Informatics College Pokhara

informatics
college   pokhara

Application Development

CS6004NI

Course Work 1

Submitted By: Latika Gurung   **Submitted To:** Ishwor Sapkota
London Met ID: Enter ID Here   Module Leader

Component Grade and Comments	
A. Implementation of Application	
User Interface and proper controls used for designing	missing controls in the interface
Manual data entry or import from csv	appropriate use of data types but missing some properties required or missing CRUD operation
Data Validation	missing most of the validation
Enrollment Report & weekly report in tabular format	very poorly executed reports and data not shown accurately
Course wise enrollment report & Chart display	Very poorly designed and only contains one report format with in appropriate data
Algorithm used for sorting & proper sorting of data	Default sorting provided by .net is used
B. Documentation	
User Manual for running the application	User Manual is below average. Is textual only.

Application architecture & description of the classes ad methods sued	average work with very limited explanation of the classes and methods used
Flow chart, algoriathms and data sctructures used	average work with very limited explanation and missing diagramatic representation.
Reflective essay	Very poorly written

C. Programming Style

Clarity of code,Popper Naming convention & comments	very poorly written code and no comments at all
System Usability	unusable system

Overall Grade:	E+ E+

Overall Comment:

Code should be self explainable with less comments. Need some proper naming of the component and require to add comments on required area.

In overall the code is working and all the functionality seems working and system can be used

Informatics College Pokhara

Information Systems

CS6004NP

Coursework 1

Submitted By:
Student Name: Latika Gurung
Student ID: NP04CP4A170053
Group: L3C2
Date: 10-Jan-2020

Submitted To:
Mr Ishwor Sapkota
Module Leader
Application Development

Contents

Table of Figure

1. Introduction

The system is Student Information system. This system is designed to store information of students in advanced and better way. It is made after undergoing various test circumstances.

This application provides user to input the id, name, address, course enroll & registration date of the student and store in a table. It also shows the overall stored data Furthermore this system also provides facility to view weekly report, import csv sorting according to name and registration date and so on.

1.1 Current Scenario

There are various student information system which stores information about students. But they are lacking in providing special features which can be essential for the data storage.

1.2 Proposed System

The proposed system provides more features like importing csv, generating a weekly report, sorting by name and date. This system also provides proper security in the login form.

2. User manual

Below are the screenshots which will show how to use the system.

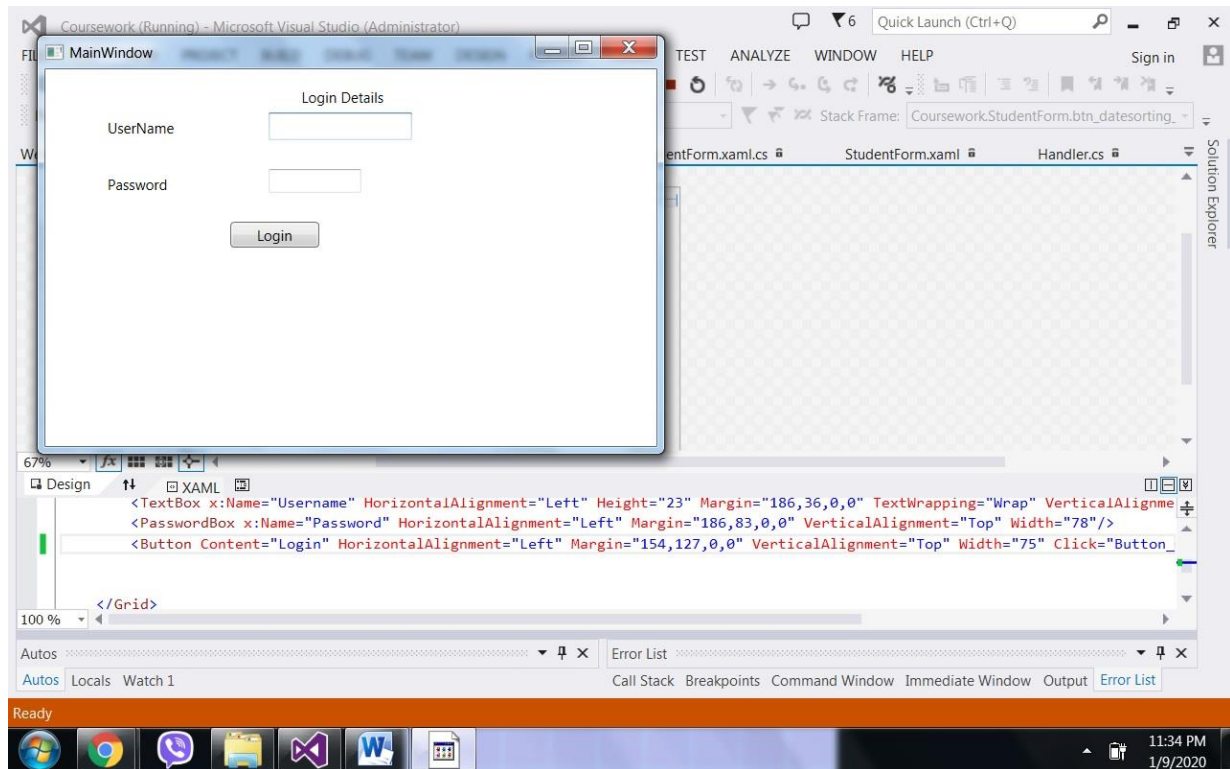The user needs to input username and password and press the login button.
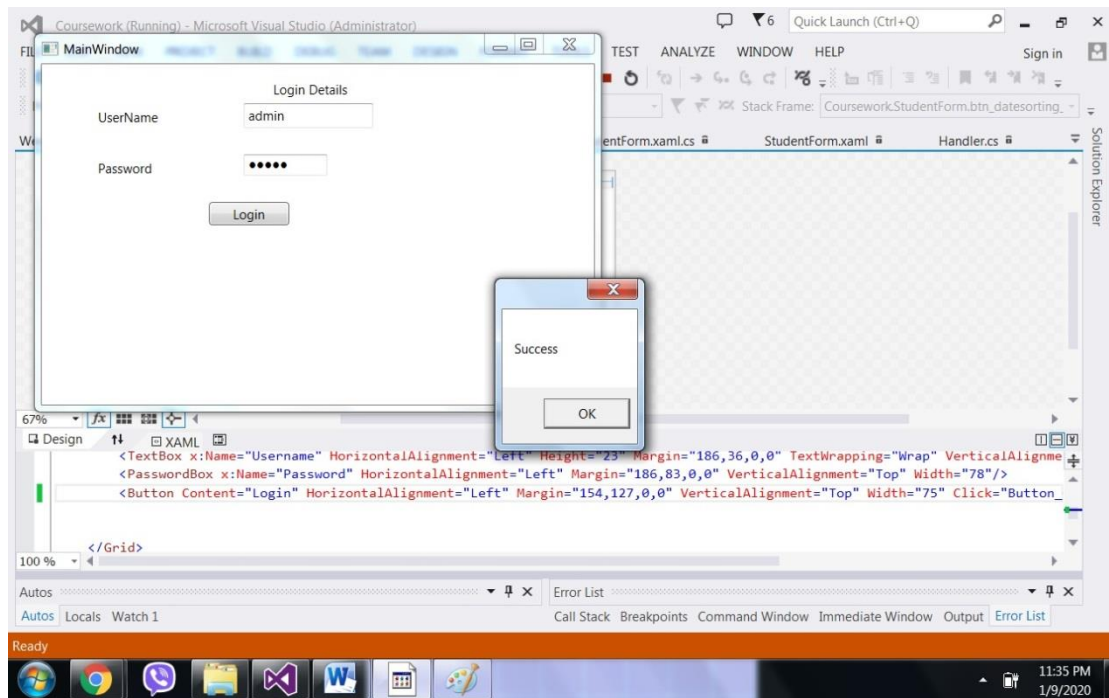

Figure 1

Figure 2

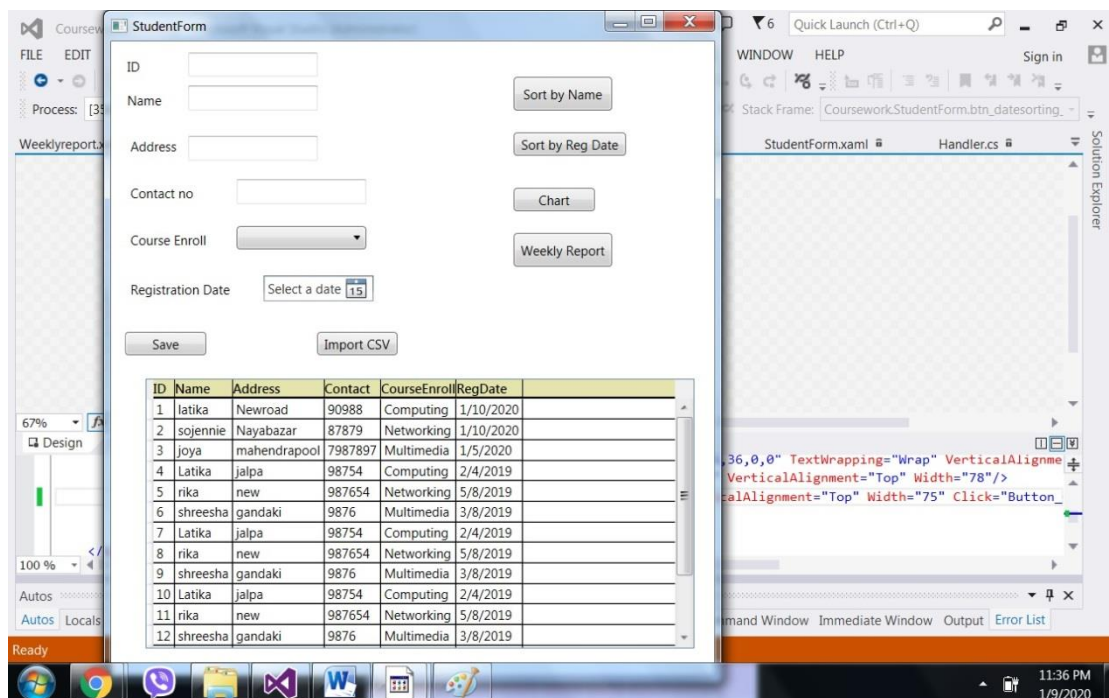After the login , an alert box appears saying success.

Figure 3

Latika Gurung

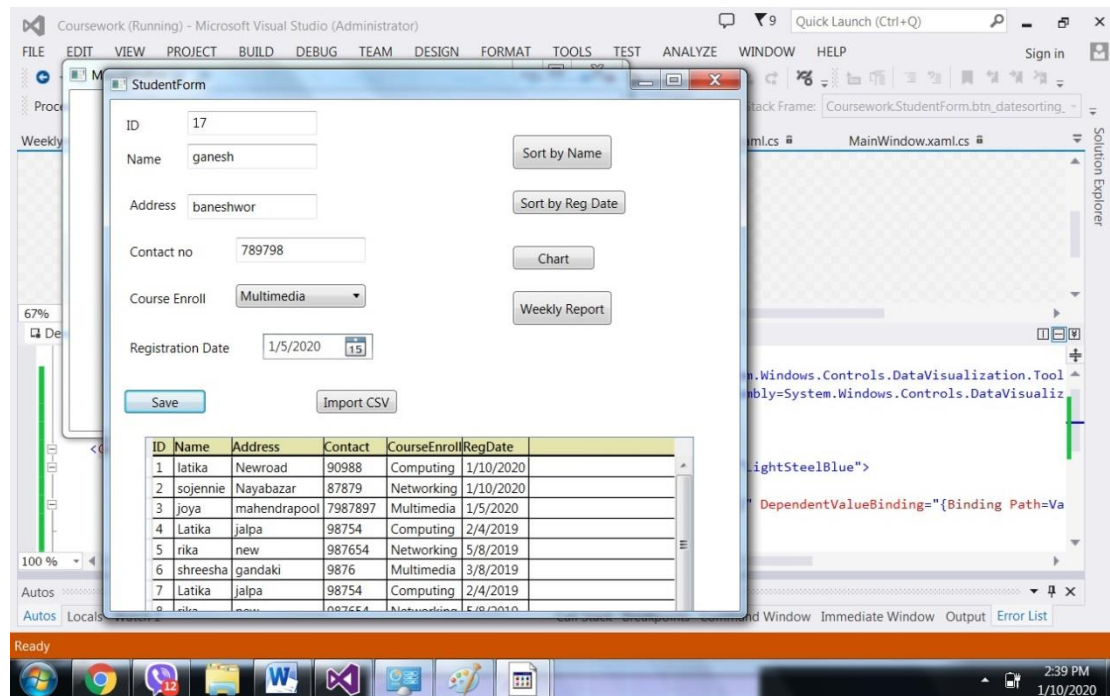Then a login form appears in which user needs to enter data.

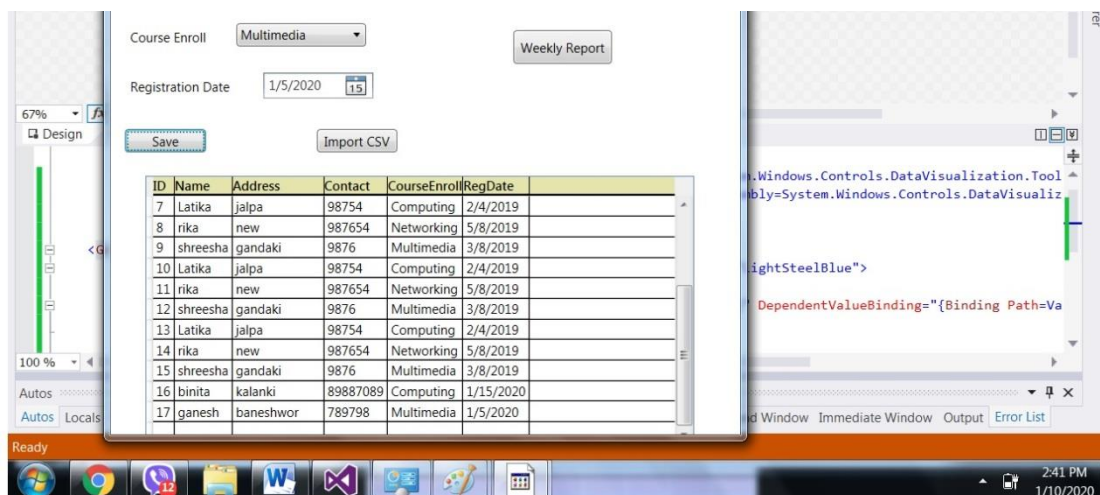*Figure 4*

Then you enter all the information and fill it.

Figure 5

Latika Gurung

And the information is added on the table.

Figure 6

Figure 7

Latika Gurung
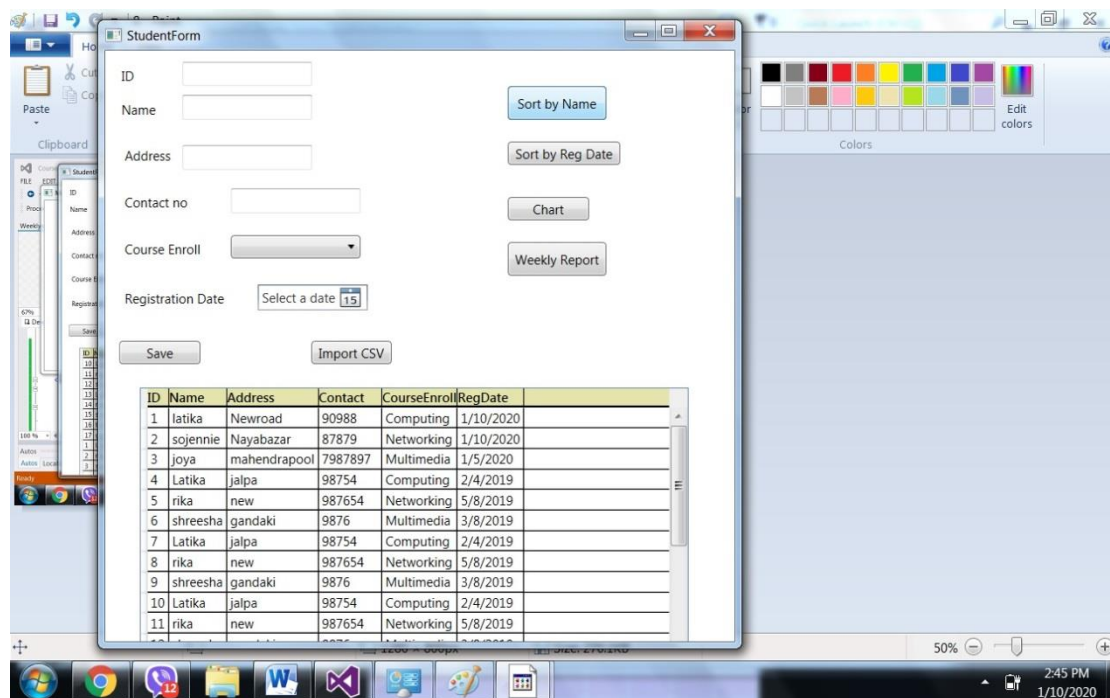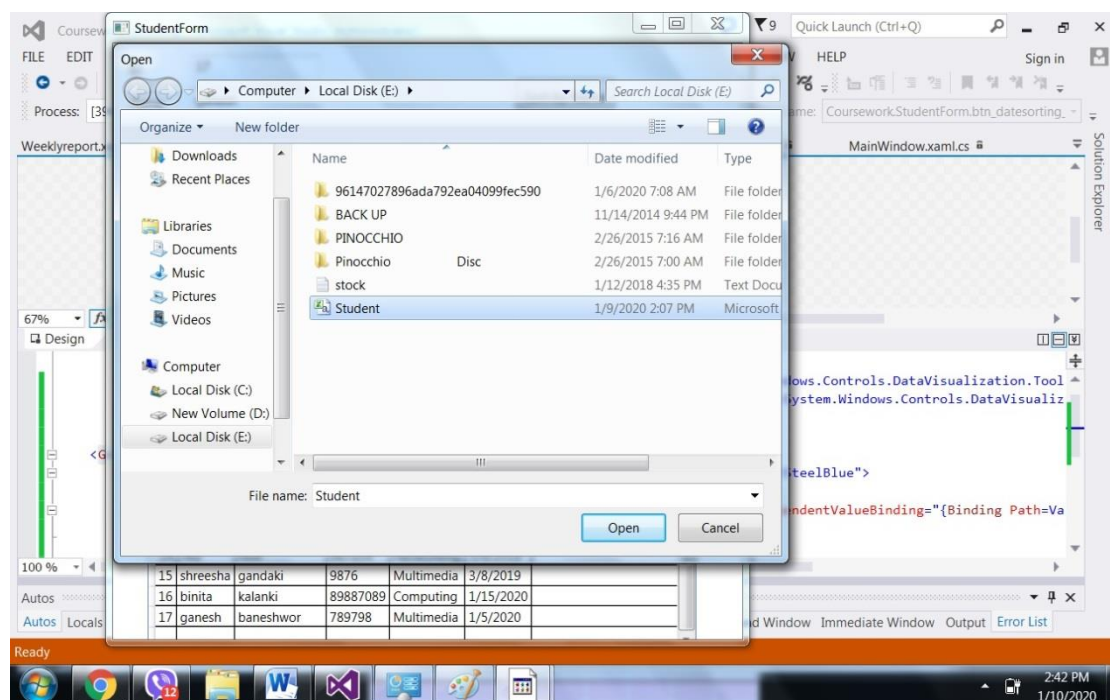
Click on the import CSV button. Then a folder opens where your student file is located which you have to add in excel.
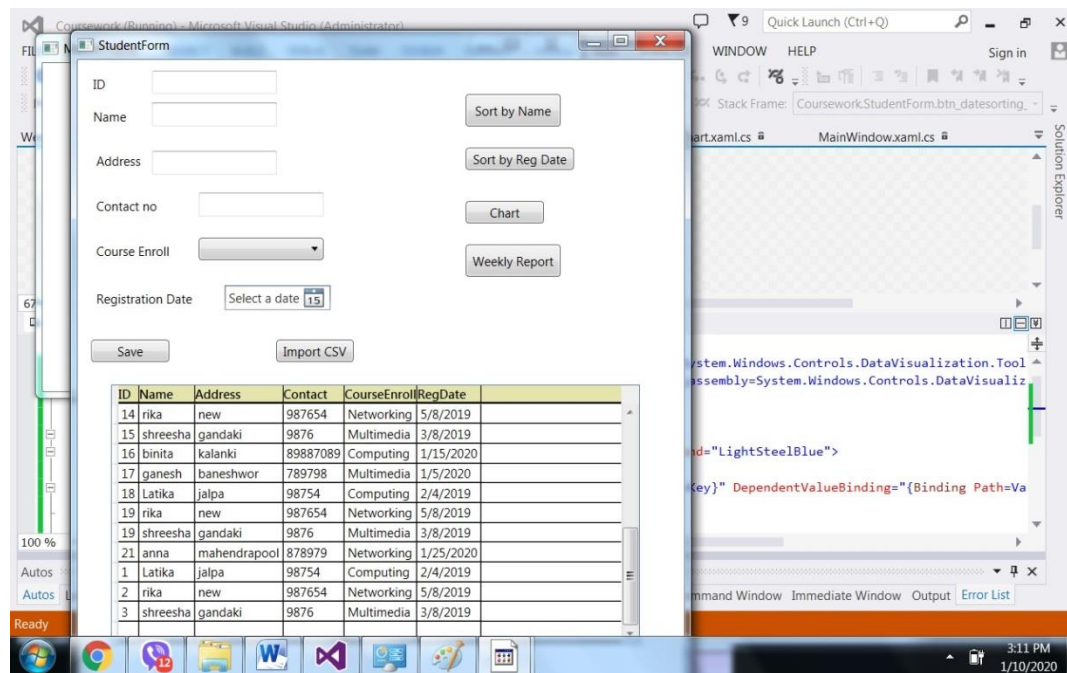
Figure 8

The information of the file is added in to the table.

Figure 9

This is the unsorted table.

Figure 10

This is the sorted table according to name.

Figure 11

This is the unsorted table.
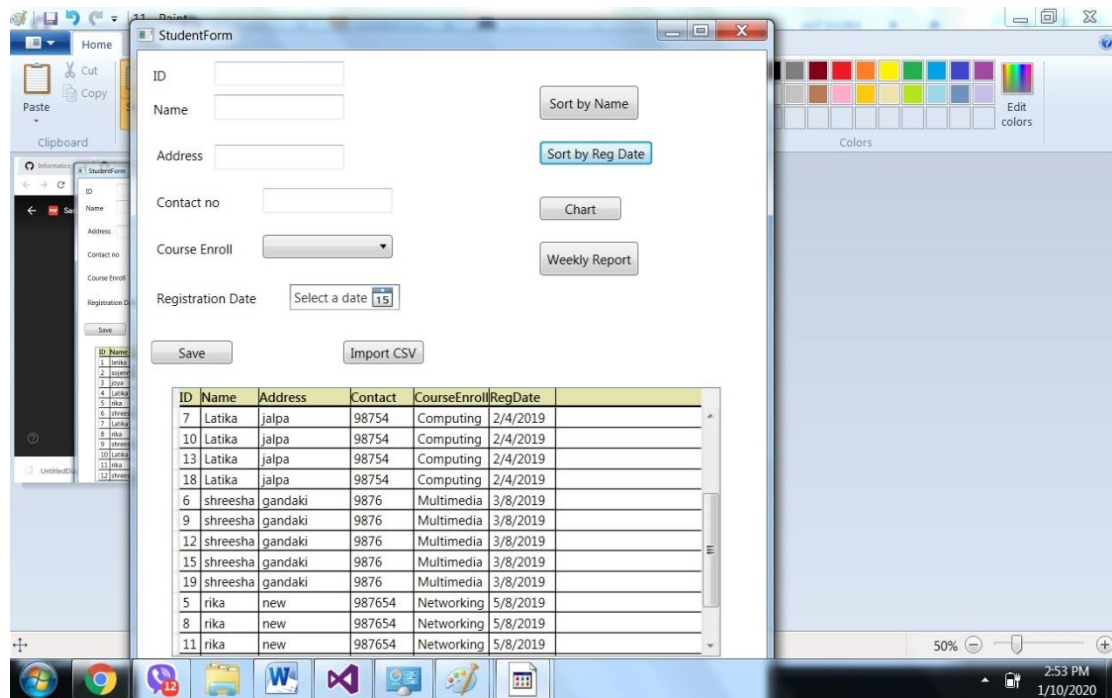
Figure 12

This is the sorted table according to date.
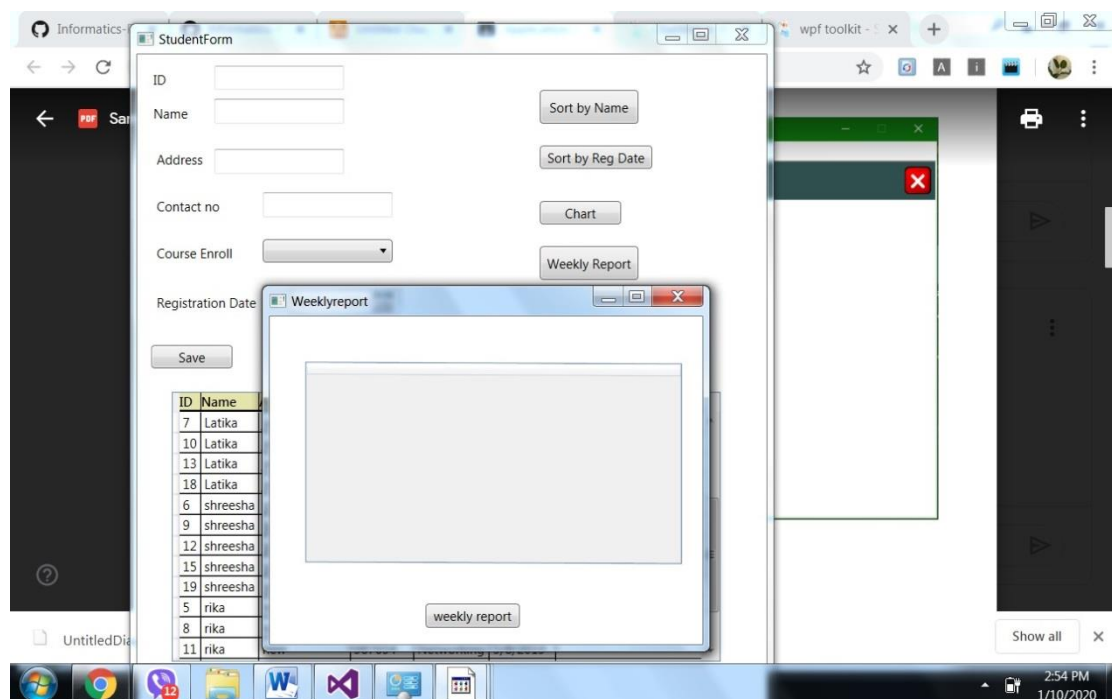
Figure 13

This is the weekly report window.

Latika Gurung
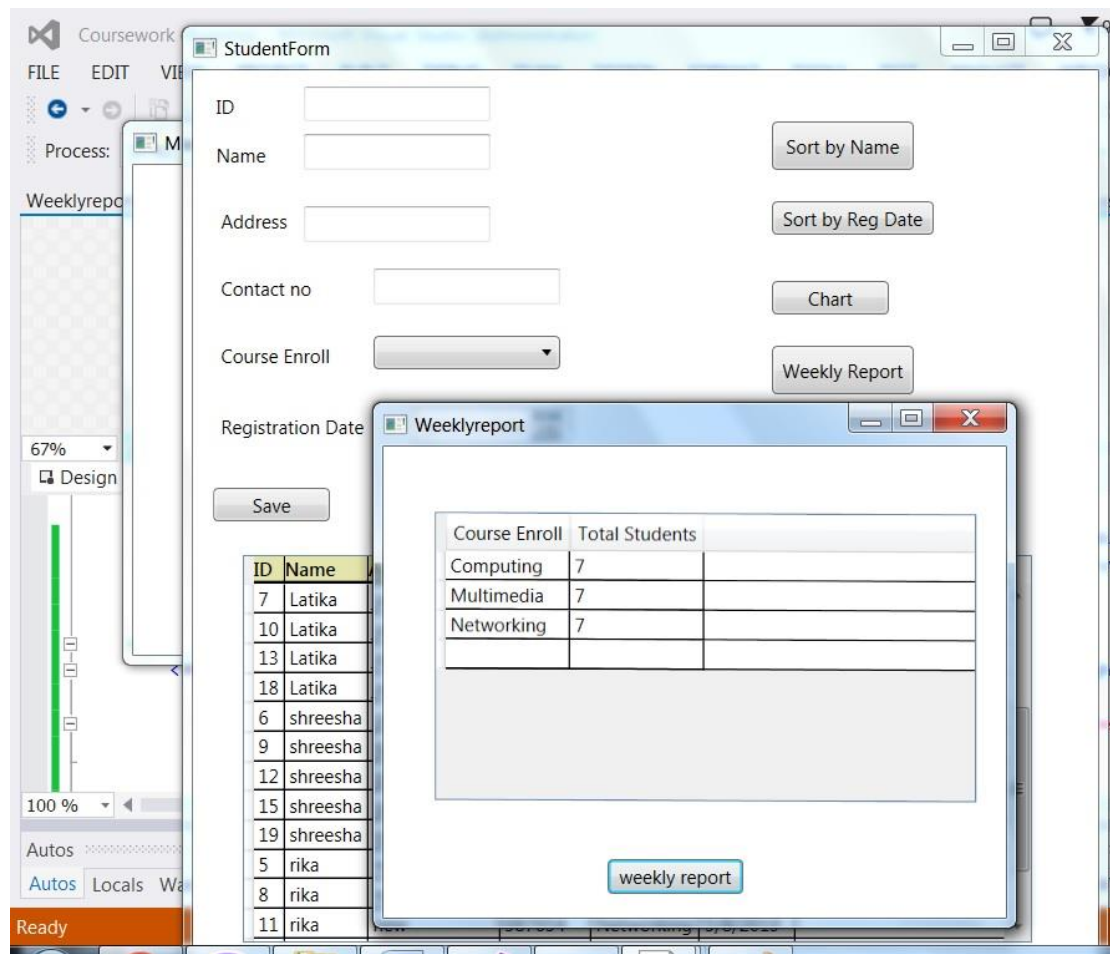
Figure 14

This shows the weekly report according to course enroll.
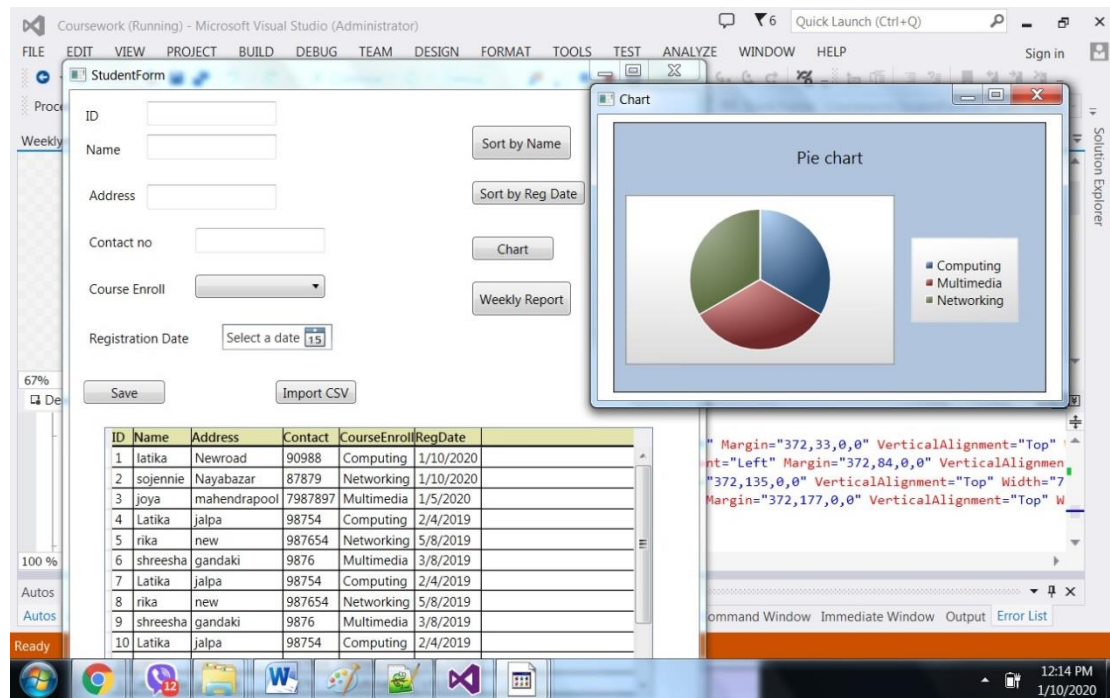
Figure 15

This shows the chart of the weekly report.

3. Journal Article

The **Student Information System (SIS)** is a fully computerized system or more precisely a database where all the student related data can be stored, retrieved, monitored and analyzed. The data is saved in one central location which can be accessed by multiple persons at the same time provided they have the login credentials; this ensures the safety of the stored information. Registration, demand generation, admission, billing, provision of financial aid to students and many other things can be managed with ease making the whole process of student enrollment quick, error free, systematic and undemanding faculty members in channelizing their concentration on the students and their

4. Architecture Design
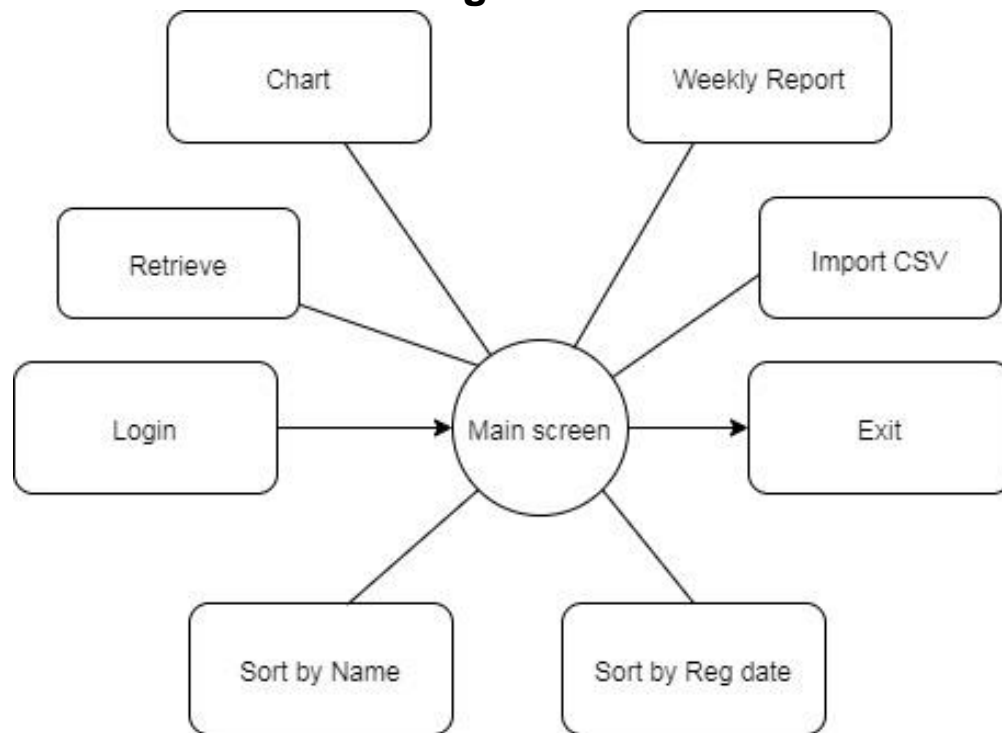
Figure 16

Class diagram
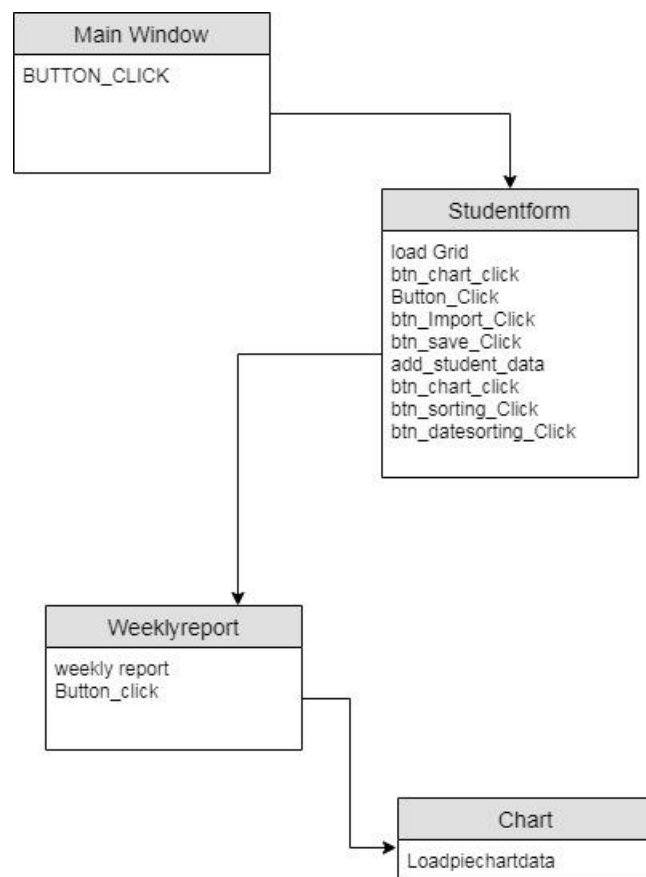
Figure 17

5. Flow chart Report

Weekly report
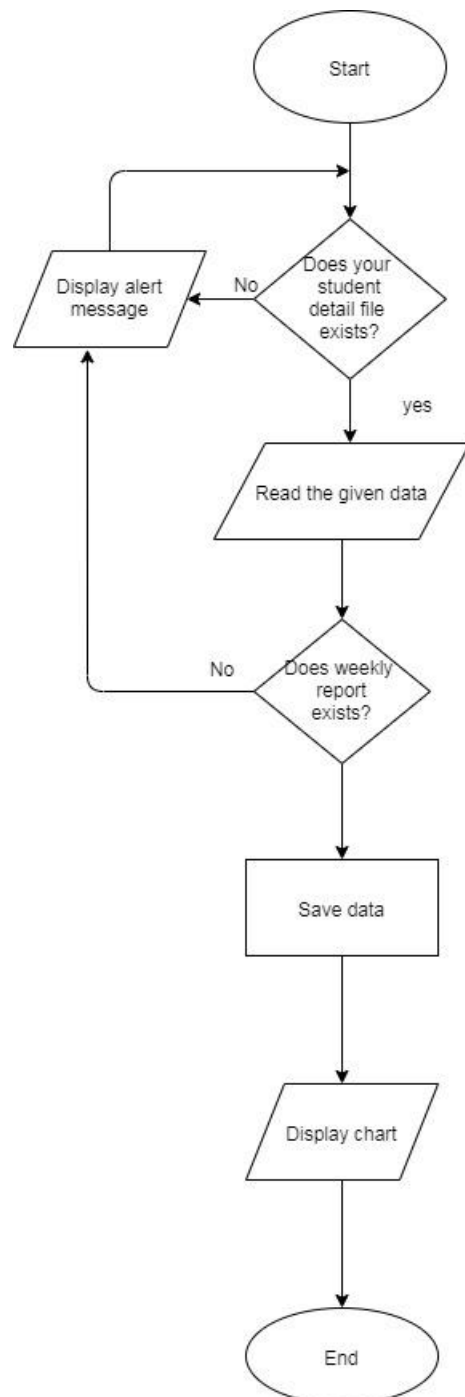
Figure 18

6. **Sorting Algorithm**

 The algorithm used in this system is bubble sort algorithm.

Bubble sort is a simple sorting algorithm. This sorting algorithm is comparison-based algorithm in which each pair of adjacent elements is compared and the elements are swapped if they are not in order. This algorithm is not suitable for large data sets as its average and worst case complexity are of O(n2) where n is the number of items (Anon., n.d.).

Implementing Bubble Sort Algorithm

Following are the steps involved in bubble sort(for sorting a given array in ascending order):

Starting with the first element(index = 0), compare the current element with the next element of the array.

If the current element is greater than the next element of the array, swap them.

If the current element is less than the next element, move to the next element. Repeat Step 1.

Let's consider an array with values {5, 1, 6, 2, 4, 3}

Below, we have a pictorial representation of how bubble sort will sort the given array.
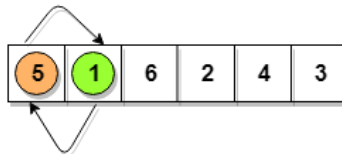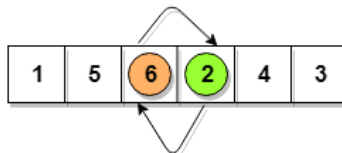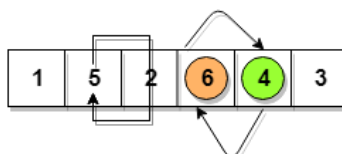
Latika Gurung
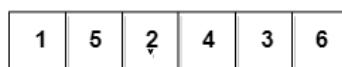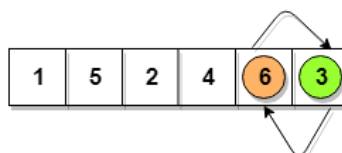
Figure 19

So as we can see in the representation above, after the first iteration, 6 is placed at the last index, which is the correct position for it.

Similarly after the second iteration, 5 will be at the second last index, and so on (Anon., n.d.).

7. Conclusion

This project was implemented using wpf.net framework. This coursework is based to implement a student information system which is used to input student's data.

It required a visual studio. The framework consists of login to provide user security.

After login, it shows the home screen which consist of login form and has various button which is used for the operation of the system.

8. Bibliography

Bibliography

Anon., n.d. *https://www.studytonight.com/data-structures/bubble-sort.* [Online]
Available at: https://www.studytonight.com/data-structures/bubble-sort

Anon., n.d.
https://www.tutorialspoint.com/data_structures_algorithms/bubble_sort_algorithm.htm
. [Online]
Available at:
https://www.tutorialspoint.com/data_structures_algorithms/bubble_sort_algorithm.htm

9. Appendix

```csharp
using System;
using System.Collections.Generic;
using System.Data;
using System.IO;
using System.Linq;
using System.Text;
using System.Threading.Tasks;
using System.Windows;
using System.Windows.Controls;
using System.Windows.Data;
using System.Windows.Documents;
using System.Windows.Input;
using System.Windows.Media;
using System.Windows.Media.Imaging;
using System.Windows.Shapes;
using DataHandler;
using Microsoft.Win32;

namespace Coursework
{

    /// <summary>
    /// Interaction logic for StudentForm.xaml
    /// </summary>
    public partial class StudentForm : Window
    {
        DataTable dataTable;

        public StudentForm()
        {
            InitializeComponent();
            viewGrid();
        }
        public void viewGrid()
        {
            if (File.Exists(@"D:\student.xml"))
            {


                var dataSet = new DataSet();
                dataSet.ReadXml(@"D:\student.xml");
                dataTable = dataSet.Tables["Student"];
                DataGridXAML.ItemsSource = dataTable.DefaultView;


            }
        }



        private void btnSave_Click(object sender, RoutedEventArgs e)
        {
            var handler = new Handler();
            var dataSet = new DataSet();

            if (File.Exists(@"D:\student.xml"))
            {
                dataSet.ReadXml(@"D:\student.xml");

            }
```

Latika Gurung

```
            else
            {
                dataSet = handler.CreateDataSet();
            }
            AddStudentData(dataSet);
            dataSet.WriteXml(@"D:\student.xml");
            viewGrid();
        }
        private void AddStudentData(DataSet dataSet)
        {
            var dr1 = dataSet.Tables["Student"].NewRow();
            dr1["ID"] = txtId.Text;
            dr1["Name"] = txtname.Text;
            dr1["Address"] = txtAddress.Text;
            dr1["Contact"] = txtContact.Text;
            dr1["CourseEnroll"] = txtCrsEnrl.Text;
            dr1["RegDate"] = txtRegidate.Text;
            dataSet.Tables["Student"].Rows.Add(dr1);
        }



        private void btn_sorting_Click(object sender, RoutedEventArgs e)
        {
            dataTable.DefaultView.Sort = "Name ASC";
            DataGridXAML.DataContext = dataTable.DefaultView;



        }

        private void btn_datesorting_Click(object sender, RoutedEventArgs e)
        {
            dataTable.DefaultView.Sort = "RegDate ASC";
            DataGridXAML.DataContext = dataTable.DefaultView;
        }

        private void btn_chart_Click(object sender, RoutedEventArgs e)
        {
            Chart chart = new Chart();
            chart.Show();

        }

        private void Button_Click(object sender, RoutedEventArgs e)
        {
            Weeklyreport weeklyReport = new Weeklyreport();
            weeklyReport.ShowDialog();
        }


        private void DataGridXAML_SelectionChanged(object sender,
SelectionChangedEventArgs e)
        {

        }

        private void btnImport_Click(object sender, RoutedEventArgs e)
        {
            var dataSet = new DataSet();
            dataSet.ReadXml(@"D:\student.xml");
```

Latika Gurung

```csharp
            OpenFileDialog openFileDialog = new OpenFileDialog();
            if (openFileDialog.ShowDialog() == true)
            {
                string filePath = openFileDialog.FileName;
                //read all std from file code copy

                using (var reader = new StreamReader(filePath))
                {
                    reader.ReadLine();
                    while (!reader.EndOfStream)
                    {
                        var line = reader.ReadLine();
                        var values = line.Split(',');
                        var newRow = dataSet.Tables["Student"].NewRow();
                        newRow["ID"] = values[0];
                        newRow["Name"] = values[1];
                        newRow["Address"] = values[2];
                        newRow["Contact"] = values[3];
                        newRow["CourseEnroll"] = values[4];
                        newRow["RegDate"] = values[5];
                        dataSet.Tables["Student"].Rows.Add(newRow);

                        dataSet.WriteXml(@"D:\student.xml");
                    }


                }
            }
        }
    }
}
```

```xml
<Window x:Class="Coursework.StudentForm"
        xmlns="http://schemas.microsoft.com/winfx/2006/xaml/presentation"
        xmlns:x="http://schemas.microsoft.com/winfx/2006/xaml"
        Title="StudentForm" Height="1405.239" Width="596.12">
    <Grid>
        <Label Content="ID" HorizontalAlignment="Left" Margin="10,10,0,0"
VerticalAlignment="Top"/>
        <Label Content="Name" HorizontalAlignment="Left" Margin="10,41,0,0"
VerticalAlignment="Top"/>
        <Label Content="Address" HorizontalAlignment="Left" Margin="13,84,0,0"
VerticalAlignment="Top"/>
        <Label Content="Contact no" HorizontalAlignment="Left"
Margin="13,127,0,0" VerticalAlignment="Top"/>
        <Label Content="Course Enroll" HorizontalAlignment="Left"
Margin="13,170,0,0" VerticalAlignment="Top"/>
        <TextBox Name="txtAddress" HorizontalAlignment="Left" Height="23"
Margin="71,87,0,0" TextWrapping="Wrap" Text="" VerticalAlignment="Top"
Width="120"/>
        <TextBox Name="txtContact" HorizontalAlignment="Left" Height="23"
Margin="116,127,0,0" TextWrapping="Wrap" Text="" VerticalAlignment="Top"
Width="120"/>
        <TextBox x:Name="txtname" HorizontalAlignment="Left" Height="23"
Margin="71,41,0,0" TextWrapping="Wrap" Text="" VerticalAlignment="Top"
Width="120"/>
        <TextBox x:Name="txtId" HorizontalAlignment="Left" Height="23"
Margin="71,10,0,0" TextWrapping="Wrap" Text="" VerticalAlignment="Top"
Width="120"/>
```

Latika Gurung

```xml
        <ComboBox Name="txtCrsEnrl" HorizontalAlignment="Left"
Margin="116,170,0,0" VerticalAlignment="Top" Width="120">
            <ComboBoxItem Content="Computing"/>
            <ComboBoxItem Content="Networking"/>
            <ComboBoxItem Content="Multimedia"/>
        </ComboBox>
        <Label Content="Registration Date" HorizontalAlignment="Left"
Margin="13,216,0,0" VerticalAlignment="Top"/>
        <Button Name="btnSave" Content="Save" HorizontalAlignment="Left"
Margin="13,268,0,0" VerticalAlignment="Top" Width="75" Click="btnSave_Click"/>
        <DatePicker x:Name="txtRegidate" HorizontalAlignment="Left"
Margin="141,216,0,0" VerticalAlignment="Top"/>
        <Button Name="btnImport" Content="Import CSV"
HorizontalAlignment="Left" Margin="190,268,0,0" VerticalAlignment="Top"
Width="75" Click="btnImport_Click" />
        <DataGrid Name="DataGridXAML"  HorizontalAlignment="Left"
Margin="32,311,0,0" VerticalAlignment="Top" Height="249" Width="507"
SelectionChanged="DataGridXAML_SelectionChanged">
            <DataGrid.Resources>
                <Style TargetType="{x:Type DataGridColumnHeader}">
                    <Setter Property="Background" Value="#FFE4E4AB"/>
                    <Setter Property="FontWeight" Value="SemiBold"/>
                    <Setter Property="BorderThickness" Value="0 0 1 2"/>
                    <Setter Property="BorderBrush" Value="Black"/>
                </Style>
            </DataGrid.Resources>

        </DataGrid>
        <Button Name="btn_sorting" Content="Sort by Name"
HorizontalAlignment="Left" Margin="372,33,0,0" VerticalAlignment="Top"
Width="91" Height="31" Click="btn_sorting_Click"/>
        <Button Name="btn_datesorting" Content="Sort by Reg Date"
HorizontalAlignment="Left" Margin="372,84,0,0" VerticalAlignment="Top"
Width="104" Click="btn_datesorting_Click"/>
        <Button Name="btn_chart" Content="Chart" HorizontalAlignment="Left"
Margin="372,135,0,0" VerticalAlignment="Top" Width="75"
Click="btn_chart_Click"/>
        <Button Name="btn_Week" Content="Weekly Report"
HorizontalAlignment="Left" Margin="372,177,0,0" VerticalAlignment="Top"
Width="91" Height="31" Click="Button_Click"/>

    </Grid>
</Window>
using System;
using System.Collections.Generic;
using System.Linq;
using System.Text;
using System.Threading.Tasks;
using System.Windows;
using System.Windows.Controls;
using System.Windows.Data;
using System.Windows.Documents;
using System.Windows.Input;
using System.Windows.Media;
using System.Windows.Media.Imaging;
using System.Windows.Navigation;
using System.Windows.Shapes;

namespace Coursework
{
    /// <summary>
```

Latika Gurung

```csharp
        /// Interaction logic for MainWindow.xaml
        /// </summary>
        public partial class MainWindow : Window
        {
            public MainWindow()
            {
                InitializeComponent();
            }

            private void Button_Click(object sender, RoutedEventArgs e)
            {
                if (Password.Password != "" && Username.Text != "")
                {
                    if (Password.Password == "admin" && Username.Text == "admin")
                    {
                        MessageBox.Show("Success");
                        StudentForm studentform = new StudentForm();
                        studentform.ShowDialog();


                    }
                }

            }
        }
    }
```

```xml
<Window x:Class="Coursework.MainWindow"
        xmlns="http://schemas.microsoft.com/winfx/2006/xaml/presentation"
        xmlns:x="http://schemas.microsoft.com/winfx/2006/xaml"
        Title="MainWindow" Height="350" Width="525">
    <Grid>
        <Label Content="Login Details" HorizontalAlignment="Left"
Margin="209,10,0,0" VerticalAlignment="Top"/>
        <Label Content="UserName"  Margin="47,36,395.4,0"
VerticalAlignment="Top"/>
        <Label Content="Password" HorizontalAlignment="Left" Margin="47,83,0,0"
VerticalAlignment="Top"/>
        <TextBox x:Name="Username" HorizontalAlignment="Left" Height="23"
Margin="186,36,0,0" TextWrapping="Wrap" VerticalAlignment="Top" Width="120"/>
        <PasswordBox x:Name="Password" HorizontalAlignment="Left"
Margin="186,83,0,0" VerticalAlignment="Top" Width="78"/>
        <Button Content="Login" HorizontalAlignment="Left" Margin="154,127,0,0"
VerticalAlignment="Top" Width="75" Click="Button_Click"/>


    </Grid>
</Window>
```

```csharp
using System;
using System.Collections.Generic;
using System.Data;
using System.Linq;
using System.Text;
using System.Threading.Tasks;
using System.Windows;
using System.Windows.Controls;
using System.Windows.Data;
using System.Windows.Documents;
using System.Windows.Input;
using System.Windows.Media;
using System.Windows.Media.Imaging;
using System.Windows.Shapes;
```

Latika Gurung

```csharp
namespace Coursework
{
    /// <summary>
    /// Interaction logic for Weeklyreport.xaml
    /// </summary>
    public partial class Weeklyreport : Window
    {
        public Weeklyreport()
        {
            InitializeComponent();
        }

        private void button_click(object sender, RoutedEventArgs e)
        {
            var dataset = new DataSet(); // declaring new data set
            dataset.ReadXml(@"D:\student.xml");  // reading main report
            DataTable stdReport = dataset.Tables[0];
            int total_Com = 0;    // assigning initial values of Course to
            int total_Mul = 0;
            int total_Net = 0;

            DataTable dt = new DataTable("tbl");
            dt.Columns.Add("Course Enroll", typeof(String));  // creating two
columns
            dt.Columns.Add("Total Students", typeof(int));

            for (int i = 0; i < stdReport.Rows.Count; i++)
            {


                String col = stdReport.Rows[i]["CourseEnroll"].ToString();
                if (col == "Computing")
                {
                    total_Com++;   // incrementing values of each course based
on user input
                }
                else if (col == "Multimedia")
                {
                    total_Mul++;
                }
                else if (col == "Networking")
                {
                    total_Net++;
                }
            }

            dt.Rows.Add("Computing", total_Com);          // final assign
            dt.Rows.Add("Multimedia", total_Mul);
            dt.Rows.Add("Networking", total_Net);


            WeeklyGrid.ItemsSource = dt.DefaultView; // is the name of data
grid


        }
    }
}
<Window x:Class="Coursework.Weeklyreport"
        xmlns="http://schemas.microsoft.com/winfx/2006/xaml/presentation"
```

Latika Gurung

```xml
        xmlns:x="http://schemas.microsoft.com/winfx/2006/xaml"
        Title="Weeklyreport" Height="338.209" Width="414.627">
    <Grid>
        <DataGrid x:Name="WeeklyGrid" HorizontalAlignment="Left"
Margin="33,42,0,0" VerticalAlignment="Top" Height="185" Width="348"
RenderTransformOrigin="0.5,0.5">
            <DataGrid.RenderTransform>
                <TransformGroup>
                    <ScaleTransform/>
                    <SkewTransform AngleY="0.263"/>
                    <RotateTransform/>
                    <TranslateTransform Y="0.592"/>
                </TransformGroup>
            </DataGrid.RenderTransform>
        </DataGrid>
        <Button Content="weekly report" HorizontalAlignment="Left"
Margin="144,265,0,0" VerticalAlignment="Top" Width="87" Height="22"
Click="button_click"/>

    </Grid>
</Window>
using System;
using System.Collections.Generic;
using System.Data;
using System.Linq;
using System.Text;
using System.Threading.Tasks;
using System.Windows;
using System.Windows.Controls;
using System.Windows.Controls.DataVisualization.Charting;
using System.Windows.Data;
using System.Windows.Documents;
using System.Windows.Input;
using System.Windows.Media;
using System.Windows.Media.Imaging;
using System.Windows.Shapes;

namespace Coursework
{
    /// <summary>
    /// Interaction logic for Chart.xaml
    /// </summary>
    public partial class Chart : Window
    {
        public Chart()
        {
            InitializeComponent();
            LoadPieChartData();
        }
        private void LoadPieChartData()
        {
            var dataset = new DataSet(); // declaring new data set
            dataset.ReadXml(@"D:\student.xml");  // reading main report
            DataTable stdReport = dataset.Tables[0];
            int total_Com = 0;   // assigning initial values of Course to
            int total_Mul = 0;
            int total_Net = 0;

            DataTable dt = new DataTable("tbl");
            dt.Columns.Add("Course Enroll", typeof(String));  // creating two
columns
```

Latika Gurung

```
            dt.Columns.Add("Total Students", typeof(int));

            for (int i = 0; i < stdReport.Rows.Count; i++)
            {


                String col = stdReport.Rows[i]["CourseEnroll"].ToString();
                if (col == "Computing")
                {
                    total_Com++;    // incrementing values of each course based
on user input
                }
                else if (col == "Multimedia")
                {
                    total_Mul++;
                }
                else if (col == "Networking")
                {
                    total_Net++;
                }
            }

            dt.Rows.Add("Computing", total_Com);            // final assign
            dt.Rows.Add("Multimedia", total_Mul);
            dt.Rows.Add("Networking", total_Net);
            ((PieSeries)Piechart).ItemsSource =
                new KeyValuePair<string, int>[]{
        new KeyValuePair<string,int>("Computing", total_Com),
        new KeyValuePair<string,int>("Multimedia", total_Mul),
        new KeyValuePair<string,int>("Networking", total_Net)};
        }
    }

}

<Window x:Class="Coursework.Chart"
        xmlns="http://schemas.microsoft.com/winfx/2006/xaml/presentation"
        xmlns:x="http://schemas.microsoft.com/winfx/2006/xaml"
        xmlns:DV="clr-
namespace:System.Windows.Controls.DataVisualization;assembly=System.Windows.Con
trols.DataVisualization.Toolkit"
        xmlns:DVC="clr-
namespace:System.Windows.Controls.DataVisualization.Charting;assembly=System.Wi
ndows.Controls.DataVisualization.Toolkit"

        Title="Chart" Height="300" Width="300">
    <Grid Margin="0,0,-115.4,0">
        <DVC:Chart Margin="0" Title="Pie chart"   Width="400" Height="250"
Background="LightSteelBlue">

            <DVC:PieSeries x:Name="Piechart" IndependentValueBinding="{Binding
Path=Key}" DependentValueBinding="{Binding Path=Value}">
                </DVC:PieSeries>
using System;
using System.Data;

namespace DataHandler
{
    public class Handler
    {
        public DataSet CreateDataSet()
```

Latika Gurung

```
        {
            var ds = new DataSet();
            ds.Tables.Add(CreateCourseTable());
            ds.Tables.Add(CreateStudentTable());
            ds.Tables.Add(CreateStudentReportTable());

            ForeignKeyConstraint courseWorkFK = new
ForeignKeyConstraint("courseWorkFK",
            ds.Tables["Course"].Columns["ID"],
            ds.Tables["Student"].Columns["CourseEnroll"]);
            courseWorkFK.DeleteRule = Rule.None;
            ds.Tables["Student"].Constraints.Add(courseWorkFK);
            return ds;
        }

        private DataTable CreateStudentTable()
        {
            var dt = new DataTable("Student");
            DataColumn dataColumn = new DataColumn("ID", typeof(int));
            dataColumn.AutoIncrement = true;
            dataColumn.AutoIncrementSeed = 1;
            dataColumn.AutoIncrementStep = 1;

            dt.Columns.Add(dataColumn);

            dt.Columns.Add("Name", typeof(string));
            dt.Columns.Add("Address", typeof(string));
            dt.Columns.Add("ContactNo", typeof(string));
            dt.Columns.Add("CourseEnroll", typeof(int));
            dt.Columns.Add("RegistrationDate", typeof(DateTime));
            //dt.Columns.Add("PermanentAddress", typeof(string));
            //dt.Columns.Add("ParentsName", typeof(string));
            //dt.Columns.Add("ParentsContact", typeof(string));
            //dt.Columns.Add("", typeof(string));
            //dt.Columns.Add("Address", typeof(string));
            //dt.Columns.Add("Address", typeof(string));
            //dt.Columns.Add("Address", typeof(string));

            dt.PrimaryKey = new DataColumn[] { dt.Columns["ID"] };
            return dt;
        }

        private DataTable CreateCourseTable()
        {
            var dt = new DataTable("Course");
            DataColumn dataColumn = new DataColumn("ID", typeof(int));
            dataColumn.AutoIncrement = true;
            dataColumn.AutoIncrementSeed = 1;
            dataColumn.AutoIncrementStep = 1;
            dt.Columns.Add(dataColumn);

            dt.Columns.Add("Name", typeof(string));
            dt.Columns.Add("DisplayText", typeof(string));
           // dt.Columns.Add("CourseDuration", typeof(string));

            dt.PrimaryKey = new DataColumn[] { dt.Columns["ID"] };
            return dt;
        }

        private DataTable CreateStudentReportTable()
        {
```

Latika Gurung

```csharp
            var dt = new DataTable("StudentReport");
            DataColumn dataColumn = new DataColumn("ID", typeof(int));
            dataColumn.AutoIncrement = true;
            dataColumn.AutoIncrementSeed = 1;
            dataColumn.AutoIncrementStep = 1;

            dt.Columns.Add(dataColumn);

            dt.Columns.Add("RegNo", typeof(string));
            dt.Columns.Add("Name", typeof(string));
            dt.Columns.Add("Address", typeof(string));
            dt.Columns.Add("ContactNo", typeof(string));
            dt.Columns.Add("CourseEnroll", typeof(int));
            dt.Columns.Add("RegistrationDate", typeof(DateTime));


            //dt.PrimaryKey = new DataColumn[] { dt.Columns["ID"] };
            return dt;
        }
    }
}
```