Informatics College Pokhara

Application Development

CS6004NA

Coursework 1

Submitted By:
Student Name: Khumraj Gurung
London Met ID: 17030716
Group: L3C2
Date: 10-Jan-2020

Submitted To:
Mr. Ishwor Sapkota

Abstract

This is an individual Coursework for the module “Application Development” for Student Management System which is developed using Visual Studio Platform using C# language. The coursework is released in the 5th week and it is supposed to be submitted in the 11th week. With the help of module leader and friend, the coursework was completed within the time.

Table of Contents

Table of Figure

Table of Tables

1.Introduction

The given coursework is related to design and implementation of Student Information System. The is desktop application based not a web-based or database application. The system is designed developed and test under various circumstances. In this application, the user will input the student personal details that will include registration date so that a system can generate a weekly enrolment report of the student. System will include details like Name, address, contact number, email, registration date. The main purpose of this system is to keep track of the student's details, program enrol and registration date during student enrolment. The features and function that are required by Student Information System are almost fulfilled by the developed system. It consists of features like generating and displaying two different report that include students' details. These two reports are generating and display by two different way: sorted by student first name and another way is sorted by registration date. Furthermore, there is a feature to view daily and weekly table and chart.

Current Scenario

There are more than more school, universities in the world. They all use student management system software. But they keep record of student data in traditional system. some of the school, collages, universities used modern system but are well lacking the features which are needed for the student information system.

Proposed System

The proposed system is digitized system which I specially designed to overcome problem mentioned above. In this system, all the features will include that needed for the system. There will be all daily report, weekly report of student who enrolled in institute. Entry of data and display of data have been made easy with the presence of easy user interface.

2. User Manual

There are Student Information System's screenshot below which will illustrate a administrator how to operate the system.

As the administrator operates the system the initial screen will be the login form. The user name and password of the system is "admin@123". Only a valid username and password can provide access to this system.

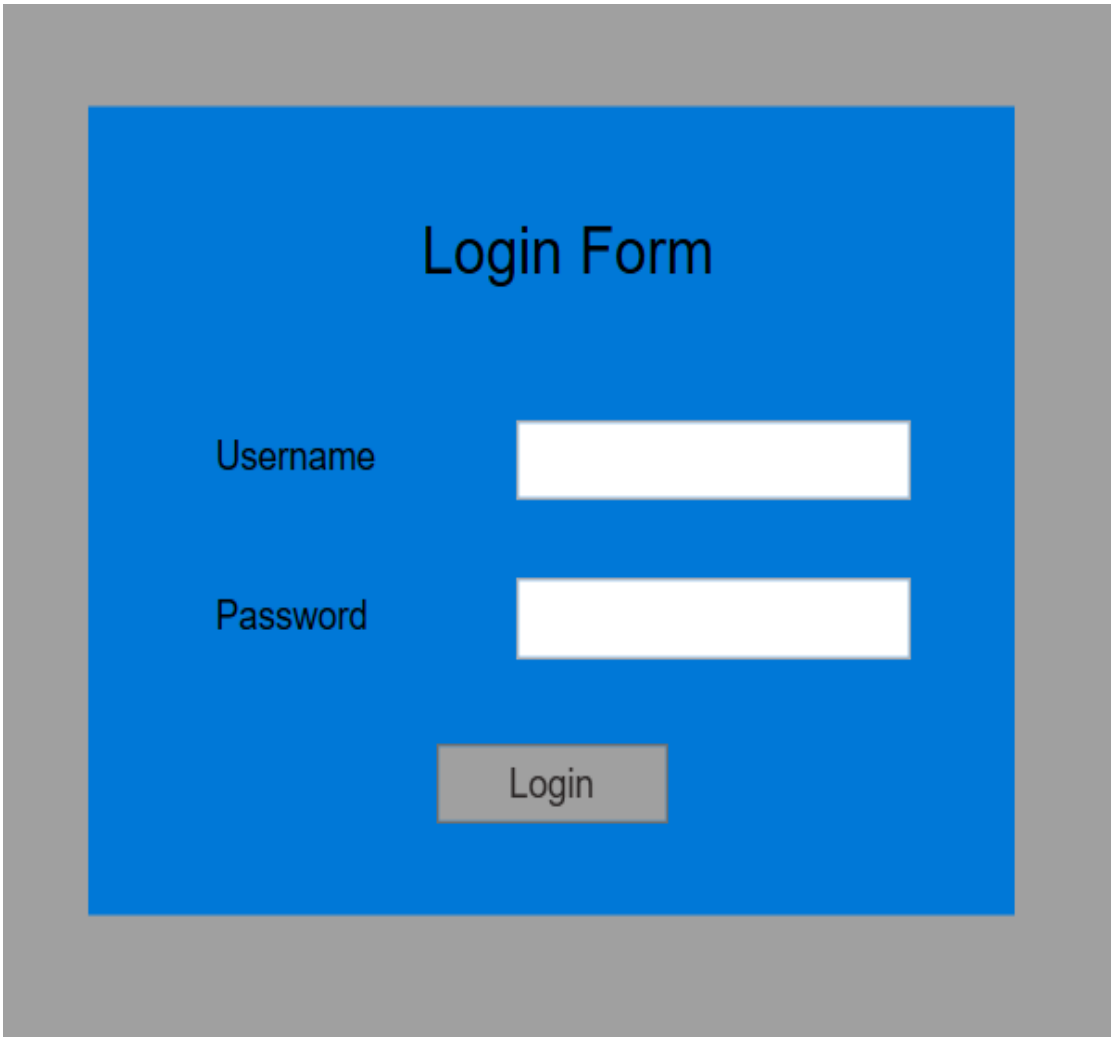


Figure 1: Login Form

After successfully login the main screen of the system will be as shown below

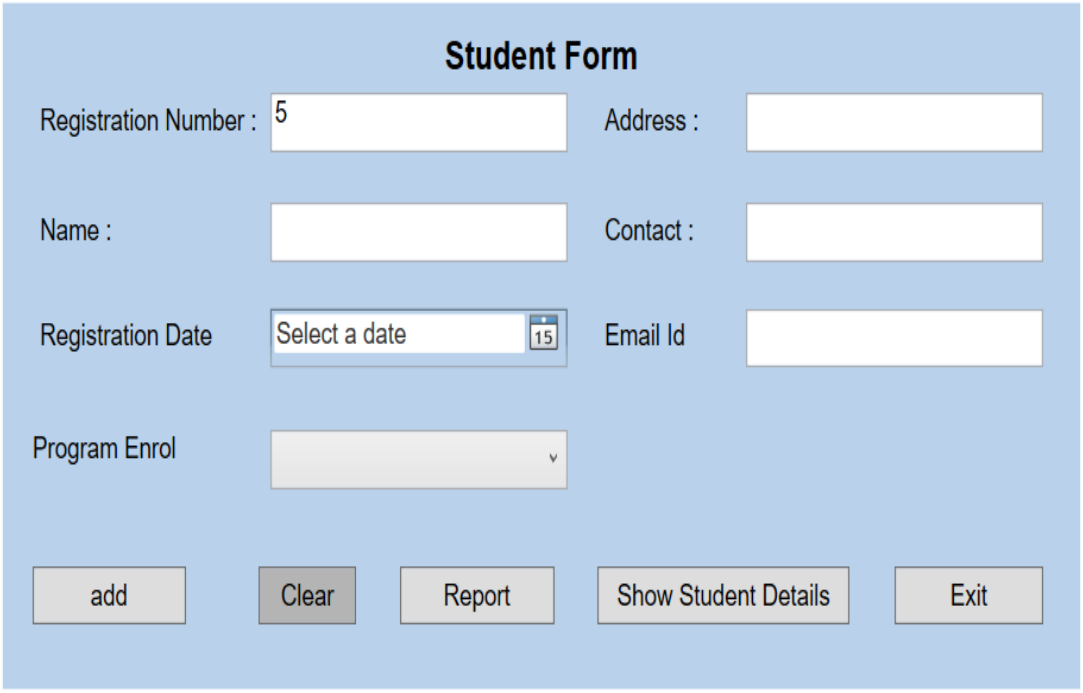


Figure 2:Student Enrolment Form

Above figure is the main windows of the system. There are student enrol form, add section, clear section, report of the student section, student details show section and exit button.

Add button is used to add student In the system. Report button is used to know weekly and total student enrol in the institute. With the help of show Student Details button we can get all the information of student who are enrolled in the institute. To close the application, there is exit button.

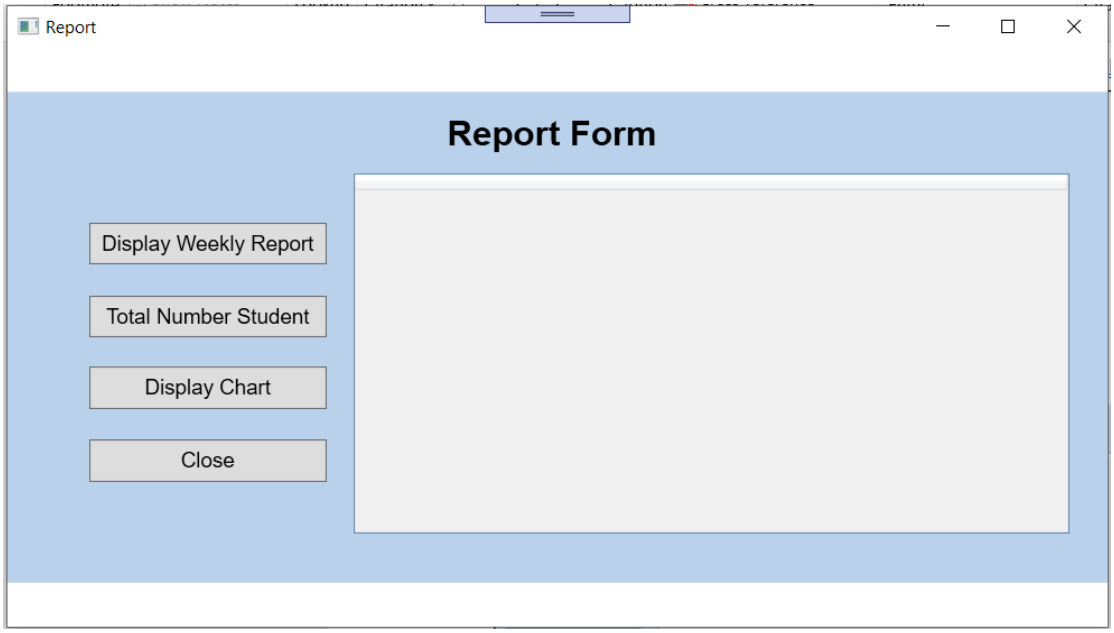


Figure 3:Report Form

In above figure there is Report form windows. There are two main button Display Weekly Report and Display chart button.

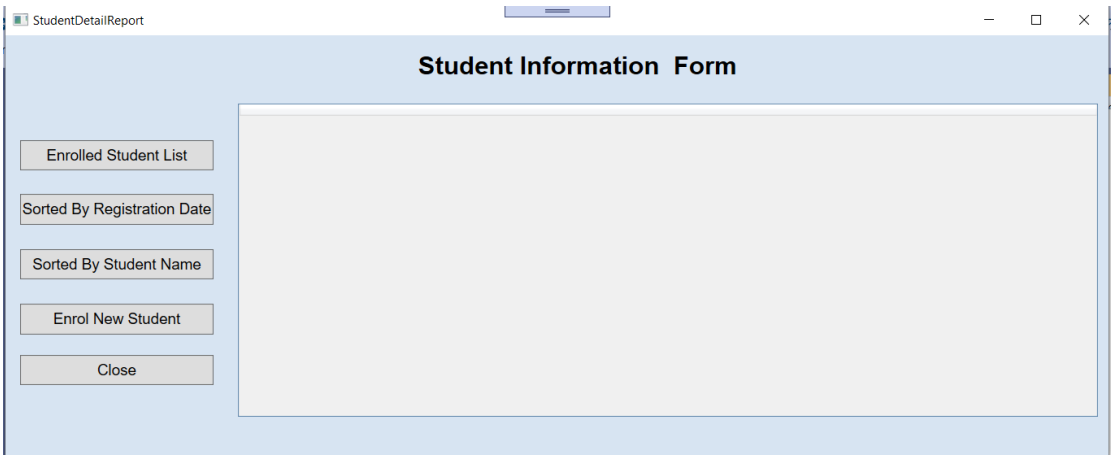


Figure 4: Student Information Form

Above figure shows Student information Form, there are mainly four important buttons i.e. Enrolled Student list, Sorted by Registration Date, Sorted by Student Name and Enrol new student button.

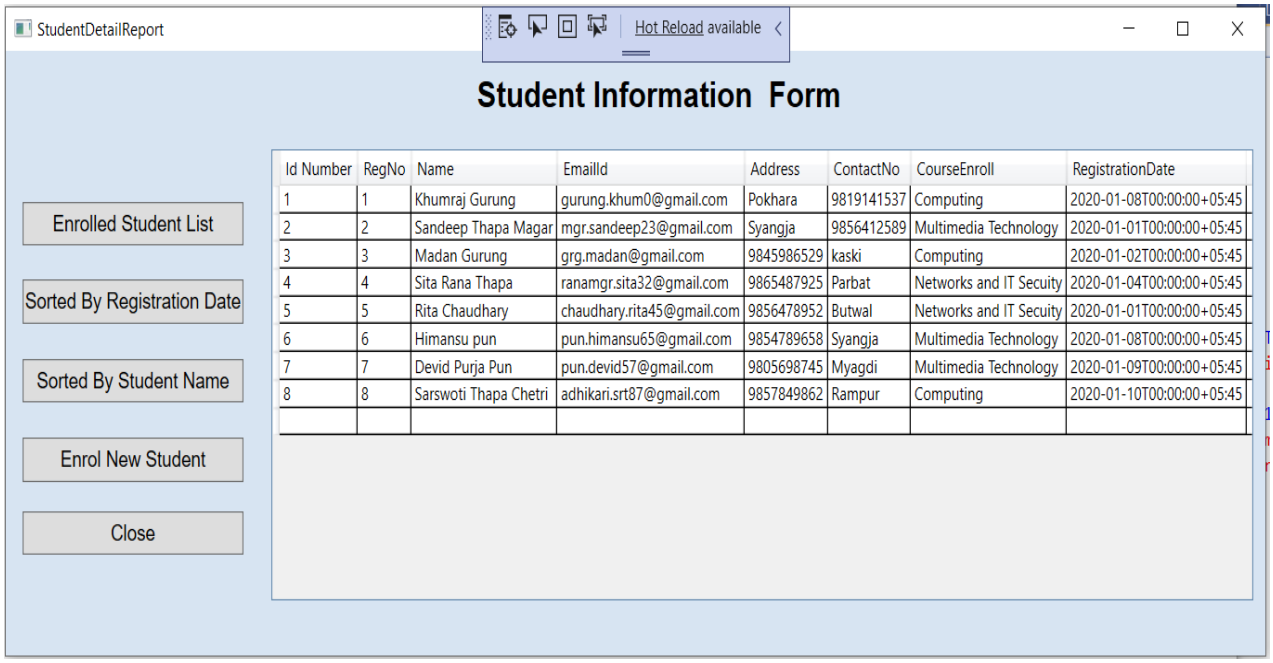


Figure 5:Enrolled Total Student List

Above figure shows enrolled student list. In the figure showing all the information of student like id number, registration number, name, email id, address, contact number, course enrol and registration date.

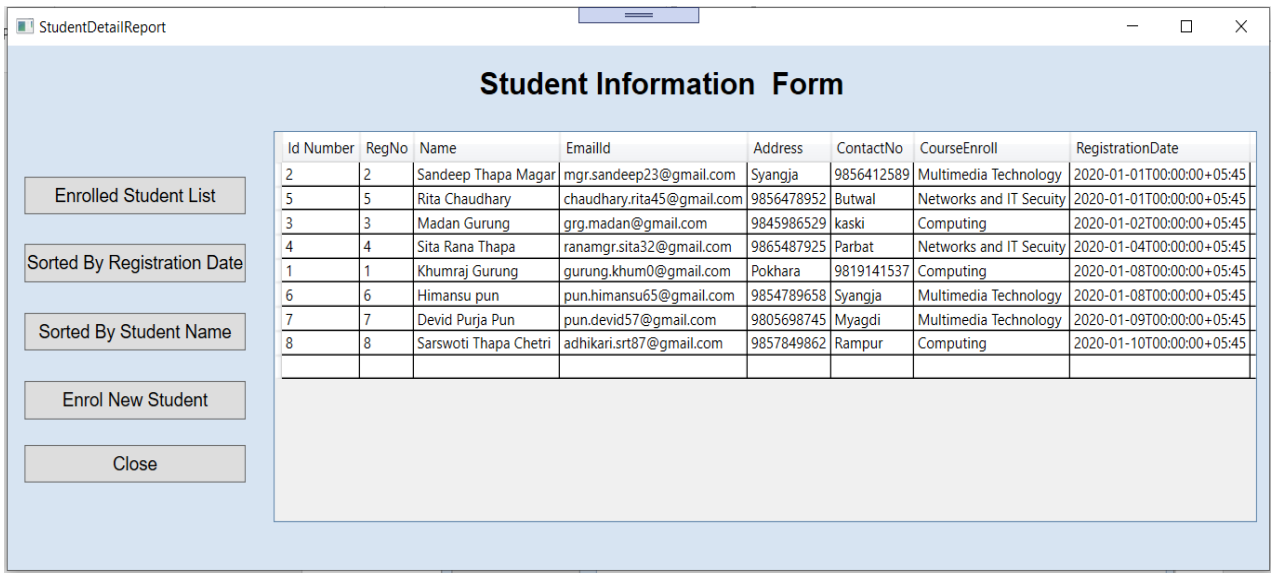


Figure 6: Student Information Showing sorted by Registration Date

In the above figure, the students are sorted by the registration date. In above figure, student details are showing in date format order.

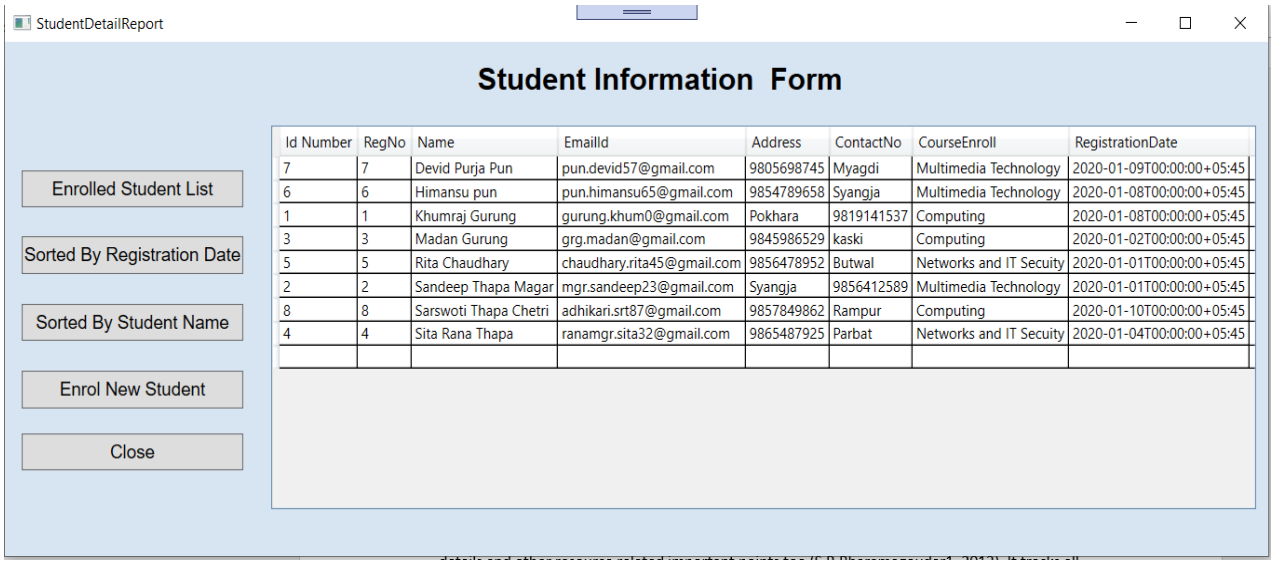


Figure 7:Student Information showing sorted by student first name

The above figure showing student's details which is sorted by name. here, students' names are showing in ascending order.

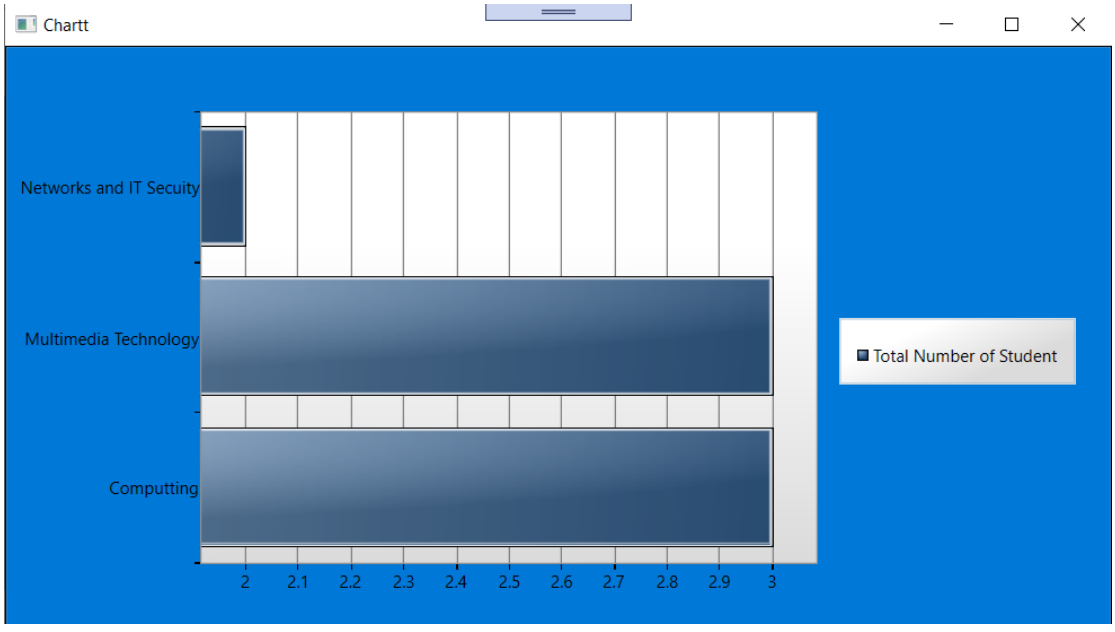


Figure 8: Chart for total number of Students

Above figure shows total number of students enrolled in the institute.

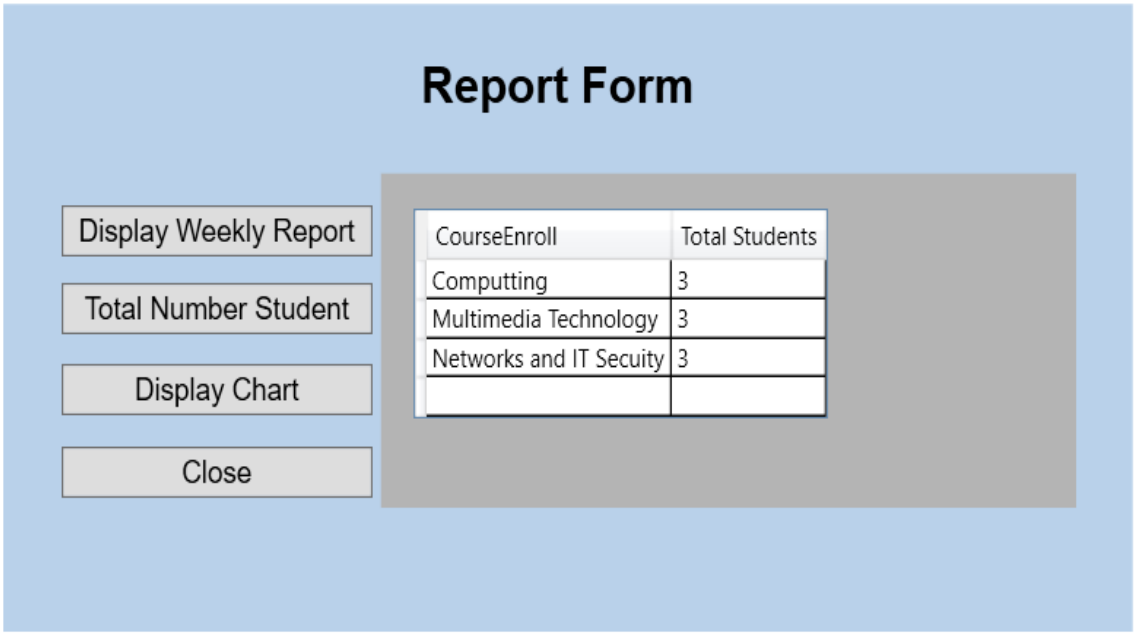


Figure 9:Weekly Report

The above figure shows total number of students enrolled in institute within a week.

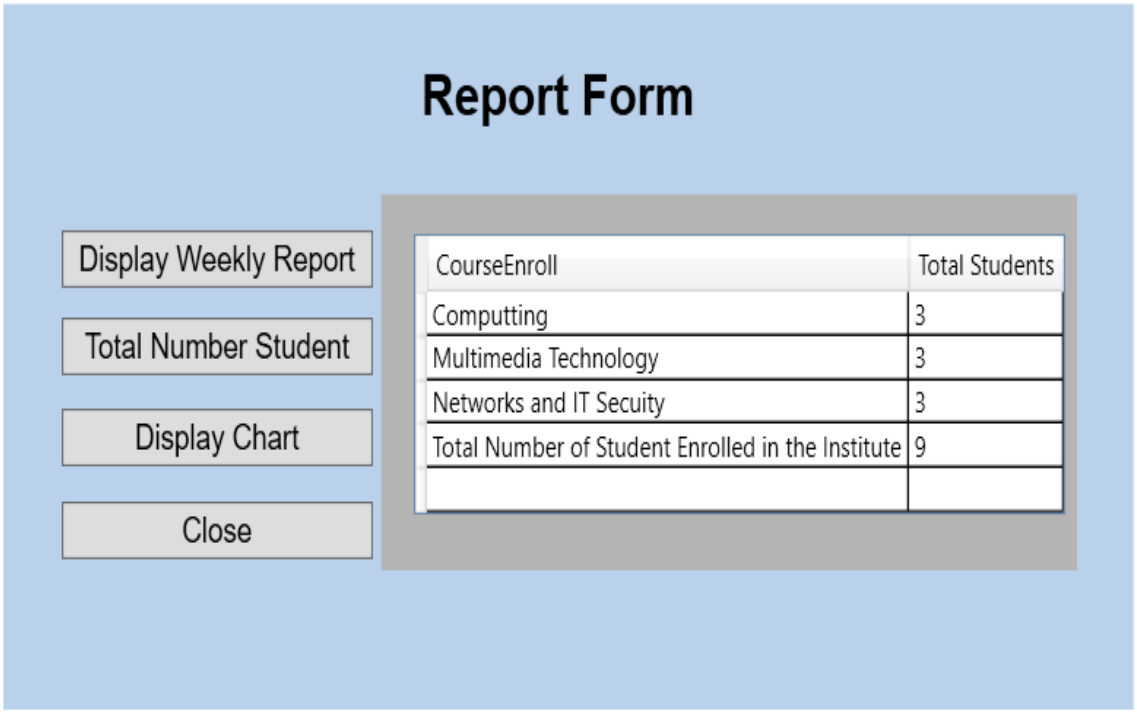


Figure 10:Total Number of Student

The above figure shows that total number of students enrolled in the institute.

Individual Diagram

Login

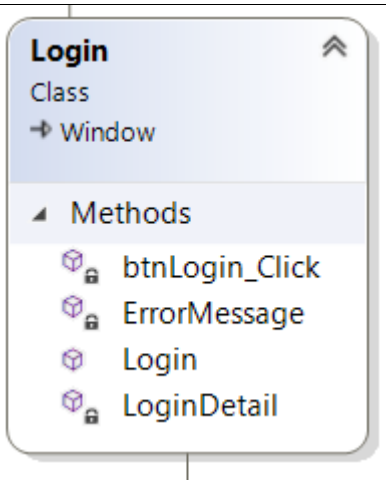
Methods	Description	Diagram
ErrorMessage	This is used to display error massage when username and password are not valid	
btnLogin_Click	This used to validate username and password to enter in the System	

Table 1:Login Individual Diagram

Main Activity

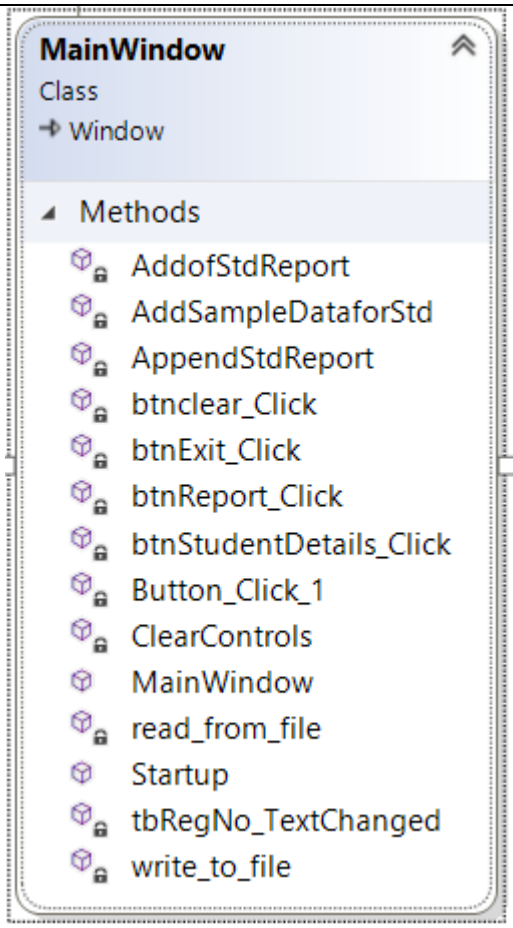
Methods	Descriptions	Diagram
AddSampleDataforStd	It is used to add student data.	
btnReport	It is used to show report of student.	
btnStudentDetail_Click	It is used to show total student enrolled in the institute	

Table 2:Main Activity Individual Diagram

Student Details Report

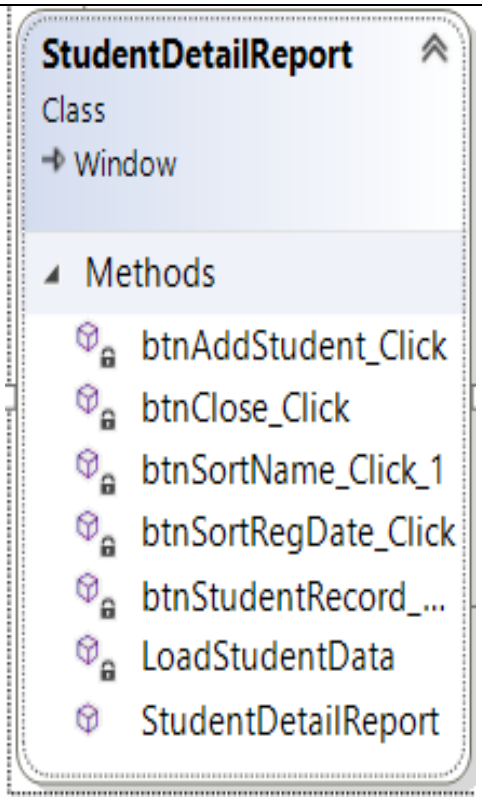
Methods	Descriptions	Diagram
btnAddStudent_Click	It is used to add student who want to enrol in the institute. In this, administrative can manually insert.	
LoadStudentData	To load student details and add in list, it is used.	
btnStudentRecord_Click	It is used for to know total student enrolled in the institute with full details.	
btnSortRegDate_Click	Sorting student data by Registration Date.	
btnSortName_Click	Sorting student details by student's first name	

Table 3:Student Details Individual Diagram

Report

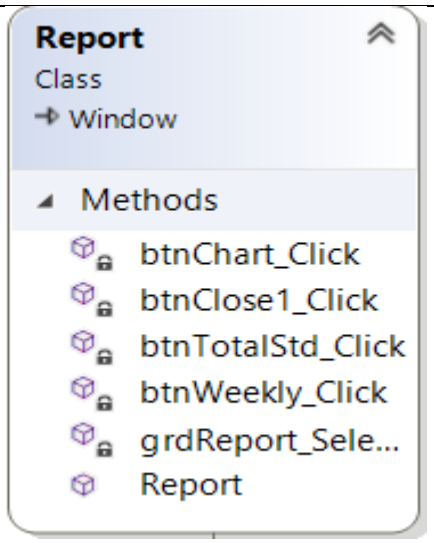
Methods	Description	Diagram
btnWeekly	It is used to show weekly enrolled student details	
btnChart	It is used to show total number of students in graph	
btnTotalStd	It is used to display total number of students	

Table 4:Report Individual Diagram

Journal Articles

1.Student Information Management System (SIMS) offers a easy interface for protection of student information. It can be used through educational institutes or faculties to maintain the files of college students easily. Student facts system offers with all kind of student details, educational related reports, college details, course details, curriculum, batch details, placement details and other resource related important points too (S.R.Bharamagoudar1, 2013). It tracks all the important points of a student from the day one to the end of the route which can be used for all reporting purpose, tracking of attendance, growth in the course, performed semesters, years, coming semester year curriculum details, exam details, project or any different mission details, ultimate exam result and all these will be available thru a secure, on-line interface embedded in the college's website. It will additionally have school details, batch execution details, students' small print in all aspects, the various educational notifications to the group of workers and college students updated with the aid of the university administration. It also facilitates us explore all the things to do happening in the college, Different reviews and Queries can be generated primarily based on significant options related to students, batch, course, faculty, exams, semesters, certification and even for the entire college (S.R.Bharamagoudar1, 2013).

2. It is the inevitable outcome of higher education reform to elevate out workplace and departmental goal responsibility system, in which statistical processing of student's data is a necessary part of student's overall performance review (Zhou, 2012). On the foundation of the evaluation of the student's evaluation, the pupil information administration database utility gadget is designed via using relational database management device software program in this paper. In order to put into effect, the function of pupil records management, the purposeful requirement, universal structure, data sheets and fields, data sheet Association and software program codes are designed in details (Zhou, 2012).

3. A cell pupil records machine (MSIS) primarily based on cellular computing and context-aware utility ideas can provide greater user-centric data services to students. The motive of this paper is to describe a gadget for imparting relevant statistics to college students on a cellular platform. Design methodology approach - The research accompanied a plan science approach, which includes surveys to argue for the relevance of the device and contrast of exceptional versions of the device the usage of a cell machine acceptance model (MSAM) (Krogstie, 2011). Findings - It used to be observed that the intention to use such services is high, in precise relative to services presenting records primarily based on localization and the non-public agenda and interests of the student. Originality price - Several cellular structures exist that furnish prevalent campus statistics for college students and this paper describes one of the few structures of this sort of personalization which have been evaluated (Krogstie, 2011).

4. Courtesy of new technologies, such as pupil facts structures (SIS), districts are opening new channels of communication, giving mother and father each time Internet get right of entry to to data they want to song their kid's progress--and affording them the probability to make a magnificent impact on their kid's mastering growth (Bird, 2006). Take what is occurring at Westside Community Schools, a school district in Omaha, Nebraska, composed of more than 6,000 college students attending 10 elementary schools, one middle school, and one high school. In its tries to keep dad and mom in touch with their child's academic performance, the district used to face the acquainted obstacles. Parents who wanted an replace on their youngster would have to call the principal's workplace or the teacher's direct line. Parent-teacher conferences got here too late to reverse a student's lack of progress. Parents of older college students have been provided few opportunities to continue to be involved (Bird, 2006).

System Architecture

Architecture Diagram

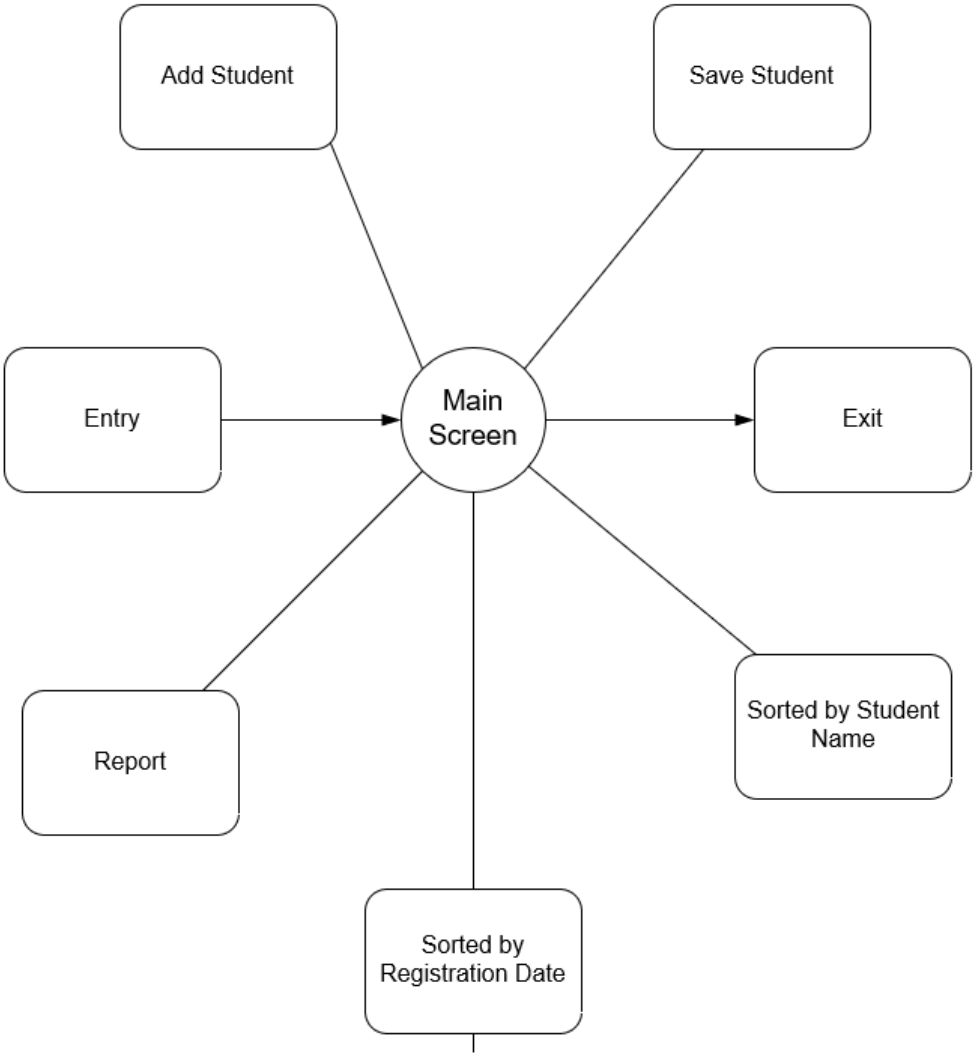


Figure 11:System Architecture Diagram

Class Diagram

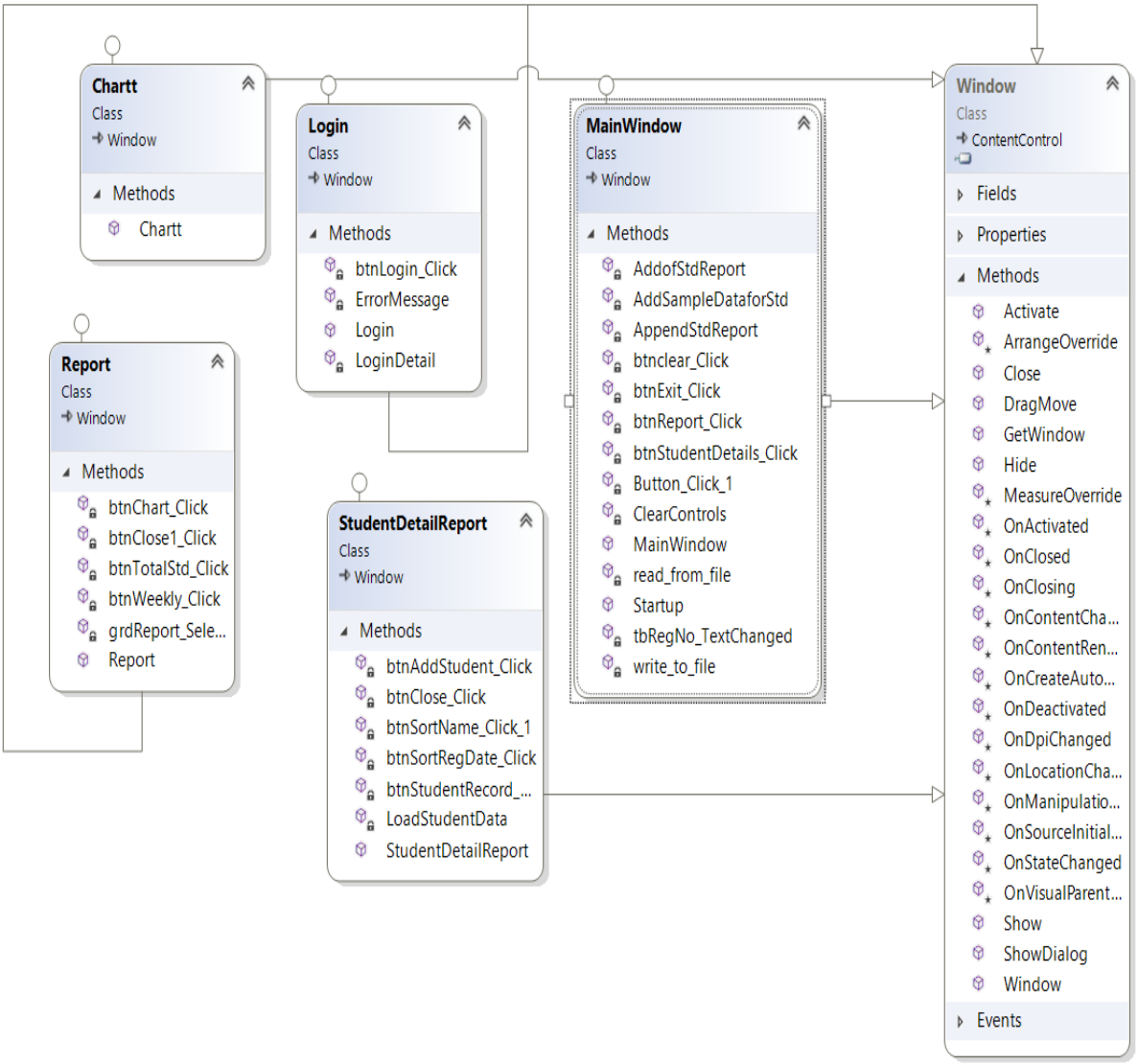


Figure 12:Class Diagram of Student Information System

Flowchart

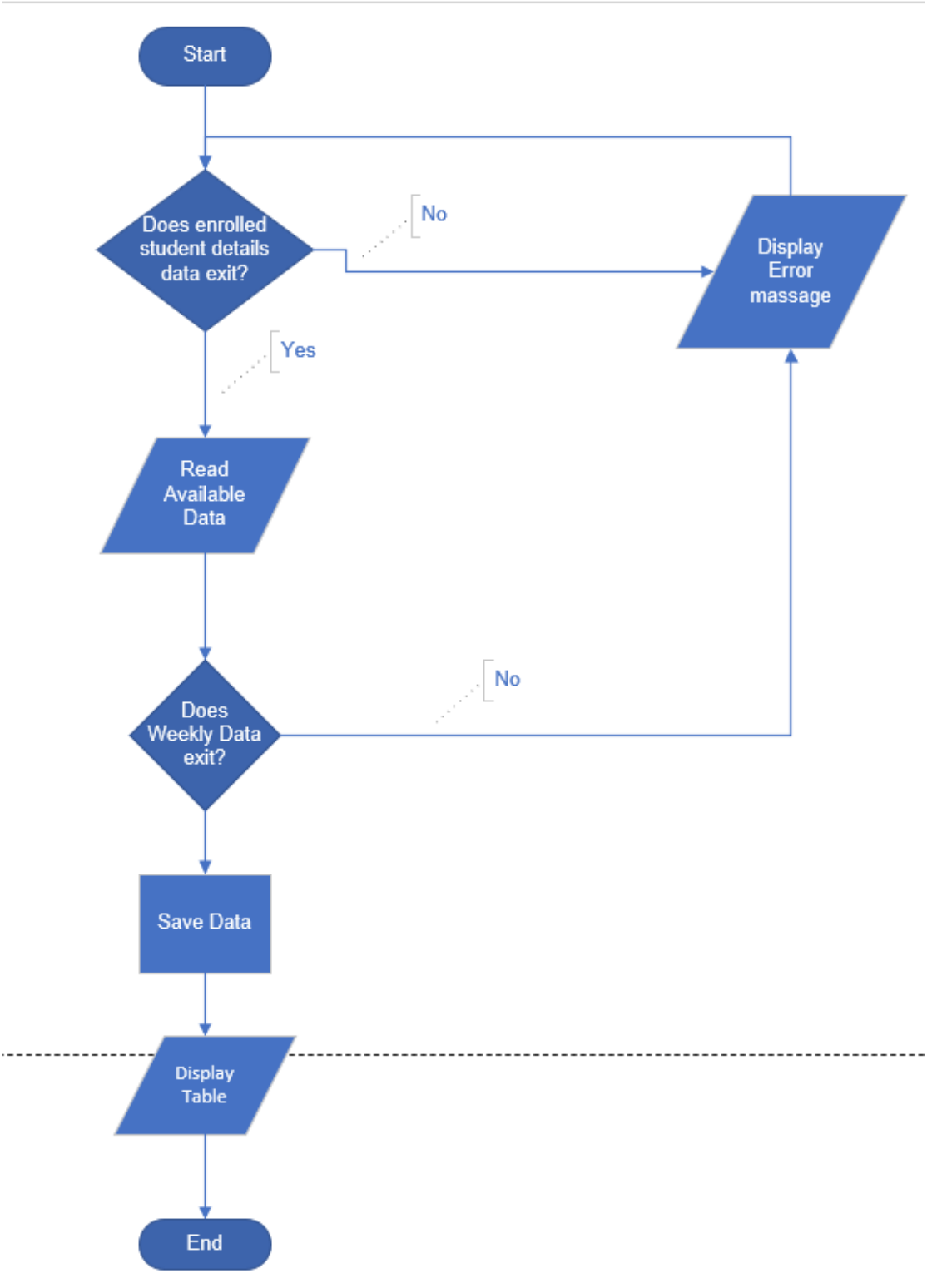


Figure 13:Flowchart of weekly enrolled student

Sorting Algorithm

The sorting Algorithm used in the Student Information System is bubble sorting algorithm.

Sorting

Sorting is ordering a list of gadgets. We can distinguish two types of sorting. If the variety of objects is small adequate to suits into the primary memory, sorting is acknowledged as interior sorting. If the huge variety of gadgets is so massive that some of them are dwelling on exterior storage in the course of the sort, it is referred to as external sorting (S.Adamchik, 2009). Different types of sorting

- Bucket sort
- Bubble sort
- Insertion sort
- Selection sort
- Heapsort
- Merge sort

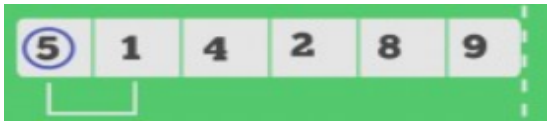
Bubble sort

Bubble Sort is the simplest sorting algorithm that works by way of repeatedly swapping the adjoining elements if they are in the wrong order (S.Adamchik, 2009).

Bubble kind is a simple arranging calculation. This arranging calculation is a correlation-based calculation in which each suit of neighbouring aspects is analysed and the aspects are swapped if they are now not altogether. This calculation isn't sensible for giant informational indexes as its normal and most pessimistic state of affairs multifaceted nature are of O(n2) where n is the extent of things.

Working Mechanism

1.First, there is unsorted array for example.



2.Here, algorithm compares the first two elements. In this case, the value 1 is smaller than the value 5 so, the two value must swap. i.e. 5>1.

3.Similarly, in the below case, the value 4 and the value 2 is smaller than 5 so both the value must swap. i.e. 5>4 and 5>2.

4.In the below case, there is the value 2 is smaller than the value 4. The two values must swap. i.e.4>2.

5.In the below case, the values are already in order (5>4), algorithm does not swap them.

6.In the below case, the values are already in order (2>1), algorithm does not swap them.

Algorithms of Reports

Weekly Report

Steps:

1. Start
2. Check whether the student entry data file exist or not
3. If it doesn’t exist, display no data
4. If exists, read the available data
5. Check whether there is student enrol or not
6. If student not enrol, display error massage
7. If student enrolled, retrieve the data
8. Display the data in the Bar chart
9. stop

Reflection

“Student Information System” is the desktop based modern system. It is developed by using Visual Studio Community 2019 with the C# programming language. The school, collage, universities and other educational institutes can use this system to record student record. The GUI designed is highly interface and user with basic system administration can operate the system.

In this system, the details of the student can be added such as: Student name, phone number, email address, program enrol, address of student. All this information can add manually by the user. Administrative can check out the daily and weekly chart along who enrolled in the institution.

I have only some experience with Visual Studio. But I have learned about visual studio with the help of module leader and got knowledge in visual studio. I started coursework. 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.

Conclusion

The coursework for the module CS6004NA “Application development” was to designed and developed the “Student Information System” desktop application which was built in Visual Studio Community 2019 using C# programming language. The framework shows all the details of student with registration date. This system also shows the student enrolment course and program. Inside the system, all and every functionality are found. Aside from various shape components, class outline for every one of the structures and classes were utilized. There was display daily and weekly tabular report showing total number of students enrolled. After doing hardworking finally, coursework was completed and thank to friend and module leader for guiding me throughout the project.

Bibliography

Bird, K. (2006). Student Information Systems: How Do You Spell Parental Involvement? S-I-S. *T.H.E. Journal*, 192-592. Retrieved 01 06, 2020, from https://www.learntechlib.org/p/77171/

Krogstie, J. (2011). Mobile student information system. *Emerald Management 120*, 5-15. Retrieved 01 06, 2020, from https://ocul-gue.primo.exlibrisgroup.com/discovery/fulldisplay?docid=emerald_s10.1108%2F10650741111097269&context=PC&vid=01OCUL_GUE:GUELPH&lang=en&search_scope=MyInst_and_CI&adaptor=Primo%20Central&tab=Everything&query=any,contains,student%20information%2

S.Adamchik, V. (2009). *CMU*. Retrieved 01 07, 2020, from CMU: https://www.cs.cmu.edu/~adamchik/15-121/lectures/Sorting%20Algorithms/sorting.html

S.R.Bharamagoudar1, G. R. (2013). Web Based Student Information Management. *International Journal of Advanced Research in Computer and Communication Engineering*, 2342. Retrieved 01 06, 2020, from https://s3.amazonaws.com/academia.edu.documents/35152249/4-shobha_bharamaoudar-WEB_BASED_STUDENT_INFORMATION.pdf?response-content-disposition=inline%3B%20filename%3DWeb_Based_Student_Information_Management.pdf&X-Amz-Algorithm=AWS4-HMAC-SHA256&X-Amz-Creden

Zhou, H. (2012). Design of Student Information Management Database Application System for Office and Departmental Target Responsibility System. *Physics Procedia*, 1660-1665. Retrieved 01 06, 2020, from https://ocul-gue.primo.exlibrisgroup.com/discovery/fulldisplay?docid=elsevier_sdoi_10_1016_j_phpro_2012_03_291&context=PC&vid=01OCUL_GUE:GUELPH&lang=en&search_scope=MyInst_and_CI&adaptor=Primo%20Central&tab=Everything&query=any,contains,student%20informat

Appendix

Login.cs

```
using System;
using System.Collections.Generic;
using System.Text;
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 StudentInformationSystems
{
    /// <summary>
    /// Interaction logic for Login.xaml
    /// </summary>
    public partial class Login : Window
    {
        public Login()
        {
            InitializeComponent();
        }
        private void LoginDetail()
        {
            string user = tbUsername.Text;
            string pass = pbPassword.Password;

            if (user == "admin" && pass == "admin@123")
            {
                this.Hide();
```

```
            MainWindow mainWindow = new MainWindow();
            mainWindow.Show();
        }
        else
        {
            ErrorMessage();
        }
    }

    private void btnLogin_Click(object sender, RoutedEventArgs e)
    {
        LoginDetail();
    }

    private void ErrorMessage()
    {
        MessageBox.Show("Invalid username or password", "Login Error",
MessageBoxButton.OK, MessageBoxImage.Error);
        tbUsername.Text = "";
        pbPassword.Password = "";

    }
  }
}
```

Main Activity.cs

```
using DataHandler;
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;
using System.Data;
using System.IO;

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

            tbRegNo.Text = read_from_file();
        }

        public void Startup()
        {

        }




        private void AddSampleDataforStd(DataSet dataSet)
```

```
        {

            var dr = dataSet.Tables["Course"].NewRow();
            dr["Name"] = "BBA";
            dr["DisplayText"] = "BBA Hons";
            dataSet.Tables["Course"].Rows.Add(dr);

            var dr1 = dataSet.Tables["Student"].NewRow();
            dr1["Name"] = tbName.Text;
            dr1["Address"] = tbAddress.Text;
            dr1["EmailId"] = tbEmail.Text;
            dr1["ContactNo"] = tbContact.Text;
            dr1["CourseEnroll"] = cbProgramEnroll.Text;
            dr1["RegistrationDate"] = dpDateTime.SelectedDate;
            dataSet.Tables["Student"].Rows.Add(dr1);
        }

        private void AddofStdReport(DataSet dataSet)
        {

            var dr1 = dataSet.Tables["StudentReport"].NewRow();

dataSet.Tables["StudentReport"].ReadXml(@"C:\Informatics\Coursework\Appl
ication Development\StudentReport.xml");

            dr1["Name"] = tbName.Text;
            dr1["Address"] = tbAddress.Text;
            dr1["EmailId"] = tbEmail.Text;
            dr1["ContactNo"] = tbContact.Text;
            dr1["CourseEnroll"] = cbProgramEnroll.Text;
            dr1["RegistrationDate"] = dpDateTime.SelectedDate;
            dataSet.Tables["StudentReport"].Rows.Add(dr1);
```

```
        }

        private void AppendStdReport(DataSet dataSet)
        {
            if                  (File.Exists(@"C:\Informatics\Coursework\Application
Development\StudentReport.xml"))
            {
                var handler = new Handler();


dataSet.Tables["StudentReport"].ReadXml(@"C:\Informatics\Coursework\Appl
ication Development\StudentReport.xml");
                var dr2 = dataSet.Tables["StudentReport"].NewRow();
                dr2["RegNo"] = tbRegNo.Text;
                dr2["Name"] = tbName.Text;
                dr2["Address"] = tbAddress.Text;
                dr2["EmailId"] = tbEmail.Text;
                dr2["ContactNo"] = tbContact.Text;
                dr2["CourseEnroll"] = cbProgramEnroll.Text;
                dr2["RegistrationDate"] = dpDateTime.SelectedDate;

                dataSet.Tables["StudentReport"].Rows.Add(dr2);

//dataSet.Tables["StudentReport"].WriteXml(@"C:\Informatics\Coursework\Ap
plication Development\StudentReport.xml");
            }
            else
            {

dataSet.Tables["StudentReport"].WriteXml(@"C:\Informatics\Coursework\Appl
ication Development\StudentReport.xml");
                AppendStdReport(dataSet);
            }
```

```
        }

        private void Button_Click_1(object sender, RoutedEventArgs e)
        {
            var handler = new Handler();
            var dataSet = handler.CreateDataSet();
            AddSampleDataforStd(dataSet);
            AppendStdReport(dataSet);

            var regno = tbRegNo.Text;
            var name = tbName.Text;

            dataSet.WriteXmlSchema(@"C:\Informatics\Coursework\Application
Development\StudentCWSchema1.xml");

dataSet.Tables["Student"].WriteXml(@"C:\Informatics\Coursework\Application
Development\" + name + "CWData" + regno + ".xml");
            dataSet.Tables[2].WriteXml(@"C:\Informatics\Coursework\Application
Development\StudentReport.xml");

            write_to_file(tbRegNo.Text);

            tbRegNo.Text = read_from_file();

            ClearControls();

            MessageBox.Show("Add Successfully");
        }

        private void write_to_file(string text)
        {
```

```
            System.IO.File.WriteAllText(@"C:\Informatics\Coursework\Application
Development\count.txt", text);



        }
        private string read_from_file()
        {
            int i = 1;
            if                  (File.Exists(@"C:\Informatics\Coursework\Application
Development\count.txt"))
            {
                string                              text                          =
File.ReadAllText(@"C:\Informatics\Coursework\Application
Development\count.txt");
                i = int.Parse(text.ToString());
                i = i + 1;
            }
            else
            {
                File.WriteAllText(@"C:\Informatics\Coursework\Application
Development\count.txt", "text");
            }
            return i.ToString();


        }

        private void ClearControls()
        {
            tbName.Text = "";
            tbAddress.Text = "";
            tbContact.Text = "";
            tbEmail.Text = "";
            cbProgramEnroll.Text = "";
```

```
    }

    private void tbRegNo_TextChanged(object sender,
TextChangedEventArgs e)
    {

    }

    private void btnclear_Click(object sender, RoutedEventArgs e)
    {
        tbName.Text = "";
        tbAddress.Text = "";
        tbContact.Text = "";
        tbEmail.Text = "";
        cbProgramEnroll.Text = "";

    }

    private void btnStudentDetails_Click(object sender, RoutedEventArgs e)
    {
        StudentDetailReport studentDetailReport = new StudentDetailReport();
        studentDetailReport.Show();
    }

    private void btnReport_Click(object sender, RoutedEventArgs e)
    {
        Report report = new Report();
        report.Show();
    }

    private void btnExit_Click(object sender, RoutedEventArgs e)
    {
        this.Close();
    }
```

```
    }
}
```

Report.cs

```
using DataHandler;
using System;
using System.Collections.Generic;
using System.Data;
using System.Text;
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 StudentInformationSystems
{
    /// <summary>
    /// Interaction logic for Report.xaml
    /// </summary>
    public partial class Report : Window
    {
        public Report()
        {
            InitializeComponent();
        }

        private void btnClose1_Click(object sender, RoutedEventArgs e)
```

```
        {
            this.Close();
        }

        private void btnWeekly_Click(object sender, RoutedEventArgs e)
        {
            var handler = new Handler();
            var dataSet = new DataSet();
            dataSet.ReadXml(@"C:\Informatics\Coursework\Application
Development\StudentReport.xml");
            DataTable dtStdReport = dataSet.Tables[0];

            int total_Computing = 0;
            int total_MultimediaTechnology = 0;
            int total_NetworksandITSecuity = 0;

            DataTable dt = new DataTable("newTable");
            dt.Columns.Add("CourseEnroll", typeof(string));
            dt.Columns.Add("Total Students", typeof(int));

            for (int i = 0; i < dtStdReport.Rows.Count; i++)
            {
                string col = dtStdReport.Rows[i]["CourseEnroll"].ToString();
                if (col == "Computing")
                {
                    total_Computing++;
                }
                else if (col == "Multimedia Technology")
                {
                    total_MultimediaTechnology++;
                }
                else if (col == "Networks and IT Secuity")
                {
                    total_NetworksandITSecuity++;
```

```
            }

        }
        dt.Rows.Add("Computting", total_Computing);
        dt.Rows.Add("Multimedia Technology", total_MultimediaTechnology);
        dt.Rows.Add("Networks and IT Secuity", total_NetworksandITSecuity);
        grdReport.ItemsSource = dt.DefaultView;
    }

    private void btnTotalStd_Click(object sender, RoutedEventArgs e)
    {
        var handler = new Handler();
        var dataSet = new DataSet();
        dataSet.ReadXml(@"C:\Informatics\Coursework\Application
Development\StudentReport.xml");
        DataTable dtStdReport = dataSet.Tables[0];

        int total_Computing = 0;
        int total_MultimediaTechnology = 0;
        int total_NetworksandITSecuity = 0;

        DataTable dt = new DataTable("newTable");
        dt.Columns.Add("CourseEnroll", typeof(string));
        dt.Columns.Add("Total Students", typeof(int));

        for (int i = 0; i < dtStdReport.Rows.Count; i++)
        {
            string col = dtStdReport.Rows[i]["CourseEnroll"].ToString();
            if (col == "Computing")
            {
                total_Computing++;
            }
            else if (col == "Multimedia Technology")
            {
```

```
                total_MultimediaTechnology++;
            }
            else if (col == "Networks and IT Secuity")
            {
                total_NetworksandITSecuity++;
            }

        }
        dt.Rows.Add("Computting", total_Computing);
        dt.Rows.Add("Multimedia Technology", total_MultimediaTechnology);
        dt.Rows.Add("Networks and IT Secuity", total_NetworksandITSecuity);
        grdReport.ItemsSource = dt.DefaultView;
    }

    private void grdReport_SelectionChanged(object sender,
SelectionChangedEventArgs e)
    {

    }

    private void btnChart_Click(object sender, RoutedEventArgs e)
    {
        Chartt chartt = new Chartt();
        chartt.Show();
    }
  }

}
```

StudentDetailsReport.cs

```
using DataHandler;
```

```
using Microsoft.Win32;
using System;
using System.Collections.Generic;
using System.Data;
using System.IO;
using System.Text;
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 StudentInformationSystems
{
    /// <summary>
    /// Interaction logic for StudentDetailReport.xaml
    /// </summary>
    public partial class StudentDetailReport : Window
    {
        public StudentDetailReport()
        {
            InitializeComponent();

        }
        private void LoadStudentData()
        {

            if       (System.IO.File.Exists(@"C:\Informatics\Coursework\Application
Development\StudentReport.xml"))
            {
                var handler = new Handler();
```

```
            var dataSet = new DataSet();

            dataSet.ReadXml(@"C:\Informatics\Coursework\Application
Development\StudentReport.xml");

            DataTable dtStdReport = new DataTable();
            dtStdReport = dataSet.Tables[0];
            grdStudentDetails.ItemsSource = dtStdReport.DefaultView;
        }

    }


    private void btnAddStudent_Click(object sender, RoutedEventArgs e)
    {
        MainWindow mainWindow = new MainWindow();
        mainWindow.Show();
    }

    private void btnStudentRecord_Click(object sender, RoutedEventArgs e)
    {
        LoadStudentData();
    }

    private void btnSortRegDate_Click(object sender, RoutedEventArgs e)
    {
        var dataSet = new DataSet();
        dataSet.ReadXml(@"C:\Informatics\Coursework\Application
Development\StudentReport.xml");
        DataTable DataTable = dataSet.Tables["StudentReport"];
        DataTable.DefaultView.Sort = "RegistrationDate Asc";
        grdStudentDetails.ItemsSource = DataTable.DefaultView;
```

```
        }

        private void btnSortName_Click_1(object sender, RoutedEventArgs e)
        {
            var dataSet = new DataSet();
            dataSet.ReadXml(@"C:\Informatics\Coursework\Application
Development\StudentReport.xml");
            DataTable DataTable = dataSet.Tables["StudentReport"];
            DataTable.DefaultView.Sort = "Name Asc";
            grdStudentDetails.ItemsSource = DataTable.DefaultView;
        }

        private void btnClose_Click(object sender, RoutedEventArgs e)
        {
            this.Close();
        }
    }
}
```

Chartt.cs

```
using DataHandler;
using System;
using System.Collections.Generic;
using System.Data;
using System.Text;
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 StudentInformationSystems
{
    /// <summary>
    /// Interaction logic for Chartt.xaml
    /// </summary>
    public partial class Chartt : Window
    {
        public Chartt()
        {
            InitializeComponent();

            var handler = new Handler();
            var dataSet = new DataSet();
            dataSet.ReadXml(@"C:\Informatics\Coursework\Application
Development\StudentReport.xml");
            DataTable dtStdReport = dataSet.Tables[0];

            int total_Computing = 0;
            int total_MultimediaTechnology = 0;
            int total_NetworksandITSecuity = 0;

            DataTable dt = new DataTable("newTable");
            dt.Columns.Add("CourseEnroll", typeof(string));
            dt.Columns.Add("Total Students", typeof(int));

            for (int i = 0; i < dtStdReport.Rows.Count; i++)
            {
                string col = dtStdReport.Rows[i]["CourseEnroll"].ToString();
                if (col == "Computing")
                {
                    total_Computing++;
                }
```

```
            else if (col == "Multimedia Technology")
            {
                total_MultimediaTechnology++;
            }
            else if (col == "Networks and IT Secuity")
            {
                total_NetworksandITSecuity++;
            }

        }
        dt.Rows.Add("Computting", total_Computing);
        dt.Rows.Add("Multimedia Technology", total_MultimediaTechnology);
        dt.Rows.Add("Networks and IT Secuity", total_NetworksandITSecuity);

        ((BarSeries) totalChart.Series[0]).ItemsSource = new
KeyValuePair<string, int>[] {
            new KeyValuePair<string, int>("Computting", total_Computing),
        new KeyValuePair<string, int>("Multimedia Technology",
total_MultimediaTechnology),
        new KeyValuePair<string, int>("Networks and IT Secuity",
total_NetworksandITSecuity)
    };
       }
    }
}
```