Module Code & Module Title

CS6004NP & Application Development

Assessment Weightage & Type

30% & Individual Coursework

3rd Year and 1st Semester

Name: Sujana Thapa

College ID: NP04CP4A170039

University ID: 17030752

Module Leader: Mr. Ishwor Sapkota

I confirm that I understand my coursework needs to be submitted online via Google Classroom under the relevant module page before the deadline in order for my assignment to be accepted and marked. I am fully aware that late submissions will be treated as non-submission and a marks of zero will be awarded.

Table of Contents

Table of Figure

1. Introduction

The designed system is a desktop application for Student Information System for a company. The system is designed in C# and test under various circumstances. The functions for the application is implemented as required for the coursework. The application allows the user to input the student personal detail including registration date so that the system can generate weekly enrolment report of the student. The application keeps track of the student's details, program enrol and registration date. The system also allow to import a record from .CSV file or to allow manually inputting details. Furthermore, there is a feature to retrieve the student enrol status with the student details and to generate weekly report chart showing total number of student on each program.

1.1 Current Scenario

In the present context, many companies have adapted digitalized system to record their data but there are still various company who keep record of their data in traditional system.

1.2 Proposed System

The proposed system is digitalized system which is designed to overcome the problem that the companies were facing while using traditional system. The system ensures security with the login section. The system is made user-friendly.

2. User Manual

The screenshot of the application are listed below which will illustrate how the application runs.

Figure 1: Login Screen

As the user operates the system, the initial screen will be the login page. The username and password of the system is "admin". Only the valid username and password can provide access to the system.

Figure 2: Login Screen if user directly tries to login without typing password

If the user tries to access the system without typing valid password, the system does not give the user to access the further system.
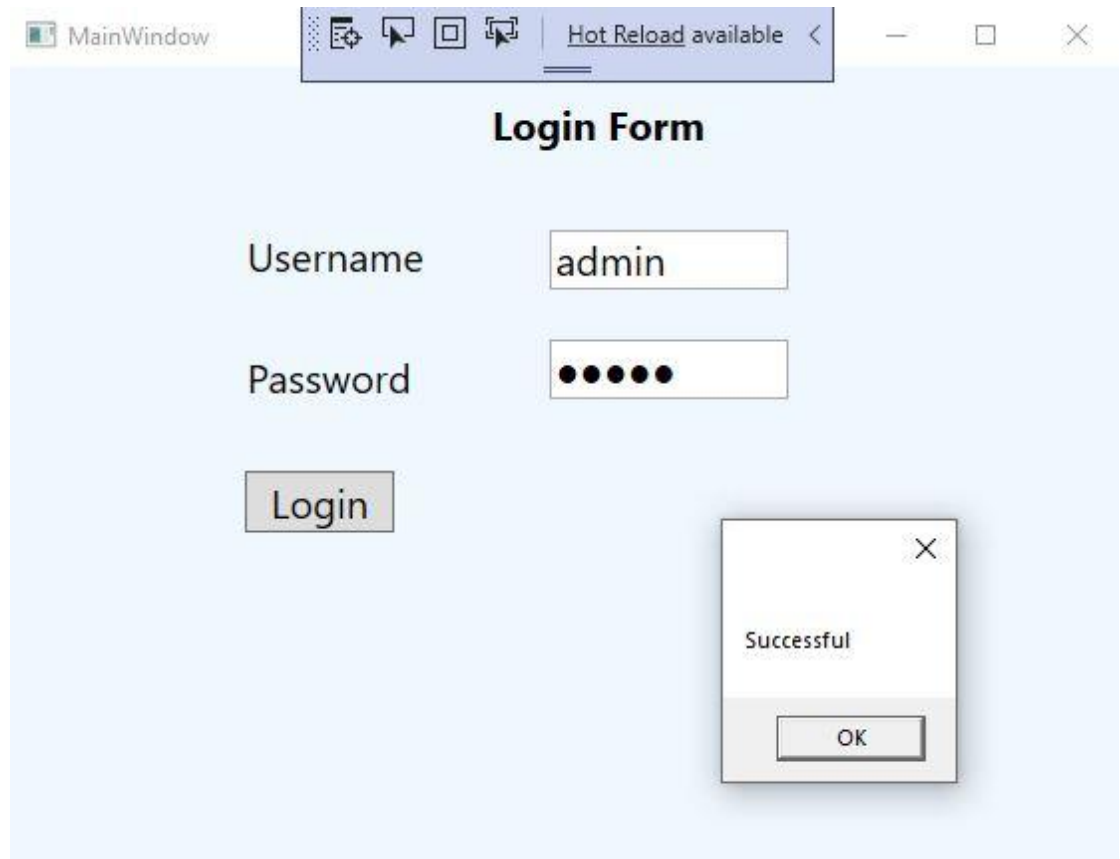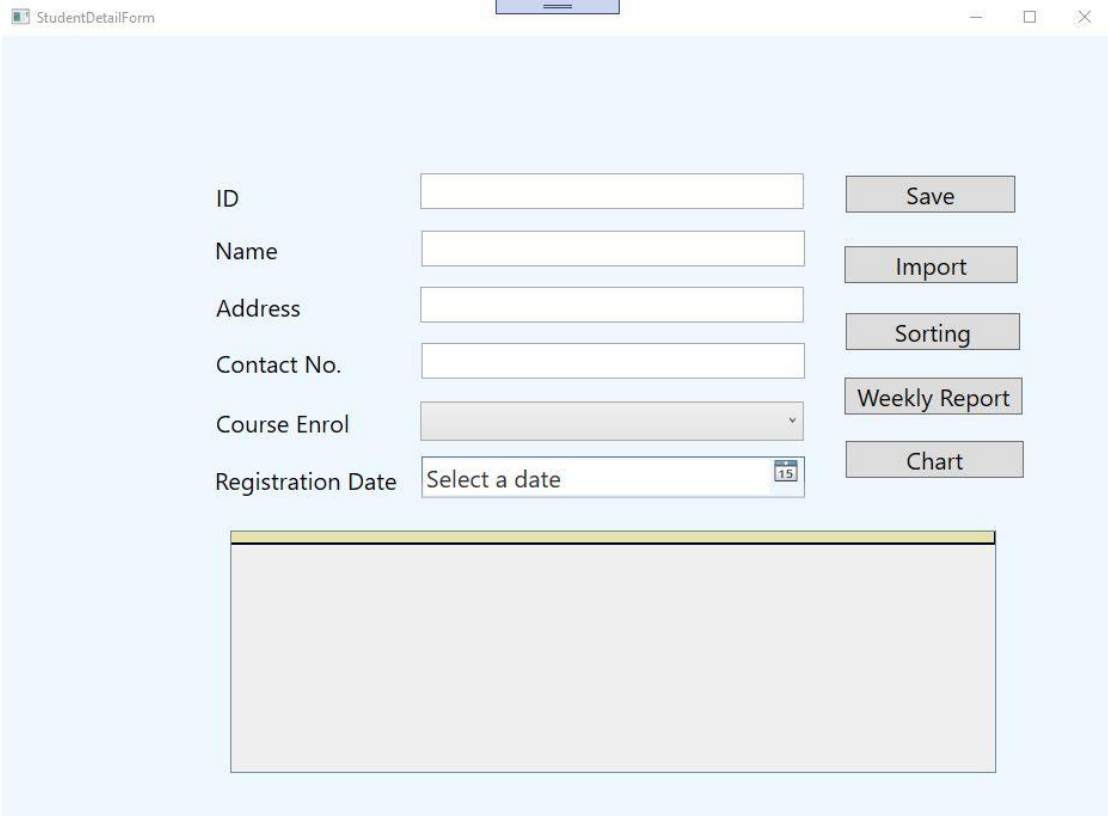
Figure 3: Login Successful

The username and password of the system is "admin". Only the valid username and password can provide access to the system.

Figure 4: Student Detail Form after successful login

After the successful login, the student detail form window of the system appears. It allows the user to fill the student's details. It also allows user to save and retrieve student's details, import a record from .CSV file, sort the student's details as per their name and registration date, generate weekly report and generate chart.

Figure 5: Student Enrolled Successfully

The system allows the feature to manually input student's detail.

Figure 6: Validating if the user fill the form or not before clicking save button

If the user tries to save the data without filling the form, the system throws a dialog box to remind to fill the form.

Figure 7: Dialog Screen for importing csv File

The system has a feature to import a .CSV file by browsing the file.

Figure 8: Data imported from CSV file

After giving the correct path, the system import the .CSV file in the datagrid.

Figure 9: Sorting Window

The system has the feature to sort the students details.

Figure 10: Unsorted Data After clicking Retrieve button

The unsorted details of students are shown in the above figure.
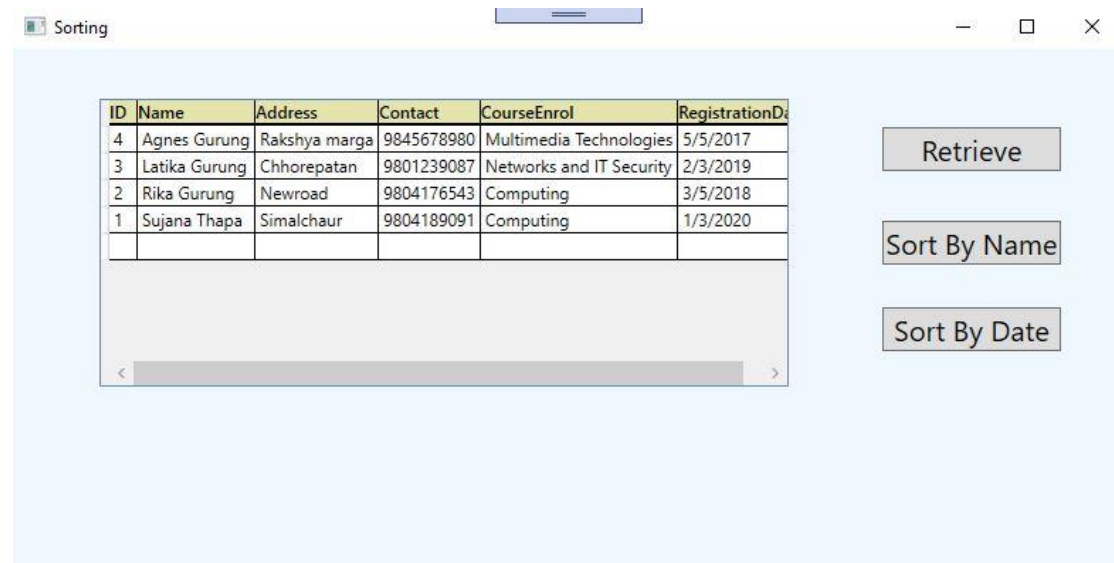
Figure 11: Sorted Data as per name

The sorted data as per the students name are listed in the above figure.
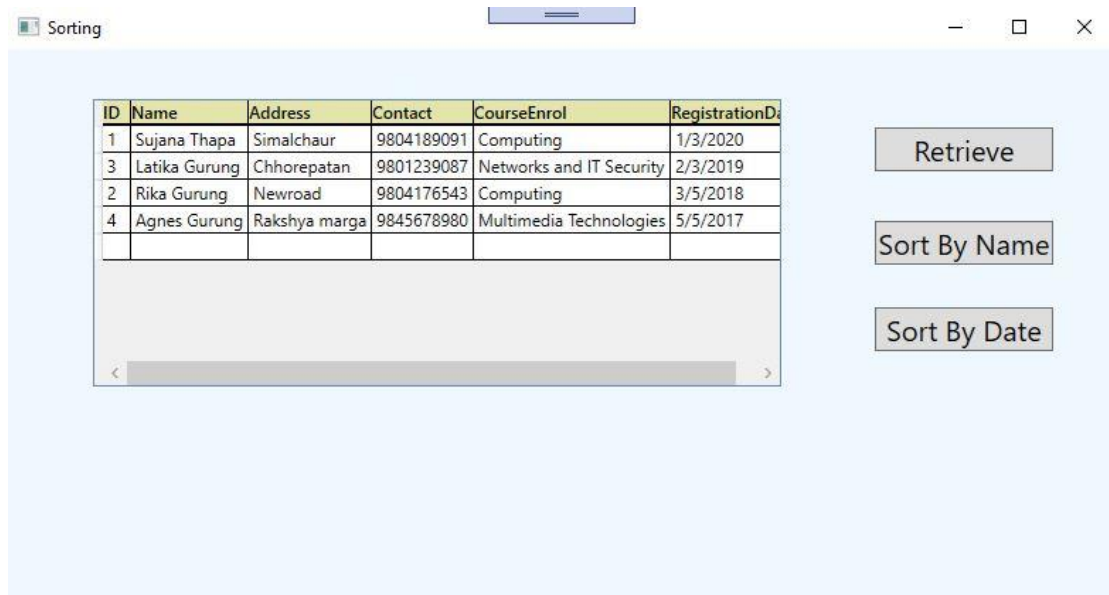
Figure 12:Sorted data as per registration date

The sorted data as per the registration date of students are listed in the above figure.
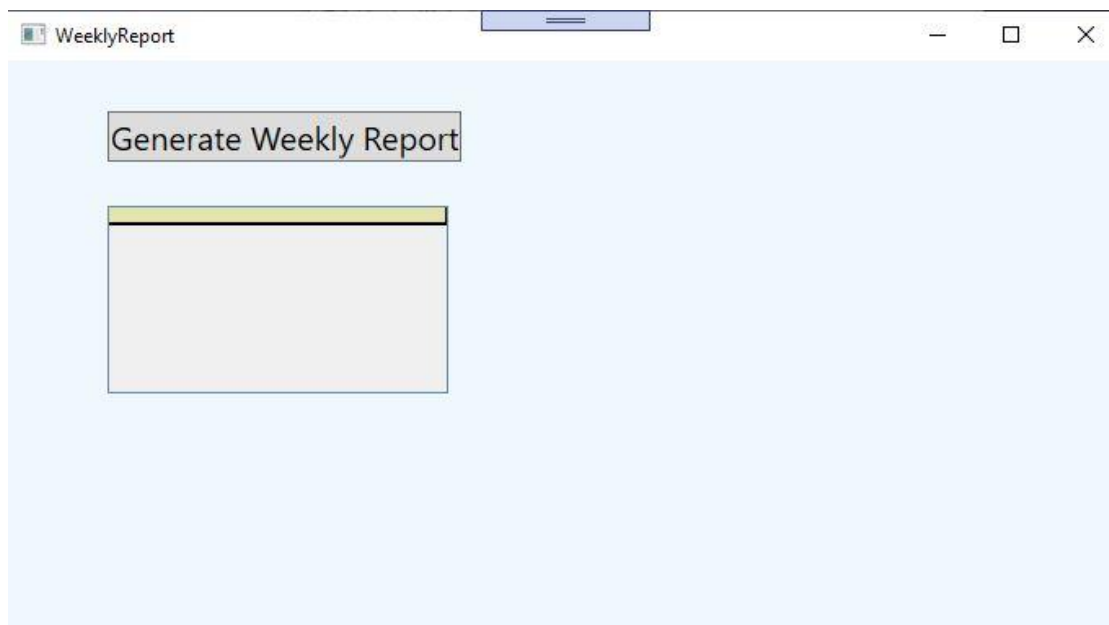
Figure 13: Weekly Report window

The weekly report window after clicking weekly report button is shown above.
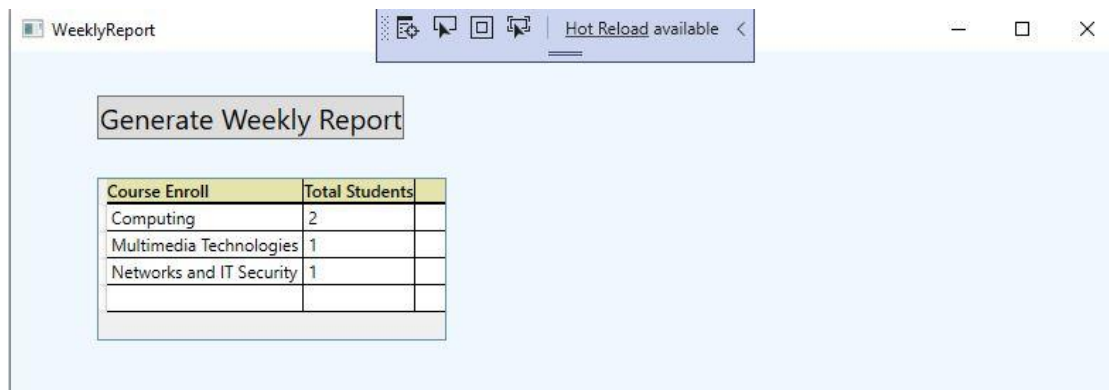
Figure 14: Generated weekly report after clicking generate weekly report

The system generate and display the weekly report.
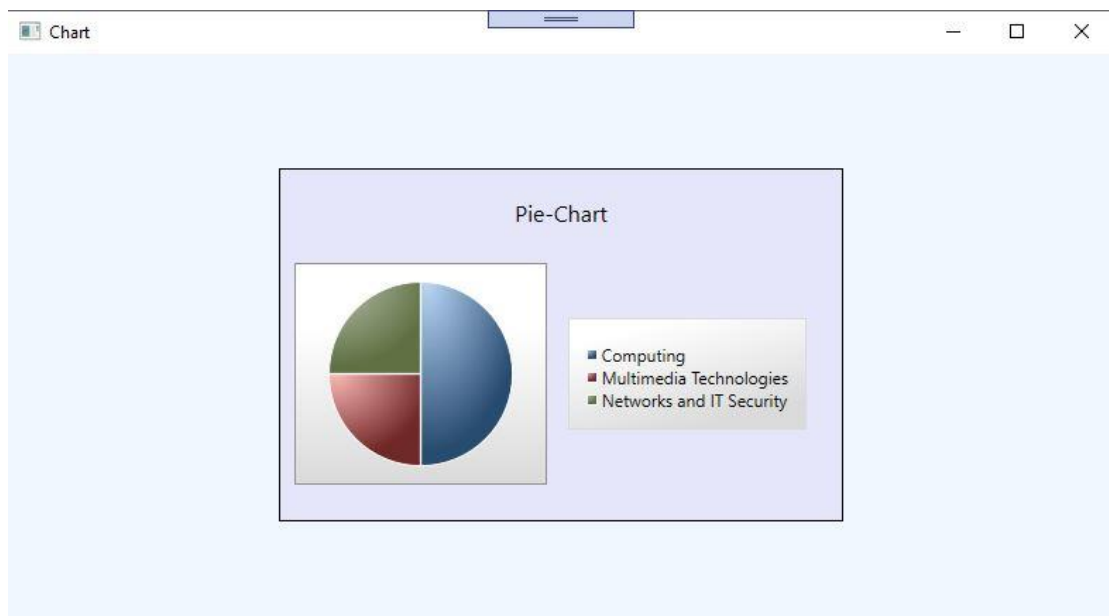
Figure 15: Pie-Chart Window after clicking chart button

The system display the chart showing total number of student on each program.

3. System Architecture

3.1 Architecture Diagram
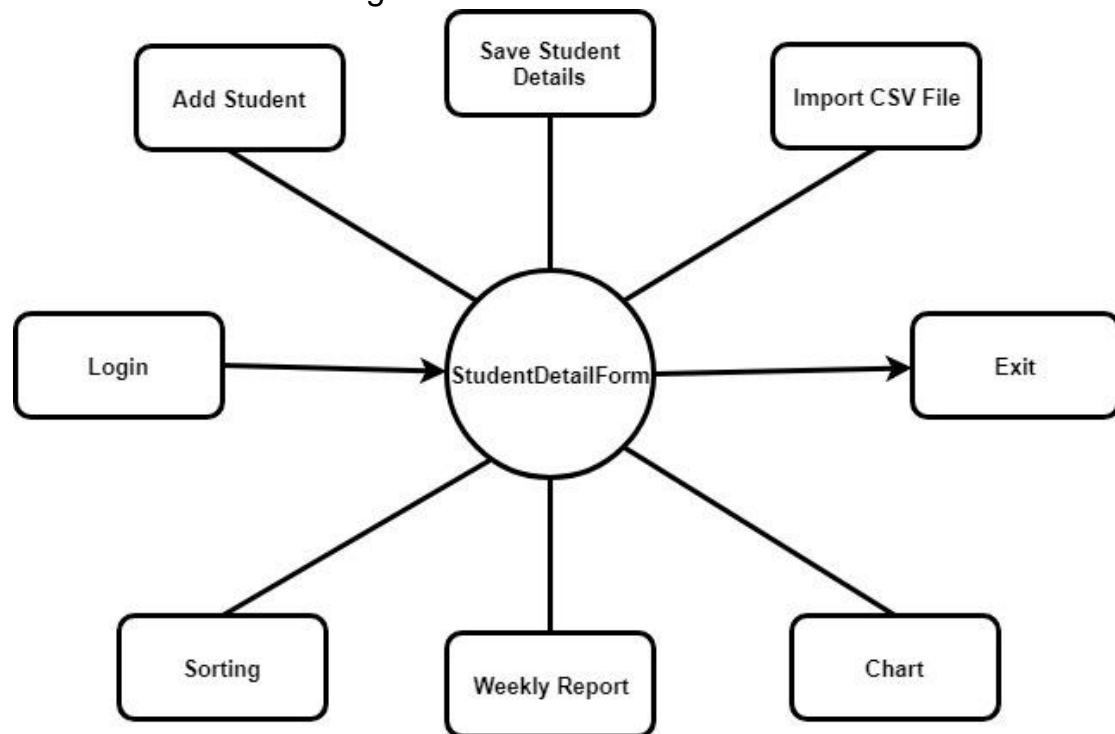
Figure 16: Architecture Diagram

3.2 Class Diagram

A class diagram is a part of a unified modelling language(UML) that defines and provides the overview and structure of a system in terms of classes, attributes, and methods and the relationships between different classes. It is used to illustrate and create a functional diagram of the classes (techopedia, 2019).
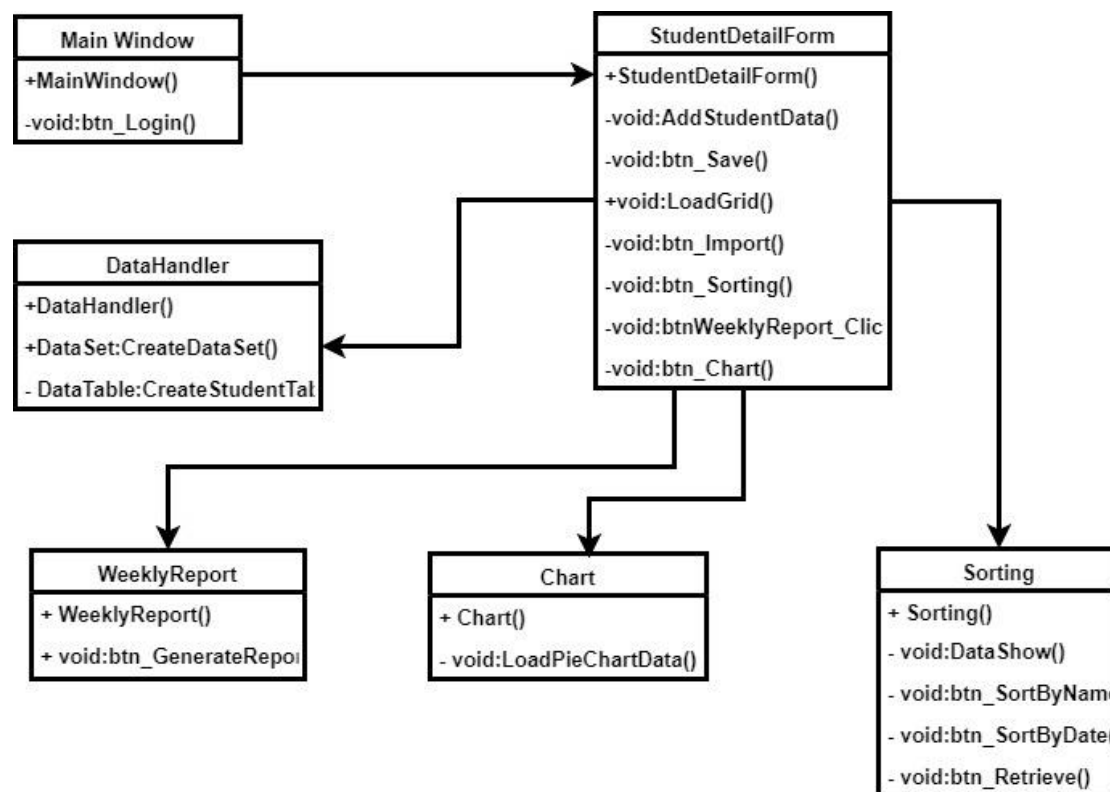
Figure 17: Class Diagram of the application

3.2 Flowchart

A flowchart is a formalized graphic representation of a logic sequence and work process. It provides a reference point when dealing with a project or process. It uses simple geometric symbols and arrows to define relationships (Margaret Rouse, August 2008).
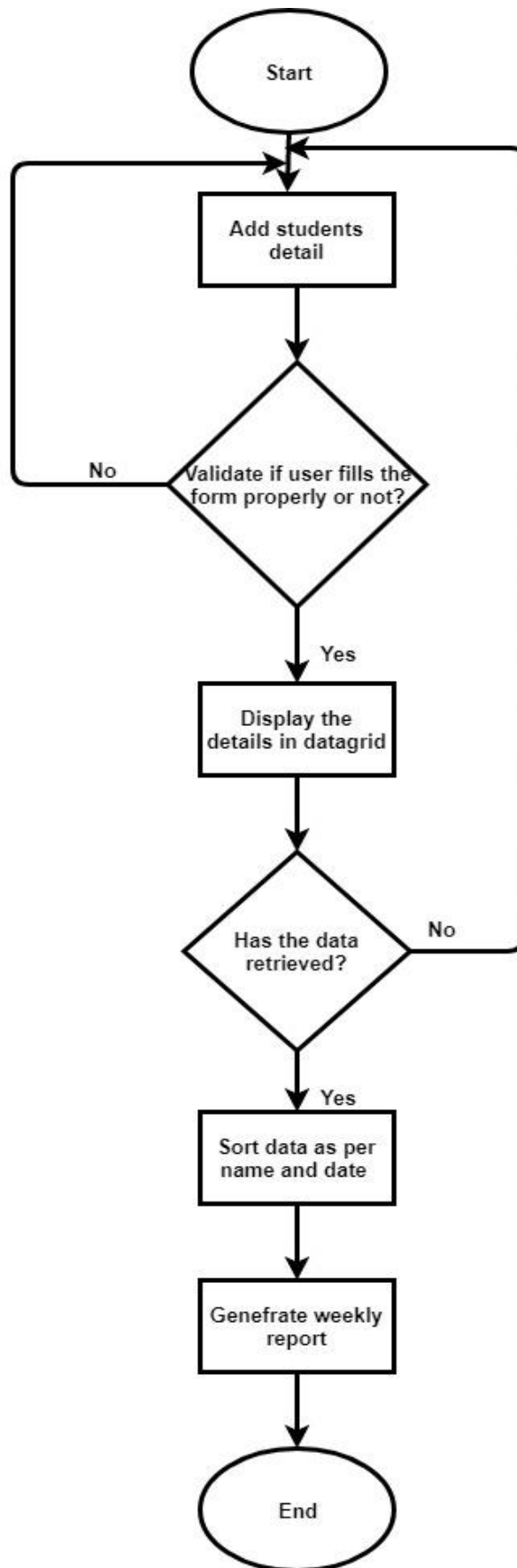
Figure 18: Flowchart of the application

4. Algorithm

Bubble sort is the simplest sorting algorithm that compares the adjacent elements and swaps their positions if they are not in the intended order. The order can be ascending or descending (Parewa Labs Pvt. Ltd, 2019). In the bubble sort technique, each of the elements in the list is compared to its adjacent element. Thus if there are n elements in list A, then A[0] is compared to A[1], A[1] is compared to A[2] and so on. After comparing if the first element is greater than the second, the two elements are swapped then (softwaretestinghelp, November 10, 2019).

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

The pictorial representation of how bubble sort will sort the given array is given below:
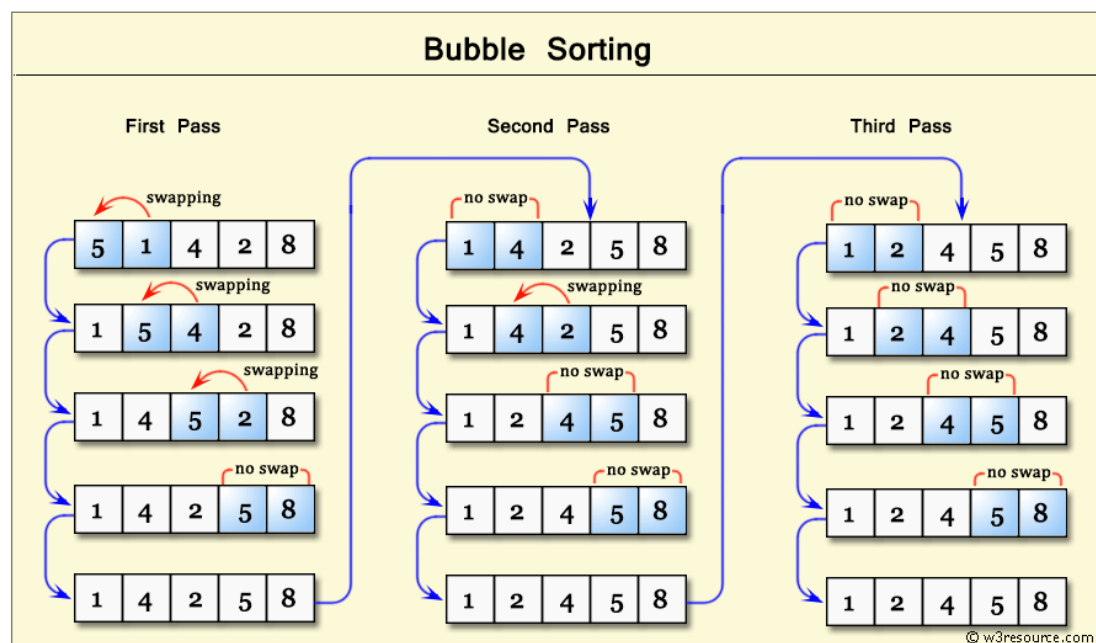
(w3resource, November 9, 2019)

Figure 19: Implementation of Bubble Sort Algorithm

5. Reflection

The project is a real-time application. The designed system is a desktop application for Student Information System for a company. The system is designed in C# and test under various circumstances. The functions for the application is implemented as required for the coursework. The application allows the user to input the student personal detail including registration date so that the system can generate weekly enrolment report of the student. The application keeps track of the student's details, program enrol and registration date.

This project has helped me enhance my skills in C#. All the tasks for project is finally completed through lots of trial and errors. Some of the tasks in the project was quite confusing. I had gone through lots of errors while doing the programs. It needed continuous practice and hard work. Lots of study and research was done on the relevant topics. The tasks done till now in the project was successfully completed at the end of the day because of the full effort that was given in each step of each question. It was difficult in the beginning but the practice of these questions were started from the first day of the release of the coursework, which helped in the successful completion of half of the project.

To sum up, though the project was to be completed and submitted on the time, it does not mean it is not useful by the time we have submitted the demo project. This project has brought a lot more improvement on us. Although it is quite difficult, all that is required is hard work and patience.

6. References

Margaret Rouse, August 2008. *whatis.techtarget.com.* [Online]
Available at: https://whatis.techtarget.com/definition/flowchart
[Accessed 28 December 2019].

Parewa Labs Pvt. Ltd, 2019. *www.programiz.com.* [Online]
Available at: https://www.programiz.com/dsa/bubble-sort
[Accessed 27 December 2019].

softwaretestinghelp, November 10, 2019. *www.softwaretestinghelp.com.* [Online]
Available at: https://www.softwaretestinghelp.com/bubble-sort/
[Accessed 27 December 2019].

techopedia, 2019. *www.techopedia.com.* [Online]
Available at: https://www.techopedia.com/definition/16466/class-diagram
[Accessed 28 December 2019].

w3resource, November 9, 2019. *www.w3resource.com.* [Online]
Available at: https://www.w3resource.com/php-exercises/searching-and-sorting-algorithm/searching-and-sorting-algorithm-exercise-6.php
[Accessed 27 December 2019].

7. Appendix

MainWindow.xaml

```xml
<Window x:Class="StudentInformationSystem.MainWindow"
        xmlns="http://schemas.microsoft.com/winfx/2006/xaml/presentation"
        xmlns:x="http://schemas.microsoft.com/winfx/2006/xaml"
        xmlns:d="http://schemas.microsoft.com/expression/blend/2008"
        xmlns:mc="http://schemas.openxmlformats.org/markup-compatibility/2006"
        xmlns:local="clr-namespace:StudentInformationSystem"
        mc:Ignorable="d"
        Title="MainWindow" Height="450" Width="575">
    <Grid Background="AliceBlue">
        <Label Content="Login Form" HorizontalAlignment="Left"
Margin="237,10,0,0" VerticalAlignment="Top" FontSize="20" FontWeight="Bold"/>
        <Label Content="Username" HorizontalAlignment="Left"
Margin="114,76,0,0" VerticalAlignment="Top" FontSize="20"/>
        <Label Content="Password" HorizontalAlignment="Left"
Margin="114,137,0,0" VerticalAlignment="Top" FontSize="20"/>
        <TextBox x:Name="txtUsername" HorizontalAlignment="Left" Height="30"
Margin="271,82,0,0" TextWrapping="Wrap" VerticalAlignment="Top" Width="120"
FontSize="20"/>
        <PasswordBox x:Name="txtPassword" HorizontalAlignment="Left"
Height="30" Margin="271,137,0,0" VerticalAlignment="Top" Width="120"
FontSize="20"/>

        <Button Content="Login" HorizontalAlignment="Left" Margin="118,203,0,0"
VerticalAlignment="Top" Width="75" FontSize="20" Click="btn_Login"/>
    </Grid>
</Window>
```

MainWindow.xaml.cs

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

```csharp
        InitializeComponent();
    }

    private void btn_Login(object sender, RoutedEventArgs e)
    {
        string user, pass;
        user = txtUsername.Text;
        pass = txtPassword.Password;
        if(user=="admin"&& pass=="admin")
        {
            MessageBox.Show("Successful");
            StudentDetailForm studentDetailForm = new StudentDetailForm();
            this.Hide();
            studentDetailForm.ShowDialog();
        }
        else
        {
            MessageBox.Show("Error");
        }
    }
}
}
```

StudentDetailForm.xaml

```xml
<Window x:Class="StudentInformationSystem.StudentDetailForm"
        xmlns="http://schemas.microsoft.com/winfx/2006/xaml/presentation"
        xmlns:x="http://schemas.microsoft.com/winfx/2006/xaml"
        xmlns:d="http://schemas.microsoft.com/expression/blend/2008"
        xmlns:mc="http://schemas.openxmlformats.org/markup-compatibility/2006"
        xmlns:local="clr-namespace:StudentInformationSystem"
        mc:Ignorable="d"
        WindowStartupLocation="CenterScreen"
        Title="StudentDetailForm" Height="700" Width="944.5">
    <Grid Margin="0,0,-44,-1" Background="AliceBlue">
        <Grid.ColumnDefinitions>
            <ColumnDefinition/>
        </Grid.ColumnDefinitions>
        <Label Content="Name" HorizontalAlignment="Left" Margin="173,160,0,0"
VerticalAlignment="Top" FontSize="20"/>
        <Label Content="Address" HorizontalAlignment="Left"
Margin="174,208,0,0" VerticalAlignment="Top" FontSize="20"
RenderTransformOrigin="0.448,1.858"/>
        <Label Content="Contact No." HorizontalAlignment="Left"
Margin="174,255,0,0" VerticalAlignment="Top" FontSize="20"
RenderTransformOrigin="0.448,1.858"/>
        <Label Content="Course Enrol" HorizontalAlignment="Left"
Margin="174,305,0,0" VerticalAlignment="Top" FontSize="20"
RenderTransformOrigin="0.448,1.858"/>
        <Label Content="Registration Date" HorizontalAlignment="Left"
Margin="173,353,0,0" VerticalAlignment="Top" FontSize="20"
RenderTransformOrigin="0.448,1.858"/>
        <Label Content="ID" HorizontalAlignment="Left" Margin="174,116,0,0"
VerticalAlignment="Top" FontSize="20" RenderTransformOrigin="0.512,-0.784"/>
```

```xml
        <TextBox x:Name="txtStdID" HorizontalAlignment="Left" Height="30"
Margin="350,115,0,0" TextWrapping="Wrap" VerticalAlignment="Top" Width="321"
FontSize="20"/>
        <TextBox x:Name="txtStdName" HorizontalAlignment="Left" Height="30"
Margin="351,163,0,0" TextWrapping="Wrap" VerticalAlignment="Top" Width="321"
FontSize="20"/>
        <TextBox x:Name="txtStdAdd" HorizontalAlignment="Left" Height="30"
Margin="350,210,0,0" TextWrapping="Wrap" VerticalAlignment="Top" Width="321"
FontSize="20"/>
        <TextBox x:Name="txtStdCont" HorizontalAlignment="Left" Height="30"
Margin="351,257,0,0" TextWrapping="Wrap" VerticalAlignment="Top" Width="321"
FontSize="20"/>
        <ComboBox x:Name="courseEnrol"  HorizontalAlignment="Left"
Margin="350,306,0,0" VerticalAlignment="Top" Width="321" FontSize="20">
            <ComboBoxItem Content="Computing"/>
            <ComboBoxItem Content="Networks and IT Security"/>
            <ComboBoxItem Content="Multimedia Technologies"/>
        </ComboBox>
        <DatePicker x:Name="registrationDate" HorizontalAlignment="Left"
Margin="351,352,0,0" VerticalAlignment="Top" FontSize="20" Width="321"/>
        <Button x:Name="btnSave" Content="Save" HorizontalAlignment="Left"
Margin="706,117,0,0" VerticalAlignment="Top" Width="142" FontSize="20"
Click="btn_Save"/>
        <Button x:Name="btnImport" Content="Import" HorizontalAlignment="Left"
Margin="705,176,0,0" VerticalAlignment="Top" Width="145" FontSize="20"
Click="btn_Import" RenderTransformOrigin="2.84,-4.806"/>
        <DataGrid x:Name="DataGridXAML"  Height="203" Margin="191,414,141,0"
VerticalAlignment="Top">
            <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 x:Name="btnSorting" Content="Sorting"
HorizontalAlignment="Left" Margin="706,232,0,0" VerticalAlignment="Top"
Width="146" FontSize="20" Click="btn_Sorting"/>
        <Button x:Name="btnWeeklyReport" Content="Weekly Report"
HorizontalAlignment="Left" Margin="705,286,0,0" VerticalAlignment="Top"
Width="149" FontSize="20" Click="btnWeeklyReport_Click" />
        <Button x:Name="btnChart" Content="Chart" HorizontalAlignment="Left"
Margin="706,339,0,0" VerticalAlignment="Top" Width="149" FontSize="20"
Click="btn_Chart" />
    </Grid>
</Window>
```

StudentDetailForm.xaml.cs

```csharp
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.Shapes;
using System.Data;
using System.IO;
using Microsoft.Win32;

namespace StudentInformationSystem
{
    /// <summary>
    /// Interaction logic for StudentDetailForm.xaml
    /// </summary>
    public partial class StudentDetailForm : Window
    {
        public StudentDetailForm()
        {
            InitializeComponent();

            //add new student

            LoadGrid();
        }

        public class Student
        {
            public string ID { get; set; }
            public string Name { get; set; }
            public string Address { get; set; }
            public string Contact { get; set; }
            public string CourseEnrol { get; set; }
            public string RegistrationDate { get; set; }
        }

        private void AddStudentData(DataSet dataSet)
        {
            var newRow = dataSet.Tables["Student"].NewRow();
            newRow["ID"] = txtStdID.Text;
            newRow["Name"] = txtStdName.Text;
            newRow["Address"] = txtStdAdd.Text;
            newRow["Contact"] = txtStdCont.Text;
            newRow["CourseEnrol"] = courseEnrol.Text;
            newRow["RegistrationDate"] = registrationDate.SelectedDate;
            dataSet.Tables["Student"].Rows.Add(newRow);
        }


        private void btn_Save(object sender, RoutedEventArgs e)
        {
```

```csharp
        if (txtStdID.Text == "" || txtStdName.Text == "" || txtStdAdd.Text
== "" || txtStdCont.Text == "" || courseEnrol.Text == "" ||
registrationDate.Text == "")
        {
            MessageBox.Show("Empty");
        }
        else
        {
            try
            {
                var handler = new DataHandler();
                var dataSet = new DataSet();

                if (File.Exists(@"D:\student.xml"))
                {
                    dataSet.ReadXml(@"D:\student.xml");
                    //dataSet.WriteXml(@"D:\Data\student.xml");
                }
                else
                {
                    dataSet = handler.CreateDataSet();
                }
                AddStudentData(dataSet);
                dataSet.WriteXml(@"D:\student.xml");
                LoadGrid();

                MessageBox.Show("Student Added Succcessfully");
            }
            catch (Exception)
            {

            }
        }


    }

    public void LoadGrid()
    {
        var dataSet = new DataSet();

        if (File.Exists(@"D:\student.xml"))
        {
            dataSet.ReadXml(@"D:\student.xml");
            DataGridXAML.ItemsSource =
dataSet.Tables["Student"].DefaultView;
        }
    }

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

```csharp
                    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["CourseEnrol"] = values[4];
                        newRow["RegistrationDate"] = values[5];
                        dataSet.Tables["Student"].Rows.Add(newRow);

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

                    }
                }
                DataGridXAML.ItemsSource =
dataSet.Tables["Student"].DefaultView;
            }
        }

        private void btn_Sorting(object sender, RoutedEventArgs e)
        {
            Sorting sorting = new Sorting();
            sorting.ShowDialog();
        }

        private void btnWeeklyReport_Click(object sender, RoutedEventArgs e)
        {
            WeeklyReport weelkyReport = new WeeklyReport();
            weelkyReport.ShowDialog();
        }

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

DataHandler.cs

```csharp
using System;
using System.Collections.Generic;
using System.Linq;
using System.Text;
using System.Threading.Tasks;
using System.Data;

namespace StudentInformationSystem
{
    class DataHandler
    {
        public DataSet CreateDataSet()
        {
            var ds = new DataSet();
```

```
                ds.Tables.Add(CreateStudentTable());
                return ds;
            }

        private DataTable CreateStudentTable()
        {
            var dt = new DataTable("Student");
            dt.Columns.Add("ID", typeof(string));
            dt.Columns.Add("Name", typeof(string));
            dt.Columns.Add("Address", typeof(string));
            dt.Columns.Add("Contact", typeof(string));
            dt.Columns.Add("CourseEnrol", typeof(string));
            dt.Columns.Add("RegistrationDate", typeof(DateTime));
            return dt;
        }
    }
}
```

Sorting.xaml

```xml
<Window x:Class="StudentInformationSystem.Sorting"
        xmlns="http://schemas.microsoft.com/winfx/2006/xaml/presentation"
        xmlns:x="http://schemas.microsoft.com/winfx/2006/xaml"
        xmlns:d="http://schemas.microsoft.com/expression/blend/2008"
        xmlns:mc="http://schemas.openxmlformats.org/markup-compatibility/2006"
        xmlns:local="clr-namespace:StudentInformationSystem"
        mc:Ignorable="d"
        Title="Sorting" Height="450" Width="800">
    <Grid Background="AliceBlue">
        <DataGrid x:Name="DataGridSorting"  Height="203" Margin="61,35,237,0"
VerticalAlignment="Top">
            <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 x:Name="btnRetrieve" Content="Retrieve"
HorizontalAlignment="Left" Margin="613,55,0,0" VerticalAlignment="Top"
Width="126" FontSize="20" Click="btn_Retrieve" />
        <Button x:Name="btnSortByName" Content="Sort By Name"
HorizontalAlignment="Left" Margin="613,121,0,0" VerticalAlignment="Top"
Width="126" FontSize="20" Click="btn_SortByName" />
        <Button x:Name="btnSortByDate" Content="Sort By Date"
HorizontalAlignment="Left" Margin="613,182,0,0" VerticalAlignment="Top"
Width="126" FontSize="20" Click="btn_SortByDate"/>
    </Grid>
</Window>
```

Sorting.xaml.cs

```
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;

namespace StudentInformationSystem
{
    /// <summary>
    /// Interaction logic for Sorting.xaml
    /// </summary>
    public partial class Sorting : Window
    {
        DataTable buffer;

        public Sorting()
        {
            InitializeComponent();
        }

        private void DataShow()
        {
            string dataXMLFile = @"D:\student.xml";
            System.Data.DataSet dataset = new DataSet();
            dataset.ReadXml(dataXMLFile);

            buffer = new DataTable("dt");
            buffer.Columns.Add("ID", typeof(String));
            buffer.Columns.Add("Name", typeof(String));
            buffer.Columns.Add("Address", typeof(String));
            buffer.Columns.Add("Contact", typeof(String));
            buffer.Columns.Add("CourseEnrol", typeof(String));
            buffer.Columns.Add("RegistrationDate", typeof(String));

            for (int i = 0; i < dataset.Tables[0].Rows.Count; i++)
            {
                string s = dataset.Tables[0].Rows[i][5].ToString();
                DateTime dtime = DateTime.Parse(s);
                buffer.Rows.Add(
                    dataset.Tables[0].Rows[i][0].ToString(),
                    dataset.Tables[0].Rows[i][1].ToString(),
                    dataset.Tables[0].Rows[i][2].ToString(),
                    dataset.Tables[0].Rows[i][3].ToString(),
                    dataset.Tables[0].Rows[i][4].ToString(),
                    dtime.ToShortDateString());
            }
            DataView dataView = new DataView(buffer);
            DataGridSorting.ItemsSource = dataView;
        }

        private void btn_SortByName(object sender, RoutedEventArgs e)
```

```
        {
            DataView dataView = new DataView(buffer);
            dataView.Sort = "Name ASC";
            DataGridSorting.ItemsSource = dataView;
        }

        private void btn_SortByDate(object sender, RoutedEventArgs e)
        {
            DataView dataView = new DataView(buffer);
            dataView.Sort = "RegistrationDate ASC";
            DataGridSorting.ItemsSource = dataView;
        }

        private void btn_Retrieve(object sender, RoutedEventArgs e)
        {
            DataShow();
        }
    }
}
```

WeeklyReport.xaml

```xml
<Window x:Class="StudentInformationSystem.WeeklyReport"
        xmlns="http://schemas.microsoft.com/winfx/2006/xaml/presentation"
        xmlns:x="http://schemas.microsoft.com/winfx/2006/xaml"
        xmlns:d="http://schemas.microsoft.com/expression/blend/2008"
        xmlns:mc="http://schemas.openxmlformats.org/markup-compatibility/2006"
        xmlns:local="clr-namespace:StudentInformationSystem"
        mc:Ignorable="d"
        Title="WeeklyReport" Height="450" Width="700">
    <Grid Background="AliceBlue">
        <DataGrid x:Name="DataGridWeeklyReport"  Height="115"
Margin="61,89,414,0" VerticalAlignment="Top">
            <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 x:Name="btnGenerateReport" Content="Generate Weekly Report"
HorizontalAlignment="Left" Margin="61,31,0,0" VerticalAlignment="Top"
Width="217" FontSize="20" Click="btn_GenerateReport" />
    </Grid>
</Window>
```

WeeklyReport.xaml.cs

```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;

namespace StudentInformationSystem
{
    /// <summary>
    /// Interaction logic for WeelkyReport.xaml
    /// </summary>
    public partial class WeeklyReport : Window
    {
        public WeeklyReport()
        {
            InitializeComponent();

        }

        private void btn_GenerateReport(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]["CourseEnrol"].ToString();
                if (col == "Computing")
                {
                    total_Com++;   // incrementing values of each course based
on user input
                }
                else if (col == "Multimedia Technologies")
                {
                    total_Mul++;
                }
                else if (col == "Networks and IT Security")
                {
                    total_Net++;
                }
            }
```

```
            dt.Rows.Add("Computing", total_Com);              // final assign
            dt.Rows.Add("Multimedia Technologies", total_Mul);
            dt.Rows.Add("Networks and IT Security", total_Net);

            DataGridWeeklyReport.ItemsSource = dt.DefaultView; // is the name
of data grid
        }
    }
}
```

Chart.xaml

```xml
<Window x:Class="StudentInformationSystem.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"
        xmlns:local="clr-namespace:StudentInformationSystem"

        Title="Chart" Height="450" Width="800">
    <Grid Background="AliceBlue">
        <DVC:Chart Margin="0" Title="Pie-Chart" Width="400" Height="250"
Background="Lavender">
            <DVC:PieSeries    x:Name="PieChart"
IndependentValueBinding="{Binding Path=Key}"   DependentValueBinding="{Binding
Path=Value}">
            </DVC:PieSeries>
        </DVC:Chart>
    </Grid>
</Window>
```

Chart.xaml.cs

```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;

namespace StudentInformationSystem
```

```csharp
{
    /// <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
            dt.Columns.Add("Total Students", typeof(int));

            for (int i = 0; i < stdReport.Rows.Count; i++)
            {
                String col = stdReport.Rows[i]["CourseEnrol"].ToString();
                if (col == "Computing")
                {
                    total_Com++;   // incrementing values of each course based
on user input
                }
                else if (col == "Multimedia Technologies")
                {
                    total_Mul++;
                }
                else if (col == "Networks and IT Security")
                {
                    total_Net++;
                }
            }

            dt.Rows.Add("Computing", total_Com);          // final assign
            dt.Rows.Add("Multimedia Technologies", total_Mul);
            dt.Rows.Add("Networks and IT Security", total_Net);

((System.Windows.Controls.DataVisualization.Charting.PieSeries)PieChart).ItemsS
ource =
                new KeyValuePair<string, int>[]{
        new KeyValuePair<string,int>("Computing", total_Com),
        new KeyValuePair<string,int>("Multimedia Technologies", total_Mul),
        new KeyValuePair<string,int>("Networks and IT Security", total_Net) };
        }
    }
}
```

student.xml

```xml
<?xml version="1.0" standalone="yes"?>
<NewDataSet>
 <Student>
  <ID>1</ID>
  <Name>Sujana Thapa</Name>
  <Address>Simalchaur</Address>
  <Contact>9804189091</Contact>
  <CourseEnrol>Computing</CourseEnrol>
  <RegistrationDate>2020-01-03T00:00:00+05:45</RegistrationDate>
 </Student>
 <Student>
  <ID>2</ID>
  <Name>Rika Gurung</Name>
  <Address>Newroad</Address>
  <Contact>9804176543</Contact>
  <CourseEnrol>Computing</CourseEnrol>
  <RegistrationDate>3/5/2018</RegistrationDate>
 </Student>
 <Student>
  <ID>3</ID>
  <Name>Latika Gurung</Name>
  <Address>Chhorepatan</Address>
  <Contact>9801239087</Contact>
  <CourseEnrol>Networks and IT Security</CourseEnrol>
  <RegistrationDate>2/3/2019</RegistrationDate>
 </Student>
 <Student>
  <ID>4</ID>
  <Name>Agnes Gurung</Name>
  <Address>Rakshya marga</Address>
```

```
  <Contact>9845678980</Contact>

  <CourseEnrol>Multimedia Technologies</CourseEnrol>

  <RegistrationDate>5/5/2017</RegistrationDate>

 </Student>

</NewDataSet>
```