

# Informatics College Pokhara



informatics  
college pokhara

**Application Development**

**CS6004NI**

**Course Work 1**

**Submitted By: Agnesh Lama**  
**London Met ID: 17030684**

**Submitted To: Ishwor Sapkota**  
**Module Leader**

Component Grade and Comments	
<b>A. Implementation of Application</b>	
<b>User Interface and proper controls used for designing</b>	User Interface is complete but not separated and have proper use of controls
<b>Manual data entry or import from csv</b>	appropriate use of data types but missing some properties required or missing CRUD operation
<b>Data Validation</b>	No validation at all
<b>Enrollment Report &amp; weekly report in tabular format</b>	very poorly executed reports and data not shown accurately
<b>Course wise enrollment report &amp; Chart display</b>	Very poorly designed and only contains one report format with in appropriate data
<b>Algorithm used for sorting &amp; proper sorting of data</b>	Default sorting provided by .net is used
<b>B. Documentation</b>	
<b>User Manual for running the application</b>	User Manual is average. Includes description for all interfaces

<b>Application architecture &amp; description of the classes and methods used</b>	very poorly explained.
<b>Flow chart, algorithms and data structures used</b>	very poorly explained and no diagrammatic representation
<b>Reflective essay</b>	Very poorly written

**C. Programming Style**

<b>Clarity of code, Proper Naming convention &amp; comments</b>	Code is poorly written and lacks comments
<b>System Usability</b>	System can't be used and have issues

<b>Overall Grade:</b>	<b>C</b>	<b>C</b>
-----------------------	----------	----------

**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



Application Development  
CS6004NP  
Coursework 1

**Submitted By:**

Student Name: Agnesh Lama

London Met ID: 17030684

Group: L3C2

Date: 10<sup>th</sup> January 2020

**Submitted To:**

Mr. Ishwor Sapkota

## Table of Contents

1. Introduction .....	1
2. User Manual.....	2
3. Journal Articles .....	7
4. System Architecture .....	10
4.1 Class Diagram .....	10
4.2 Flow Chart .....	14
5. Sorting Algorithm.....	16
6. Conclusion.....	18
7. Bibliography .....	19
8. Appendix .....	20

## List of Figure

Figure 1 LogIn .....	2
Figure 2 MainWindow.....	2
Figure 3 Student Details .....	3
Figure 4 Import Student Details.....	3
Figure 5 Import CSV file .....	4
Figure 6 Total Students .....	4
Figure 7 Report .....	5
Figure 8 Sort By Name .....	5
Figure 9 Sort By Date.....	6
Figure 10 Chart .....	6
Figure 11 Journal 1 .....	7
Figure 12 Journal 2 .....	8
Figure 13 Journal 3 .....	9
Figure 14 Class Diagram.....	10
Figure 15 Enroll student flowchart.....	14
Figure 16 CSV file flowchart.....	15
Figure 17 Bubble Sort .....	17

## List of Table

Table 1 MainWindow.....	10
Table 2 BarDiagram .....	11
Table 3 EnrollStudent.....	11
Table 4 Home.....	12
Table 5 Report .....	12
Table 6 ImportStudentDetails.....	13
Table 7 TotalStudent.....	13

## 1. Introduction

This course work is about Student Information System where you can enrol different students and can choose different courses to enrol. You can generate report of all details and can look at number of students in each course. There is also a chart that shows the details of the student.

This course work is done in Visual Studio using C#. This project is based on WPF(Windows Presentation Foundation).

Visual Studio is an Integrated Development Environment(IDE) developed by Microsoft to develop GUI(Graphical User Interface), console, Web applications, web apps, mobile apps, cloud, and web services, etc. With the help of this IDE, you can create managed code as well as native code. It uses the various platforms of Microsoft software development software like Windows store, Microsoft Silverlight, and Windows API, etc. It is not a language-specific IDE as you can use this to write code in C#, C++, VB(Visual Basic), Python, JavaScript, and many more languages. It provides support for 36 different programming languages. It is available for Windows as well as for macOS (Anon., n.d.).

Windows Presentation Foundation(WPF) is a development framework used to create a desktop application. It is a part of the .NET framework. The WPF has a resolution-independent and vector-based rendering engine which is helpful to deal with modern graphics hardware. The latest version of WPF is 4.6. In this framework, UI of the application is designed in XAML language and Application logic is Written in C# programming language (Anon., n.d.).

## 2. User Manual

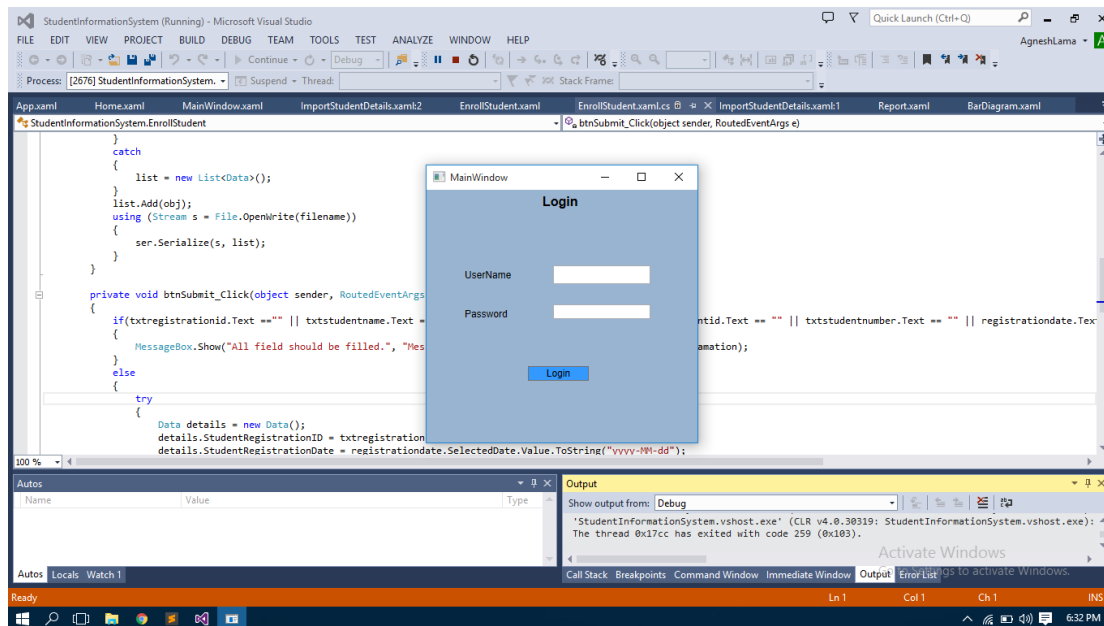


Figure 1 Login

This is the first window that appears after we run the code. This is the login form where the user enters the username and the password to login. After you are logged in, you will see the main window appear.

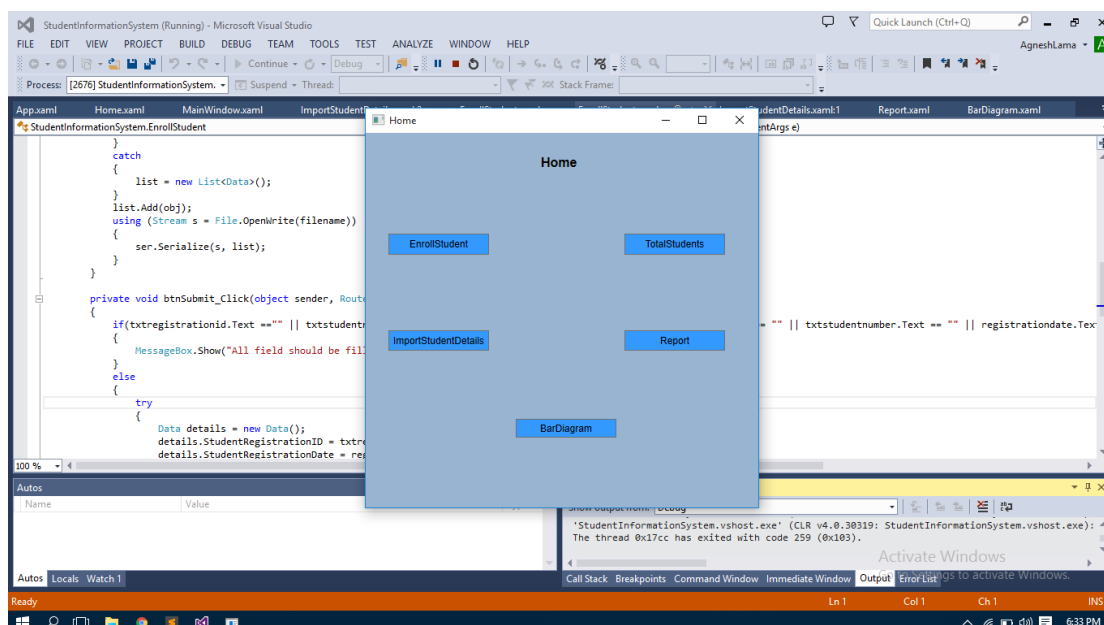


Figure 2 MainWindow

This is the main window of the project. From this window you can go to all the other windows and do as the user wants.



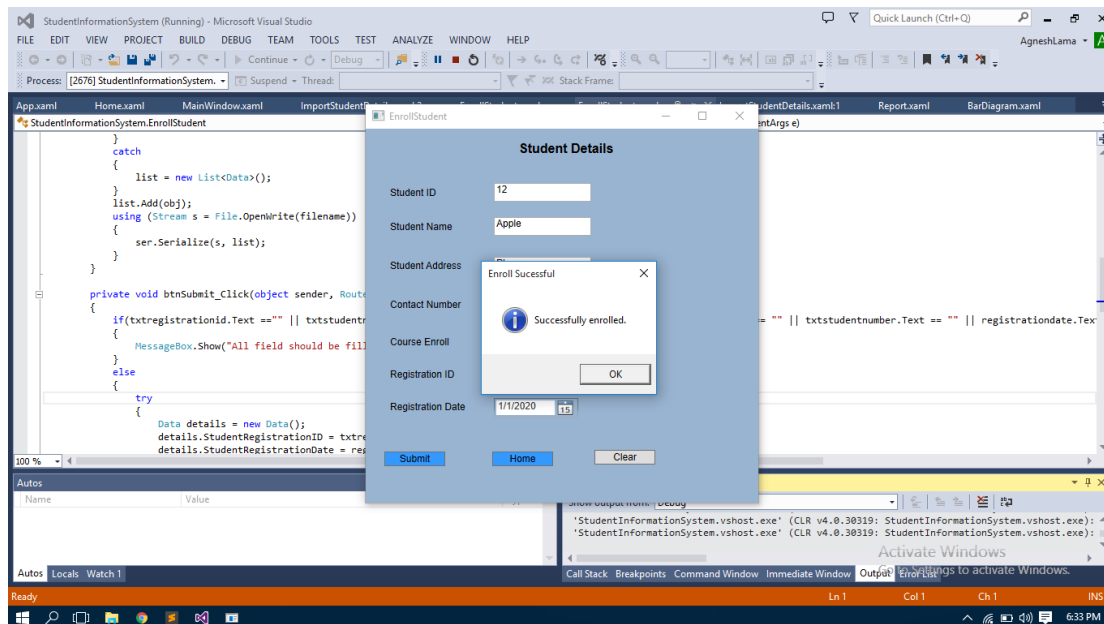


Figure 3 Student Details

In this window you can enroll the student by filling all the fields and clicking on the submit button. After you click on the submit button, a dialog box will appear that will have some message written to show that the student is enrolled successfully.

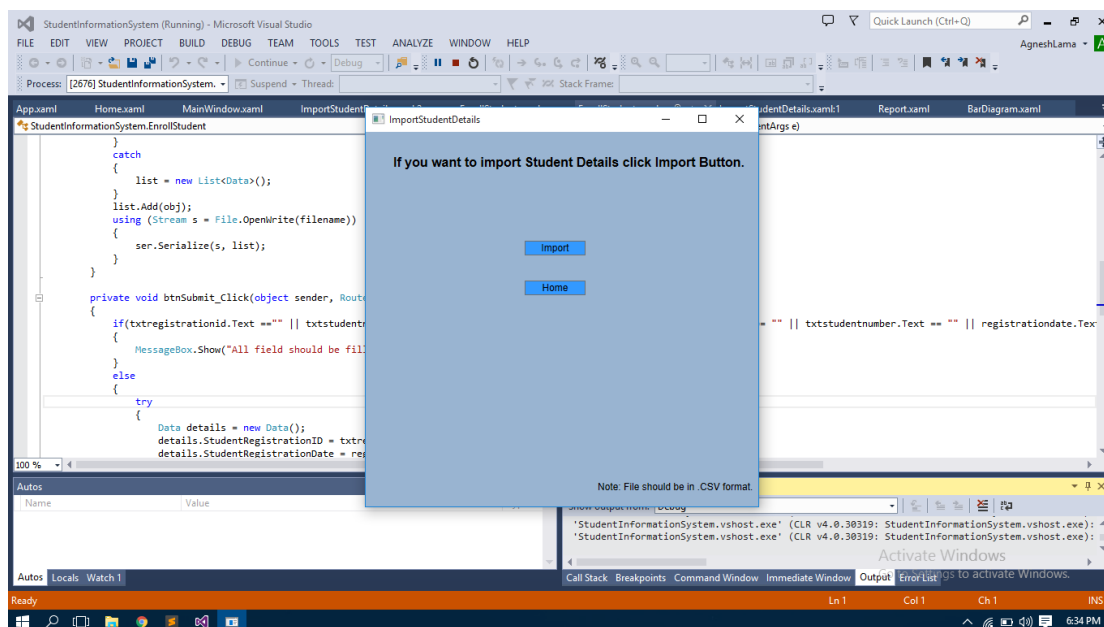


Figure 4 Import Student Details

In this window, if the user wants to import student details from CSV file, they can import that file by clicking on the import button. Or user can go back into the main page by clicking on the home button.

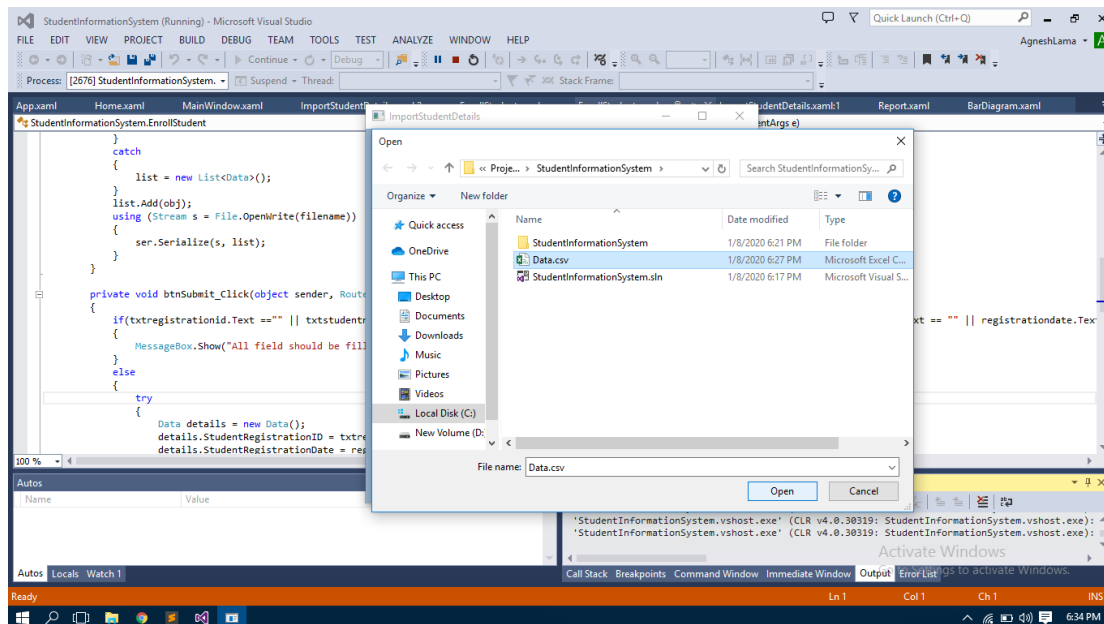


Figure 5 Import CSV file

This shows how CSV file is imported into the project. Once you choose the required CSV file, you click on open and then the file will be imported.

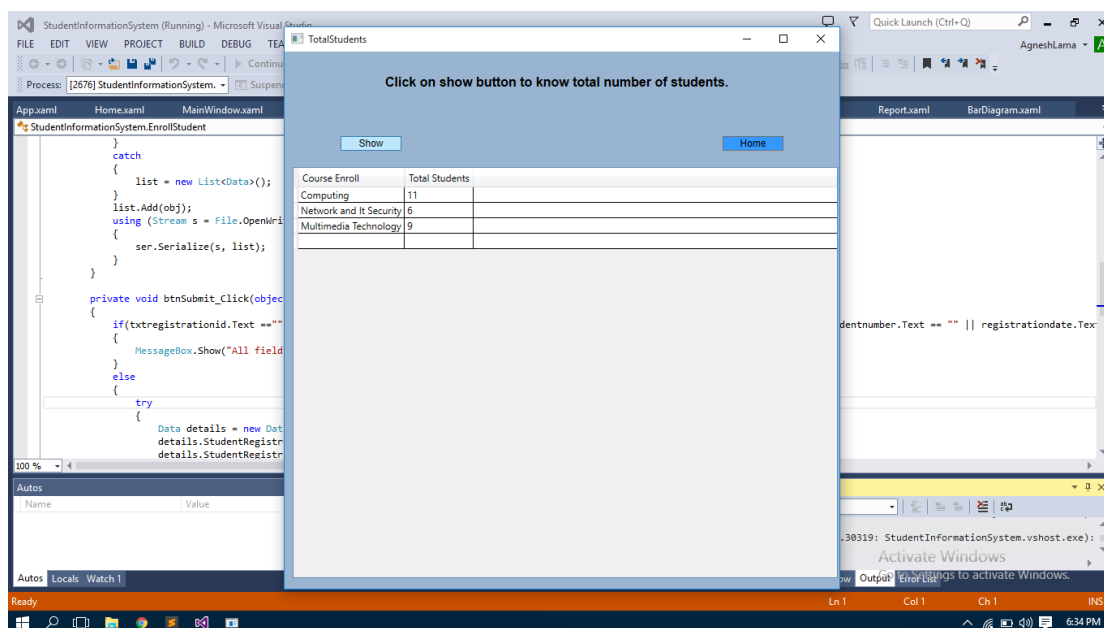


Figure 6 Total Students

This is the windows which shows total number of students enrolled in each course. User can look at this information once the user clicks on the show button on the screen. And can go back to the main page by clicking on home button.

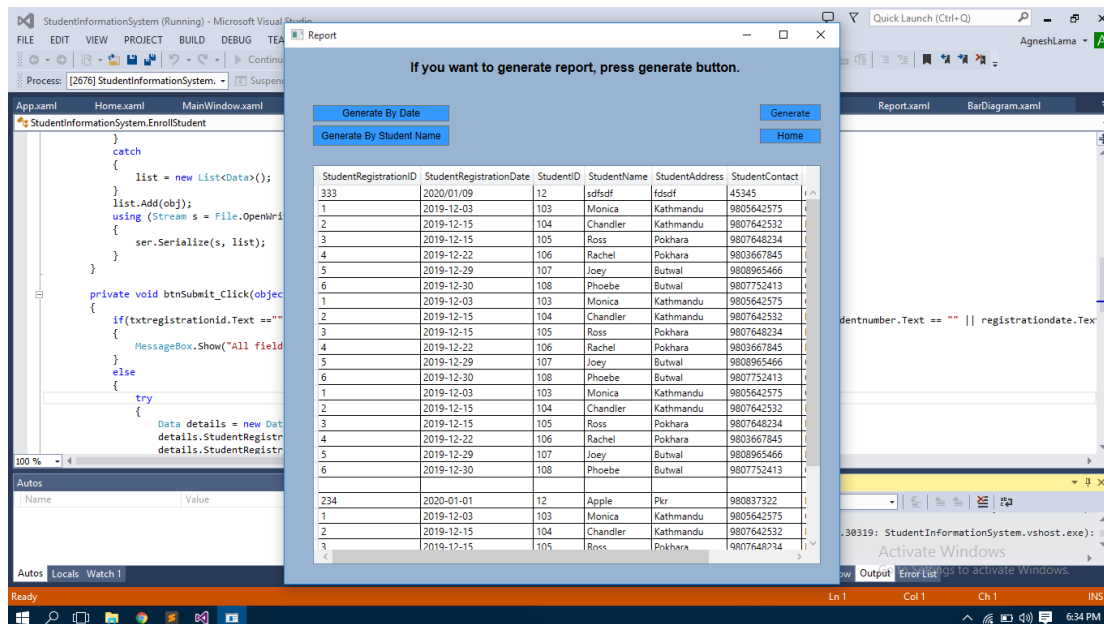


Figure 7 Report

This is the window that shows report of all the students enrolled. This information is showed once the user clicks on generate button.

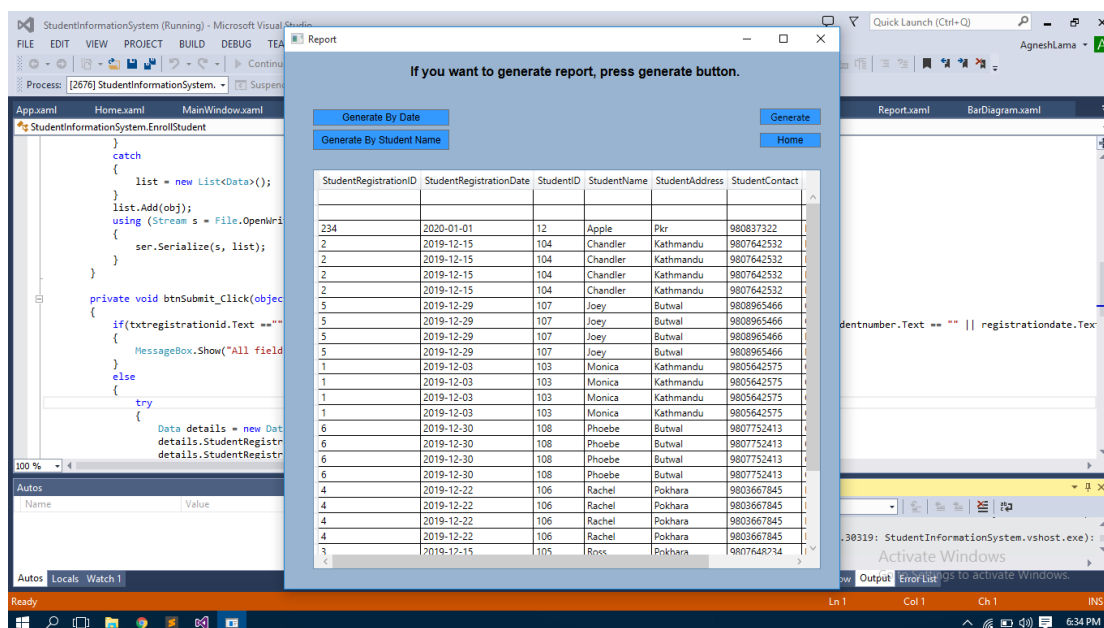


Figure 8 Sort By Name

When the report is generated, all the information showed is not in order. You can sort them by pressing on generate by name button if user want to sort by alphabetical order.

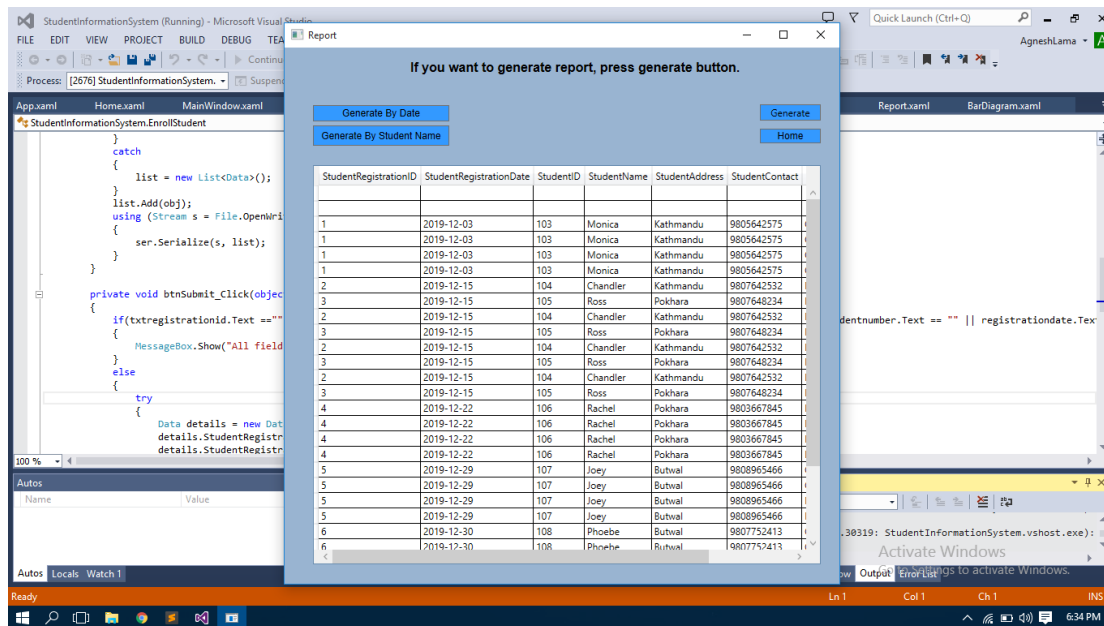


Figure 9 Sort By Date

If user wants to generate report by date then you can click on generate by date button. And if you want to go back you can press home button.

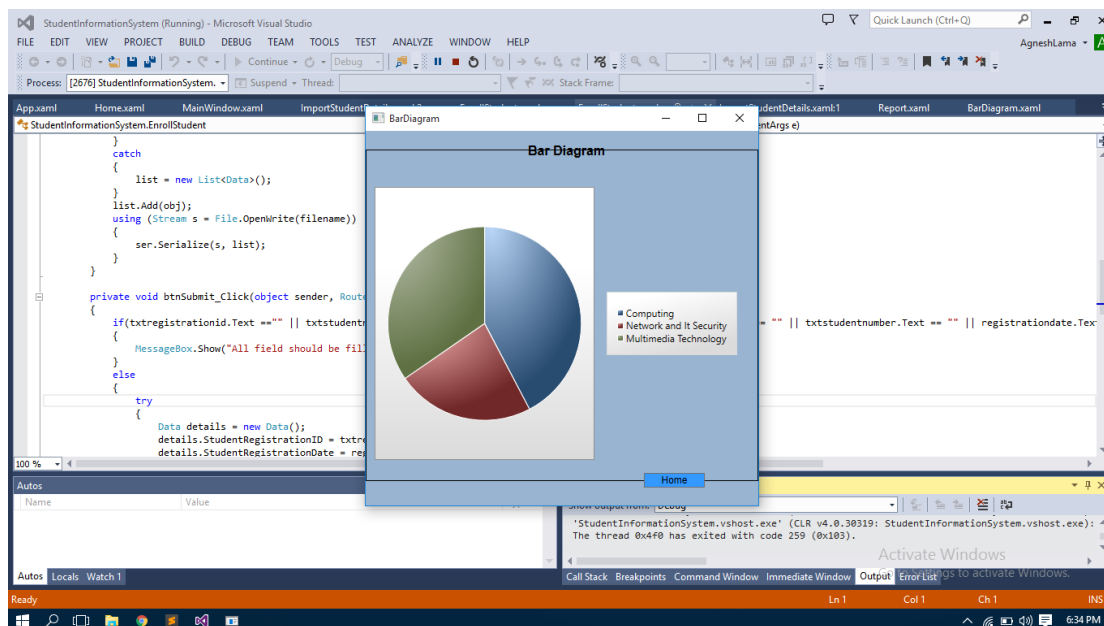


Figure 10 Chart

This is pie chart of all the student details.

### 3. Journal Articles

These are the few journals I went through to help me finish my coursework. All these journals are based on Student Information System which is similar to our coursework. I got some ideas after reading these journals on how these types of system help in managing student's details and data.

- WEB BASED STUDENT INFORMATION MANAGEMENT SYSTEM (Lubanga, 2018).

#### **WEB BASED STUDENT INFORMATION MANAGEMENT SYSTEM IN UNIVERSITIES: EXPERIENCES FROM MZUZU UNIVERSITY**

**Symon C. Lubanga**

Research Assistant

Department of Library and Information Science Mzuzu University, Mzuzu, Malawi

**Winner Dominic Chawinga**

Lecturer, Department of Library and Information Science  
Mzuzu University, Mzuzu, Malawi

**Felix Majawa**

Senior Lecturer, Department of Library and Information Science  
Mzuzu University, Mzuzu, Malawi

and

**Sellina Kapondera**

Lecturer, Department of Library and Information Science  
Mzuzu University, Mzuzu, Malawi

#### **Abstract**

Over the past few decades, universities all over the world have been experiencing new paradigms in the way they handle and manage students' information due to the proliferation of ICTs and its applications such as web-based student information systems. With the adoption of such systems as the Online Student Information System (OSIS) in academic institutions, the experience is that it has now become easy to harness and fast track all students' records in one centralized database via the internet technology. While the benefits of OSIS seem to be celebrated, it has been a hustle for most universities in Africa to completely go digital in their operations due poor ICT infrastructures that seem to be prevalent in the region. In Malawi, a social survey was conducted with the aim of assessing the Mzuzu University Student Online Management System (SOMS) from the perspective of students. The study applied the principles of both qualitative and quantitative research approaches. The principal data collection methods were questionnaires and follow up interviews. The study population was made up of third year students in the Faculty of Information Science and Communications and the Director of ICT services at Mzuzu University. The quantitative data collected were analysed and presented using Microsoft Excel Package. Thematic analysis technique was used to analyse the data collected through interviews. The study revealed that Mzuzu University SOMS has one prime service which is online registration and admissions, with online examination results access, student profile and finance as add-ons. The system benefits students as it has cut the time spent during registration periods in every new semester. Students faced the following major challenges when using the system; server loads as more students

655

*Figure 11 Journal 1*

- A Student Information Management System Based on Fingerprint Identification (Pengtao Yang, 2017).

**A Student Information Management System Based on Fingerprint Identification and Data Security Transmission**

Pengtao Yang, Guiling Sun, Jingfei He, Peiyao Zhou, and Jiangjiang Liu

College of Electronic Information and Optical Engineering, Nankai University, Tianjin 300350, China

Correspondence should be addressed to Guiling Sun; [sungl@nankai.edu.cn](mailto:sungl@nankai.edu.cn)

Received 18 February 2017; Revised 17 July 2017; Accepted 17 August 2017; Published 19 September 2017

Academic Editor: Liangmin Wang

Copyright © 2017 Pengtao Yang et al. This is an open access article distributed under the [Creative Commons Attribution License](#), which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

Full-Text XML

Linked References

How to Cite this Article

Complete Special Issue

Order Reprints

Views

Citations

ePub

PDF

2,734

0

15

648

**Abstract**

A new type of student information management system is designed to implement student information identification and management based on fingerprint identification. In order to ensure the security of data transmission, this paper proposes a data encryption method based on an improved AES algorithm. A new S-box is cleverly designed, which can significantly reduce the encryption time by improving ByteSub, ShiftRow, and MixColumn in the round transformation of the traditional AES algorithm with the process of look-up table. Experimental results show that the proposed algorithm can significantly improve the encryption time compared with the traditional AES algorithm.

Figure 12 Journal 2

- Student Information Systems Are Integrating More Functions (Willis, n.d.).

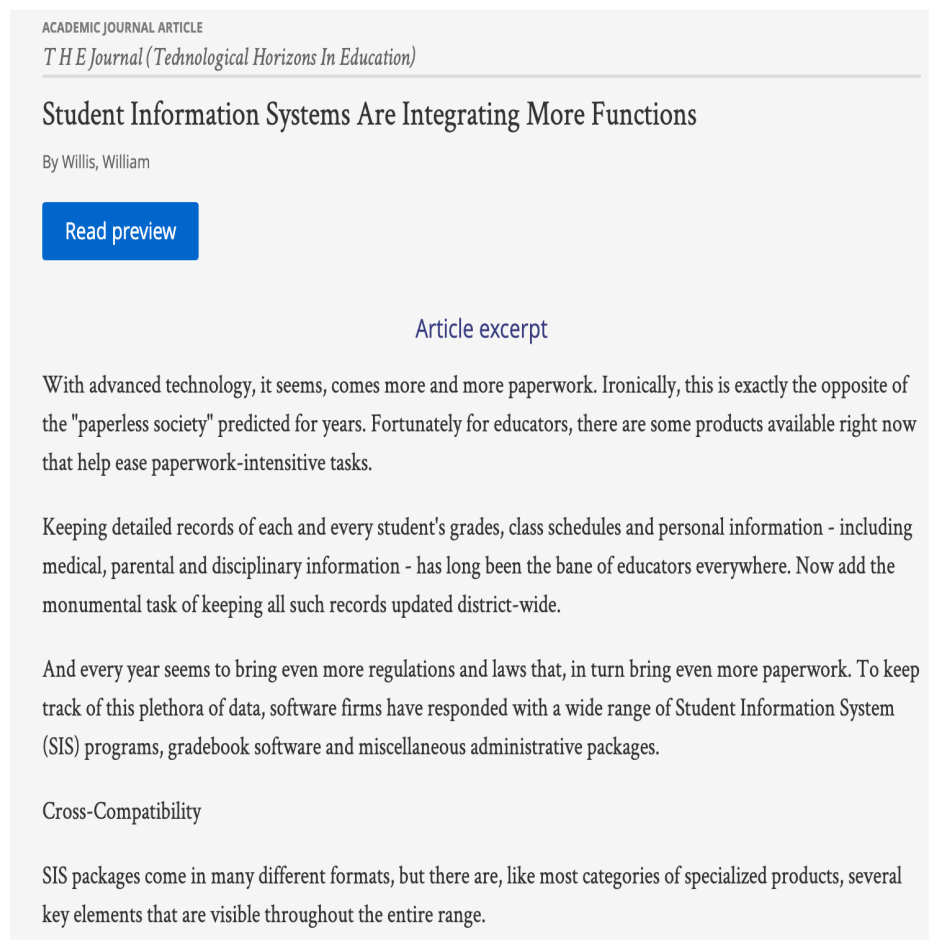


Figure 13 Journal 3

4. System Architecture

4.1 Class Diagram

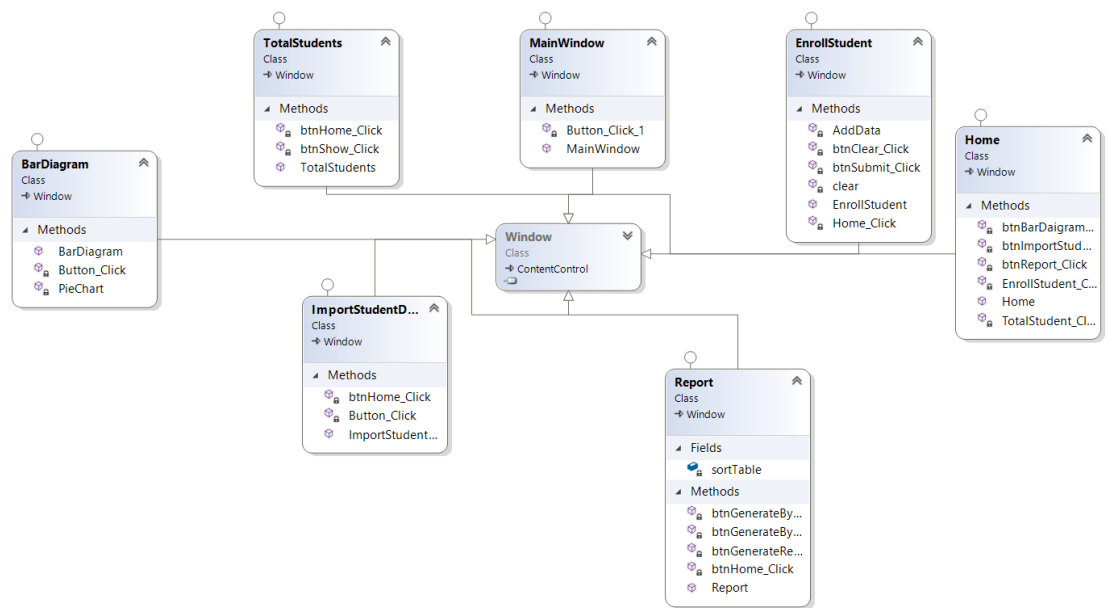


Figure 14 Class Diagram

This is the main class diagram which shows all the classes used in the project. There are total of seven main classes. They are: TotalStudents, MainWindow, EnrollStudent, Home, BarDiagram, ImportStudentDetails, Report.

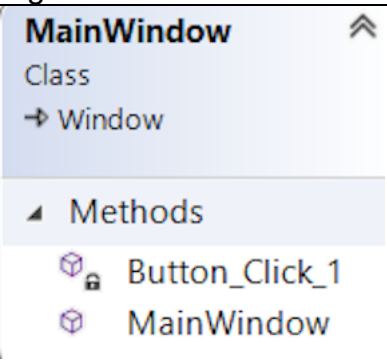
Methods	Description	Figure
Button_Click_1	The Button_Click_1 method deals with all the action which is done when clicking on that button.	

Table 1 MainWindow



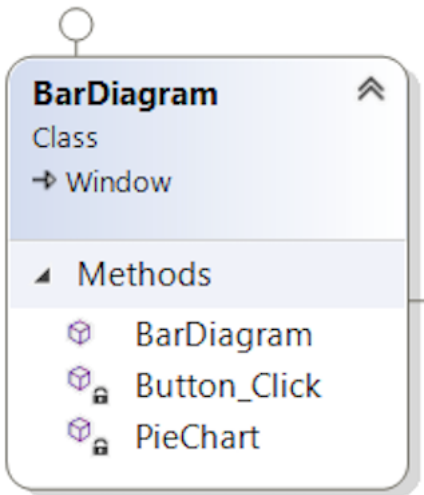
Methods	Description	Figure
-Button_Click_1 -PieChart	- The Button_Click_1 method deals with all the action which is done when clicking on that button. - The PieChart method shows the chart for the report.	

Table 2 BarDiagram

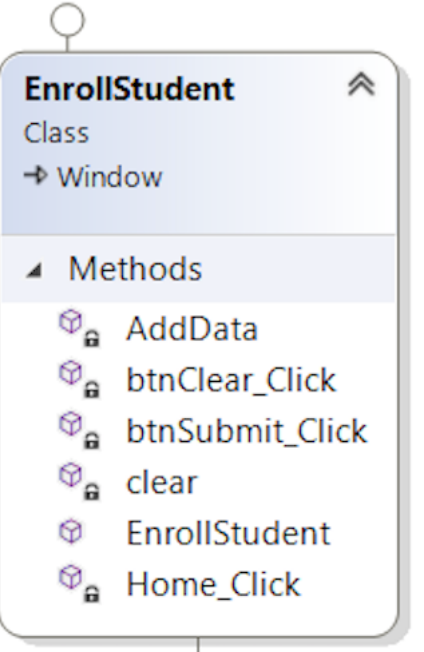
Methods	Description	Figure
-AddData -btnClear_Click -btnSubmit_Click -clear -EnrollStudent -Home_Click	- AddData method adds data into the file. - btnClear_Click method clears the data written in the form for further use. - btnSubmit_Click method submits the data entered by the user. - EnrollStudent method tells if the student is enrolled successfully or not. - Home_Click method takes you back into home page.	

Table 3 EnrollStudent

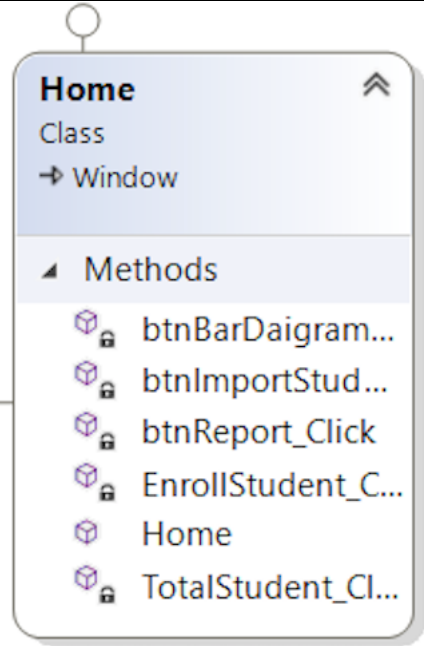
Methods	Description	Figure
-btnBar_Diagram -btnImportStudent -btnReport_Click -Home - -EnrollStudent_Click -TotalStudent_Click	- btnBar_Diagram takes you to bar diagram window. - btnImportStudent takes you to import student window. - btnReport_Click takes you to report window. -EnrollStudent_Click takes you to enroll window. - TotalStudent_Click takes you to total student window.	 <p>The figure shows a UML Class Diagram for the 'Home' class. It is a 'Class' and a 'Window'. The 'Methods' section lists: btnBarDaigram..., btnImportStud..., btnReport_Click, EnrollStudent_C..., Home, and TotalStudent_Cl... Each method is preceded by a small icon representing a button.</p>

Table 4 Home

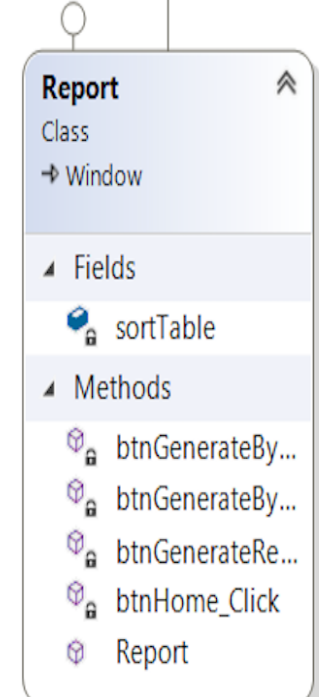
Methods	Description	Figure
-btnGenerateByDate - -btnGenerateByName -btnGenerateReport -btnHome_Click -Report	- btnGenerateByDate generates report in order of date. - btnGenerateByName generates report in alphabetical order. - btnGenerateReport generates report of all the student. -btnHome_Click takes you back to home page.	 <p>The figure shows a UML Class Diagram for the 'Report' class. It is a 'Class' and a 'Window'. The 'Fields' section lists: sortTable. The 'Methods' section lists: btnGenerateBy..., btnGenerateBy..., btnGenerateRe..., btnHome_Click, and Report. Each method is preceded by a small icon representing a button.</p>

Table 5 Report

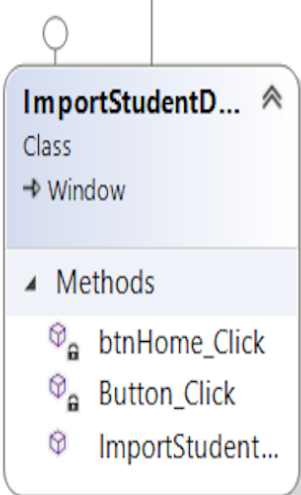
Methods	Description	Figure
- ImportStudentDetails_Click -btnHome_Click -Button_Click	- ImportStudentDetails_Click imports CVS file. -btnHome_Click takes you back to home page.	 <p>The figure shows a UML Class Diagram for the <b>ImportStudentDetails</b> class. The class is labeled as a <b>Class</b> and has a <b>Window</b> icon. It contains three methods: <b>btnHome_Click</b>, <b>Button_Click</b>, and <b>ImportStudent...</b>. Each method is preceded by a small icon representing a button click.</p>

Table 6 ImportStudentDetails

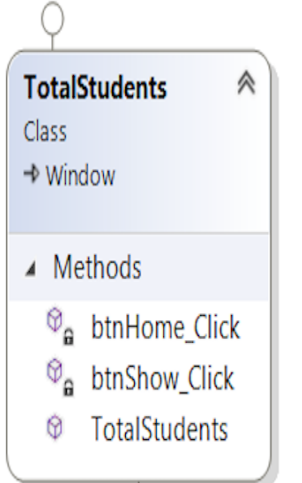
Methods	Description	Figure
-btnHome_Click -btnShow_Click -TotalStudents	-btnShow_Click shows total number of students on each course. -btnHome_Click takes you back to home page.	 <p>The figure shows a UML Class Diagram for the <b>TotalStudents</b> class. The class is labeled as a <b>Class</b> and has a <b>Window</b> icon. It contains three methods: <b>btnHome_Click</b>, <b>btnShow_Click</b>, and <b>TotalStudents</b>. Each method is preceded by a small icon representing a button click.</p>

Table 7 TotalStudent

## 4.2 Flow Chart

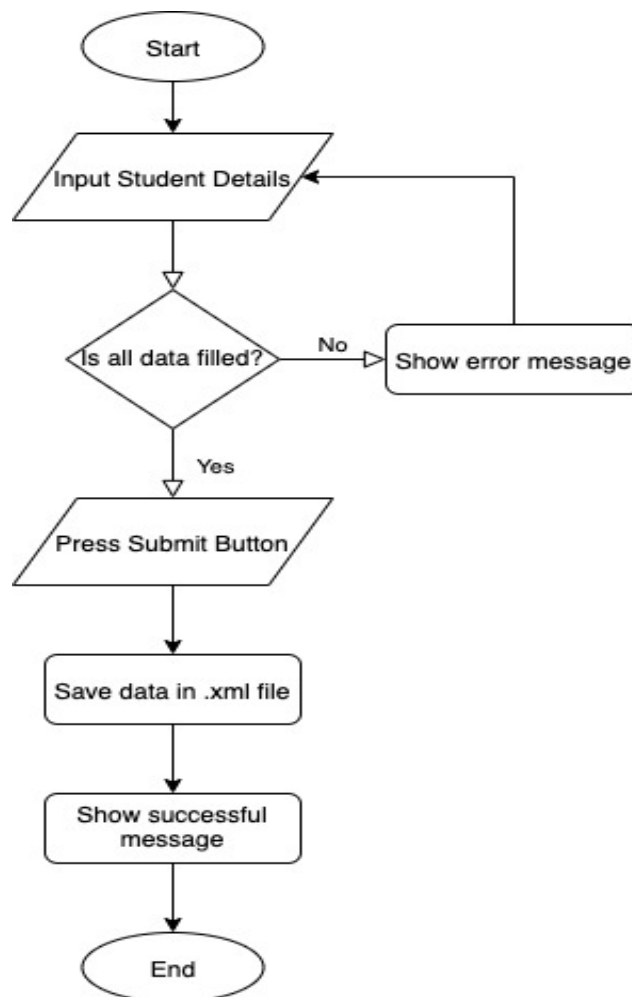


Figure 15 Enroll student flowchart

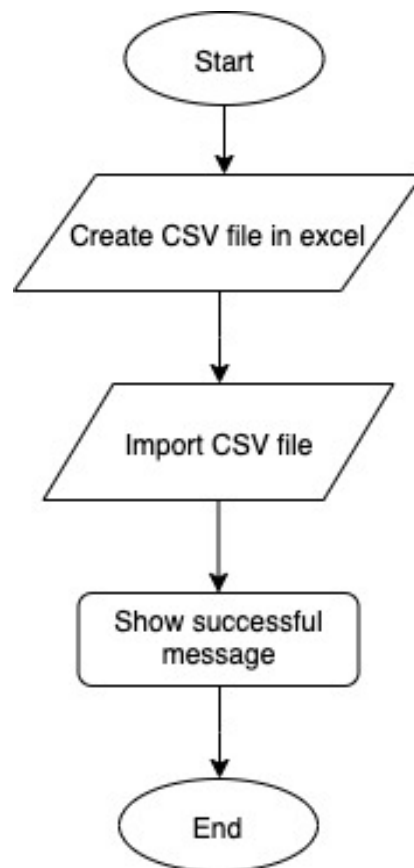


Figure 16 CSV file flowchart

## 5. Sorting Algorithm

Bubble Sort:

Bubble Sort is the simplest sorting algorithm that works by repeatedly swapping the adjacent elements if they are in wrong order.

Example:

First Pass:

( 5 1 4 2 8 )  $\rightarrow$  ( 1 5 4 2 8 ), Here, algorithm compares the first two elements, and swaps since  $5 > 1$ .

( 1 5 4 2 8 )  $\rightarrow$  ( 1 4 5 2 8 ), Swap since  $5 > 4$

( 1 4 5 2 8 )  $\rightarrow$  ( 1 4 2 5 8 ), Swap since  $5 > 2$

( 1 4 2 5 8 )  $\rightarrow$  ( 1 4 2 5 8 ), Now, since these elements are already in order ( $8 > 5$ ), algorithm does not swap them.

Second Pass:

( 1 4 2 5 8 )  $\rightarrow$  ( 1 4 2 5 8 )

( 1 4 2 5 8 )  $\rightarrow$  ( 1 2 4 5 8 ), Swap since  $4 > 2$

( 1 2 4 5 8 )  $\rightarrow$  ( 1 2 4 5 8 )

( 1 2 4 5 8 )  $\rightarrow$  ( 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 Pass:

( 1 2 4 5 8 )  $\rightarrow$  ( 1 2 4 5 8 )

( 1 2 4 5 8 )  $\rightarrow$  ( 1 2 4 5 8 )

( 1 2 4 5 8 )  $\rightarrow$  ( 1 2 4 5 8 )

( 1 2 4 5 8 )  $\rightarrow$  ( 1 2 4 5 8 ) (Anon., n.d.)

i = 0	j	0	1	2	3	4	5	6	7
	0	5	3	1	9	8	2	4	7
	1	3	5	1	9	8	2	4	7
	2	3	1	5	9	8	2	4	7
	3	3	1	5	9	8	2	4	7
	4	3	1	5	8	9	2	4	7
	5	3	1	5	8	2	9	4	7
	6	3	1	5	8	2	4	9	7
i = 1	j	0	1	2	3	4	5	6	7
	0	3	1	5	8	2	4	7	9
	1	1	3	5	8	2	4	7	
	2	1	3	5	8	2	4	7	
	3	1	3	5	8	2	4	7	
	4	1	3	5	2	8	4	7	
	5	1	3	5	2	4	8	7	
i = 2	j	0	1	2	3	4	5	6	7
	0	1	3	5	2	4	7	8	
	1	1	3	5	2	4	7		
	2	1	3	5	2	4	7		
	3	1	3	2	5	4	7		
	4	1	3	2	4	5	7		
i = 3	j	0	1	2	3	4	5	6	7
	0	1	3	2	4	5	7		
	1	1	3	2	4	5			
	2	1	2	3	4	5			
	3	1	2	3	4	5			
i = 4	j	0	1	2	3	4	5	6	7
	0	1	2	3	4	5			
	1	1	2	3	4				
	2	1	2	3	4				
i = 5	j	0	1	2	3	4	5	6	7
	0	1	2	3	4				
	1	1	2	3					
i = 6	j	0	1	2	3	4	5	6	7
	0	1	2	3					
	1	1	2						

Figure 17 Bubble Sort

## **6. Conclusion**

This course work taught me a lot about C# and how to use Visual Studio. I got to learn many new things while doing this course work. I had to do a lot of research for this course work to be completed. I went through many articles and websites. I also watched a lot of videos on YouTube to learn to do this course work. Though it was difficult at first, I was able to complete this coursework on time. This kind of system should be more implemented on daily basis because it is very useful and is very easy and less time consuming to do your task. These kinds of system may be a great help in schools and colleges to store data of their students.



## 7. Bibliography

Anon., n.d. *GeeksforGeeks*. [Online]

Available at: <https://www.geeksforgeeks.org/bubble-sort/>

[Accessed 2020].

IQVIS, 2017. *IQVIS*. [Online]

Available at: <https://www.iqvis.com/blog/9-powerful-examples-of-artificial-intelligence-in-use-today/>

[Accessed 10 12 2019].

Lubanga, S. C., 2018. Research Gate. *WEB BASED STUDENT INFORMATION MANAGEMENT SYSTEM IN UNIVERSITIES: EXPERIENCES FROM MZUZU UNIVERSITY*.

Pengtao Yang, G. S. J. H. P. Z. a. J. L., 2017. A Student Information Management System Based on Fingerprint Identification and Data Security Transmission. *Journal of Electrical and Computer Engineering*, Volume 2017, p. 6.

Tuts+, E., 2019. *Envato Tuts+*. [Online]

Available at: <https://code.tutsplus.com/tutorials/introduction-to-unity3d--mobile-10752>

[Accessed 10 12 2019].

Willis, W., n.d. Student Information Systems Are Integrating More Functions. *T H E Journal (Technological Horizons In Education)*.

## 8. Appendix

- Bar Diagram

```

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.Windows.Controls.DataVisualization.Charting;

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

        private void PieChart()
        {
            var dataSet = new DataSet();

            dataSet.ReadXml(@"C:\Users\Shristita
Kunwar\Documents\Visual Studio
2013\Projects\StudentInformationSystem\StudentInformationSystem\
\bin\Debug\StudentDetails.xml");

            DataTable dataTable = dataSet.Tables[0];
            int total_Com = 0;
            int total_Net = 0;
            int total_Mul = 0;

            DataTable dt = new DataTable("tbl");

```

```

dt.Columns.Add("Course Enroll", typeof(String));
dt.Columns.Add("Total Students", typeof(int));

for (int i = 0; i < dataTable.Rows.Count; i++)
{
    String col = dataTable.Rows[i][6].ToString();

    if (col == "Computing")
    {
        total_Com++;
    }
    else if (col.Equals("Network and It Security"))
    {
        total_Net++;
    }
    else if (col.Equals("Multimedia Technologies"))
    {
        total_Mul++;
    }
}

((PieSeries)pieChart).ItemsSource =
    new KeyValuePair<string, int>[]{
        new KeyValuePair<string,int>("Computing", total_Com),
        new KeyValuePair<string,int>("Network and It Security",
total_Net),
        new KeyValuePair<string,int>("Multimedia Technology",
total_Mul) };
}

private void Button_Click(object sender, RoutedEventArgs e)
{
    Home home = new Home();
    home.Show();
    this.Close();
}
}
}

```

- Enroll Student

```
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.Xml.Serialization;
using System.IO;

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

        private void Home_Click(object sender, RoutedEventArgs e)
        {
            Home home = new Home();
            home.Show();
            this.Close();
        }

        static void AddData(Data obj, string filename)
        {
            XmlSerializer ser = new XmlSerializer(typeof(List<Data>));
            List<Data> list = null;
            try
            {
                using (Stream s = File.OpenRead(filename))
                {
                    list = ser.Deserialize(s) as List<Data>;
                }
            }
            catch
            {
            }
        }
    }
}
```

```

        list = new List<Data>();
    }
    list.Add(obj);
    using (Stream s = File.OpenWrite(filename))
    {
        ser.Serialize(s, list);
    }
}

private void btnSubmit_Click(object sender, RoutedEventArgs e)
{
    if(txtregistrationid.Text == "" || txtstudentname.Text == "" ||
txtstudentaddress.Text == "" || txtstudentid.Text == "" ||
txtstudentnumber.Text == "" || registrationdate.Text == "" ||
courseenroll.Text == "")
    {
        MessageBox.Show("All field should be filled.", "Message",
MessageBoxButton.OK, MessageBoxImage.Exclamation);
    }
    else
    {
        try
        {
            Data details = new Data();
            details.StudentRegistrationID = txtregistrationid.Text;
            details.StudentRegistrationDate =
registrationdate.SelectedDate.Value.ToString("yyyy-MM-dd");
            details.StudentID = txtstudentid.Text;
            details.StudentName = txtstudentname.Text;
            details.StudentAddress = txtstudentaddress.Text;
            details.StudentContact = txtstudentnumber.Text;
            details.StudentCourse = courseenroll.Text;
            AddData(details, "StudentDetails.xml");

            MessageBox.Show("Successfully enrolled.", "Enroll
Sucessful", MessageBoxButton.OK, MessageBoxImage.Asterisk);
        }
        catch (Exception ex)
        {
            MessageBox.Show(ex.Message);
        }
    }
    clear();
}

private void clear()
{
    txtregistrationid.Text = "";
    txtstudentaddress.Text = "";
    txtstudentname.Text = "";

```

```

        txtstudentnumber.Text = "";
        txtstudentid.Text = "";
    }

    private void btnClear_Click(object sender, RoutedEventArgs e)
    {
        clear();
    }

}
}

```

- Home

```

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.IO;

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

        private void EnrollStudent_Click(object sender, RoutedEventArgs
e)
        {
            EnrollStudent enrollStudent = new EnrollStudent();
            enrollStudent.Show();
            this.Close();
        }
    }
}

```

```

        private void TotalStudent_Click(object sender, RoutedEventArgs
e)
        {
            TotalStudents totalStudents = new TotalStudents();
            totalStudents.Show();
            this.Close();
        }

        private void btnImportStudentDetails_Click(object sender,
RoutedEventArgs e)
        {
            ImportStudentDetails importStudentDetails = new
ImportStudentDetails();
            importStudentDetails.Show();
            this.Close();
        }

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

        private void btnBarDaigram_Click(object sender,
RoutedEventArgs e)
        {
            BarDiagram barDiagram = new BarDiagram();
            barDiagram.Show();
            this.Close();
        }
    }
}

```

- Import Student Details

```

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.IO;
using Microsoft.Win32;
using System.Xml.Serialization;
using System.Data;

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

        private void Button_Click(object sender, RoutedEventArgs e)
        {
            try
            {
                var dataSet = new DataSet();
                dataSet.ReadXml(@"StudentDetails.xml");
                Microsoft.Win32.OpenFileDialog openFileDialog = new
                Microsoft.Win32.OpenFileDialog();

                if (openFileDialog.ShowDialog() == true)
                {
                    string filePath = openFileDialog.FileName;
                    using (var scan = new StreamReader(filePath))
                    {
                        scan.ReadLine();
                        while (!scan.EndOfStream)
                        {
                            var line = scan.ReadLine();
                            var values = line.Split(',');
                            var newRow = dataSet.Tables["Data"].NewRow();
                            newRow["StudentRegistrationID"] = values[0];
                            newRow["StudentRegistrationDate"] = values[1];
                            newRow["StudentID"] = values[2];
                            newRow["StudentName"] = values[3];
                            newRow["StudentAddress"] = values[4];
                            newRow["StudentContact"] = values[5];
                            newRow["StudentCourse"] = values[6];
                            dataSet.Tables["Data"].Rows.Add(newRow);

                            dataSet.WriteXml(@"StudentDetails.xml");
                        }
                    }
                }
            }
        }
    }
}
```



```

        MessageBox.Show("Student details successfully
imported.", "Import Sucessful.", MessageBoxButton.OK,
MessageBoxImage.Asterisk);
    }
}
catch (Exception ex)
{
    MessageBox.Show(ex.Message);
}
}

```

```

private void btnHome_Click(object sender, RoutedEventArgs e)
{
    Home home = new Home();
    home.Show();
    this.Close();
}
}
}

```

- Main Window

```

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

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

```

```

        private void Button_Click_1(object sender, RoutedEventArgs e)
        {
            if (txtUserName.Text.Trim() == "" &&
txtPassword.Password.Trim() == "")
            {
                MessageBox.Show("Invalid Username or Password");
            }
            else
            {
                if (txtUserName.Text.Trim() == "admin" &&
txtPassword.Password.Trim() == "admin")
                {
                    Home homeWindow = new Home();
                    homeWindow.Show();
                    this.Close();

                }
                else
                {
                    MessageBox.Show("Invalid Username or Password");
                }
            }
        }
    }
}

```

- Report

```

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;

namespace StudentInformationSystem
{

```

```
/// <summary>
/// Interaction logic for Report.xaml
/// </summary>
public partial class Report : Window
{

    public Report()
    {
        InitializeComponent();
    }

    private void btnHome_Click(object sender, RoutedEventArgs e)
    {
        Home home = new Home();
        home.Show();
        this.Close();
    }

    private void btnGenerateReport_Click(object sender,
RoutedEventArgs e)
    {
        try
        {
            DataSet ds = new DataSet();
            ds.ReadXml("StudentDetails.xml");
            DataView dv = new DataView();
            dv = ds.Tables[0].DefaultView;
            this.DataGridReport.ItemsSource = dv;

            MessageBox.Show("Student details report is generated.",
"Report Generated!!!", MessageBoxButton.OK,
MessageBoxImage.Asterisk);
        }
        catch (Exception ex)
        {
            MessageBox.Show(ex.Message);
        }
    }

    DataTable sortTable;
    private void btnGenerateByDate_Click_1(object sender,
RoutedEventArgs e)
    {
        try
        {
            if (System.IO.File.Exists(@"StudentDetails.xml"))
```

```

        {
            var dataSet = new DataSet();
            dataSet.ReadXml(@"StudentDetails.xml");

            //DataTable StudentDT = new DataTable("dt");
            sortTable = dataSet.Tables[0];
            sortTable.DefaultView.Sort = "StudentRegistrationDate
ASC";
            DataGridReport.Items.Refresh();
            DataGridReport.ItemsSource = sortTable.DefaultView;
        }
    }
    catch (Exception ex)
    {
        MessageBox.Show(ex.Message);
    }
}

private void btnGenerateByName_Click(object sender,
RoutedEventArgs e)
{
    try
    {
        if (System.IO.File.Exists(@"StudentDetails.xml"))
        {
            var dataSet = new DataSet();
            dataSet.ReadXml(@"StudentDetails.xml");

            //DataTable StudentDT = new DataTable("dt");
            sortTable = dataSet.Tables[0];
            sortTable.DefaultView.Sort = "StudentName ASC";
            DataGridReport.Items.Refresh();
            DataGridReport.ItemsSource = sortTable.DefaultView;
        }
    }
    catch (Exception ex)
    {
        MessageBox.Show(ex.Message);
    }
}
}
}
}

```

- Total Students

```
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.Xml.Serialization;

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

        private void btnHome_Click(object sender, RoutedEventArgs e)
        {
            Home home = new Home();
            home.Show();
            this.Close();
        }

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

            dataSet.ReadXml(@"C:\Users\Shristita
Kunwar\Documents\Visual Studio
2013\Projects\StudentInformationSystem\StudentInformationSystem\
\bin\Debug\StudentDetails.xml");

            DataTable dataTable = dataSet.Tables[0];
            int total_Com = 0;
            int total_Net = 0;
```

```
int total_Mul = 0;

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

for (int i = 0; i < dataTable.Rows.Count; i++)
{
    String col = dataTable.Rows[i][6].ToString();

    if (col == "Computing")
    {
        total_Com++;
    }
    else if (col.Equals("Network and It Security"))
    {
        total_Net++;
    }
    else if (col.Equals("Multimedia Technologies"))
    {
        total_Mul++;
    }

}

dt.Rows.Add("Computing", total_Com);
dt.Rows.Add("Network and It Security", total_Net);
dt.Rows.Add("Multimedia Technology", total_Mul);

Console.WriteLine(dt.Rows.Count);

dG.DataContext = dt.DefaultView;
    }
}
```