Informatics College Pokhara

informatics
college   pokhara

Application Development

CS6004NI

Course Work 1

Submitted By: **Pratik Thapa**   **Submitted To:** Ishwor Sapkota
London Met ID: Enter ID Here   Module Leader

Component Grade and Comments	
A. Implementation of Application	
User Interface and proper controls used for designing	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	Sorting not implemented
B. Documentation	
User Manual for running the application	User Manual is below average. Is textual only.

Application architecture & description of the classes ad methods sued	average work with very limited explanation of the classes and methods used
Flow chart, algoriathms and data sctructures used	average work with very limited explanation and missing diagramatic representation.
Reflective essay	Average work with un clear learnings, experience or findings.

C. Programming Style

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

Overall Grade:	D	D

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

LONDON
METROPOLITAN
UNIVERSITY

informatics
college • pokhara

Application development

CS6004NA

Coursework 1

Submitted By: **Submitted To:**

Student Name: Pratik Thapa   **Ishwor Sapkota**

London Met ID: 17030525   **Application Development**

Group: L3C2

Date:1/10/2020

Table of Contents

List of Figure

Introduction

The designed system is Student Information System. The system is highly designed developed and test under various circumstances. It is a desktop based application for a company and it is not a web based or database application which is done is C#. In this application users must input the student personal detail and registration date so that a system can generate a weekly enrolment report of the student. In addition to that user can add their information details like Name, address, contact number, registration date and course enrol.

This application is designed to keep tracking of the details of the student, course enrol and registration date.

Current Scenario

There are numerous Student Information System in schools, collages who keeps their record of data in a Paper Based System. There are only few numbers of schools, collages with Student Information System but lacked to keep track of the students details, course enrol and registration date.

Proposed System

The proposed system is designed to allow the user to input the student personal detail including registration date so that a system can generate a weekly enrolment report of the student and this system is also keep tracking of the students details, course enrol and registration date.

User Manual

There are screenshot below which will show a user how to operate the system.

1. As the user operates the system the initial screen will be the security screen where user have to login and the username and password of the system is "admin". Only a valid username and password can provide access to the system. As there is Exit button where you can exit the system.
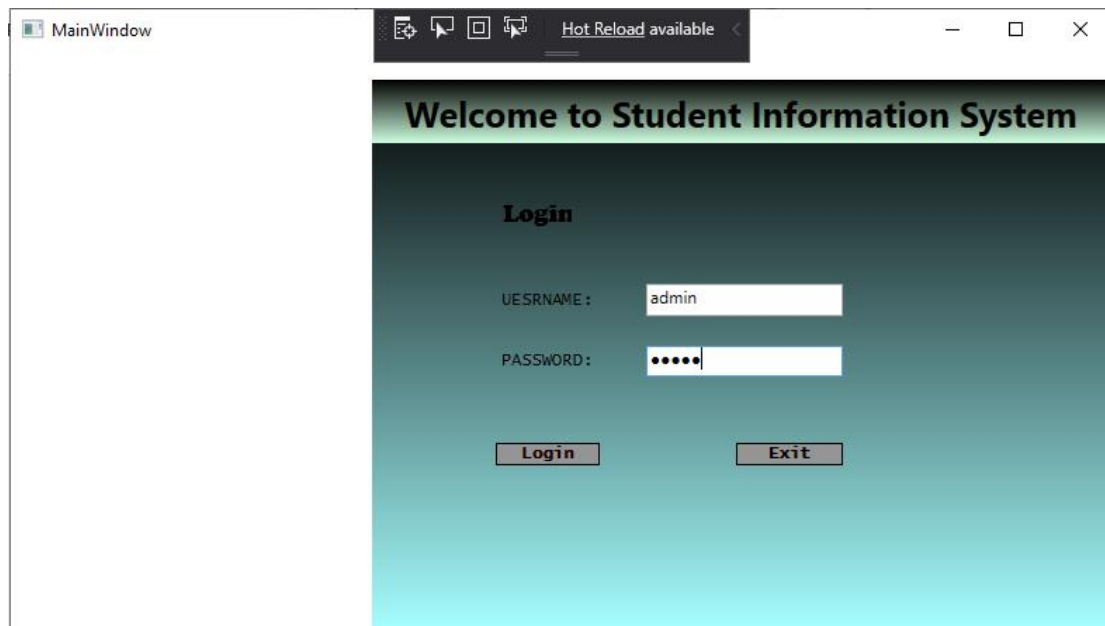

Figure 1 login

2. As the user select button Login this popup message will be displayed. After selecting this you will entered to the student details page.
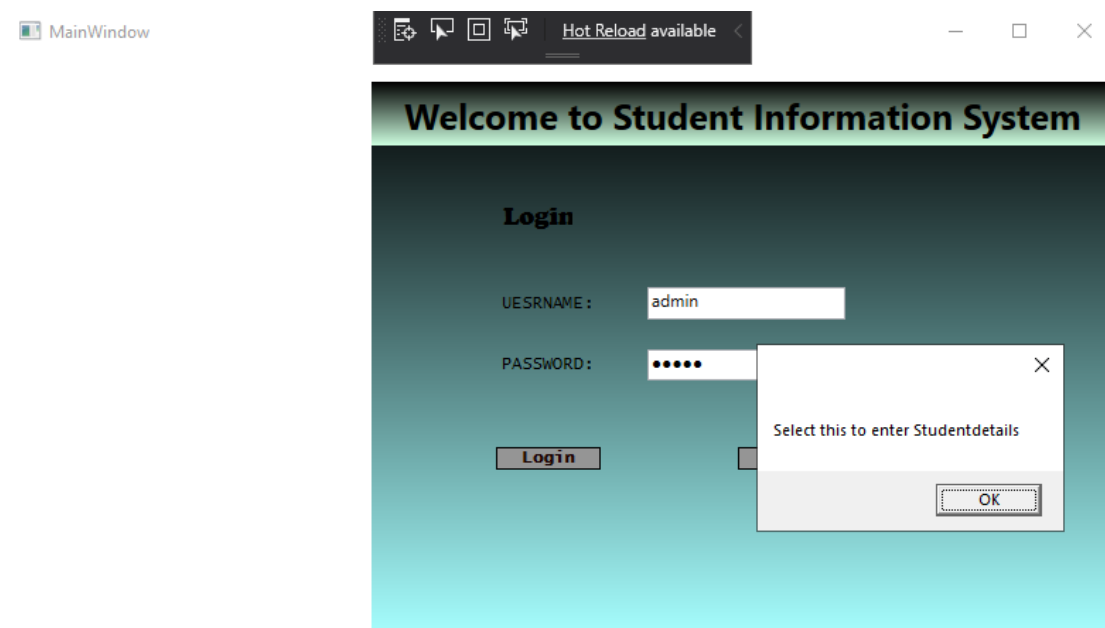

Figure 2 popup to enter studentdetails

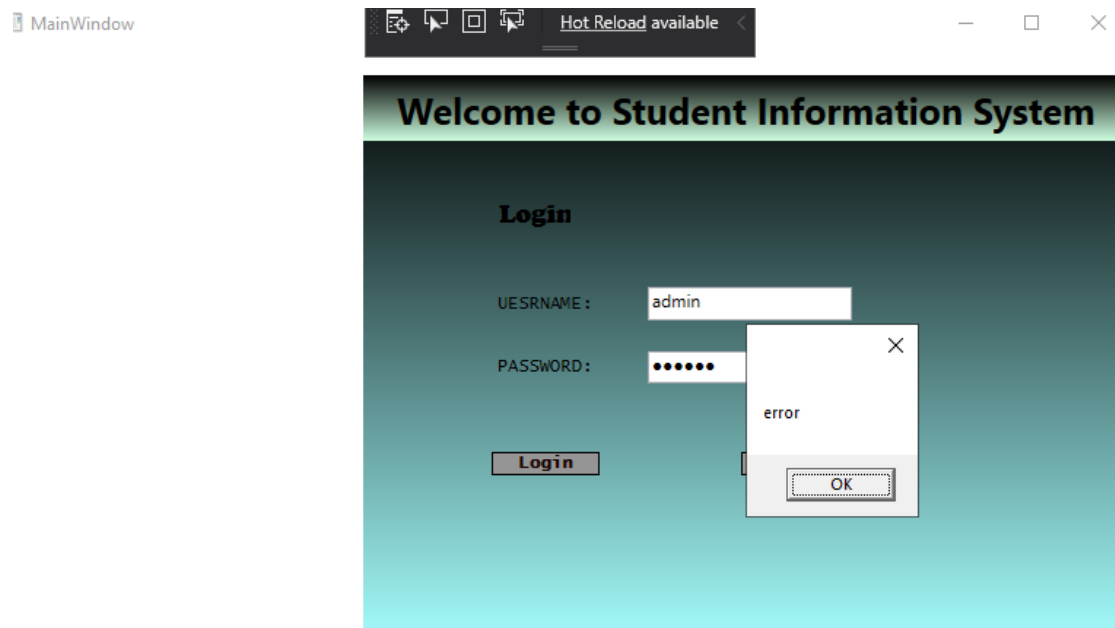3. If user enter the wrong username or password than this error popup message will be displayed.

Figure 3 error while login

4. After user login the system user have to enter the details.

Figure 4 student details

5. Sort by name

ID	RegNo	Name	Address	ContactNo	CourseEnroll	RegistrationDate
4	5	pratikmadi	lekhnath	97777	BBA	2020-01-11T00:00:00+05:45
3	4	Satkar	Bagmara	9000	BBS	2020-01-10T00:00:00+05:45
2	3	thapaPratik	lekh	9888	BBA	2020-01-11T00:00:00+05:45

Figure 5 sort by name

6. Sort by date

ID	RegNo	Name	Address	ContactNo	CourseEnroll	RegistrationDate
3	4	Satkar	Bagmara	9000	BBS	2020-01-10T00:00:00+05:45
2	3	thapaPratik	lekh	9888	BBA	2020-01-11T00:00:00+05:45
4	5	pratikmadi	lekhnath	97777	BBA	2020-01-11T00:00:00+05:45

Figure 6 sort by date

7. This is the weekly report of the overall users of course enrolled. And there is back button which goes back to the student details.

Courses Enrolled	Overall Student		
BBA	2		
BIT	1		
BBS	2		

Back

Figure 7 weekly report

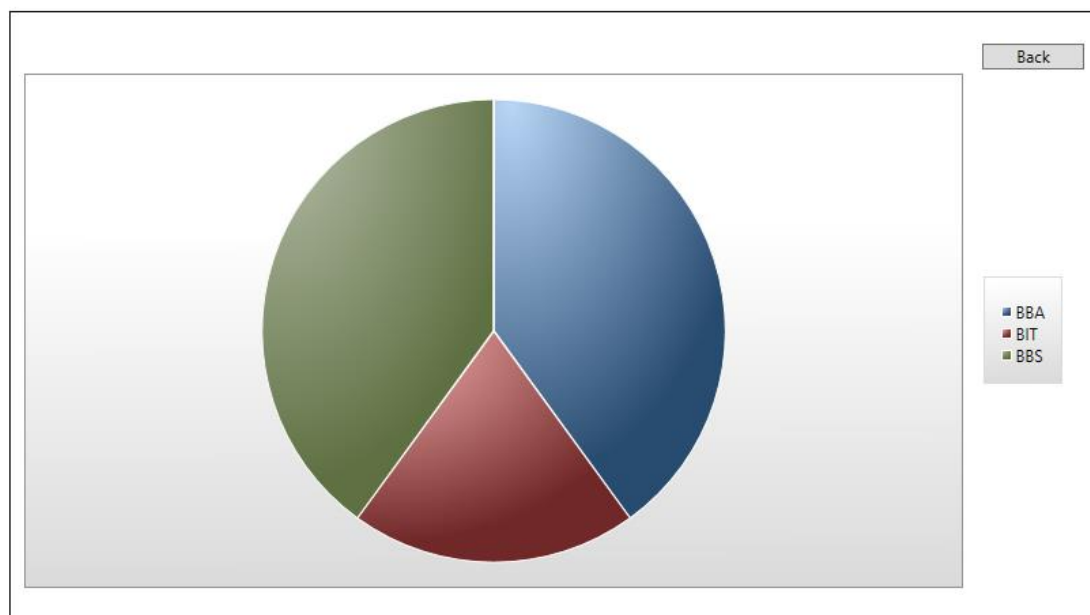8. in this figure chart showing that total number of student on each course.

Figure 8 chart

Journal Articles

1. Learning programming is thought to be troublesome. One doable reason why students don't do well in programming is expounded to the very fact that traditional way of learning within the lecture hall adds more stress on students in understanding the Material rather than applying the Material to a true application. For a few students, this teaching model might not catch their interest. As a result, they'll not offer their best effort to grasp the Material given. Seeing however the information is applied to real issues will increase student interest in learning. As a consequence, this may increase their effort to be taught. In the current paper, we try to help students learn C# programming language using Intelligent Tutoring System. This ITS was developed using ITSB authoring tool to be able to help the student learn programming efficiently and make the learning procedure very pleasing. A knowledge base using ITSB authoring tool style was used to represent the student's work and to give customized feedback and support to students. (Abu-Naser, 2017)

2. The paper presents a review of the concept of Information Management Systems, and its applicability in the Educational System context. After presenting several existing solutions on the market, the current solution is presented. The system is supporting the academic activity of both students and staff, aiming to eliminate routine and non-efficient activities. The technical structure is server based and the support is Visual Studio 2008 IDE + Mono Tool and C#. The user interface and the functionalities are based on a survey on a targeted student group. (Giurgiu & Stoicu-Tivadar, 2010)

3. The new C# language and Internet software services have received much of the attention surrounding Microsoft's new .NET environment. However, Microsoft has also redesigned the way Windows desktop applications will be created and deployed in the future. Intended as a tutorial for C++ and Java programmers at all levels, this book shows how C# and the .NET framework can be used to develop Windows applications with .NET. As a way to

demonstrate how Windows Forms applications are constructed, the book provides a chapter-by-chapter guide to building an image application to view, share, and manage digital photos stored on a hard drive or on removable media. Also discussed are Visual Studio 7.0, C# language syntax and usage, Windows Forms controls, ADO.NET, and other topics vital to the creation and deployment of Windows applications in this new environment. (Brown, March 2002)

4. Intelligent Tutoring Systems (ITS) has a wide influence on the exchange rate, education, health, training, and educational programs. In this paper we describe an intelligent tutoring system that helps student study computer networks. The current ITS provides intelligent presentation of educational content appropriate for students, such as the degree of knowledge, the desired level of detail, assessment, student level, and familiarity with the subject. Our Intelligent tutoring system was developed using ITSB authoring tool for building ITS. A preliminary evaluation of the ITS was done by a group of students and teachers. The results were acceptable. (Abu-Naser, Learning Computer Networks Using Intelligent Tutoring System, 2017)

System Architecture
Architecture Diagram
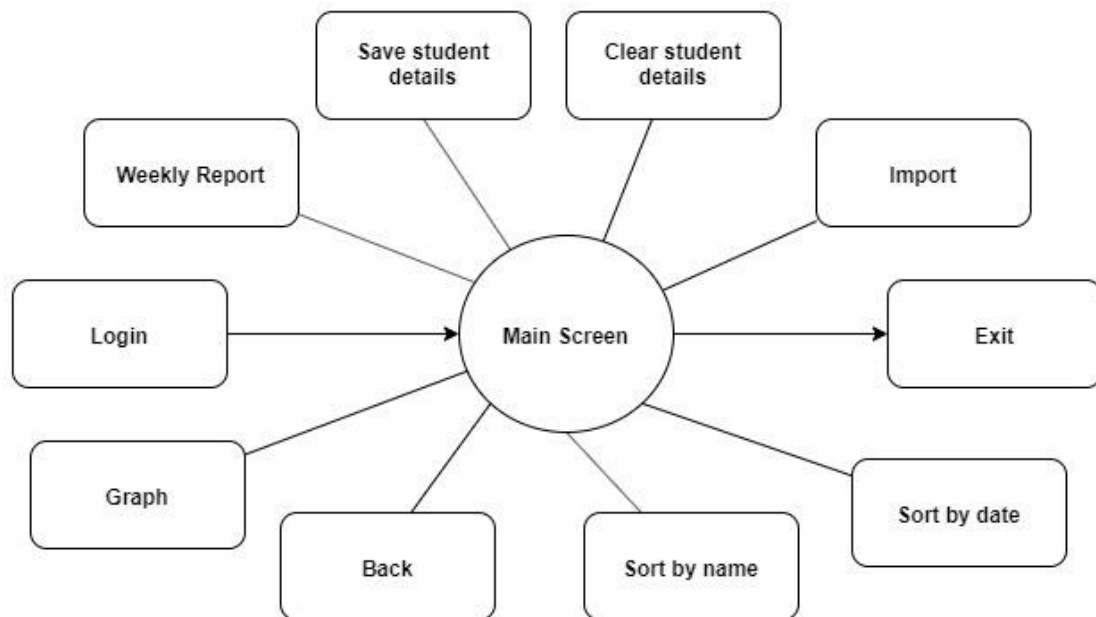
Figure 9 Architecture diagram

Flowchart of Weekly Report
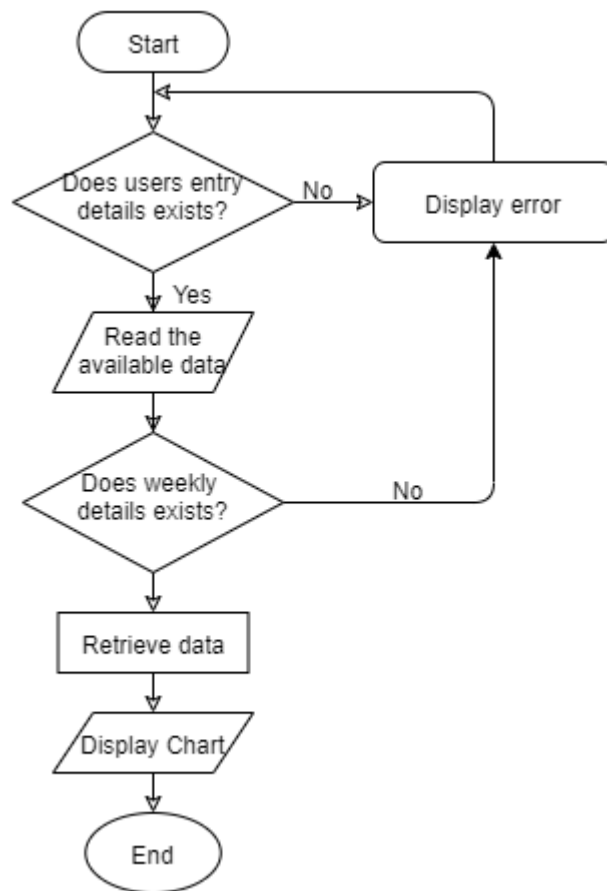
Figure 10 flowchart of weekly report

Algorithm of Weekly Report
Steps:

1. Start

2. Check whether the user entry the details exists or not.

3. If details doesn't exists then display the error message and user have to restart.

4. If details exists then read the available data.

5. Check whether weekly details exists or not.

6. If weekly details doesn't exists then display the error message and user have to restart.

7. If details exists then retrieve the data.

8. Display the data in chart

9. Stop

Reflection

The developed system is Student Information System. It is developed using Visual Studio with the help of C# language. The GUI designed is highly user friendly which is easy to use and understanding and user with basic system administration can operate the system.

The system allow the user to input the student personal details including registration date so that a system can generate a weekly enrolment report of the student. Also users can input details like name, address, contact number, course enrol and contact. And the system keep tracking the details. In addition to that, a user can check out the weekly chart along with the list. The system has also login page which we have to enter the username and password. And the default username and password is "admin".

I have very few experiences with the visual studio which makes difficulties to complete this project. Additionally feature like chart, import files, sorting and validation are completely new for me. To complete this project I liked to thanks my module leader and friend who helps me a lot.

Conclusion

As the given coursework for the module CS6004NA Application Development was to build up for the user to input the student details. It required a long time to build up the task by using Visual studio with the help of C# language. The system has login page which we have to enter the username and password. This coursework help me to gain knowledge of C# language, .ASP and more. To complete this project I liked to thanks my module leader and friend who helps me a lot.

Bibliography

Abu-Naser, S. S. (2017). Design and Development of an Intelligent Tutoring System for C# Language. *EUROPEAN ACADEMIC RESEARCH.*

Abu-Naser, S. S. (2017). Learning Computer Networks Using Intelligent Tutoring System.

Brown, E. (March 2002). Windows Forms Programming with C#.

Giurgiu, L., & Stoicu-Tivadar, L. (2010). An Information Management System supporting academic activity.

Appendix

Homepage.cs

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

namespace CW1Appdevelopment
{
    /// <summary>
    /// Interaction logic for Homepage.xaml
    /// </summary>
    public partial class Homepage : Page
    {
        public Homepage()
        {
            InitializeComponent();
```

```csharp
        }

        private void Login(object sender, RoutedEventArgs e)
        {
            {
                //.Content = new StudentDetail();
                //this.NavigationService.Navigate(new Studentdetails());
            }
            string user, pass;
            user = txtuser.Text;
            pass = txtPass.Password;


            if (user == "admin" && pass == "admin")
            {
                MessageBox.Show("Select this to enter Studentdetails");
                this.NavigationService.Navigate(new Studentdetails());

            }
            else
            {
                MessageBox.Show("error");
            }
        }


        private void Exit(object sender, RoutedEventArgs e)
        {
            System.Windows.Application.Current.Shutdown();
        }


    }
}
```

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

```csharp
namespace CW1Appdevelopment
{
    /// <summary>
    /// Interaction logic for MainWindow.xaml
    /// </summary>
    public partial class MainWindow : Window
    {
        public MainWindow()
        {
            InitializeComponent();
            main.Content = new Homepage();
        }
    }
}
```

Studentdetails.cs

```csharp
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 Microsoft.Win32;
using System.Windows.Controls.DataVisualization.Charting;

namespace ADCourseWork
{
    /// <summary>
    /// Interaction logic for Chart.xaml
    /// </summary>
    public partial class Chart : Page
    {
        public Chart()
        {
            InitializeComponent();
            LoadPieChartData();
        }
        public void LoadPieChartData()
        {
            var dataSet = new DataSet();
            if (System.IO.File.Exists(@"file\StudentReport.xml"))
```

```csharp
        {
            dataSet.ReadXml(@"file\StudentReport.xml");

            DataTable dtStdReport = dataSet.Tables[0];

            int Total_BBA = 0;
            int Total_BEIT = 0;
            int Total_BBS = 0;

            DataTable Week = new DataTable("WeekTable1");
            Week.Columns.Add("Courses Enrolled", typeof(String));
            Week.Columns.Add("Overall Student", typeof(int));


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

                String column = dtStdReport.Rows[i]["CourseEnroll"].ToString();

                if (column == "BBA")
                {
                    Total_BBA++;

                }
                else if (column == "BEIT")
                {
                    Total_BEIT++;

                }
                else if (column == "BBS")
                {
                    Total_BBS++;

                }
            }
            Week.Rows.Add("BBA", Total_BBA);
            Week.Rows.Add("BEIT", Total_BEIT);
            Week.Rows.Add("BBS", Total_BBS);

            ((PieSeries)chartEnroll).ItemsSource =
            new KeyValuePair<string, int>[]{
            new KeyValuePair<string,int>("BBA", Total_BBA),
            new KeyValuePair<string,int>("BEIT", Total_BEIT),
            new KeyValuePair<string,int>("BBS", Total_BBS)};

        }
        else
        {
            MessageBox.Show("No data to show!");
        }

    }

    private void BackToRegistration(object sender, RoutedEventArgs e)
    {
        this.NavigationService.Navigate(new StudentDetail()) ;
    }
  }
}
```

Weeklyreport.cs

```csharp
using DataHandler;
```

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

namespace ADCourseWork
{
    /// <summary>
    /// Interaction logic for WeeklyStudentReport.xaml
    /// </summary>
    public partial class WeeklyStudentReport : Page
    {
        public WeeklyStudentReport()
        {
            InitializeComponent();
            weeklyStudentList();
        }
        private void weeklyStudentList()
        {
```

```
var dataSet = new DataSet();


dataSet.ReadXml(@"file\StudentReport.xml");


DataTable dtStdReport = dataSet.Tables[0];


int Total_BBA = 0;
int Total_BEIT = 0;
int Total_BBS = 0;


DataTable Week = new DataTable("WeekTable1");
Week.Columns.Add("Courses Enrolled", typeof(String));
Week.Columns.Add("Overall Student", typeof(int));



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

    String column = dtStdReport.Rows[i]["CourseEnroll"].ToString();

    if (column == "BBA")
    {
        Total_BBA++;


    }
    else if (column == "BEIT")
    {
        Total_BEIT++;


    }
```

```
        else if (column == "BBS")

        {

            Total_BBS++;


        }

    }

    Week.Rows.Add("BBA", Total_BBA);

    Week.Rows.Add("BEIT", Total_BEIT);

    Week.Rows.Add("BBS", Total_BBS);



    StudentReport.DataContext = Week.DefaultView;




}


private void Button_Back(object sender, RoutedEventArgs e)

{

    this.NavigationService.Navigate(new StudentDetail());

}
    }
}
```

Chart.cs

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

```csharp
using System.Windows.Media.Imaging;
using System.Windows.Shapes;

namespace CW1Appdevelopment
{
    /// <summary>
    /// Interaction logic for Chart.xaml
    /// </summary>
    public partial class Chart : Page

    {
        public Chart()
        {
            InitializeComponent();
            LoadPieChartData();
        }
        public void LoadPieChartData()
        {
            var dataSet = new DataSet();
            if (System.IO.File.Exists(@"files/StudentReport.xml"))
            {
                dataSet.ReadXml(@"files/StudentReport.xml");


                DataTable dtStdReport = dataSet.Tables[0];

                int Total_BBA = 0;
                int Total_BIT = 0;
                int Total_BBS = 0;

                DataTable Week = new DataTable("WeekTable1");
                Week.Columns.Add("Courses Enrolled", typeof(String));
                Week.Columns.Add("Overall Student", typeof(int));


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

                    String column =
dtStdReport.Rows[i]["CourseEnroll"].ToString();

                    if (column == "BBA")
                    {
                        Total_BBA++;

                    }
                    else if (column == "BIT")
                    {
                        Total_BIT++;

                    }
                    else if (column == "BBS")
                    {
                        Total_BBS++;

                    }
                }
                Week.Rows.Add("BBA", Total_BBA);
                Week.Rows.Add("BIT", Total_BIT);
                Week.Rows.Add("BBS", Total_BBS);

                ((PieSeries)chartEnroll).ItemsSource =
```

```
            new KeyValuePair<string, int>[]{
            new KeyValuePair<string,int>("BBA", Total_BBA),
            new KeyValuePair<string,int>("BIT", Total_BIT),
            new KeyValuePair<string,int>("BBS", Total_BBS)};


            }
            else
            {
                MessageBox.Show("No data!");
            }

        }

        private void back(object sender, RoutedEventArgs e)
        {

            this.NavigationService.Navigate(new Studentdetails());
        }
    }
}
```