Password: " & txtPassword.Text)
147        profileListBox.Items.Add("Major: " & cmbMajorEdit.Text)
148        profileListBox.Items.Add("Phone Number: " & maskedtxtPhoneNumber.Text)
149        profileListBox.Items.Add("Address: " & vbCrLf & txtAddress.Text)
150    End Sub
151 End Class

```

c. Rent a Book Form

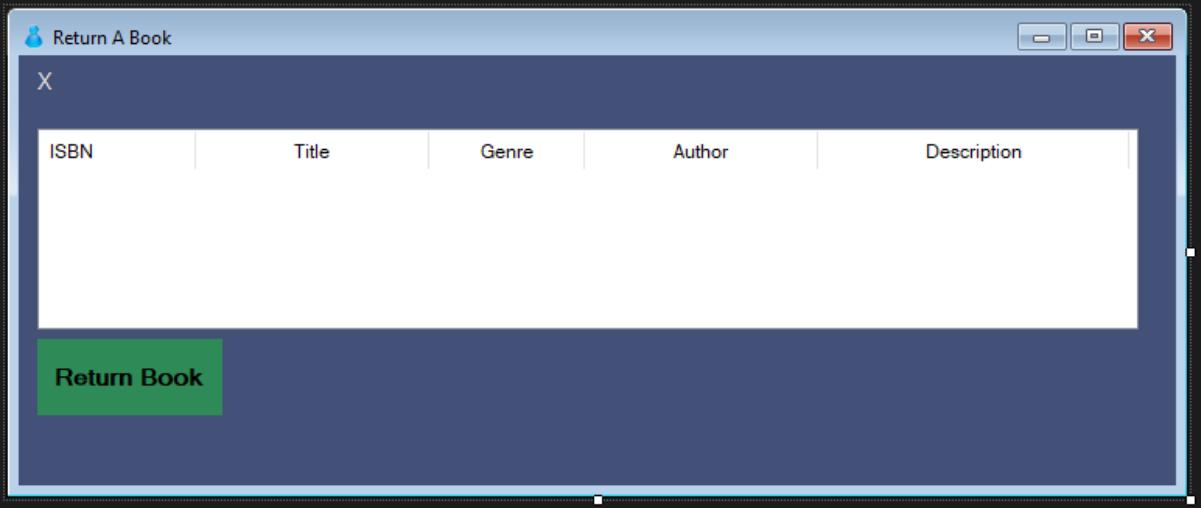


```

1 Imports System.Data.OleDb
2 3 references
3 Public Class RentBook
4     Dim id As String
5     0 references
6     Private Sub RentBook_Load(sender As Object, e As EventArgs) Handles MyBase.Load
7         Me.Retrieve()
8     End Sub
9
10    2 references
11    Private Sub Retrieve()
12        Dim dt As New DataTable
13        dt = LibraryFunction.ShowBook(1)
14        For Each row In dt.Rows
15            Call Populate(row(0), row(1), row(3), row(2), row(7))
16        Next
17        dt.Rows.Clear()
18    End Sub
19
20    1 reference
21    Private Sub Populate(isbn As String, name As String, genre As String, author As String, desc As String)
22        'Row Array
23        Dim row As String() = New String() {isbn, name, genre, author, desc}
24        Dim item As ListViewItem = New ListViewItem(row)
25
26        'Add rows to collection
27        availableBookListView.Items.Add(item)
28    End Sub
29
30    0 references
31    Private Sub btnRentBook_Click(sender As Object, e As EventArgs) Handles btnRentBook.Click
32        'Dim tanggal As Dateime
33        'tanggal = System.DateTime.Now.ToString("mm/dd/yyyy")
34        'tanggal = tanggal.AddDays(7)
35        'MsgBox(tanggal)
36        LibraryFunction.RentBook(id, userForm.lblIDStud.Text)
37        availableBookListView.Items.Clear()
38        Me.Retrieve()
39    End Sub
40
41    0 references
42    Private Sub btnClear_Click(sender As Object, e As EventArgs) Handles btnClear.Click
43        Call ClearField()
44    End Sub
45
46    0 references
47    Private Sub btnCancel_Click(sender As Object, e As EventArgs) Handles btnCancel.Click
48        Me.Close()
49    End Sub
50
51    0 references
52    Private Sub btnClose_Click(sender As Object, e As EventArgs) Handles btnClose.Click
53        Me.Close()
54    End Sub
55
56    0 references
57    Private Sub availableBookListView_SelectedIndexChanged(sender As Object, e As EventArgs) Handles availableBookListView.SelectedIndexChanged
58        For i As Integer = 0 To availableBookListView.Items.Count - 1
59            If availableBookListView.Items(i).Selected = True Then
60                id = availableBookListView.Items(i).SubItems(0).Text
61            End If
62        Next
63        Dim dt As New DataTable
64        Call Connection()
65        cmd = New OleDbCommand("SELECT * FROM Tbl_Book WHERE [Book_ISBN] = '" & id & "'", conn)
66        dr = cmd.ExecuteReader()
67        dt.Load(dr)
68        conn.Close()
69        dr.Close()
70        For Each row In dt.Rows
71            txtISBN.Text = row(0)
72            txtName.Text = row(1)
73            txtAuthor.Text = row(2)
74            txtGenre.Text = row(3)
75            txtPublisher.Text = row(4)
76            txtPublishYear.Text = row(5)
77            txtPrice.Text = row(6)
78            txtDescription.Text = row(7)
79        Next
80    End Sub
81
82    1 reference
83    Private Sub ClearField()
84        txtISBN.Clear()
85        txtName.Clear()
86        txtAuthor.Clear()
87        txtGenre.Clear()
88        txtPublisher.Clear()
89        txtPublishYear.Clear()
90        txtPrice.Clear()
91        txtDescription.Clear()
92    End Sub
93
94    End Class

```

d. Return a Book Form

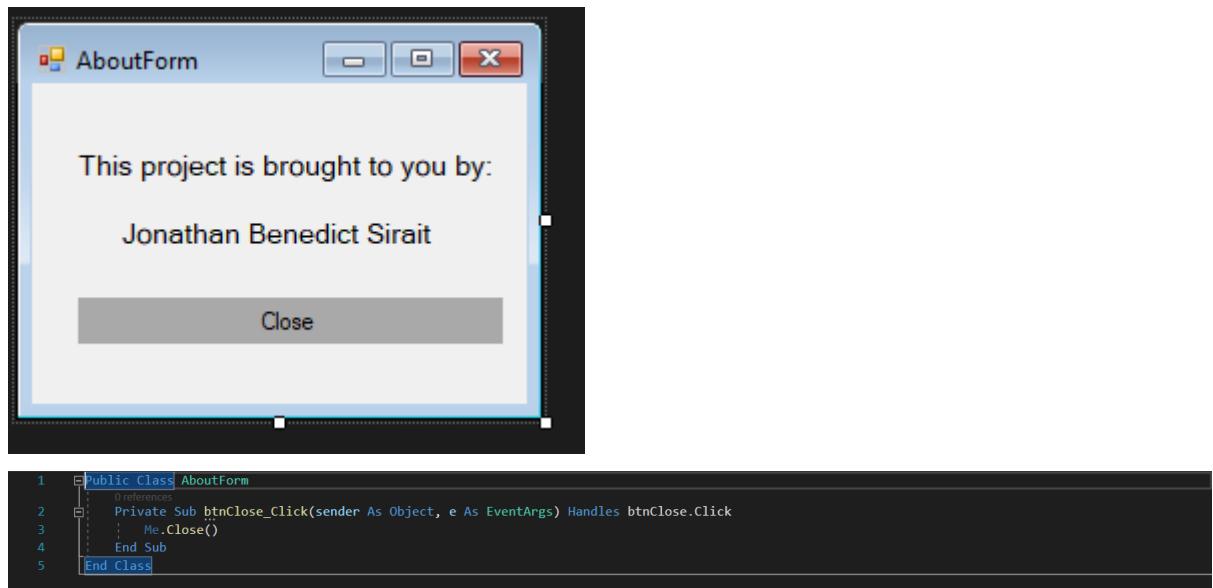


```

1 Imports System.Data.OleDb
2 Public Class ReturnBook
3     Dim id As String
4     Private Sub ReturnBook_Load(sender As Object, e As EventArgs) Handles MyBase.Load
5         Me.Retrieve()
6     End Sub
7     Private Sub Retrieve()
8         Dim dt As New DataTable
9         Call Connection()
10        cmd = New OleDbCommand("SELECT * FROM Tbl_Book WHERE [Student_ID] = '" & userForm.lblIDStud.Text & "'", conn)
11        dr = cmd.ExecuteReader()
12        dt.Load(dr)
13        conn.Close()
14        dr.Close()
15        For Each row In dt.Rows
16            Call PopulateRented(row(0), row(1), row(3), row(2), row(7))
17        Next
18        dt.Rows.Clear()
19    End Sub
20    Private Sub PopulateRented(isbn As String, book_name As String, genre As String, author As String, desc As String)
21        'Row Array
22        Dim row As String() = New String() {isbn, book_name, genre, author, desc}
23        Dim item As ListViewItem = New ListViewItem(row)
24
25        'Add rows to collection
26        rentedBookListView.Items.Add(item)
27    End Sub
28    Private Sub btnClose_Click(sender As Object, e As EventArgs) Handles btnClose.Click
29        Me.Close()
30    End Sub
31
32    Private Sub rentedBookListView_SelectedIndexChanged(sender As Object, e As EventArgs) Handles rentedBookListView.SelectedIndexChanged
33        For i As Integer = 0 To rentedBookListView.Items.Count - 1
34            If rentedBookListView.Items(i).Selected = True Then
35                id = rentedBookListView.Items(i).SubItems(0).Text
36            End If
37        Next
38    End Sub
39
40    Private Sub btnRent_Click(sender As Object, e As EventArgs) Handles btnRent.Click
41        LibraryFunction.ReturnBook(id)
42        rentedBookListView.Items.Clear()
43        Me.Retrieve()
44    End Sub
45 End Class

```

e. About Form

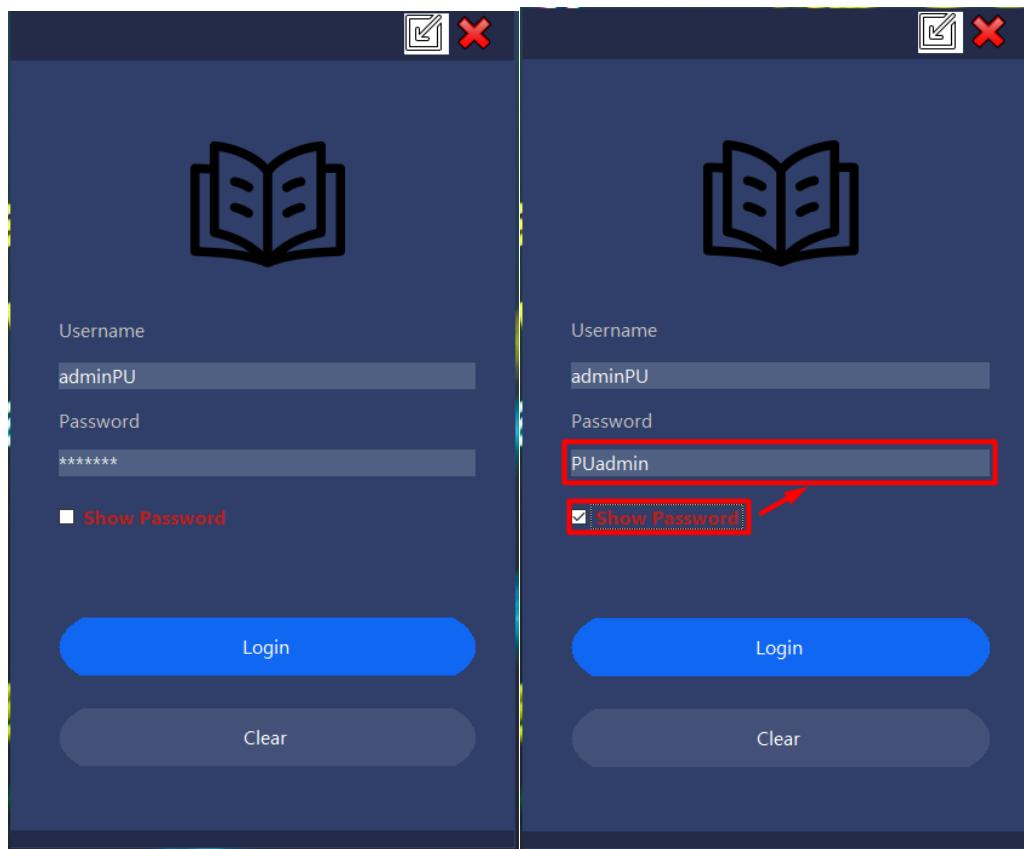


2. Output

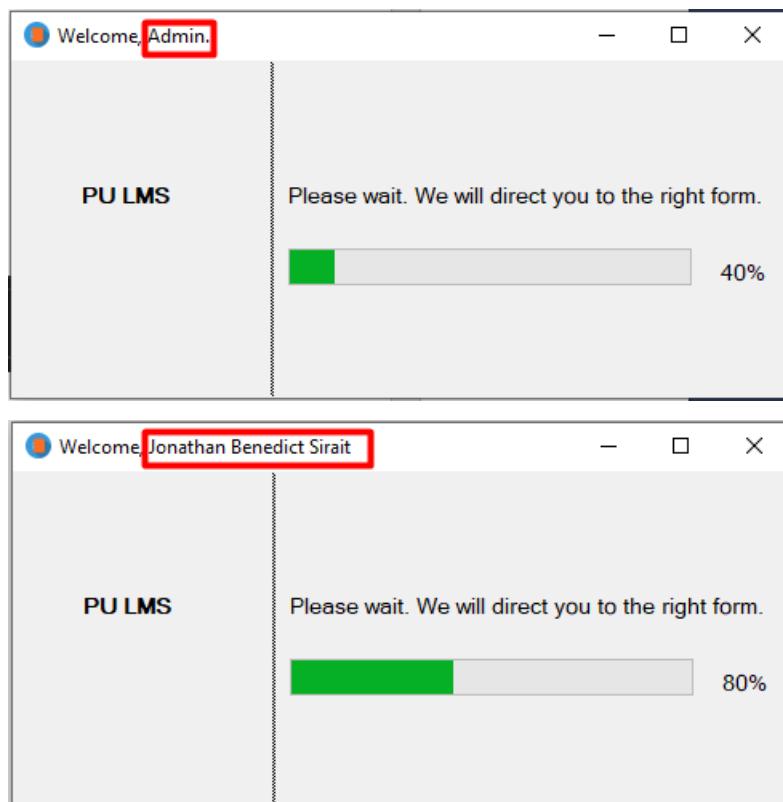
1) Login

So basically, there are **2 roles** in PU LMS, **Admin** and **User** (PU Student). When inputted correct **username and password for Admin**, the flow will **direct** you **to Admin-Related Form**. Otherwise, when inputted correct **username and password for User**, the flow will **direct** you **to User-Related Form** (the loading form will display the text property with the full name of the user).

a. Login Form



b. Loading

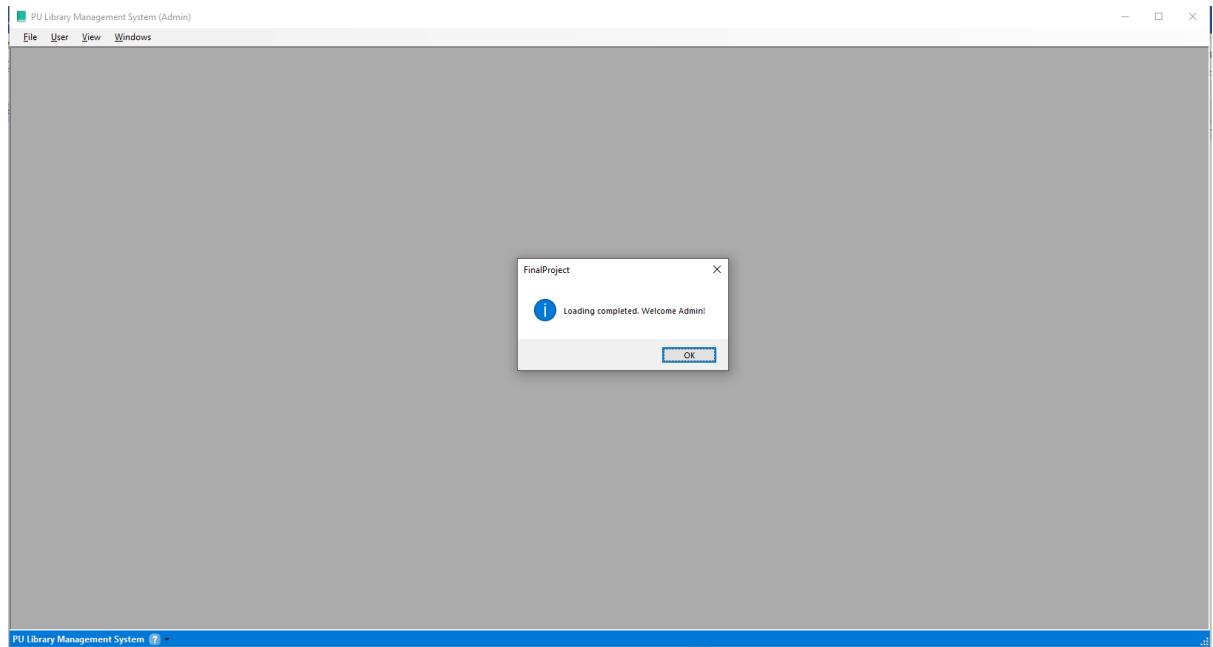


2) Admin Role

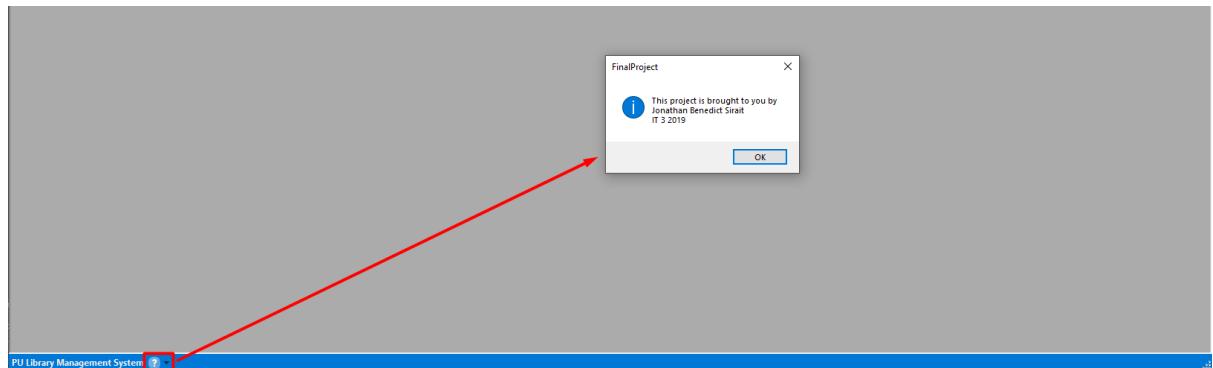
In **Admin-Related Form** it contains all form that refer to **Admin feature** or we can say what Admin can do in this application.

a. Main Form (MainForm)

When **success login** it will **direct to Main Form**.

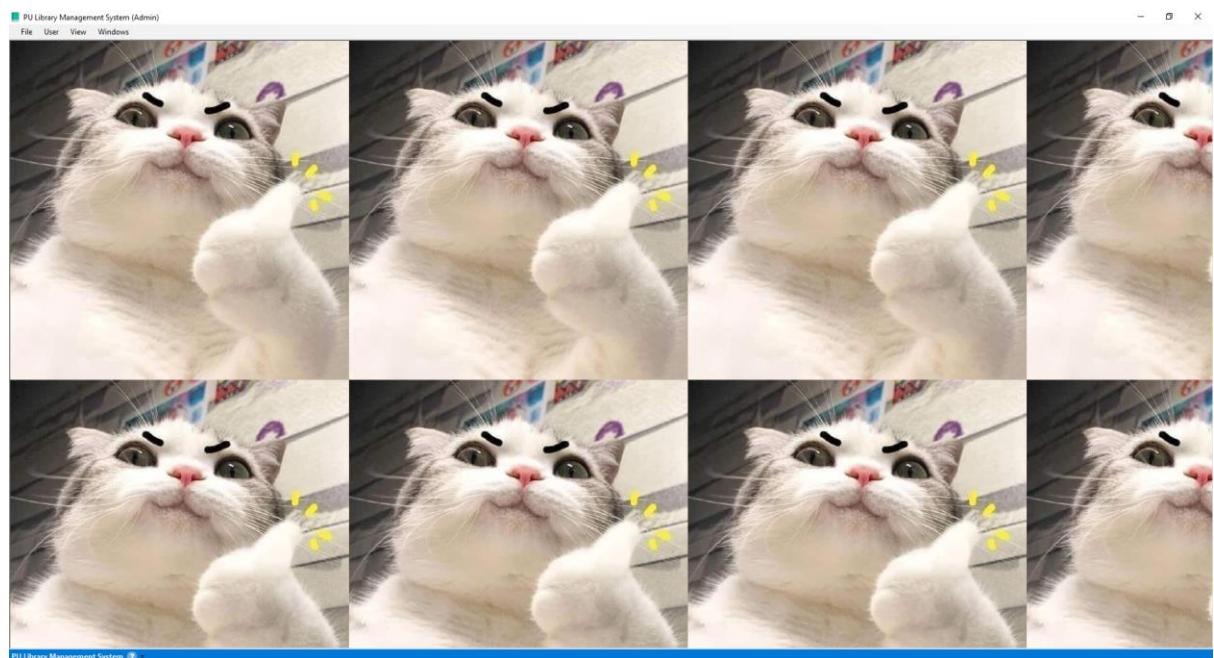
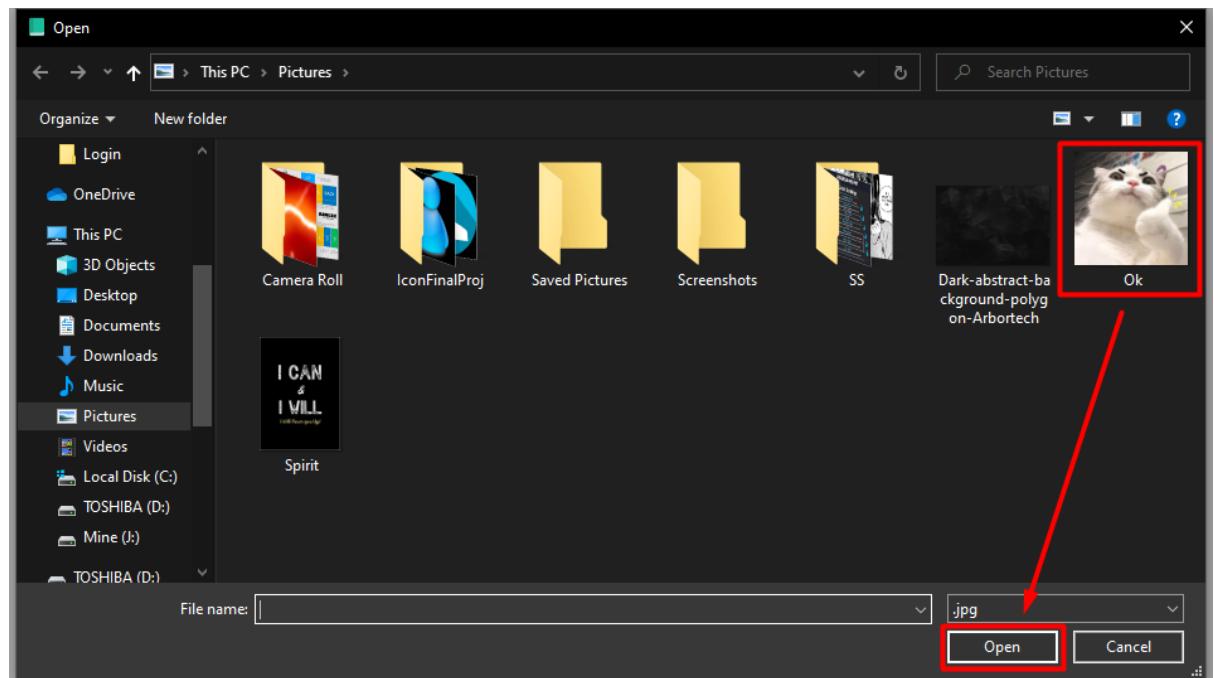


When **click** the **About Button** (Status Strip's Button) below, it will **show a Message Box**.

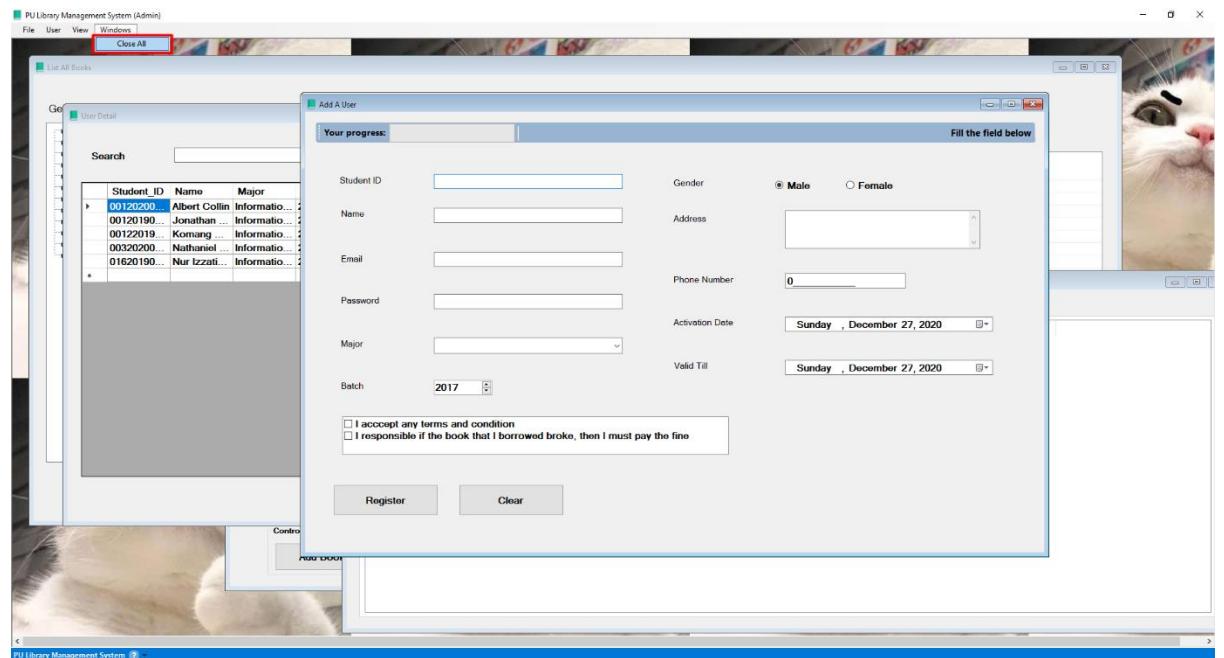


When doing **right-click**, it will give an option, which is **Upload an Image**. It will **pop up a File Explorer** window and let you **choose a picture file** to be the **background image** of the **Main Form**.

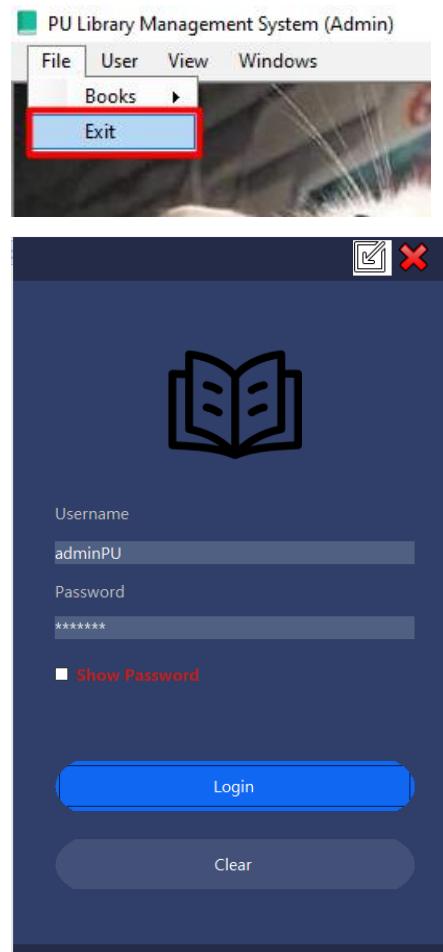




Main Form able to Close All Windows simultaneously, in situation where Admin open a lot of windows in the Main Form.



When click **Exit** (Menu Strip's Option), it will close the Main Form and **open the Login Form**.



b. Add a Book Form (AddBook)

This form will add a book. Every time Admin **fill the field** the **Progress Bar's value** will be **increased**. When **click Add Book**, the **inputted data will be stored to the database** (in this case Access DB called PULibrary).

The image displays two screenshots of a Windows application window titled "Add A Book".

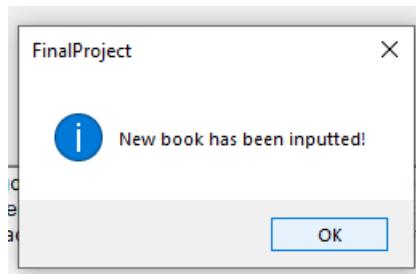
Screenshot 1 (Top): The progress bar at the top is at 0%. The form contains the following data:

Field	Value
ISBN	9785491049506
Book Name	Classroom of Elite
Genre	
Author	
Description	(Empty text area)

Screenshot 2 (Bottom): The progress bar is now at 100%. The form contains the following data:

Field	Value
ISBN	9789375123163
Book Name	Classroom of elite
Genre	Drama, Romance
Author	Kinugasa Shogo
Description	A Japanese light novel series written by Shogo Kinugasa and illustrated by Shunsaku Tomose. A manga adaptation by Yuyu Ichino began its serialization in Media Factory Monthly Comic Alive on January 27, 2016. An anime television series adaptation by Lerche aired from July to September 2017.

In both screenshots, the "Control" section at the bottom includes three buttons: "Add Book", "Clear", and "Cancel". The "Add Book" button is highlighted with a red border in the second screenshot.



When Admin has inputted the book, Admin can **click Clear** button to **clear all field**.

Add A Book

Your progress: | Fill the field below

Add a book here:

ISBN	9789375123163	Publisher	Seven Seas Entertainment
Book Name	Classroom of elite	Publishing Year	2014
Genre	Drama, Romance	Price	76000
Author	Kinugasa Shogo		

Description

A Japanese light novel series written by Shogo Kinugasa and illustrated by Shunsaku Tomose. A manga adaptation by Yuyu Ichino began its serialization in Media Factory Monthly Comic Alive on January 27, 2016. An anime television series adaptation by Lerche aired from July to September 2017.

Control

Add Book **Clear** Cancel

Clear all the field

Add A Book

Your progress: | Fill the field below

Add a book here:

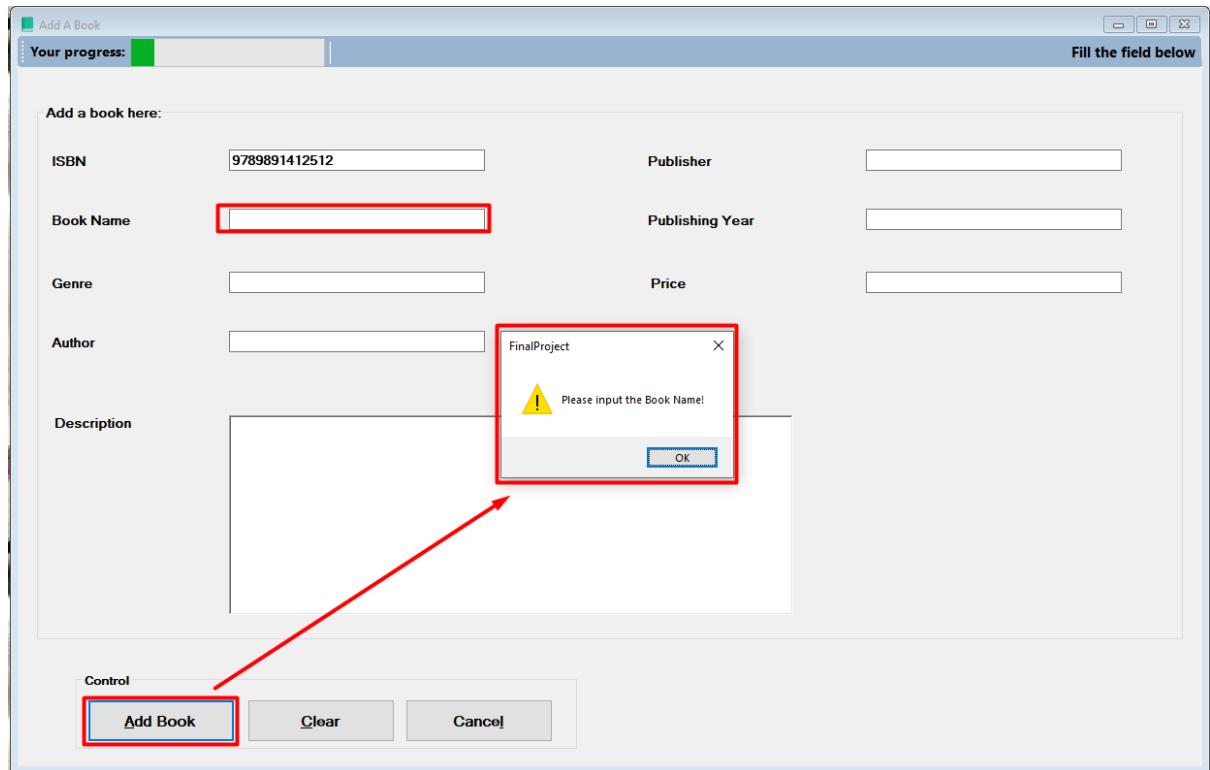
ISBN	978_____	Publisher	
Book Name		Publishing Year	
Genre		Price	
Author			

Description

Control

Add Book **Clear** Cancel

If Admin **doesn't input the required field** sequentially, it will **show a Warning Message Box**.



***Note:**

- The **Progress Bar will add a value when Admin leave the textbox or not use the textbox.** It's because on every **textbox**, it **uses Leave event**. Hence, **if Admin input a wrong value at a certain field, the Progress Bar will still add a value**. Therefore, **the best way is to clear all the field using Clear button and start input the value from the beginning. Otherwise the Progress Bar will still add a value, and when it exceeds the maximum value, the application will stop and show an error.**
- **When fill a value or text at Description's textbox, make sure the sentences or word didn't contain single quote (').** Otherwise, the application will stop and show an error. It's because the **textbox didn't have escaped character so it cannot handle the single quote.**

c. List All Book Form (ListBook)

This form will **list all the books from the database to the ListView**. It also provides search feature. It will search based on **Book Name, Author, Genre, and Publish Year**.

PU Library Management System (Admin)

File User View Windows

Books > Add a book View books Exit

List All Books

Genres:

- Action
- Adventure
- Biography
- Comedy
- Encyclopedia
- Fantasy
- Horror
- Mystery
- Romance
- Sci-Fi
- Slice of Life
- Supernatural

Search:

Search Delete

ISBN	Title	Author	Genre	Publisher	Publish Year	Status
9781974708185	Ao Haru Ride	Io Sakisaka	Slice of Life	Cobalt	2011	Availa...
9789375123163	Classroom of elite	Kinugasa Shogo	Drama, Romance	Seven Seas Entertainment	2014	Availa...
9786020626703	Kartasis	Anastasia Aemilia	Mystery	Gramedia Pustaka Utama	2014	Availa...
9784041026229	Kimi no na wa	Makota Shinka	Sci-Fi, Romance, Comedy	Kadokawa Corporation	2016	Availa...
9784575239058	Kimi no Suizo wo Tabetai	Yoru Sumino	Drama	Futabasha	2015	Availa...
978902349273	Murder On The Orient Express	Agatha Christie	Crime, Mystery	Agatha Christie Inc.	1980	Rented
978404512625	Oreagain Vol. 1	Watan Wataru	Romance	Naruto	2011	Availa...
9784049127966	Oresuki	Rakuda	Romance	Sasuke	2019	Availa...
9780136019722	The ABC Murders	Agatha Christie	Mystery	Denvers Inc	1978	Availa...
9789234236097	The Dark Tower	Jogn Boyega	Adventure, Fantasy	The Fox	2016	Availa...

List All Books

Genres:

- Action
- Adventure
- Biography
- Comedy
- Encyclopedia
- Fantasy
- Horror
- Mystery
- Romance
- Sci-Fi
- Slice of Life
- Supernatural

Search:

Search Delete

ISBN	Title	Author	Genre	Publisher	Publish Year	Status
9780136019722	The ABC Murders	Agatha Christie	Mystery	Denvers Inc	1978	Available

If you want to **display the whole book after done searching**, just **empty the textbox** and **click Search button**. It will show all the book from DB to the ListView.

List All Books

Genres:

- Action
- Adventure
- Biography
- Comedy
- Encyclopedia
- Fantasy
- Horror
- Mystery
- Romance
- Sci-Fi
- Slice of Life
- Supernatural

Search:

Search Delete

ISBN	Title	Author	Genre	Publisher	Publish Year	Status
9781974708185	Ao Haru Ride	Io Sakisaka	Slice of Life	Cobalt	2011	Available
9789375123163	Classroom of elite	Kinugasa Shogo	Drama, Romance	Seven Seas Entertainment	2014	Available
9786020626703	Kartasis	Anastasia Aemilia	Mystery	Gramedia Pustaka Utama	2014	Available
9784041026229	Kimi no na wa	Makota Shinka	Sci-Fi, Romance, Comedy	Kadokawa Corporation	2016	Available
9784575239058	Kimi no Suizo wo Tabetai	Yoru Sumino	Drama	Futabasha	2015	Available
978902349273	Murder On The Orient Express	Agatha Christie	Crime, Mystery	Agatha Christie Inc.	1980	Rented
978404512625	Oreagain Vol. 1	Watan Wataru	Romance	Naruto	2011	Available
9784049127966	Oresuki	Rakuda	Romance	Sasuke	2019	Available
9780136019722	The ABC Murders	Agatha Christie	Mystery	Denvers Inc	1978	Available
9789234236097	The Dark Tower	Jogn Boyega	Adventure, Fantasy	The Fox	2016	Available

Last but not least, this form **provides Delete** feature. Just **select the preferred cell** and **click Delete button**. It will **show a message box**, and **click Yes**. Once you clicked, it will show a success Message Box. And done, the selected book is deleted.

ISBN	Title	Author	Genre	Publisher	Publish Year	Status
9781974708185	Ao Hau Ride	Io Sakisaka	Slice of Life	Cobalt	2011	Available
9789375123163	Classroom of elite	Kinugasa Shogo	Drama, Romance	Seven Seas Entertainment	2014	Available
9786020626703	Kartasis	Anastasia Aemilia	Mystery	Gramedia Pustaka Utama	2014	Available
9784041026229	Kimi no na wa	Makoto Shinka	Sci-Fi, Romance, Comedy	Kadokawa Corporation	2016	Available
9784575239058	Kimi no Suizo wo Tabetai	You Sumino	Drama	Futabasha	2015	Available
9789023489273	Murder On The Orient Express	Agatha Christie	Crime, Mystery	Agatha Christie Inc.	1980	Rented
9784094512625	Oregairu Vol. 1	Watari Wataru	Romance	Naruto	2011	Available
9784049127966	Oresuki	Rakuda	Romance	Sasuke	2019	Available
9780136019722	The ABC Murders	Agatha Christie	Mystery	Denvers Inc	1978	Available
9789234296097	The Dark Tower	Jigen Boyenga	Adventure, Fantasy	The Fox	2016	Available

Are you sure want to delete it?

FinalProject

i Selected book has been deleted!

d. Book Status Form (StatusBook)

This form will show 2 different tabs :

1. Rented Book: Show all books that been rented by User (PU Students)
2. Available Book: Show all available books

The screenshot displays three windows of the PU Library Management System:

- Main Menu Window:** Shows the title "PU Library Management System (Admin)" and a menu bar with File, User, View, Windows, and Status Books. The "Status Books" option is highlighted.
- Rented Book Status Window:** A table showing details for a single rented book. The table has columns: ISBN, Title, Author, Genre, Student ID, Name, Major, and Batch. The data is as follows:

ISBN	Title	Author	Genre	Student ID	Name	Major	Batch
9789023489273	Murder On The Orient Express	Agatha Christie	Crime, Mystery	016201900069	Nur Izzati Ramadhan	Information System	2019

- Available Book Status Window:** A table showing multiple available books. The table has columns: ISBN, Title, Author, and Genre. The data is as follows:

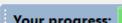
ISBN	Title	Author	Genre
9781974708185	Ao Haru Ride	Io Sakisaka	Slice of Life
9789375123163	Classroom of elite	Kinugasa Shogo	Drama, Romance
9786020626703	Kartasis	Anastasia Aemilia	Mystery
9784041026229	Kimi no na wa	Makota Shinka	Sci-Fi, Romance, Comedy
9784575239058	Kimi no Suizo wo Tabetai	Yoru Sumino	Drama
9784094512625	Oregairu Vol. 1	Watari Wataru	Romance
9784049127966	Oresuki	Rakuda	Romance
9780136019722	The ABC Murders	Agatha Christie	Mystery

e. Add a User Form (AddUser)

This form will add a user. Every time Admin **fill the field** the **Progress Bar's value** will be **increased**. When **click Register**, the **inputted data** will be **stored to the database** (in this case Access DB called PULibrary).

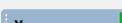
The screenshot shows the User menu from the PU Library Management System. The menu items are File, User, View, Windows, Add User, and View User. The "Add User" option is highlighted.

Add A User

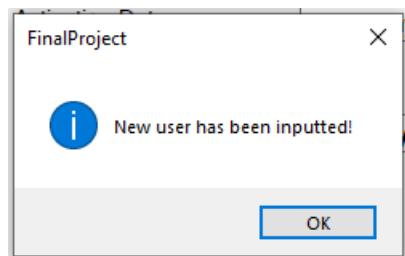
Your progress:  Fill the field below

Student ID	<input type="text" value="017202000025"/>	Gender	<input checked="" type="radio"/> Male <input type="radio"/> Female
Name	<input type="text" value="Gideon Ari Saputro"/>	Address	<input type="text"/>
Email	<input type="text" value="gideon@gmail.com"/>	Phone Number	<input type="text" value="0"/>
Password	<input type="password"/>	Activation Date	<input type="text" value="Sunday , January 3, 2021"/>
Major	<input type="text"/>	Valid Till	<input type="text" value="Sunday , January 3, 2021"/>
Batch	<input type="text" value="2017"/>		
<input type="checkbox"/> I accept any terms and condition <input type="checkbox"/> I responsible if the book that I borrowed broke, then I must pay the fine			
<input type="button" value="Register"/>		<input type="button" value="Clear"/>	

Add A User

Your progress:  Fill the field below

Student ID	<input type="text" value="017202000025"/>	Gender	<input checked="" type="radio"/> Male <input type="radio"/> Female
Name	<input type="text" value="Gideon Ari Saputro"/>	Address	<input type="text" value="Perumahan Kintamani Blok F7"/>
Email	<input type="text" value="gideon@gmail.com"/>	Phone Number	<input type="text" value="082365156313"/>
Password	<input type="password" value="gideon2020"/>	Activation Date	<input type="text" value="Sunday , January 3, 2021"/>
Major	<input type="text" value="Law"/>	Valid Till	<input type="text" value="Friday , January 3, 2025"/>
Batch	<input type="text" value="2020"/>		
<input checked="" type="checkbox"/> I accept any terms and condition <input checked="" type="checkbox"/> I responsible if the book that I borrowed broke, then I must pay the fine			
<input type="button" value="Register"/>		<input type="button" value="Clear"/>	



When Admin has inputted the user, Admin can click **Clear** button to **clear all field**.

Add A User

Your progress: | Fill the field below

Student ID	017202000025	Gender	<input checked="" type="radio"/> Male <input type="radio"/> Female
Name	Gideon Ari Saputro	Address	Perumahan Kintamani Blok F7
Email	gideon@gmail.com	Phone Number	082365156313
Password	gideon2020	Activation Date	Sunday , January 3, 2021
Major	Law	Valid Till	Friday , January 3, 2025
Batch	2020		

I accept any terms and condition
 I responsible if the book that I borrowed broke, then I must pay the fine

Register **Clear**

Add A User

Your progress: | Fill the field below

Student ID	017202000025	Gender	<input checked="" type="radio"/> Male <input type="radio"/> Female
Name	Gideon Ari Saputro	Address	Perumahan Kintamani Blok F7
Email	gideon@gmail.com	Phone Number	082365156313
Password	gideon2020	Activation Date	Sunday , January 3, 2021
Major	Law	Valid Till	Friday , January 3, 2025
Batch	2020		

I accept any terms and condition
 I responsible if the book that I borrowed broke, then I must pay the fine

Register **Clear**

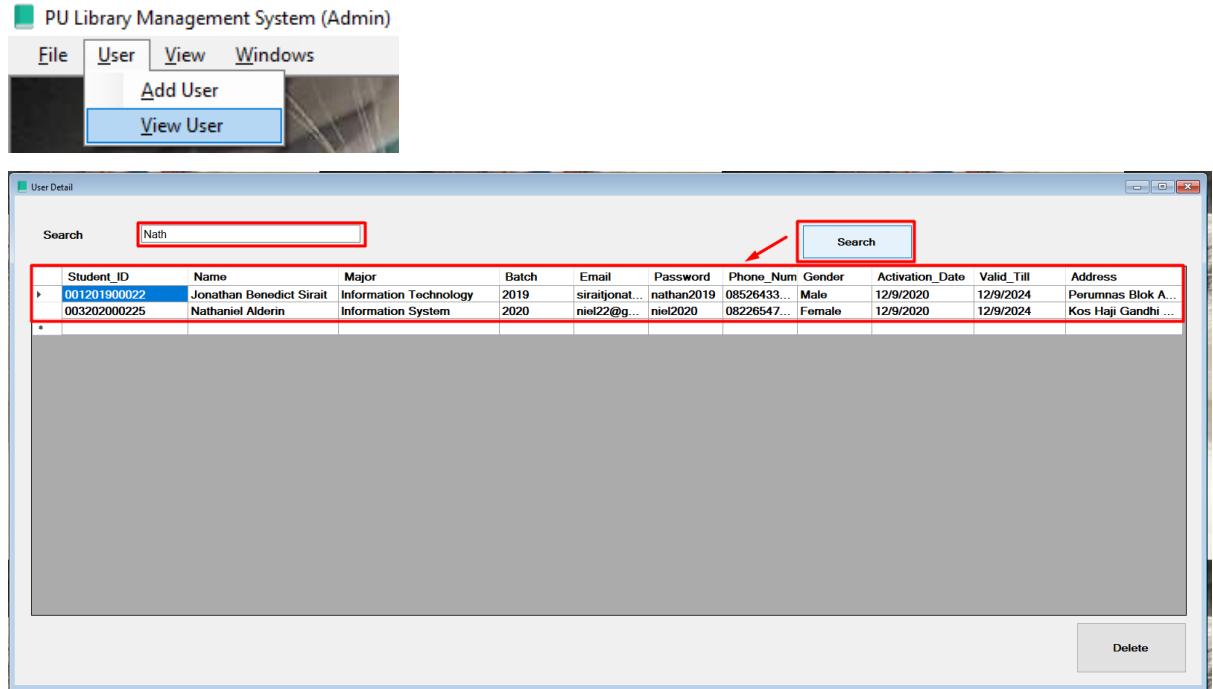
If Admin **doesn't input the required field**, it will **show a Warning Message Box**.

***Note:**

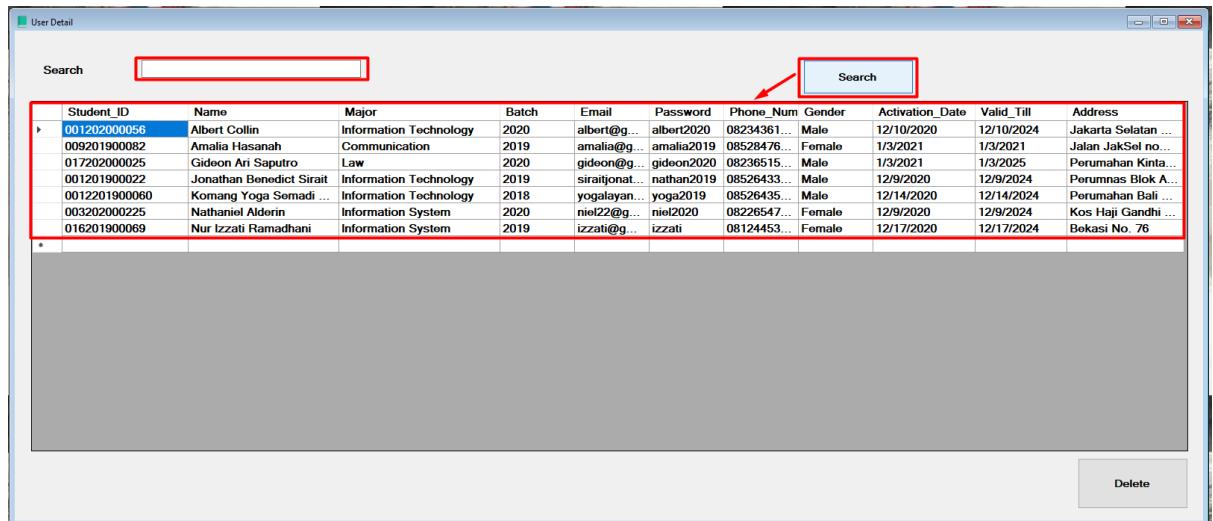
- The **Progress Bar** will add a value when Admin leave the control. It's because on every **control**, it uses **Leave** event. Hence, if Admin has inputted a wrong value at a certain control or field, the **Progress Bar** will still add a value. Therefore, the best way is to clear all the field using **Clear** button and start input the value from the beginning. Otherwise the **Progress Bar** will still add a value, and when it exceeds the maximum value, the application will stop and show an error.
- When fill a value or text at Address textbox, make sure the sentences or word didn't contain single quote ('). Otherwise, the application will stop and show an error. It's because the **textbox** didn't have escaped character so it cannot handle the single quote.

f. User Detail Form (UserDetail)

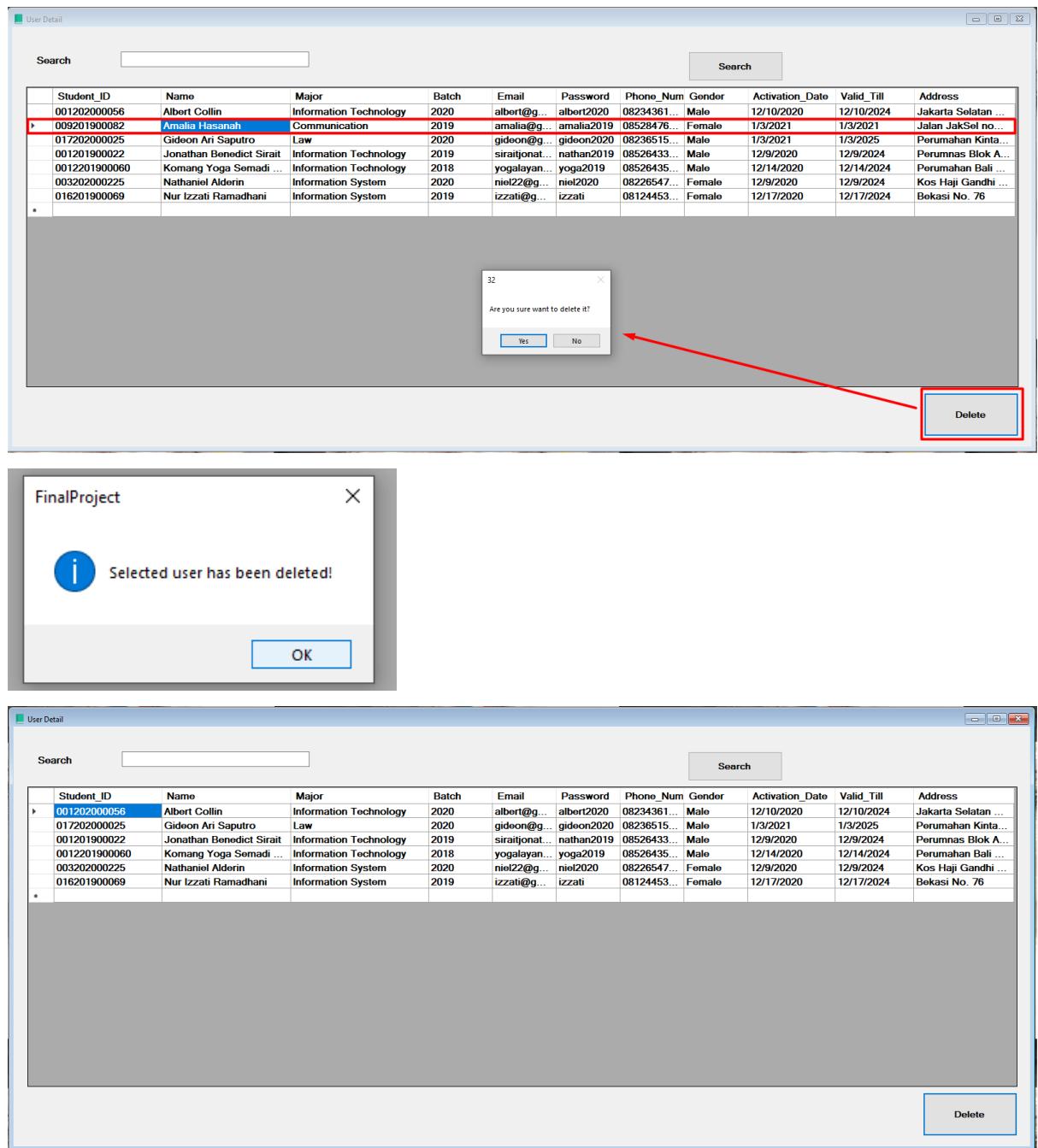
This form will **list all the users from the database to the DataGridView**. It also provides **search feature**. It will search based on **Name, Major, Email, Batch and Gender**.



If you want to **display the whole user after done searching**, just **empty the textbox** and **click Search** button. It will show all the user from DB to the DataGridView.



Last but not least, this form **provides Delete feature**. Just **select the preferred cell** and **click Delete** button. It will **show a message box**, and **click Yes**. Once you clicked, it will show a success Message Box. And done, the selected user is deleted.

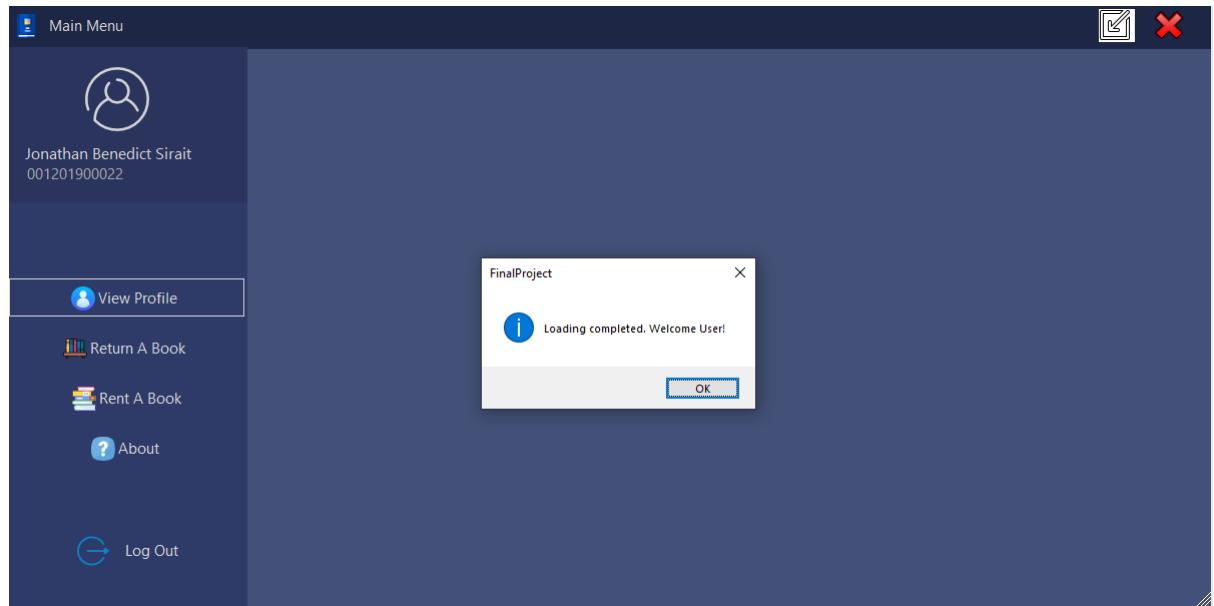


3) User Role

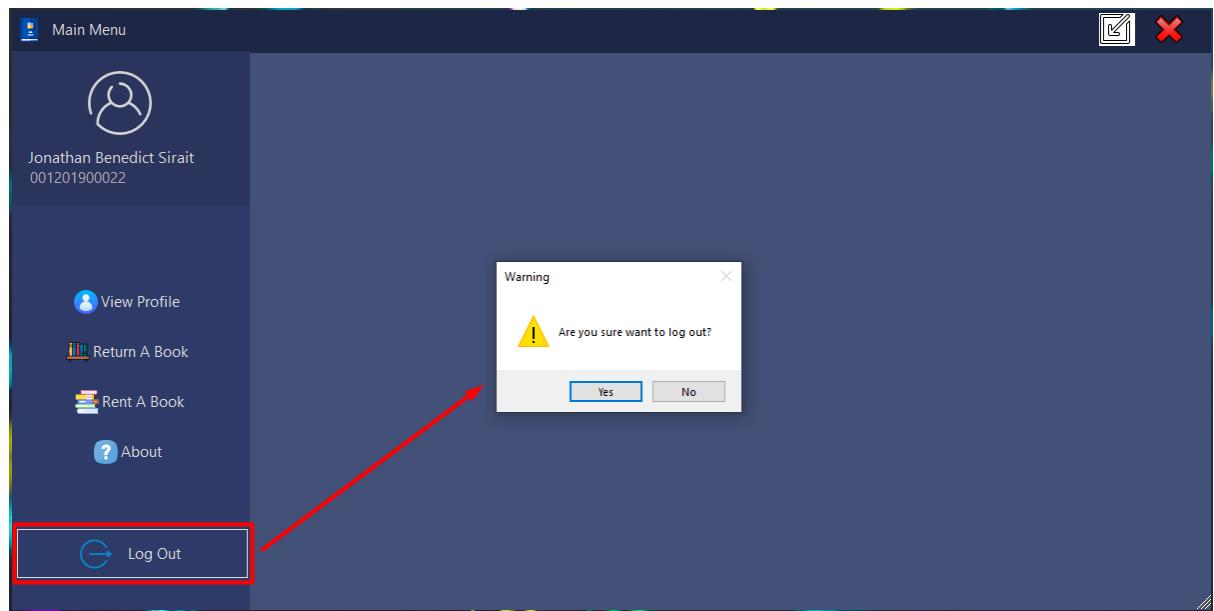
In **User-Related Form** it contains all form that refer to **User feature** or we can say what User can do in this application.

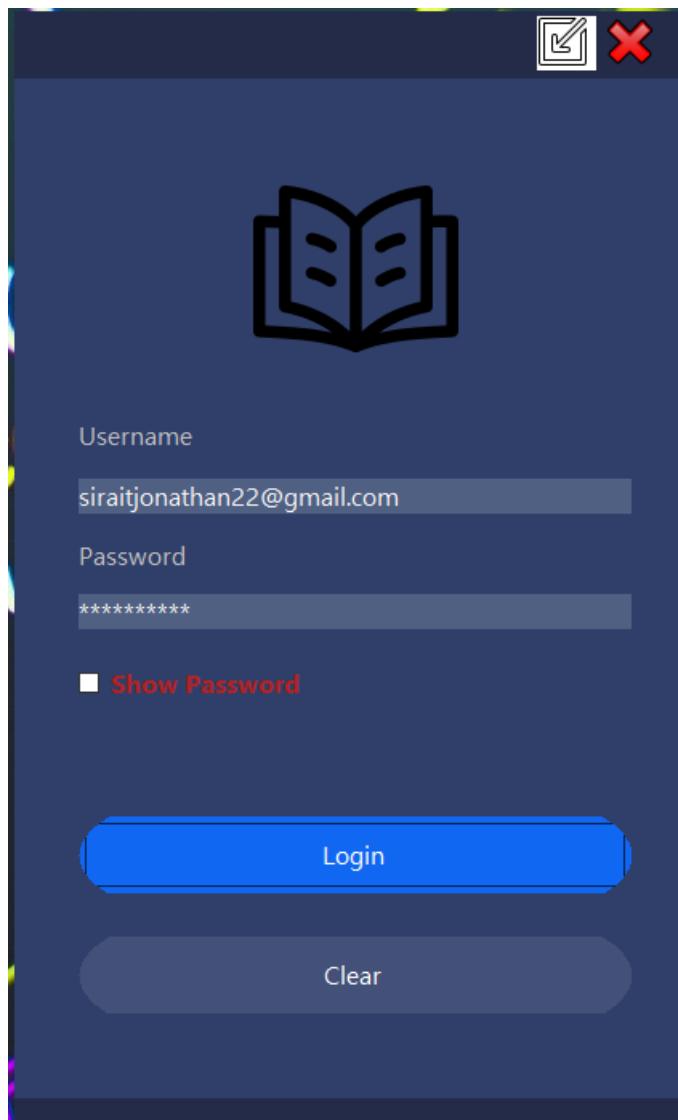
a. User Form

When **success login** it will direct to **User Form**.



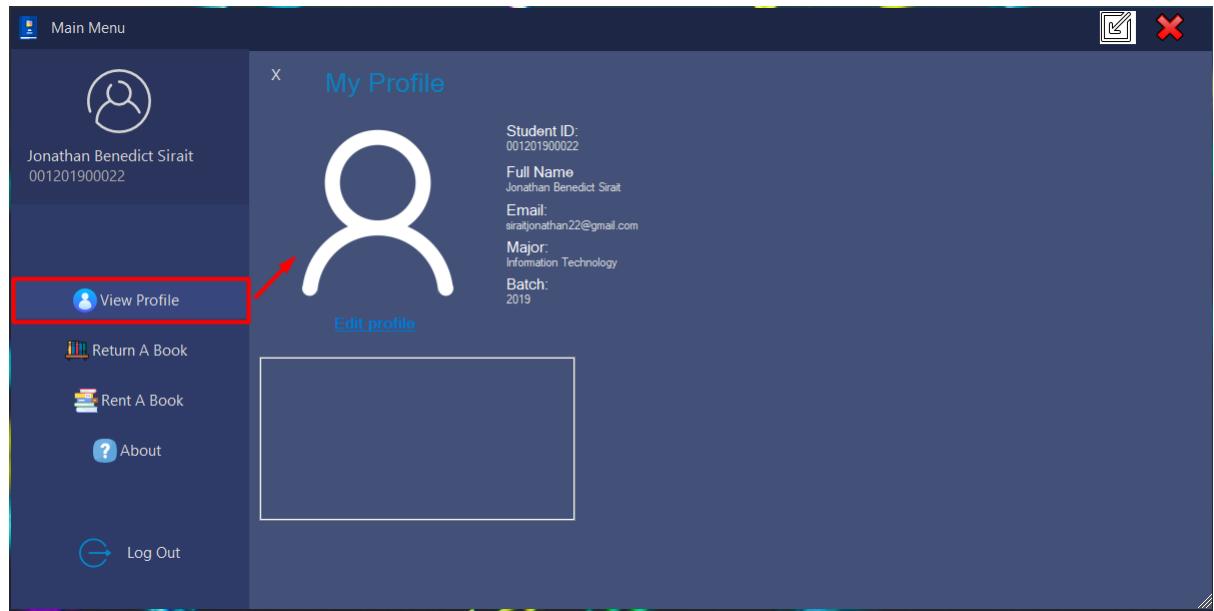
When User **click Log Out**, it will show a **Warning Messasge Box**. If click **Yes**, then it will close the User Form and **show the Login Form**. If click **No**, the it will **stay**.



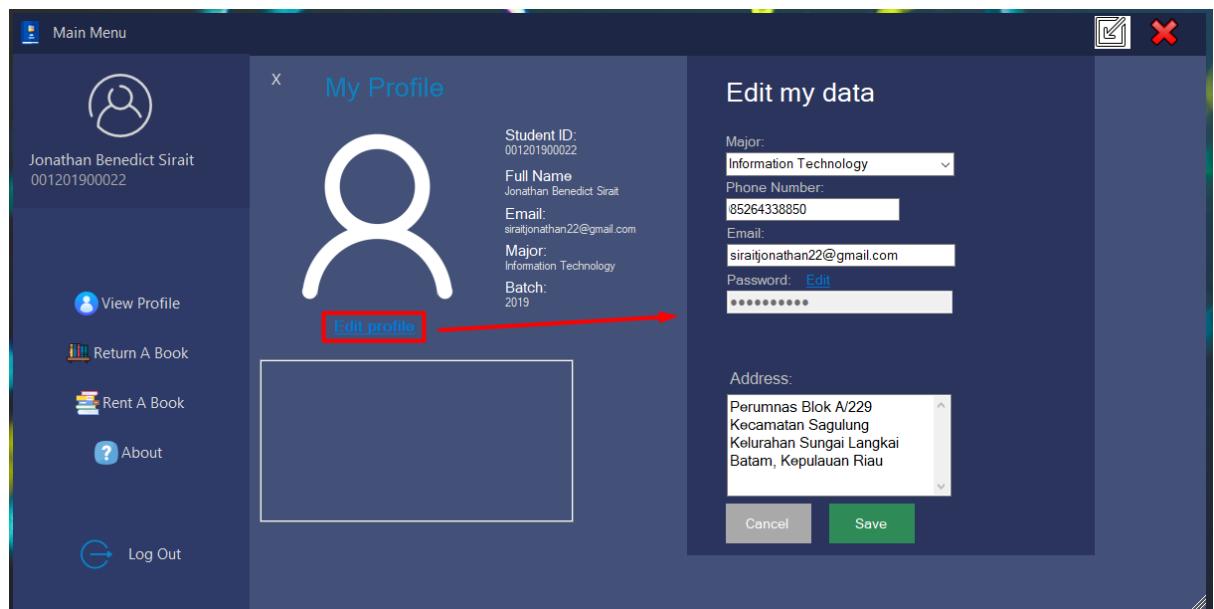


b. User Profile Form

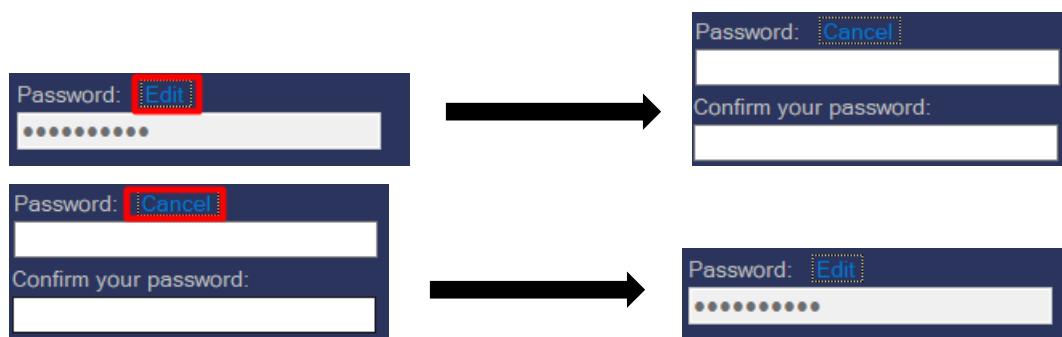
When **click View Profile** button at User Form, it will **show a form that display the user information (My Profile)**.



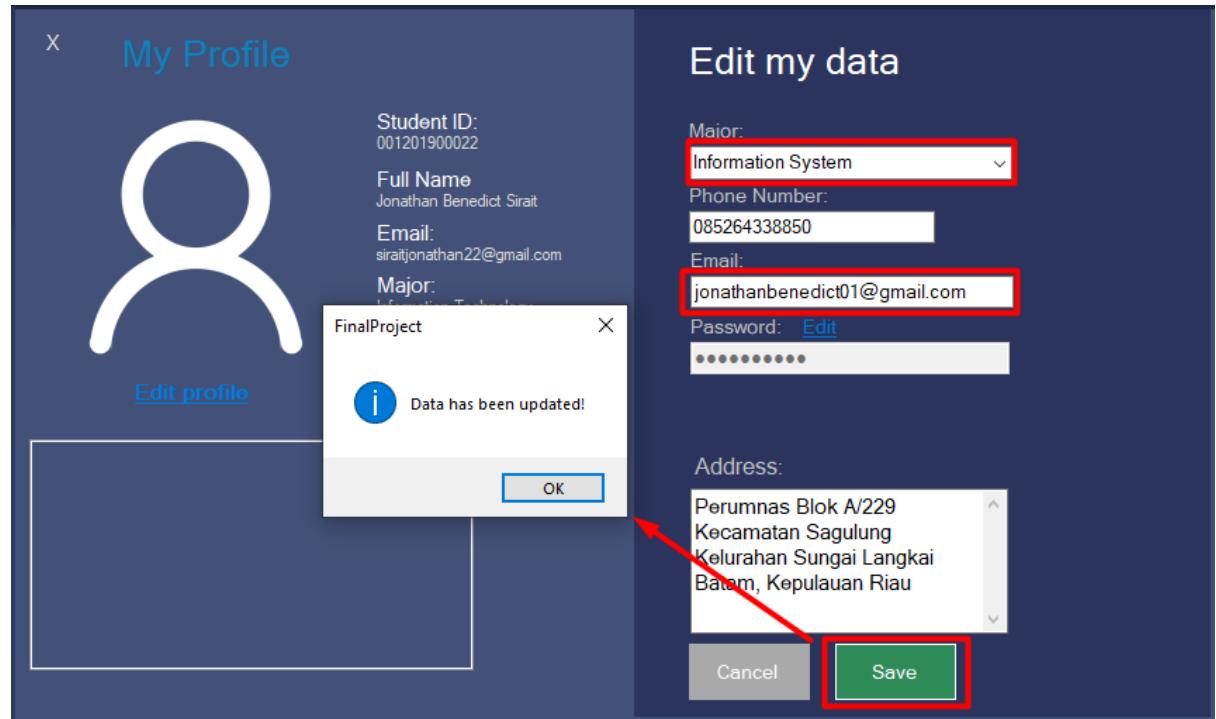
When click **Edit Profile**, it will perform a slide show and it will provide a form to edit user data.



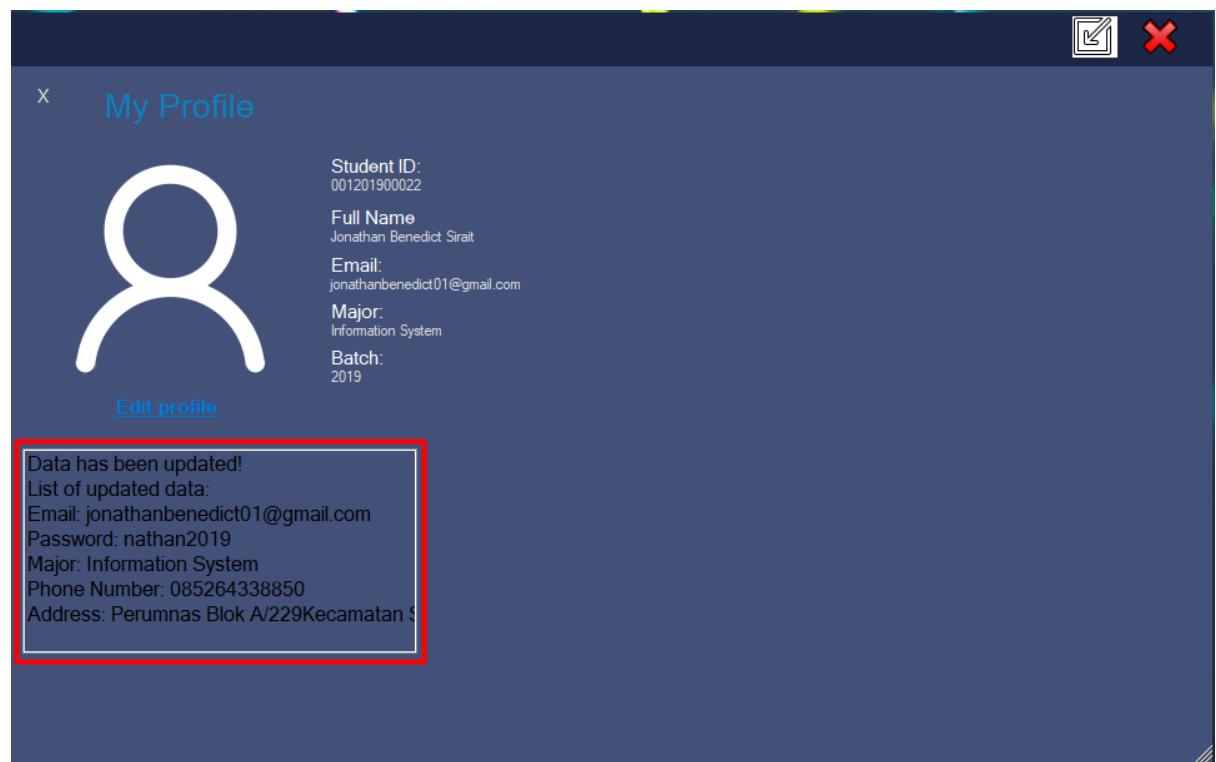
The form also provides edit password.



The User can freely edit the field based on their need. In this case, the User only updated the Major and Email (To **update major**, User must **select the item from Combo Box not type it**. Otherwise, it will show a **Warning Message Box**).

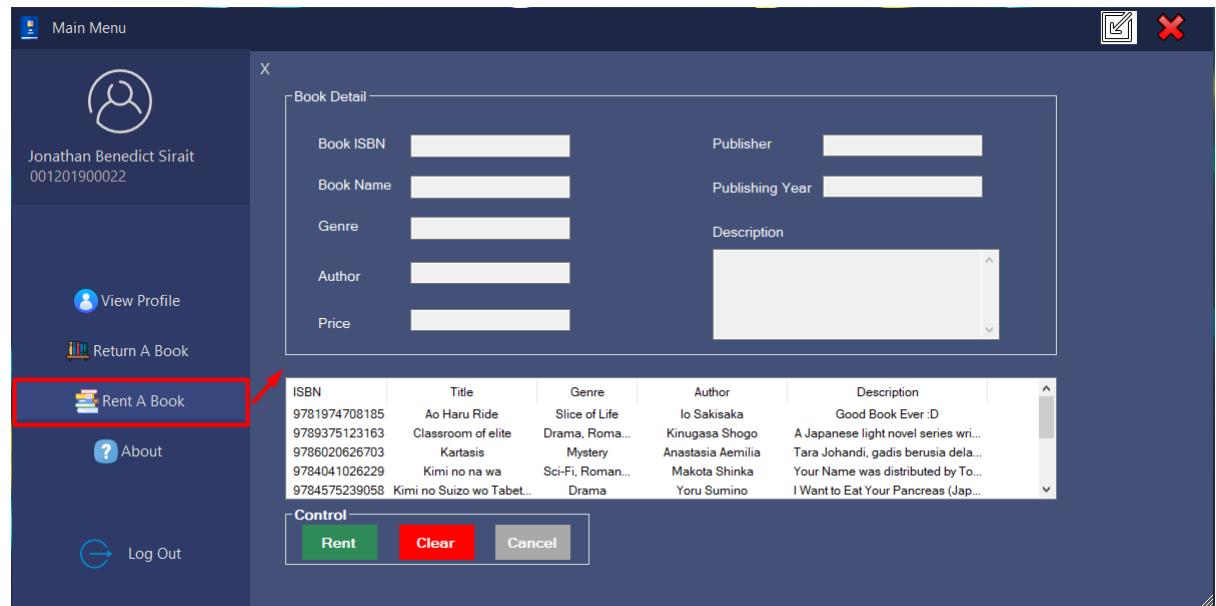


Once it finished, the data will show at the ListBox.

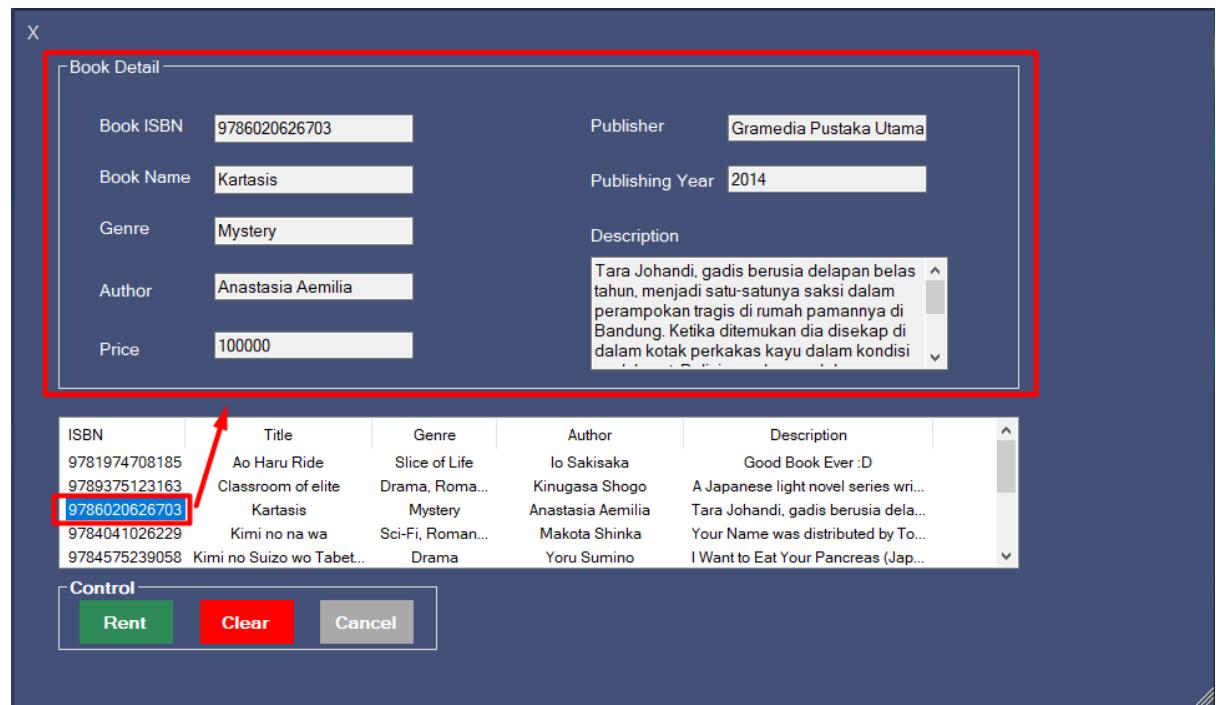


c. Rent a Book Form

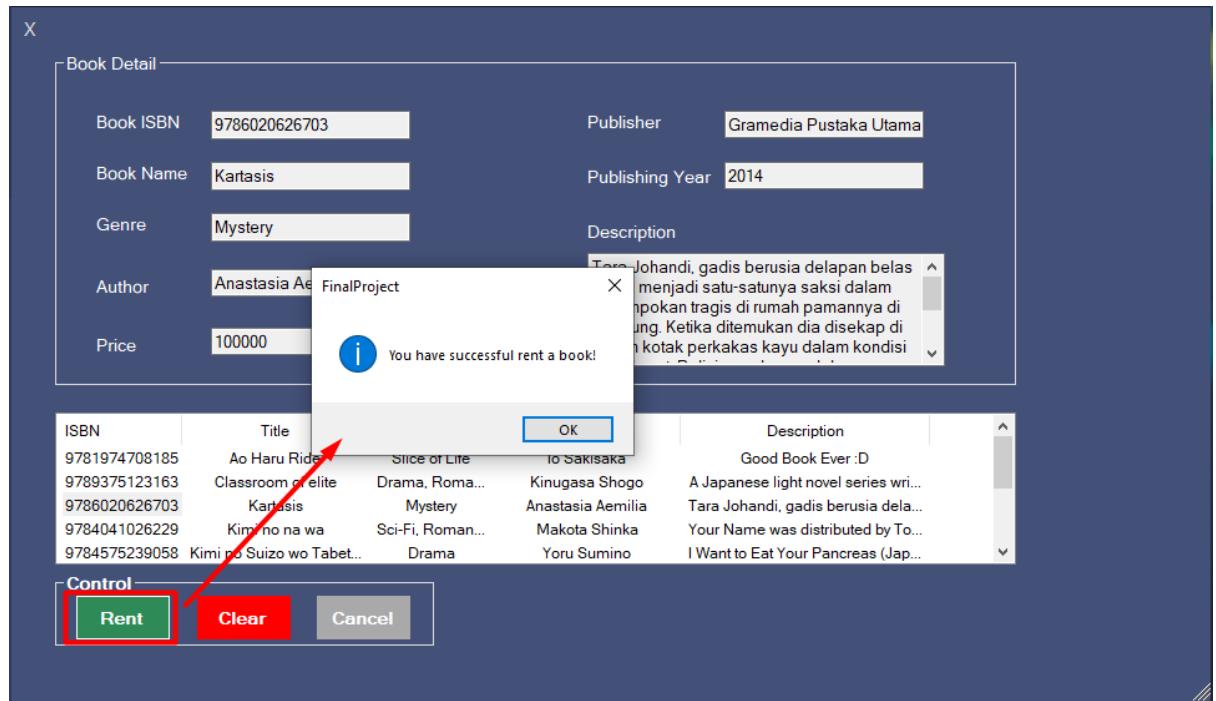
When click Rent A Book button at User Form, it will show a form that display all available book that can be borrowed.



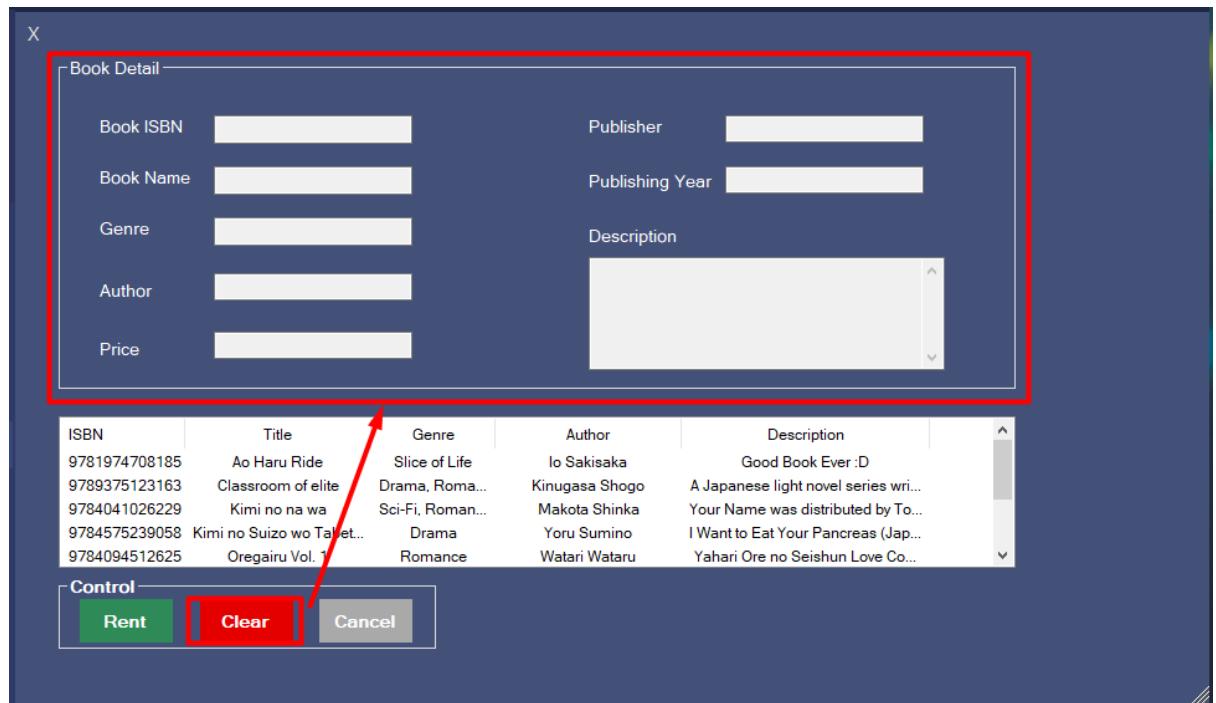
To rent a book you need, to select one of the ISBN. When the User click ISBN, it will show the book information at Book Detail section.



Once the User click Rent, it will show a Successful Message Box.



Once it rented, then the book will not available and it will not display on the ListView. To clear the field at Book Detail, click Clear.

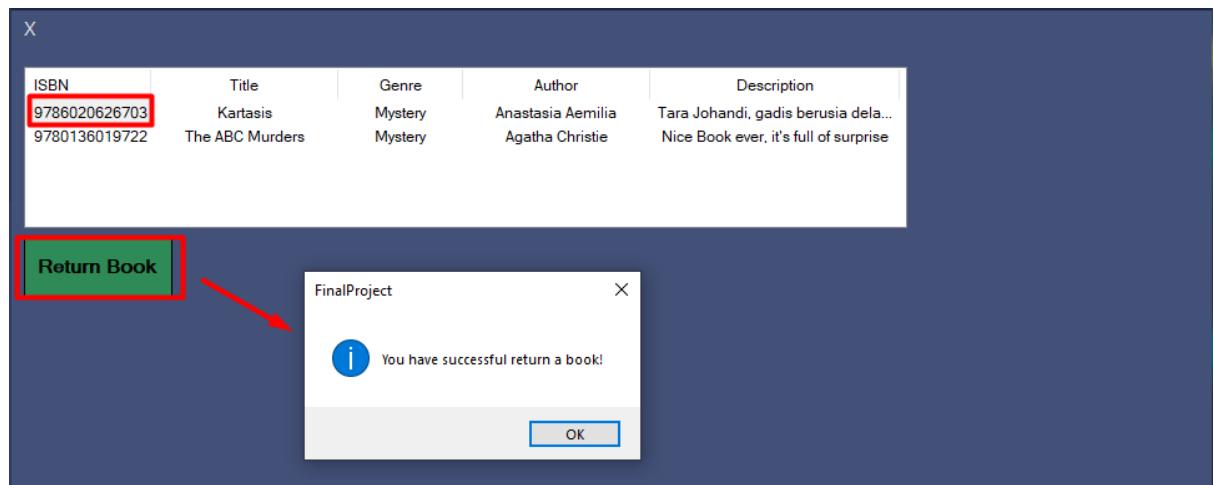


d. Return a Book Form

When click **Return A Book** button at User Form, it will show a form that display all rented book that been borrowed by the active user.



To return a book you need, to select one of the ISBN. Once the User click Return Book, it will show a Successful Message Box. Then the book will be available again, and it will not display at the ListView.



e. About Form

When click **About** button at User Form, it will **show a form that display a brief information** about the project information.

