Marking Scheme

Informatics College Pokhara

informatics
college pokhara

Application Development

CS6004NI

Course Work 1

Submitted By: **Sneha B.K.**
London Met ID: Enter ID Here

Submitted To: Ishwor Sapkota
Module Leader

Component Grade and Comments	
A. Implementation of Application	
User Interface and proper controls used for designing	User Interface is complete but not separated and have proper use of controls
Manual data entry or import from csv	appropriate use of data types but missing some properties required or missing CRUD operation
Data Validation	Only basic 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	very poorly explained.
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 poor code
System Usability	very poorly developed application

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

Module Code & Module Title

CS6004NP Application Development

Assessment Weightage & Type

30% Individual Coursework

Year and Semester

2020 L3C2

Name: Sneha B.K

College ID: 17031923

University ID: NP04CP4S180021

Table of Contents

Table of Figure

1. Introduction

In the last 30 years, Technology has grown in ways we never could've imagined and has an impact on just about every area of our everyday lives. No generation has felt the impact of the technological advances like this generation. Technology practically runs our lives, with apps for everything imaginable, even some accessible to young children. Each and every member of society, regardless of age, is now connected to the world around us because of technology. It has very vital role in education too. Likewise, the oldest way of keeping the record of student's information has been turned into digitalized form.

This project is an desktop application that keep track of the student's details, program enrol and registration date. To input the details manually we have to import a csv file. And we need to generate and display two different reports such as listing the student details and other sorting by students first name and date of registration. And after that we had to display weekly tabular report showing total number of students enrolled so far in each program offered by the institution.

1.1 Current Scenario

As per talking about the scenario, most of the colleges are still using traditional method (record in register) of record keeping which are unsafe and risky to maintain it for long period of time. Using the traditional method, the total number of students, enrolled course and weekly report are also difficult to generate.

1.2 Proposed System

The proposed software can be used in colleges to keep record of the new enrolled students. The main objective of the developed system is to keep track of the new students as well as old too, generate weekly report, can sort students by the time duration of the registration. The system is simple and easy to use and a user with little computer knowledge and training can run it.

2. User Manual

The detailed information to run the program along with proper screenshot is as below:

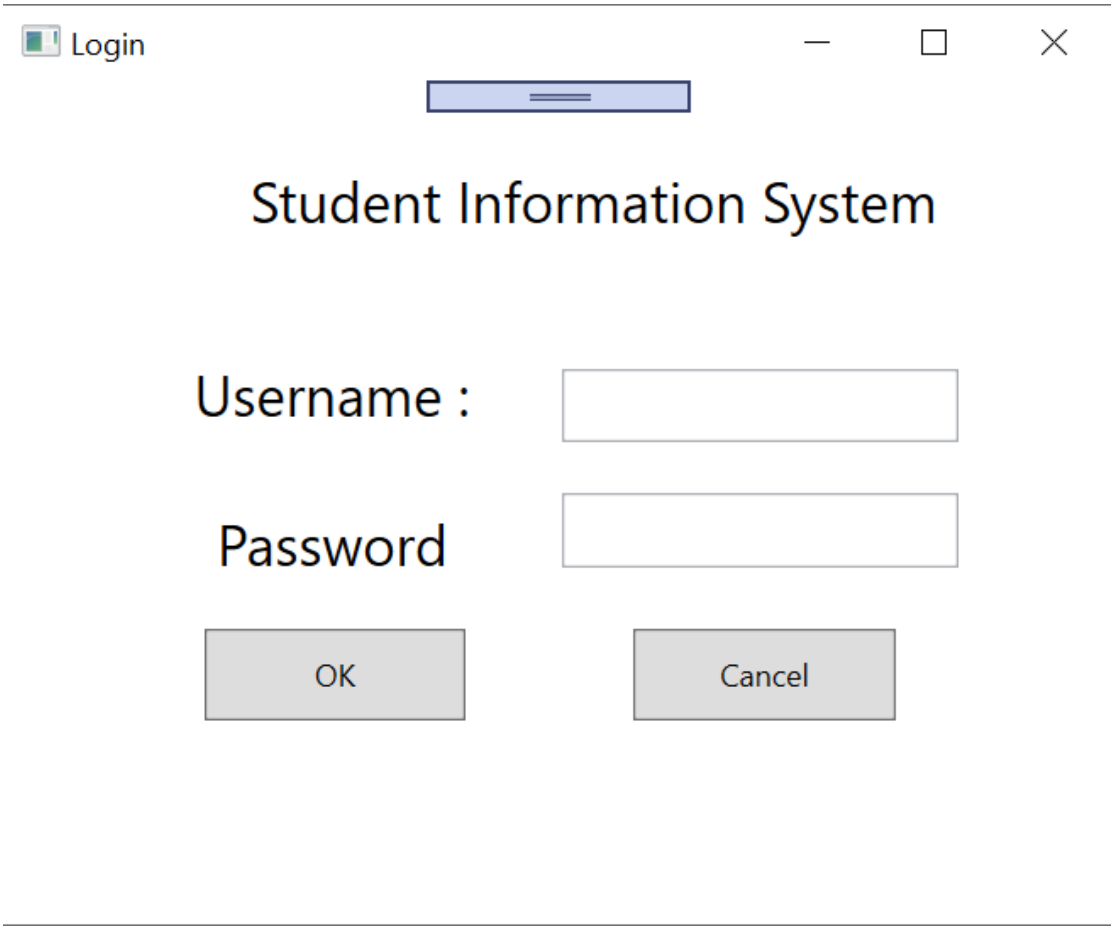

Figure 1: Login for Student Information System

- Main Page

After logging into the system with correct credentials, the main window will open where the employee had to input all the data that are required. There are 5 buttons where the employee can import students details and can save it.

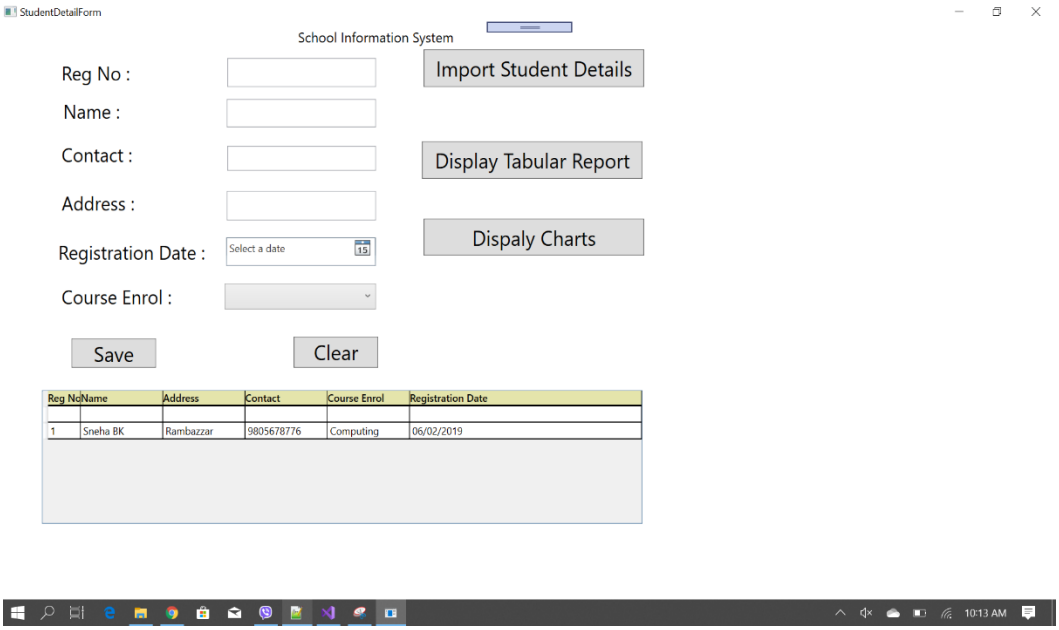

Figure 2: Student Detail Form

Data are displayed in the Excel as shown below:

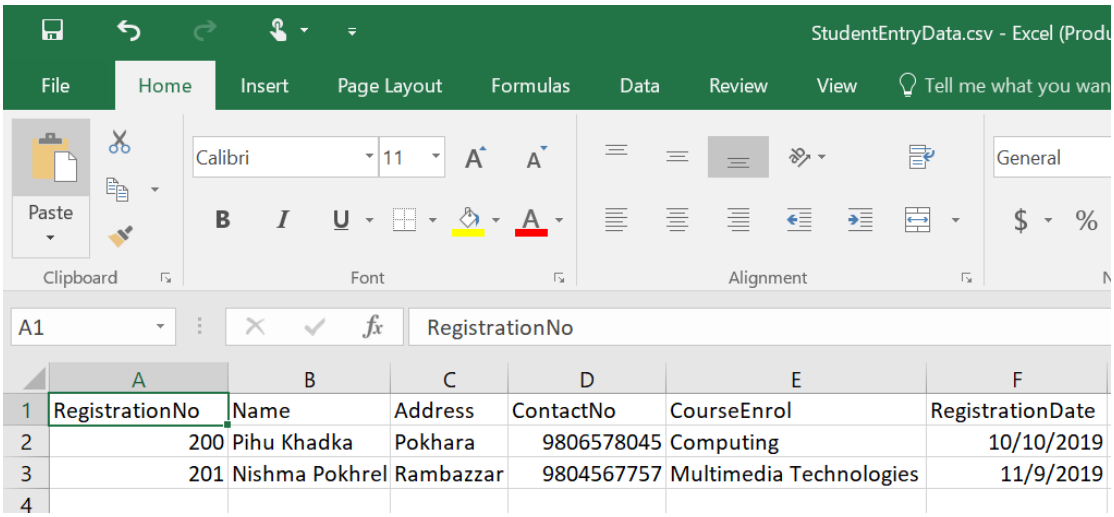

Figure 3: CSV data in Microsoft Excel

A user can import the data from CSV file into the grid as below.

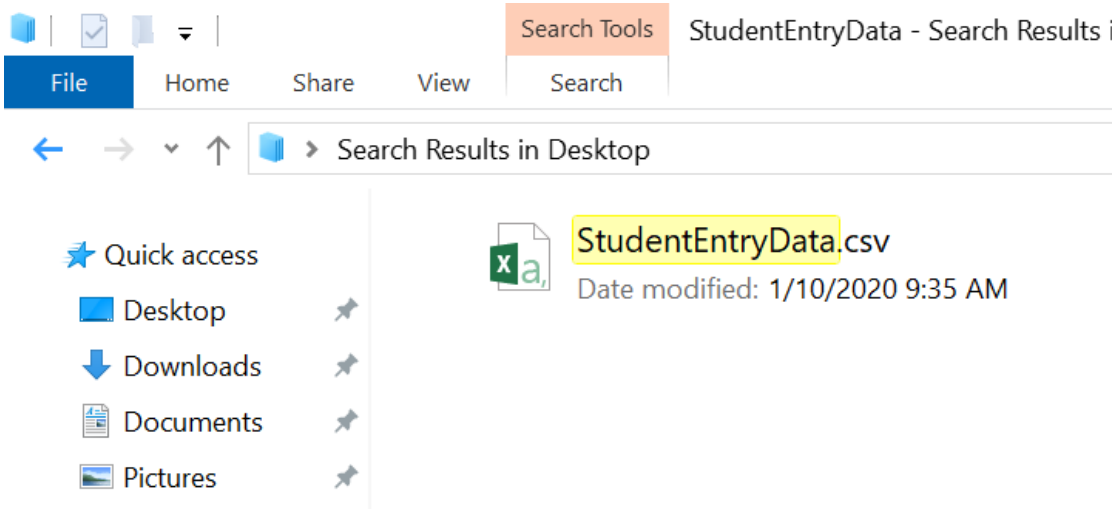

Figure 4: Import from CSV Dialog

Chart Button

Once the display chart button is clicked on, it displays a chart. This chart shows the total no. of students in terms of course enrolled.

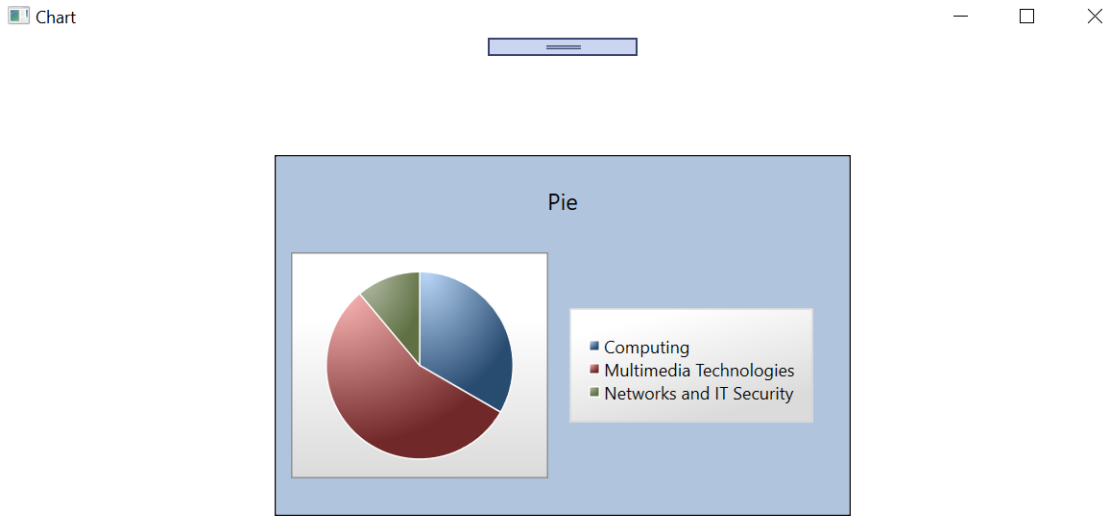

Figure 5: Chart

3. Flowchart

The above figure represents the flowchart of the developed system. At first, user needs to login to the system for which the user needs to input the correct credentials. After logging into the system with correct credentials, the system will display the main form which is the main panel of the developed system. Using the menu stripe on the main form, the user can record the student name, contact number, address course enrol. Moreover, the user can generate the report of the student in pie chart.

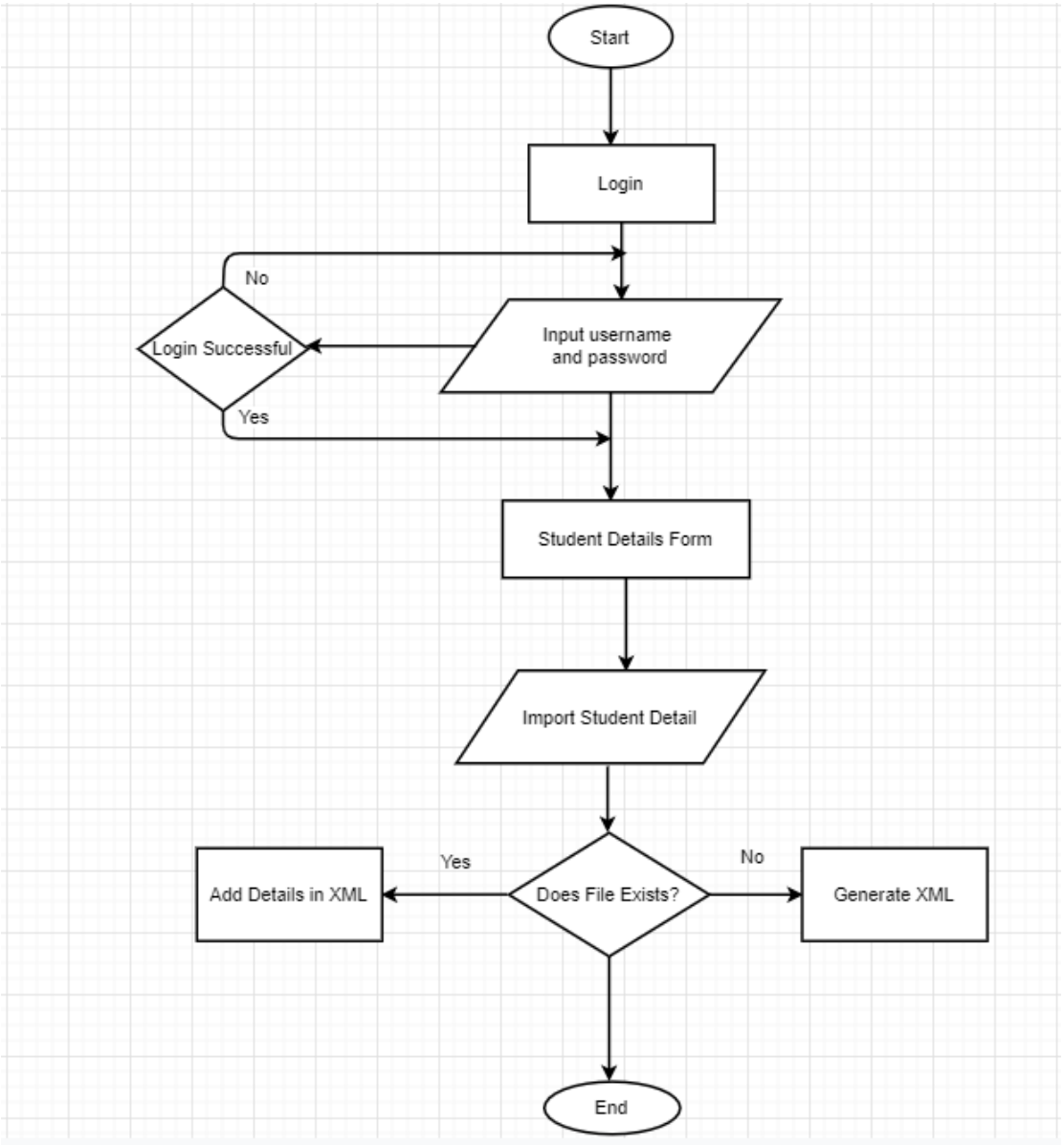

Figure 6: Flow chart of Student Information System

4. Class Diagram

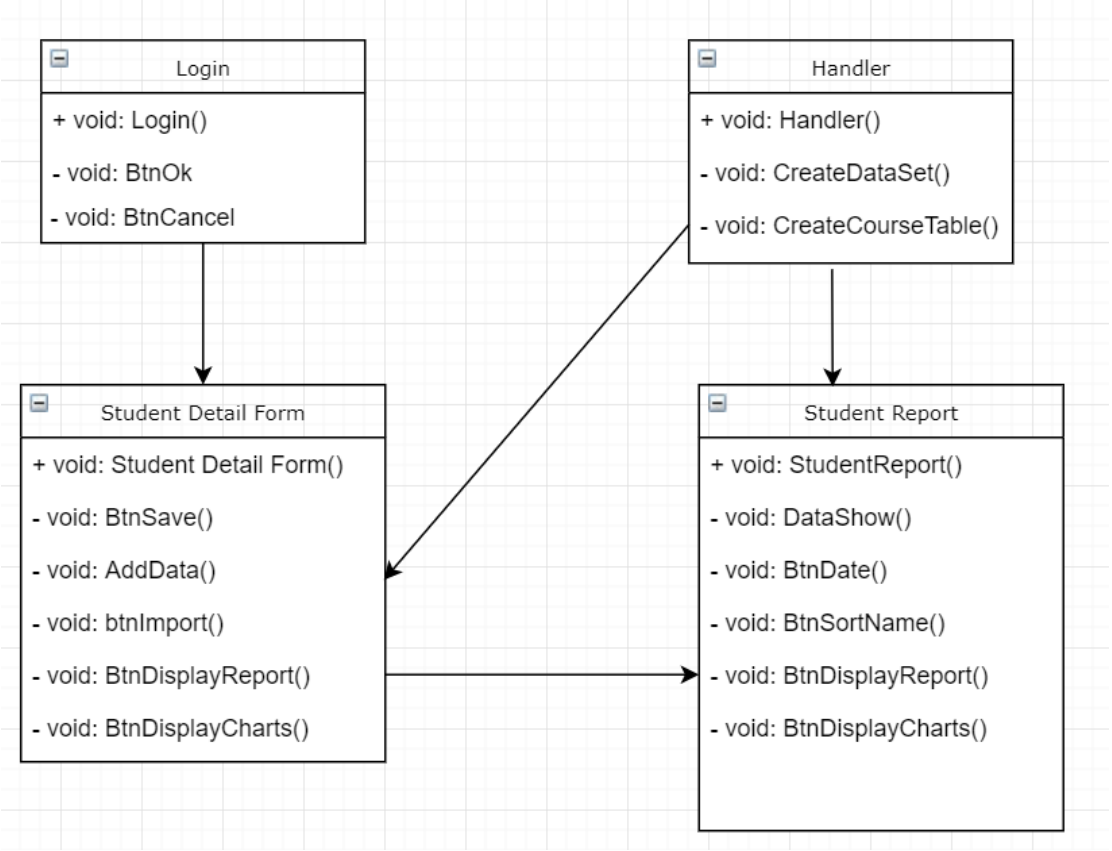

Figure 7: Class Diagram

5. Algorithm

Bubble Sort

Bubble Sort is the simplest sorting algorithm that works by repeatedly swapping the adjacent elements if they are in wrong order. The input to this algorithm is an unsorted list. This unsorted list compares each element with the right side of it's neighbour in order to sort the data. The element which is smaller will shift towards the left side. After one iteration one of the element will be in it's the correct position. This iteration continues until all the elements are in their correct position. (Bubble Sort - GeeksforGeeks, 2019).

First Phase:

First, create an unsorted list. Create a function that takes this unsorted list as an argument. Create a method inside the loop with loop variable ***i*** that counts the length of the list. Create an inner loop inside the loop with loop variable j which counts the elements from zero to i. If the inner loop contains elements which are not in order that means list[j] and list[j+1] are out of order. If in any one of the iteration there are no swaps then the list is sorted and returns the sorted list.

(5 1 4 2 8) –> (1 5 4 2 8), Here, algorithm compares the first two elements, and swaps since 5 > 1.
(1 5 4 2 8) –> (1 4 5 2 8), Swap since 5 > 4
(1 4 5 2 8) –> (1 4 2 5 8), Swap since 5 > 2
(1 4 2 5 8) –> (1 4 2 5 8), Now, since these elements are already in order (8 > 5), algorithm does not swap them.

Second Phase:
(1 4 2 5 8) –> (1 4 2 5 8)
(1 4 2 5 8) –> (1 2 4 5 8), Swap since 4 > 2
(1 2 4 5 8) –> (1 2 4 5 8)
(1 2 4 5 8) –> (1 2 4 5 8)
Now, the array is already sorted, but our algorithm does not know if it is

completed. The algorithm needs one whole pass without any swap to know it is sorted.

Third Phase:

(1 2 4 5 8) –> (1 2 4 5 8)

(1 2 4 5 8) –> (1 2 4 5 8)

(1 2 4 5 8) –> (1 2 4 5 8)

(1 2 4 5 8) –> (1 2 4 5 8)

6. Reflection

The overall system is an desktop application where Student Information System is required to build. It is developed using Visual Studio 2017 with the C# language version 7.3. The GUI designed is highly user interface and user with basic system administration can operate the system.

An end user can have the facilities to add the student details of the student. The application is required to input the student personal detail including registration date so that a system can generate a weekly enrolment report of the student. System is required to include detail like Name, address, contact no, email, program enrol, and registration date. The application is to keep track of the student's details, program enrol and registration date.

Developing the system in Microsoft Visual Studios 2017 keeping C# as primary programming language isn't new experience for me. But developing in C# environment is new for me. Features like creating chart generating list in addition to that, sorting of data form the grid was a new thing for me. Furthermore, import and exporting to CSV file was new aspect for me.

7. Testing

When the development phase is completed, testing phase is started. Testing ensures that there are no bugs or errors which lets the system run as a bug free application.

Test case 1

Objective: To analyze whether logging in with wrong username or password is possible or not.

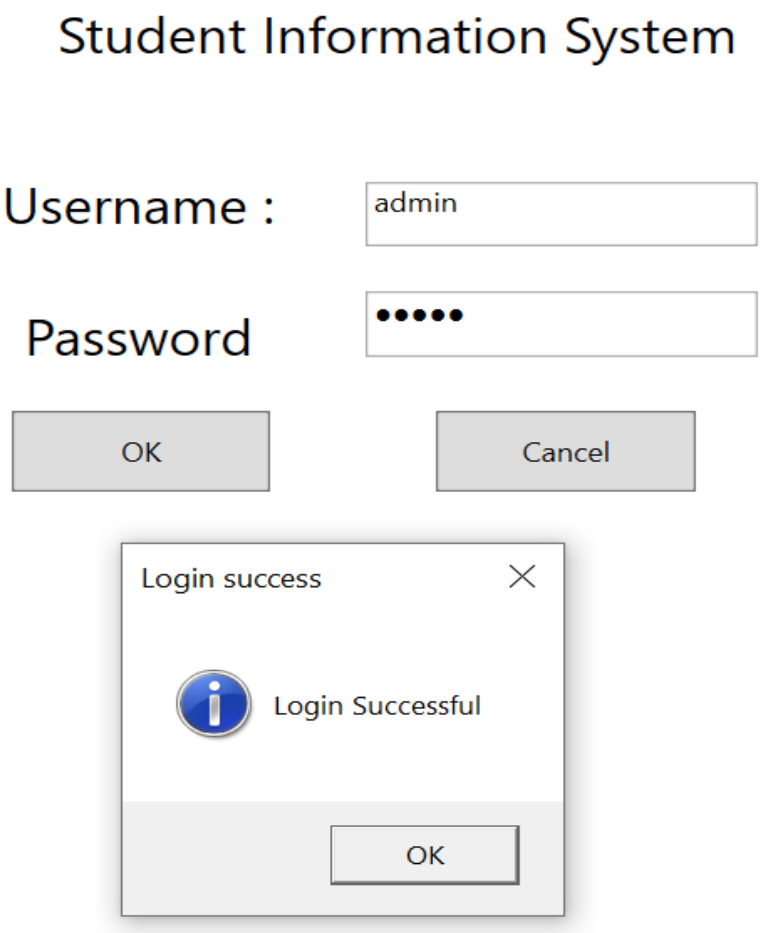

Figure 8:Login Successful

Result: logging in with correct username and password is possible.

Test is successful.

Test case 3

Objective: To analyze whether save button works or not

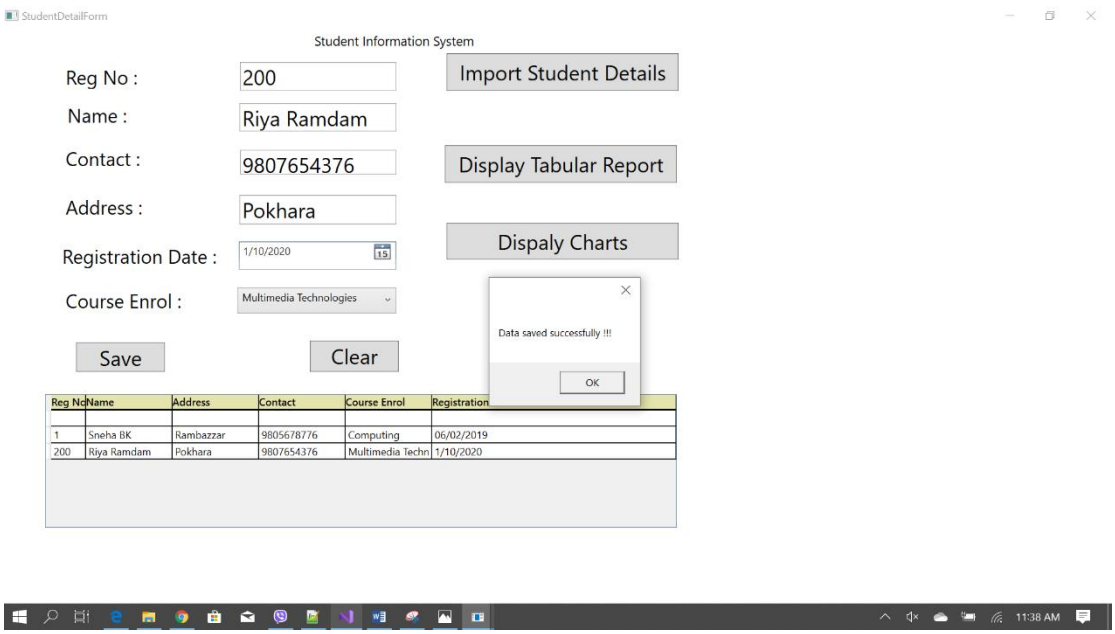

Figure 9: Students Details Save

Result: Save button works. The test is successful

Test case 4

Objective: To analyze whether clear button works or not

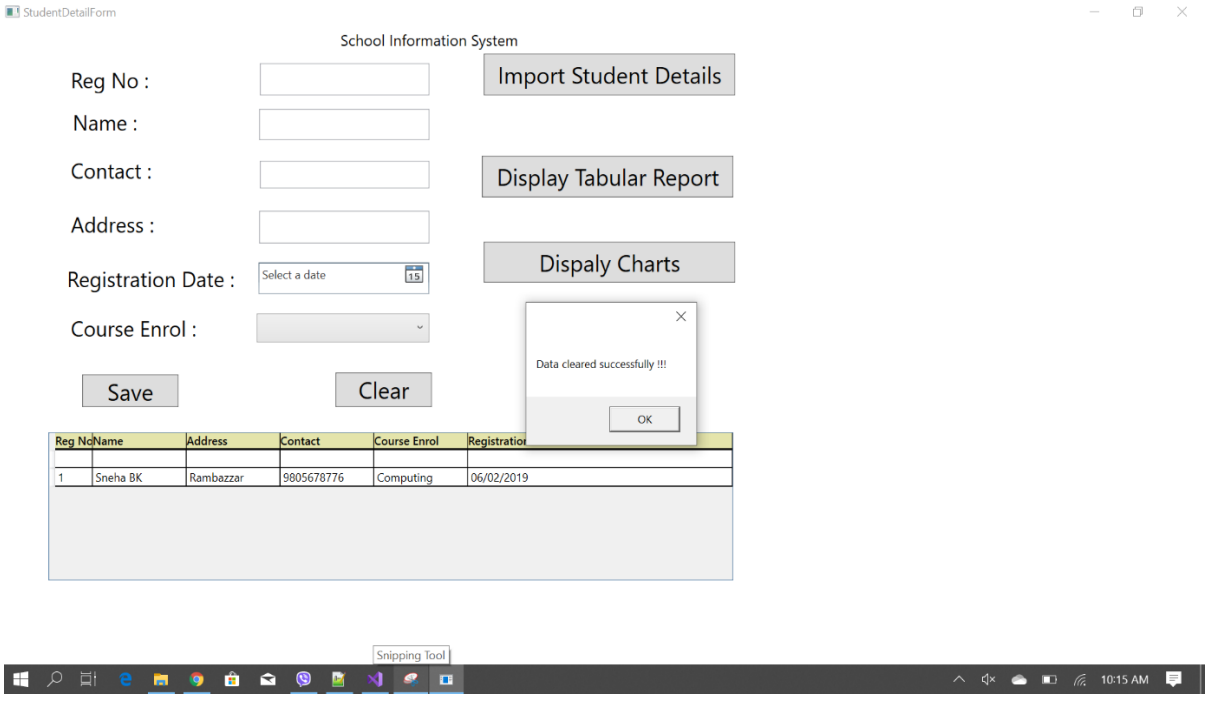

Figure 10: Data cleared

Result: Clear button works. The test is successful

Test case 5

Objective: To analyze whether Import students details button works or not

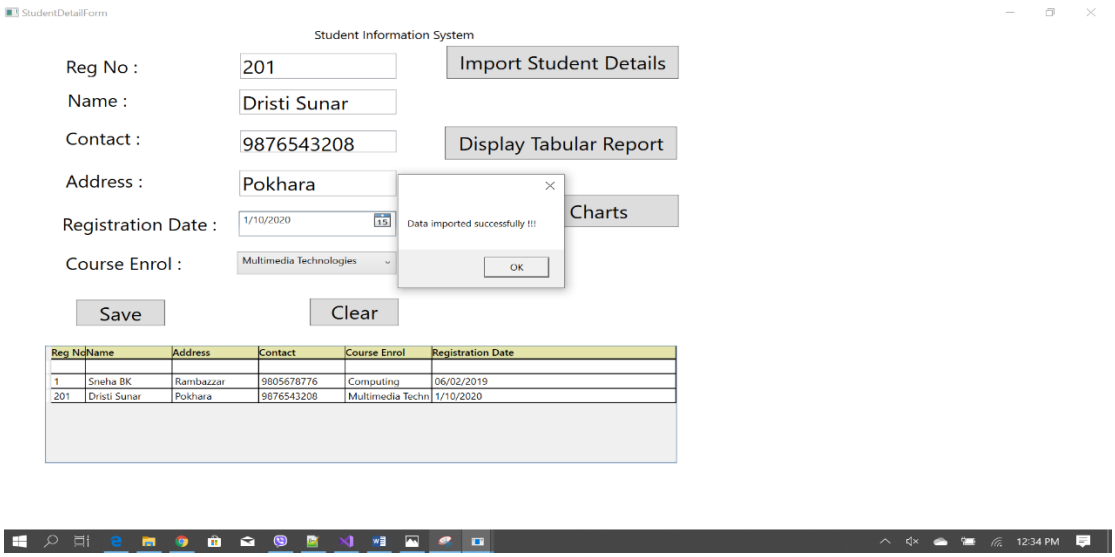

Figure 11: Import Student details

Test case 7

Objective: To analyze whether Retrieve button works or not

By clicking into Display Tabular Report button another page is open where there are 5 button such as weekly report, sort by name, sort by date, retrieve data and lastly import from CSV.

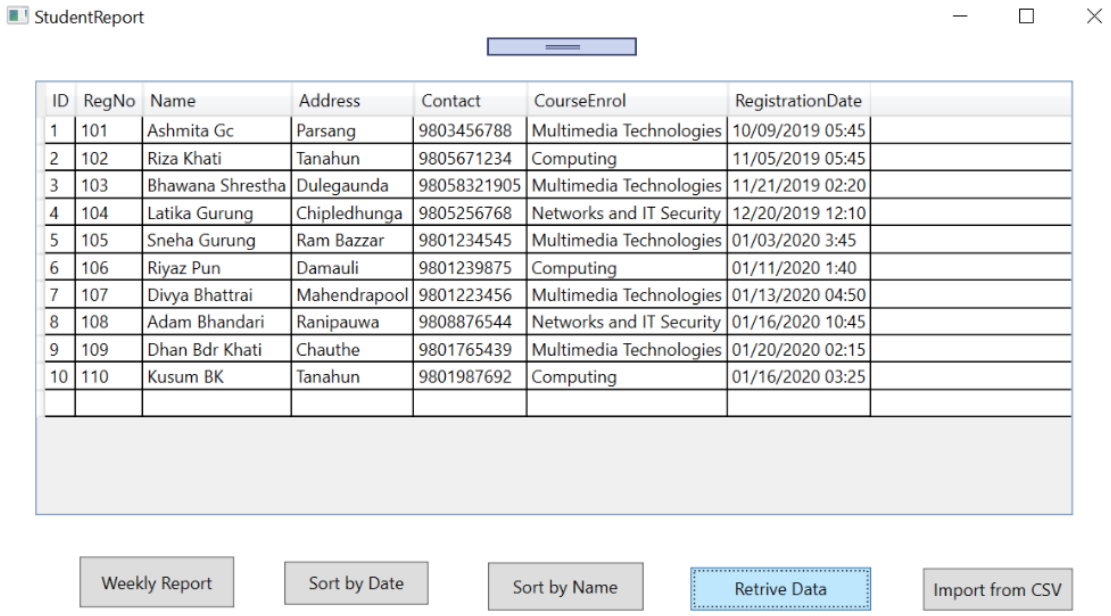

Figure 12: Retrieve Data

Result: Retrieve button works.

The test is successful

Test case 6

Objective: To analyze whether Sort By Name button works or not

All the data will be sort according to name. The figure of sort by name is shown below:-

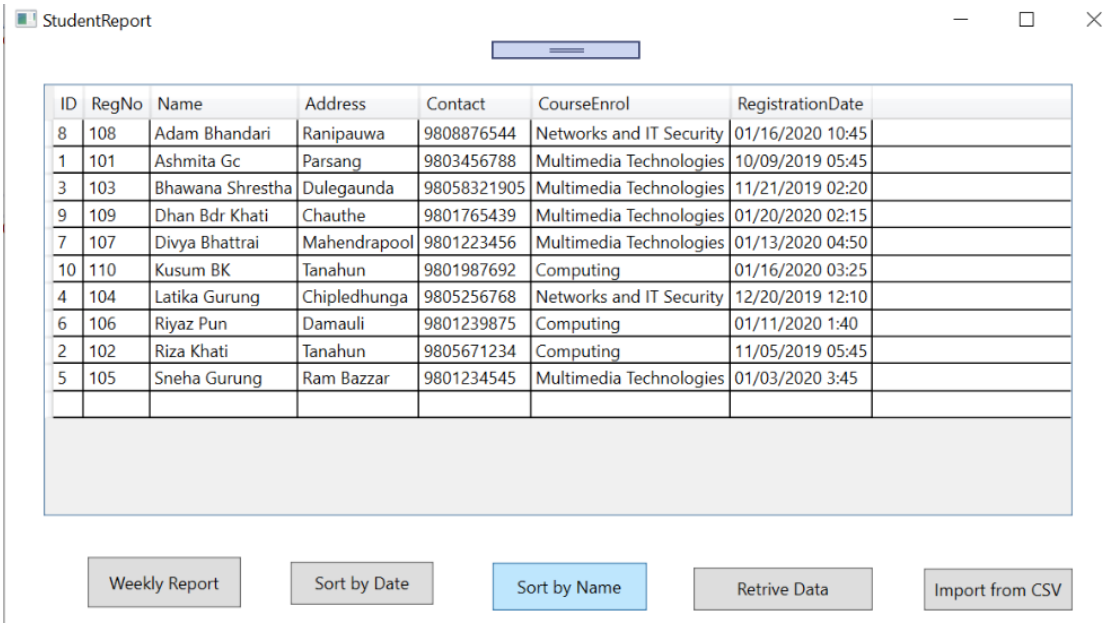

Figure 13: Sort by Name

Result: Sort by Name button works. The test is successful

Test case7

Objective: To analyze whether Sort By Date button works or not

All the data will be sort according to date. The figure of sort by date is shown below:-

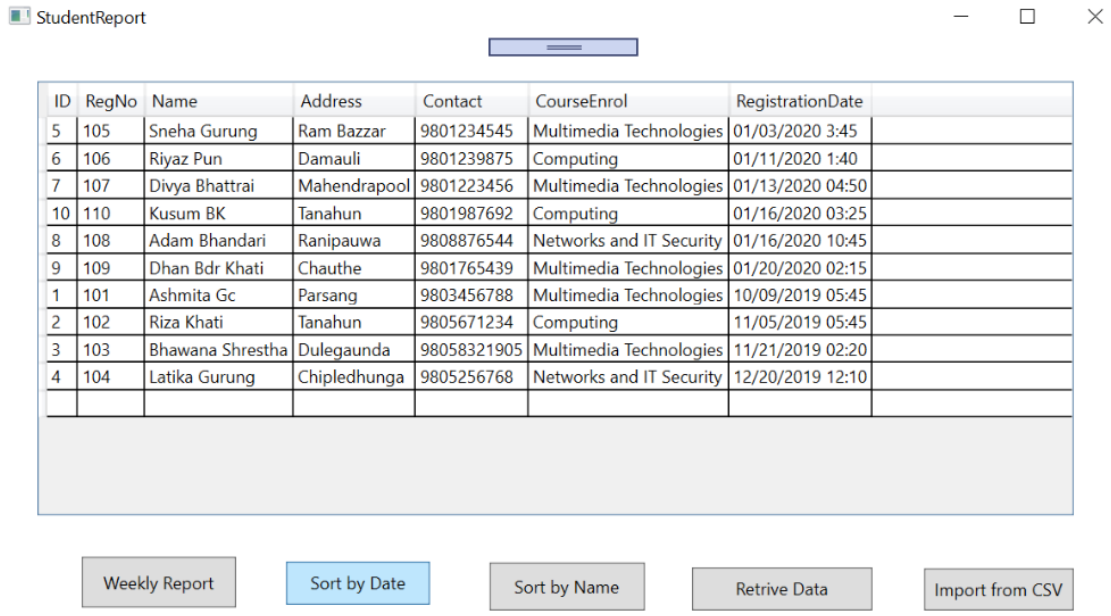

Figure 14: Sort By Date

Result: Sort by Date button works. The test is successful

Test case 8

Objective: To analyze whether Weekly Report button works or not

The report will display weekly tabular report showing total number of students enrolled so far in each program offered by the institution.

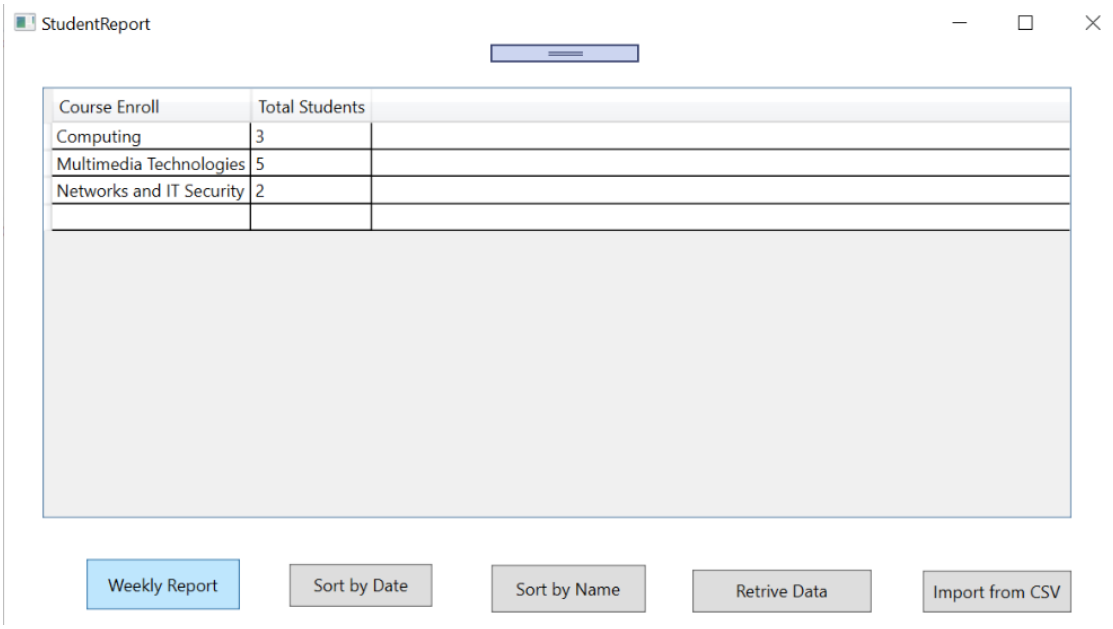

Figure 15: Weekly report

Test case 9

Objective: To analyze whether Import from CSV button works or not

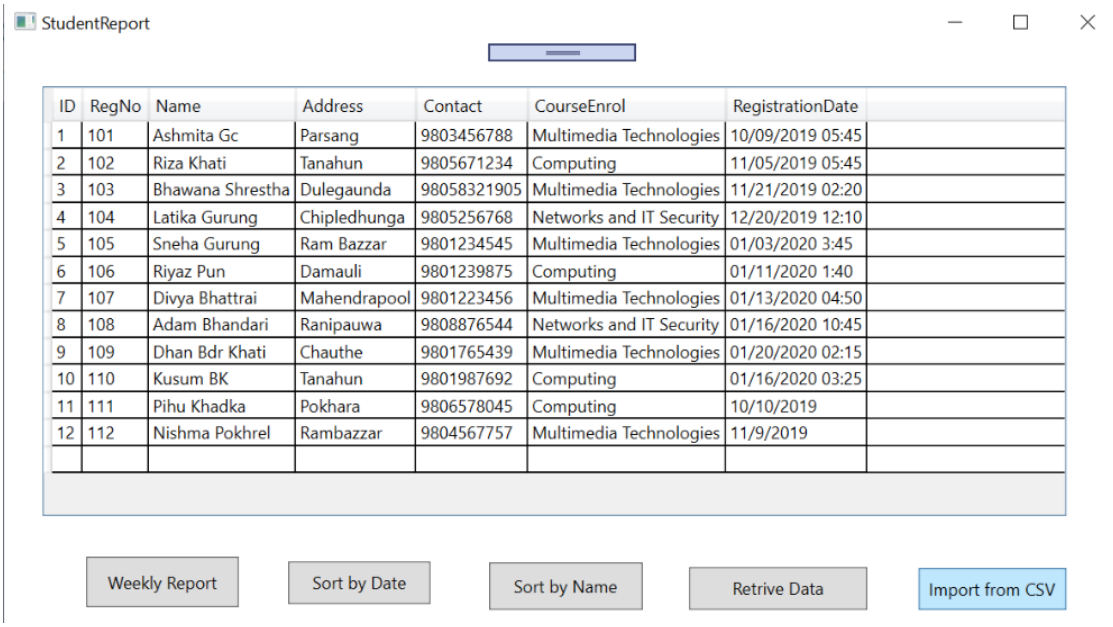

Figure 16: Import from CSV

8. Appendix

Login.cs

```
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;

namespace CourseWork1
{
    /// <summary>
    /// Interaction logic for Login.xaml
    /// </summary>
    public partial class Login : Window
    {
        public Login()
        {
            InitializeComponent();
        }

        private void BtnOk_Click(object sender, RoutedEventArgs e)
        {
            var user = txtName.Text;
            var pass = txtPassword.Password;
            if (user.Equals("") || pass.Equals(""))
            {
                MessageBox.Show("You can't pass empty value", "Login Error",
                    MessageBoxButton.OK, MessageBoxImage.Error);
            }
            else if (user != "admin" || pass != "admin")
            {
                MessageBox.Show("Username or passwoard didn't match!", "Login
Error",
                MessageBoxButton.OK, MessageBoxImage.Error);
                txtName.Clear();
                txtPassword.Clear();
            }
            else
            {
                MessageBox.Show("Login Successful", "Login success",
                MessageBoxButton.OK, MessageBoxImage.Information);
                var StudentDetailForm = new StudentDetailForm();
                StudentDetailForm.Show();
            }
        }
        private void BtnCancel_Click(object sender, RoutedEventArgs e)
        {
                Close();
           }
    }
}
```

Student Detail Form

```
using DataHandler;
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.Navigation;
using System.Windows.Shapes;



namespace CourseWork1
{
    /// <summary>
    /// Interaction logic for StudentDetailForm.xaml
    /// </summary>
    public partial class StudentDetailForm : Window
    {
        public StudentDetailForm()
        {
            InitializeComponent();



            //add new student

            Student student = new Student();

            //add details
            student.RegNo = "1";
            student.Name = "Sneha BK";
            student.Address = "Rambazzar";
            student.Contact = "9805678776";
            student.CourseEnrol = "Computing";
            student.RegistrationDate = "06/02/2019";

            grdStd.Items.Add(student);

        }

        public class Student
        {
            public string RegNo { 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 BtnSave_Click(object sender, RoutedEventArgs e)
{
    var handler = new Handler();
    var dataSet = handler.CreateDataSet();
    AddSampleData(dataSet);
    MessageBox.Show("Data saved successfully !!!");
    if (File.Exists(@"D:\studentData.xml"))
    {
        dataSet = new DataSet();
        dataSet.ReadXml(@"D:\studentData.xml");
        AddSampleData(dataSet);
        dataSet.WriteXml(@"D:\studentData.xml");
    }
    else
    {
        dataSet = handler.CreateDataSet();
        AddSampleData(dataSet);
        dataSet.WriteXml(@"D:\studentData.xml");
    }



}

private void AddSampleData(DataSet dataSet)
{
    var dr1 = dataSet.Tables["Student"].NewRow();
    //dr1["RegNo"] = regNo.Text;
    dr1["Name"] = txtName.Text;
    dr1["Address"] = txtAddress.Text;
    dr1["Contact"] = txtContact.Text;
    dr1["CourseEnrol"] = courseEnrol.Text;
    dr1["RegistrationDate"] = registrationDate.Text;
    dataSet.Tables["Student"].Rows.Add(dr1);
}




private void BtnClear_Click(object sender, RoutedEventArgs e)
{
    regNo.Text = "";
    txtName.Text = "";
    txtAddress.Text = "";
    txtContact.Text = "";
    courseEnrol.Text = "";
    registrationDate.Text = "";

    MessageBox.Show("Data cleared successfully !!!");

}

private void BtnImport_Click(object sender, RoutedEventArgs e)
{
    Student dataStudent = new Student();
    dataStudent.RegNo = regNo.Text;
    dataStudent.Name = txtName.Text;
    dataStudent.Address = txtAddress.Text;
    dataStudent.Contact = txtContact.Text;
```

```
            dataStudent.CourseEnrol = courseEnrol.Text;
            dataStudent.RegistrationDate = registrationDate.Text;
            grdStd.Items.Add(dataStudent);
            MessageBox.Show("Data imported successfully !!!");

        }

        private void BtnDispalyReport_Click(object sender, RoutedEventArgs e)
        {
            StudentReport studentReport = new StudentReport();
            studentReport.Show();


        }

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

Student Report

```
using Microsoft.Win32;
using System;
using System.Collections;
using System.Collections.Generic;
using System.Data;
using System.Data.OleDb;
using System.Globalization;
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;

namespace CourseWork1
{

    public partial class StudentReport : Window
    {
        private DataTable buffer;


        public StudentReport()
        {
            InitializeComponent();
        }


        private void DataShow()
        {
```

```
            string dataXMLFile = @"D:\studentData.xml";
            System.Data.DataSet dataset = new DataSet();
            dataset.ReadXml(dataXMLFile);
            buffer = dataset.Tables[0];
            //buffer = new DataTable("dt");

            //buffer.Columns.Add("Reg No", 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.Today;
            //    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);
            DataGrid2.ItemsSource = dataView;
        }
        private void BtnDate_Click(object sender, RoutedEventArgs e)
        {
            DataView dataView = new DataView(buffer);
            dataView.Sort = "RegistrationDate ASC";
            DataGrid2.ItemsSource = dataView;
        }

        private void BtnSortName_Click(object sender, RoutedEventArgs e)
        {
            DataView dataView = new DataView(buffer);
            dataView.Sort = "Name ASC";
            DataGrid2.ItemsSource = dataView;
        }

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

        private void BtnWeekReport_Click(object sender, RoutedEventArgs e)
        {

            var dataset = new DataSet(); // declaring new data set
            dataset.ReadXml(@"D:\studentData.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);


            DataGrid2.ItemsSource = dt.DefaultView; // is the name of data grid
        }



        private void BtnCsv_Click(object sender, RoutedEventArgs e)
        {
            var dataSet = new DataSet();
            dataSet.ReadXml(@"D:\studentData.xml");
            OpenFileDialog openFileDialog = new OpenFileDialog();
            openFileDialog.Filter = "CSV Files|*.csv";
            bool? fileselect = openFileDialog.ShowDialog();
            if (fileselect == 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["Reg No"] = 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:\studentData.xml");
```

```
                    DataGrid2.ItemsSource =
dataSet.Tables["Student"].DefaultView;
                }

            }
        }
    }
}
```

Handler.cs

```
using System;
using System.Data;

namespace DataHandler
{
    public class Handler
    {
        public DataSet CreateDataSet()
        {
            var ds = new DataSet();
            ds.Tables.Add(CreateCourseTable());
            ds.Tables.Add(CreateStudentTable());
            //ForeignKeyConstraint courseWorkFK = new
ForeignKeyConstraint("courseWorkFK",
            //ds.Tables["Course"].Columns["ID"],
            //ds.Tables["Student"].Columns["CourseEnrol"]);
            //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("Reg No", 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));
            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.PrimaryKey = new DataColumn[] { dt.Columns["ID"] };
            return dt;
        }
    }
}
```

9. Bibliography

Bubble Sort - GeeksforGeeks. (2019). Retrieved from GeeksforGeeks: https://www.geeksforgeeks.org/bubble-sort/