Informatics College Pokhara

informatics
college pokhara

Application Development
CS6004NI
Course Work 1

Submitted By: **Kriti Gurung**
London Met ID: Enter ID Here

Submitted To: Ishwor Sapkota
Module Leader

Component Grade and Comments	
A. Implementation of Application	
User Interface and proper controls used for designing	User Interface is complete but not separated and have proper use of controls
Manual data entry or import from csv	not properly saved or imported data
Data Validation	Only basic validation
Enrollment Report & weekly report in tabular format	very poorly executed reports and data not shown accurately
Course wise enrollment report & Chart display	Very poorly designed and only contains one report format with in appropriate data
Algorithm used for sorting & proper sorting of data	Default sorting provided by .net is used
B. Documentation	
User Manual for running the application	User Manual is average. Includes description for all interfaces

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	unusable system

Overall Grade:	C	C

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

Informatics College Pokhara

Application Development

CS6004NP

Coursework 1

Submitted by		**Submitted to**
Name:	Kriti Gurung	Name: IshworSapkota
ID:	17030720	Application Development
Group:	L3C1	
Date:	2020/01/10	

Table of Contents

List of Figures

1. Introduction

Student Information System is a desktop application which is designed and implemented in C#. This system is responsible for tracking the student details, module selected registration, registration date, weekly reports and so on. The applicationallows the user to input the student personal detail including registration date sothat a system can generate a weekly enrolment report of the student. System includes detail like name, address, contact no, email, program enroll, registration date. Moreover, this system brings ease to keep the record of the students.

Some of the key features of this system are listed below:

- A login window.
- Register or enroll new students.
- Create a new xml file consisting of student registration details.
- Display registered students in grid view.
- Import a CSV file and update the xml file of student registration details.
- Generate reports on the basis of requirements.
- Report could be sorted by student first name or registration date.
- Display the total number of students enrolled in a program.
- Display a pie chart of total number of student on each program.

2. User Manual

For using the system properly instructions are given below with pictorial descriptions.

2.1 Execution of program

In order to run the program click on ‘StudentInformationSystem.exe’ file.This file is located inside ‘StudentInformationSystem/bin/debug’ folder.

2.2 Login

When the system loads a login form is displayed on the screen. The form asks the user to enter username and password. The username and password for this system is admin. When the username or password is entered incorrectly, the system displays a dialogue box for the error.

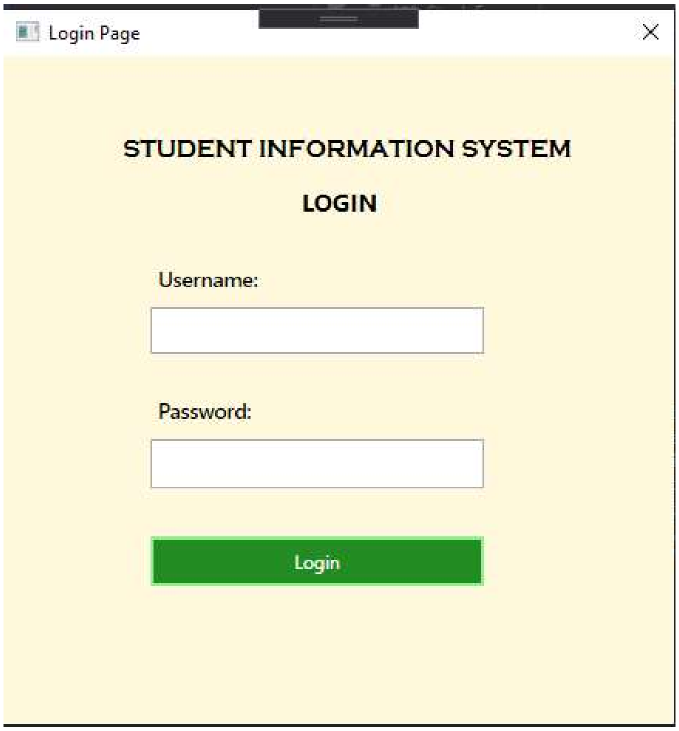
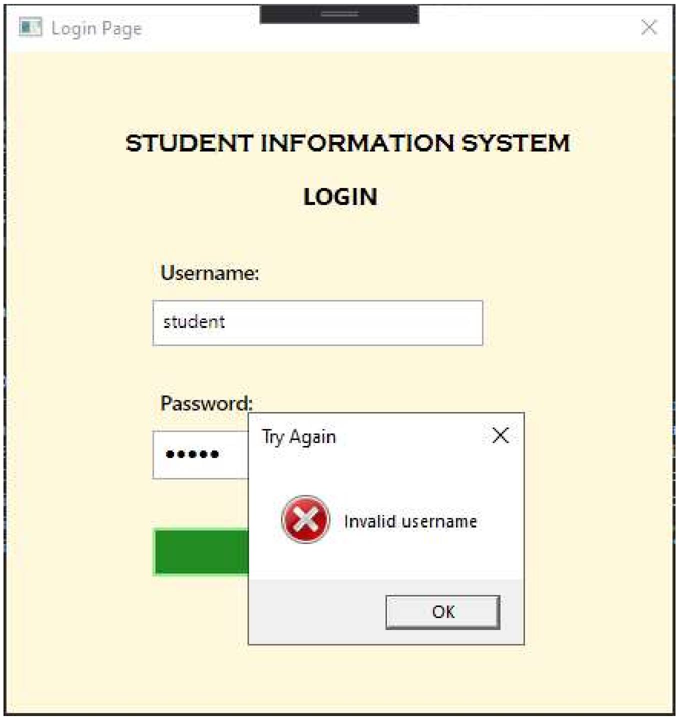


Figure 1: Login Page of system



Login Page

STUDENT INFORMATION SYSTEM

LOGIN

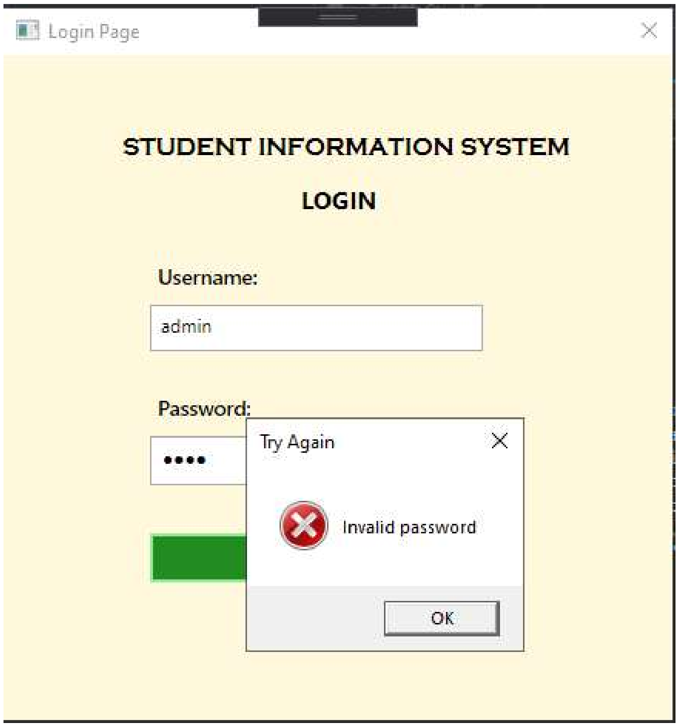
Username: student

Password: •••••

Try Again

Invalid username

Figure 2: Invalid username error



Login Page

STUDENT INFORMATION SYSTEM

LOGIN

Username: admin

Password: ••••

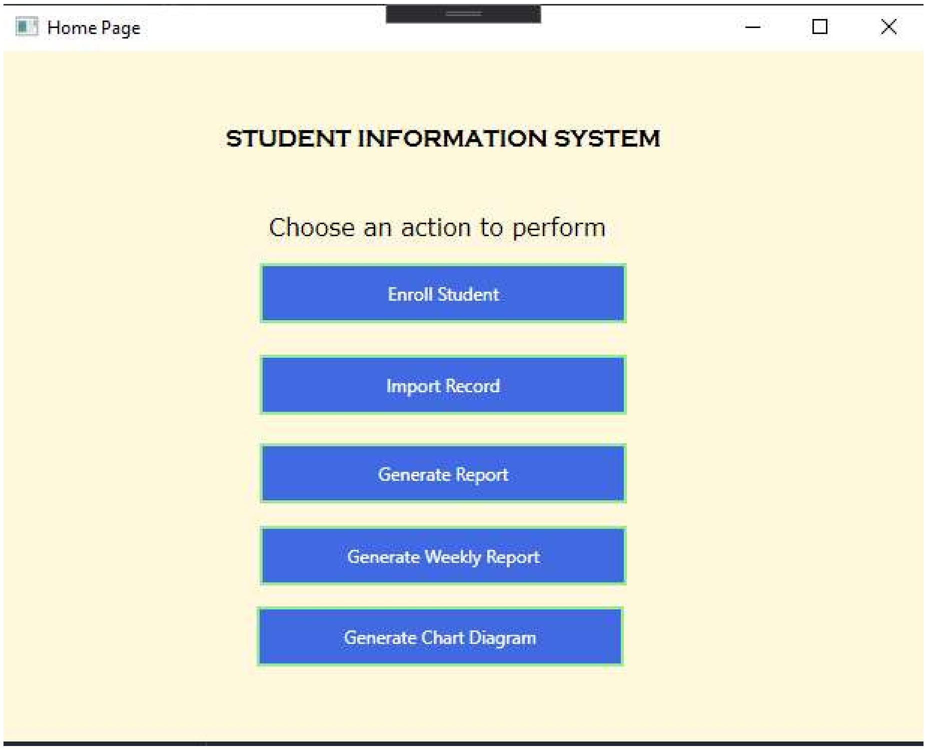
Try Again

Invalid password

Figure 3: Invalid password error

2.3 Home Window

This is the main form of the system. It consists of a group of button to navigate through next forms. There are five buttons to navigate through pages which are: enroll student, import record, generate record, generate report, generate weekly report and generate pie chart diagram. Each have their own functions.



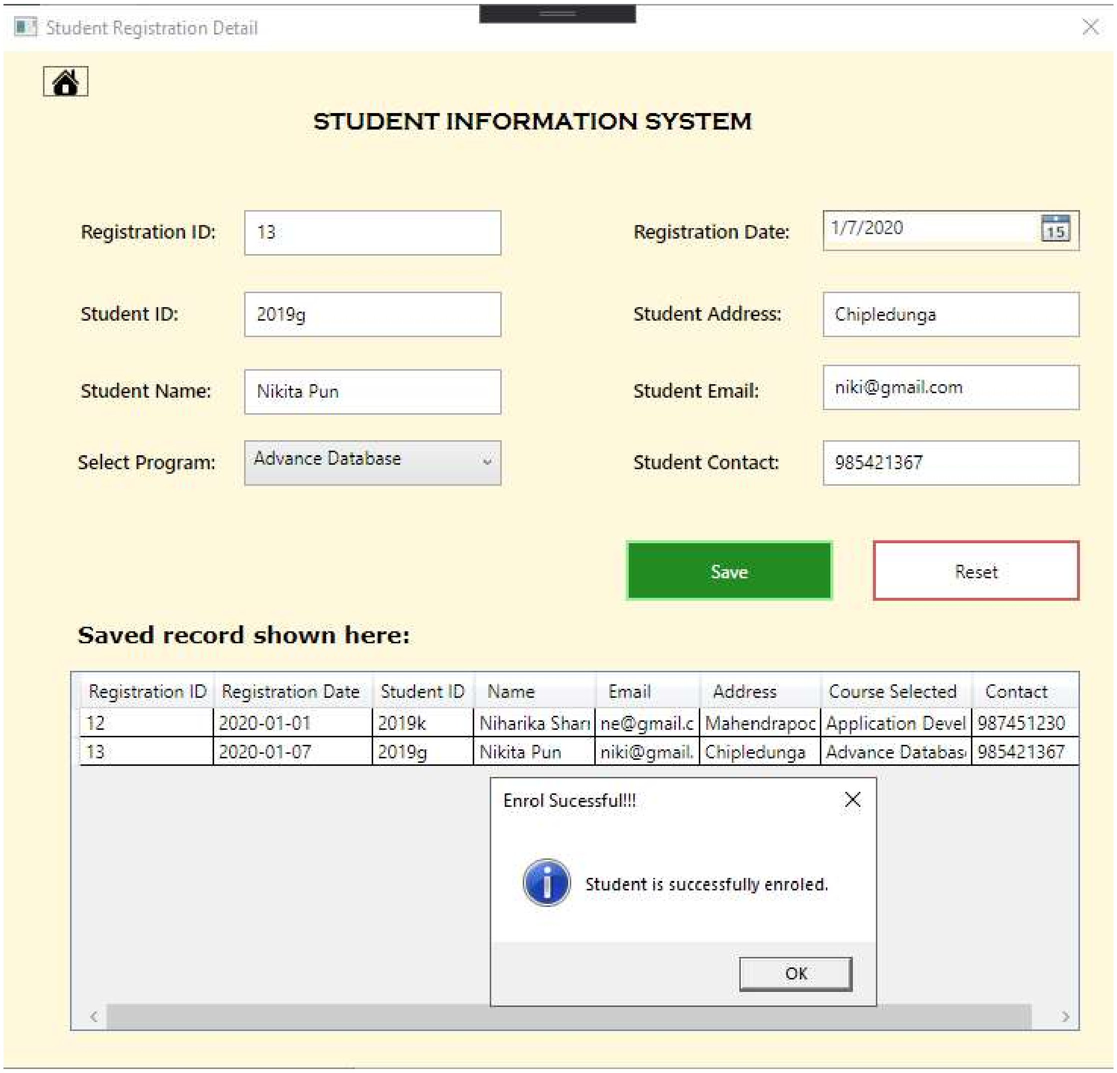
STUDENT INFORMATION SYSTEM

Choose an action to perform

Figure 4: Home page

2.4 Enroll Student

This form is responsible for registration of new students. It contains text input text boxes for registration id, registration date, student id, student name, student address, student contact number and module selected. All the fields are must except course. There is also a reset button which is to clear all the values of the fields. After the entering of genuine data, the data is displayed in the grid below the form.


Student Registration Detail

STUDENT INFORMATION SYSTEM

Registration ID:	13	Registration Date:	1/7/2020
Student ID:	2019g	Student Address:	Chipledunga
Student Name:	Nikita Pun	Student Email:	niki@gmail.com
Select Program:	Advance Database	Student Contact:	985421367

Saved record shown here:

Registration ID	Registration Date	Student ID	Name	Email	Address	Course Selected	Contact
12	2020-01-01	2019k	Niharika Shar	ne@gmail.c	Mahendrapoc	Application Devel	987451230
13	2020-01-07	2019g	Nikita Pun	niki@gmail.	Chipledunga	Advance Databas	985421367

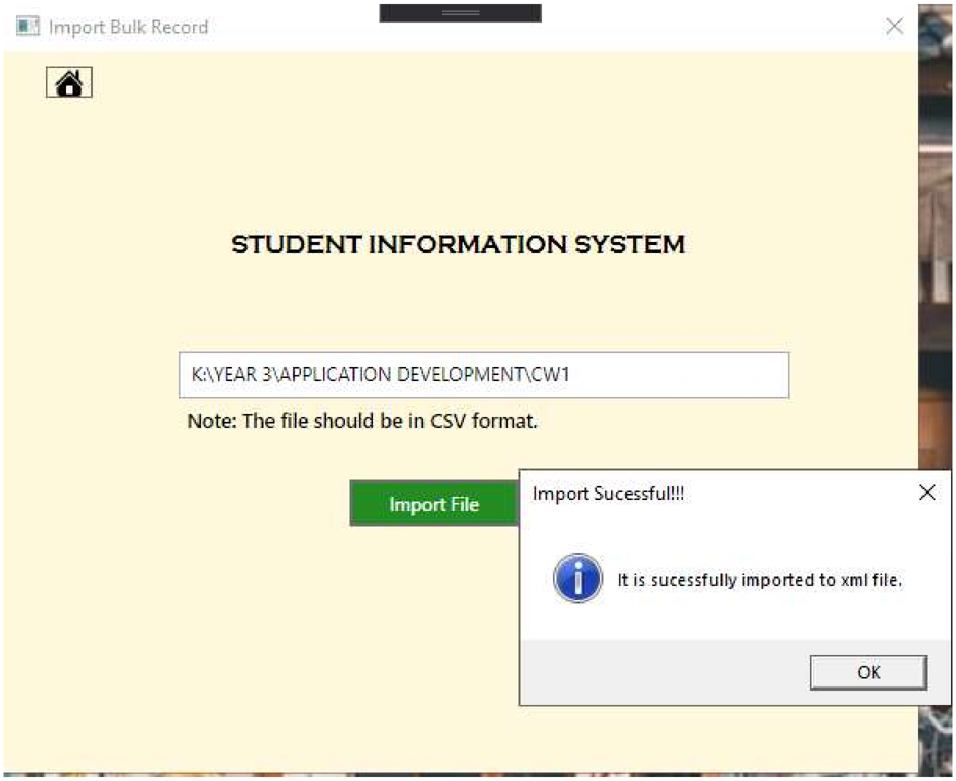
Enrol Sucessful!!!

Student is successfully enroled.

Figure 5: Student registration page

2.5 Import Record

Inside this page, a text file consisting of a bulk data which is in the CVS format can be imported. A green button named as import file helps to import the file. The data is updated to the existing XML file of student details. If the file is not in CVS format then it is not able to import the details. So the file should be in CSV format.



Import Bulk Record

STUDENT INFORMATION SYSTEM

K:\YEAR 3\APPLICATION DEVELOPMENT\CW1

Note: The file should be in CSV format.

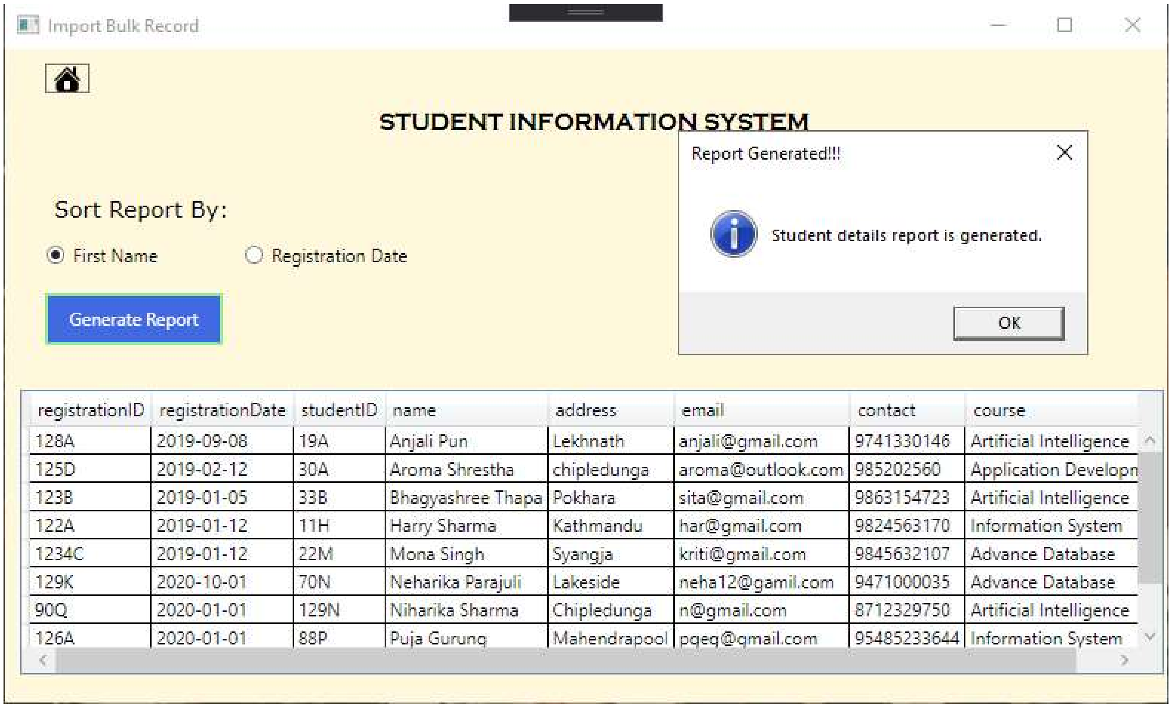
Import Sucessful!!!

It is sucessfully imported to xml file.

Figure 6: Import CSV file

2.6 Generate Report

After the registration of a student, a XML file is created. With the help of XML the report is created otherwise it is unable to show the report. The report is shown in grid view. The display can be done by two methods. One is sorted by the first name and another by registration date. For sorting two radio buttons are provided and a button for generating report.



Report Generated!!!

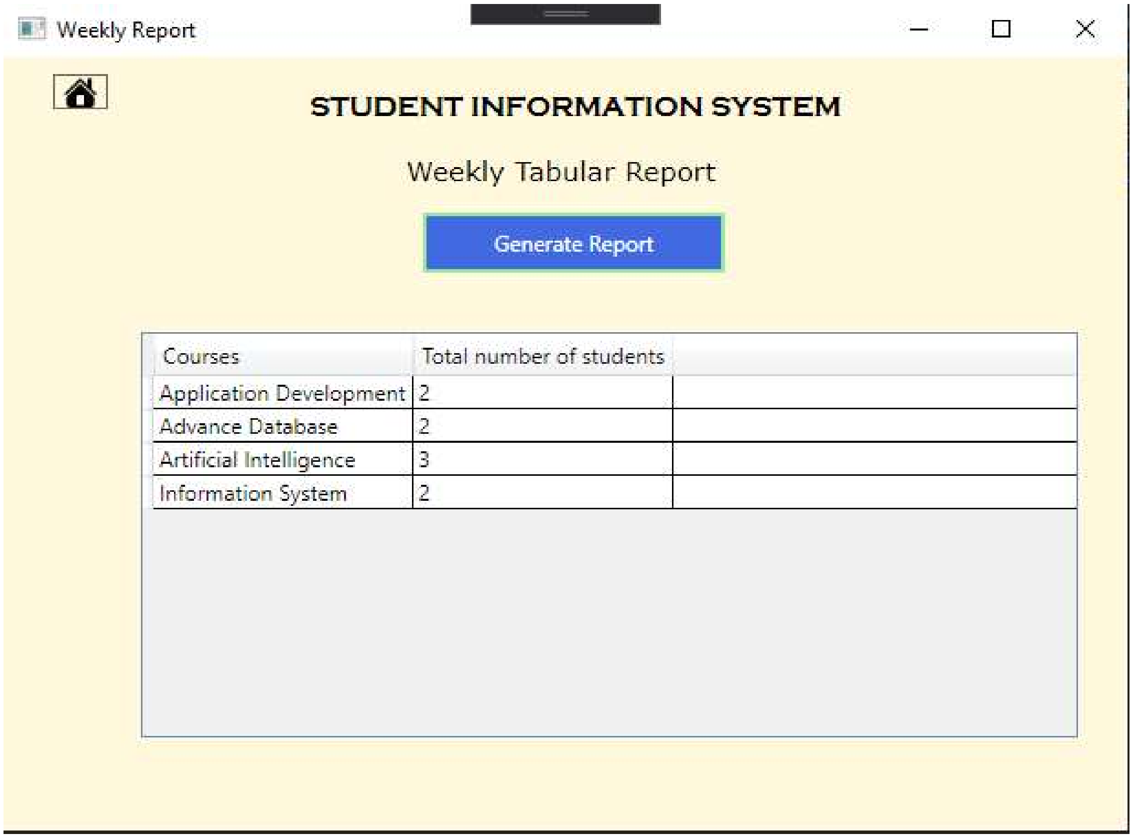
Student details report is generated.

registrationID	registrationDate	studentID	name	address	email	contact	course
128A	2019-09-08	19A	Anjali Pun	Lekhnath	anjali@gmail.com	9741330146	Artificial Intelligence
125D	2019-02-12	30A	Aroma Shrestha	chipledunga	aroma@outlook.com	985202560	Application Developn
123B	2019-01-05	33B	Bhagyashree Thapa	Pokhara	sita@gmail.com	9863154723	Artificial Intelligence
122A	2019-01-12	11H	Harry Sharma	Kathmandu	har@gmail.com	9824563170	Information System
1234C	2019-01-12	22M	Mona Singh	Syangja	kriti@gmail.com	9845632107	Advance Database
129K	2020-10-01	70N	Neharika Parajuli	Lakeside	neha12@gamil.com	9471000035	Advance Database
90Q	2020-01-01	129N	Niharika Sharma	Chipledunga	n@gmail.com	8712329750	Artificial Intelligence
126A	2020-01-01	88P	Puja Gurung	Mahendrapool	pqeq@gmail.com	95485233644	Information System

Figure 7: Report generation

2.7 Generate Weekly Report

It also displays a report of the students with the help of XML data. It shows the weekly tabular report showing total number of students enrolled so farin each program offered by the institution. The data is retrieved and shown in the grid view. If there is no XML file then it shows an error message for not being able to generate the record.



Courses	Total number of students
Application Development	2
Advance Database	2
Artificial Intelligence	3
Information System	2

Figure 8: Weekly report

2.8 Generate Chart Diagram

With the help of the data from XML, apie chart is displayed showing total number of student on each program. When the cursor is hovered on the chart one can see the total number of students and percentage of student enrolled in the course.

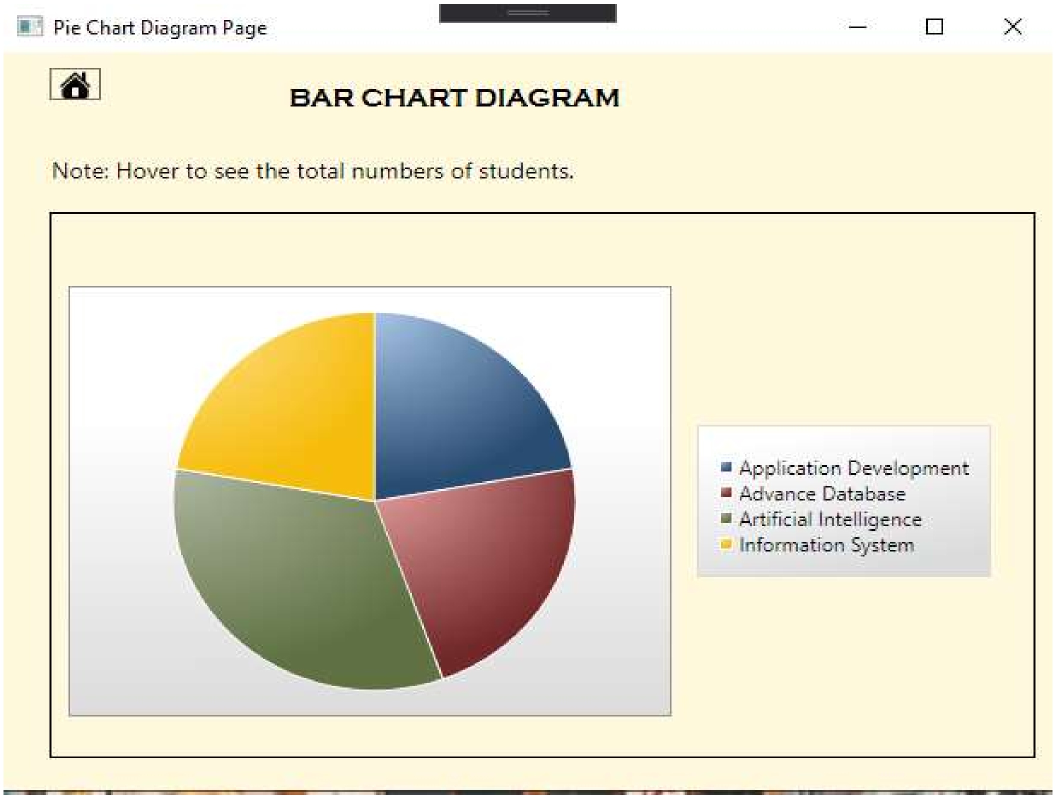


Figure 9: Display Pie chart

3. Flow Chart

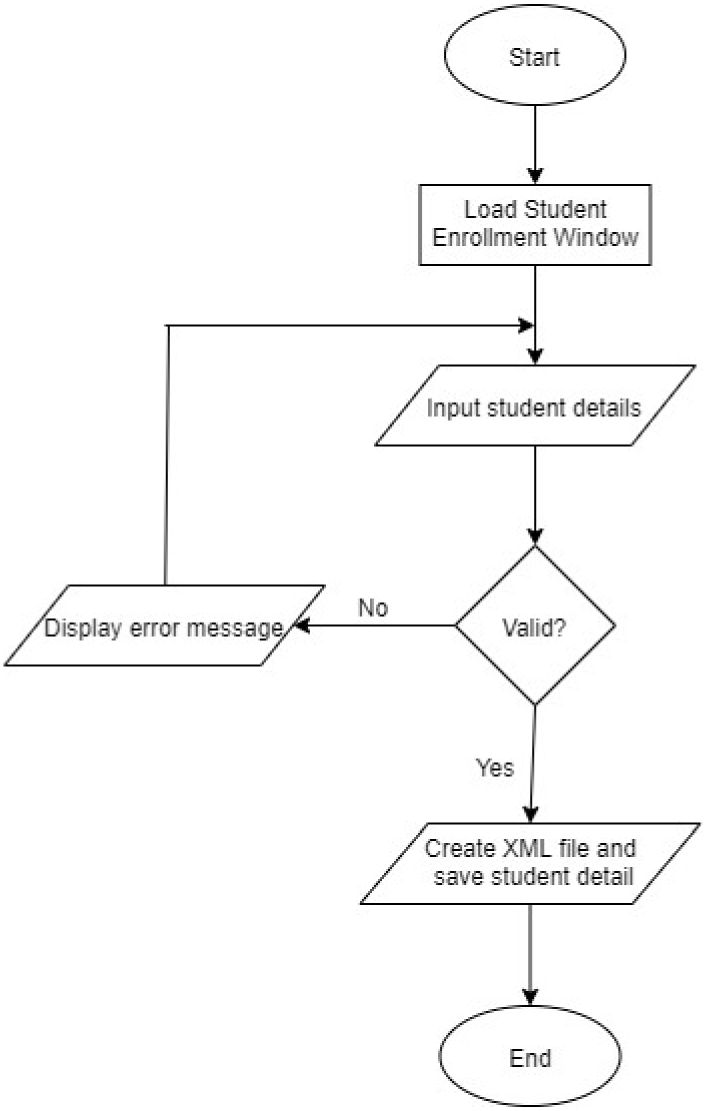

Start
Load Student Enrollment Window
Input student details
Valid?
No
Display error message
Yes
Create XML file and save student detail
End

Figure 10: Flowchart of enrollment of students

In the above figure, a flow chart of student enrollment is given. Firstly, the system displays the student enrollment form. The user will input the details. If the details are left blank or not valid then the system displays a dialog box of the error message. If the details are valid, the system creates a XML file and saves the details to the file.

4. Method Description

- **Login Page**

Method	Description
login_button	If the username and password matches then it navigates to home page. When the username and password mismatches then an error message is displayed.

- **Home Page**

Method	Description
navigate_to_registration	This method navigates to student registration page.
import_bulk_record	This method navigates to CSV file import page.
generate_report	This method navigates to report generation page.
generate_weekly_report	This method navigates to weekly report generation page.
generate_bar_diagram	This method navigates to pie chart generation page.

- **StudentRegistrationDetailsPage**

Method	Description
saveStdDetails	Firstly, it gets the user input details and checks whether the data is null or not. After the data is validated a XML file is created and the data is displayed to data grid.
AppendData	It appends the data to the current existing XML file. The data is deserialized first and serialized. After that it is added to the existing XML file.
Button_Click	It navigates to home page.
btn_reset_Click	Clears the text fields.

- **Import Bulk Record**

Method	Description
btn_save_csv_file_Click	It opens the dialogue box and the csv file is read by the stream reader. Then is breaks it and adds to defined tags in the existing XML file.
Button_Click	It navigates to home page.

- **GenerateReport**

Method	Description
generateStudentReport	It reads the XML file with the help of data set and displays in grid. The data is also sorted by first name and registration date. There are two radio buttons for selecting the sorting process.
Button_Click	It navigates to home page.

- **Weekly Report**

Method	Description
generate_weekly_report	It reads the XML file with the help of data set. It counts the total number of students on each course. And it is displayed to data grid.
Button_Click	It navigates to home page.

- **Chart Diagram**

Method	Description
ChartDiagram	It reads the XML file with the help of data set. It counts the total number of students on each course. And it is generates a pie chart. The pie chart is created with the help of installing a package of data visualization.
Button_Click	It navigates to home page.

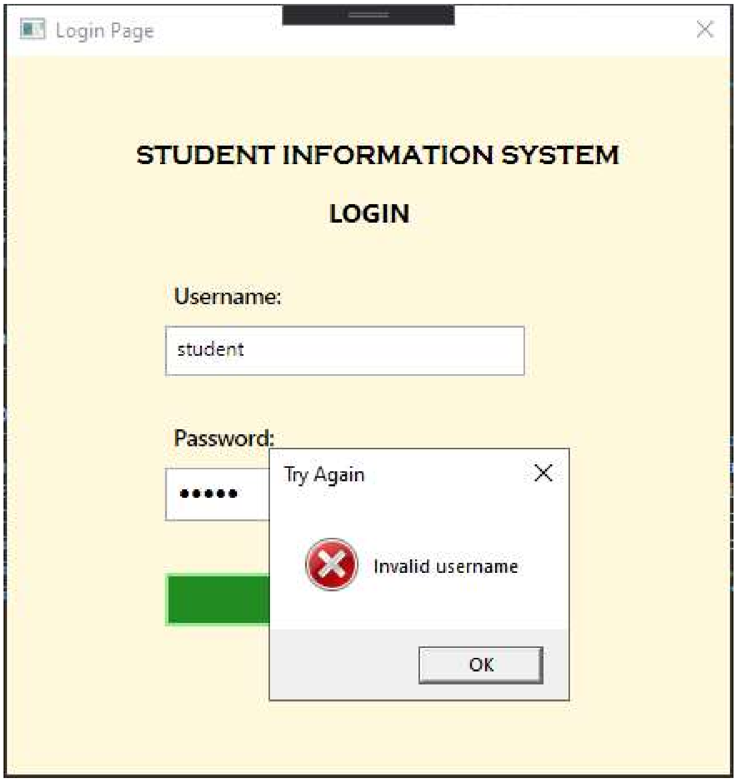
5. Testing

5.1 Testing 1

Objective: To display error message when the username or password is not valid.

Output: Displays error message for the invalid username or password.

Result: Successful.



STUDENT INFORMATION SYSTEM

LOGIN

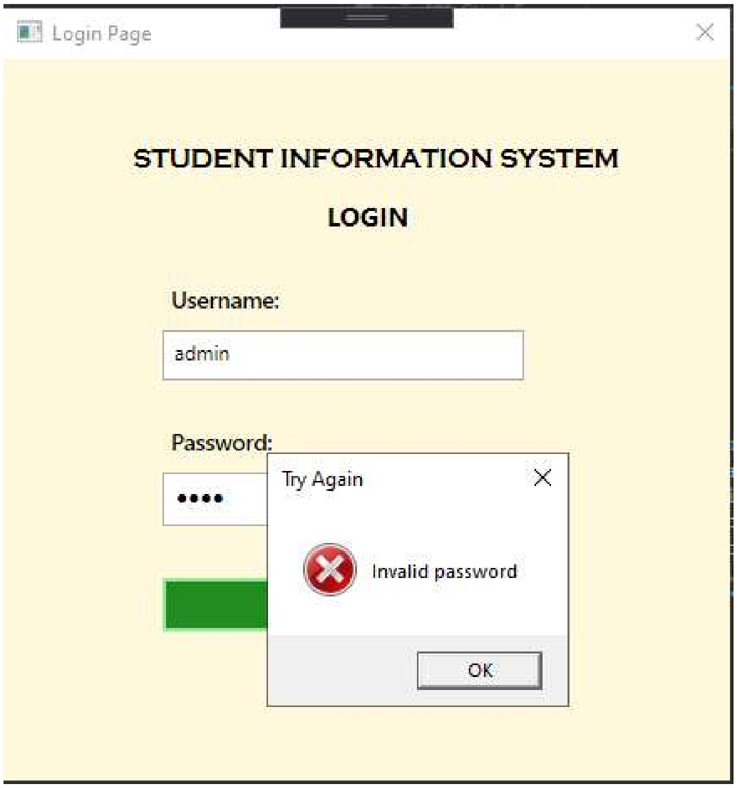
Username: student

Password: •••••

Try Again

Invalid username

Figure 11: Testing 1



Login Page

STUDENT INFORMATION SYSTEM

LOGIN

Username:

admin

Password:

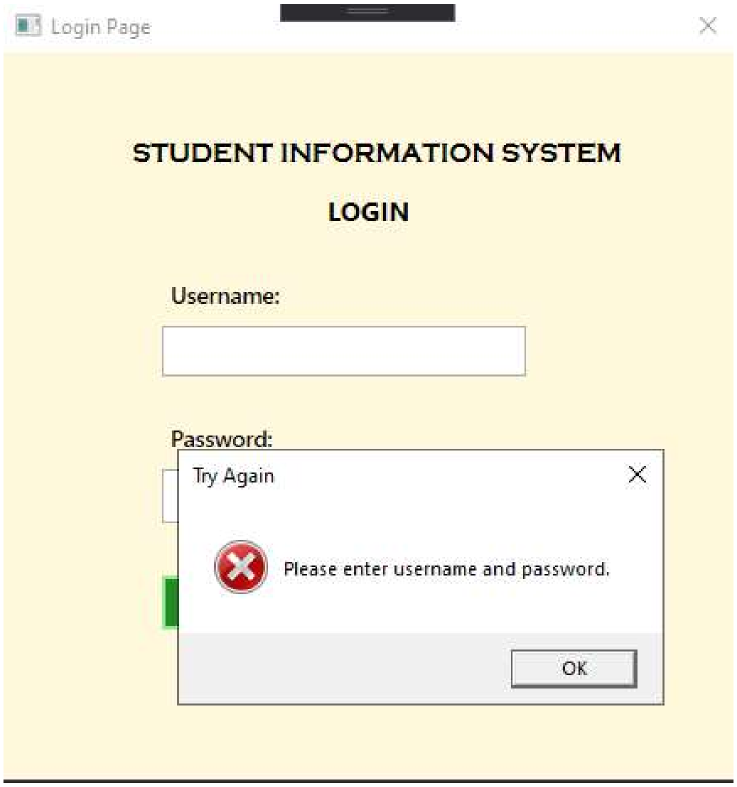
••••

Try Again

Invalid password

OK

Figure 12: Testing 1.1



Please enter username and password.

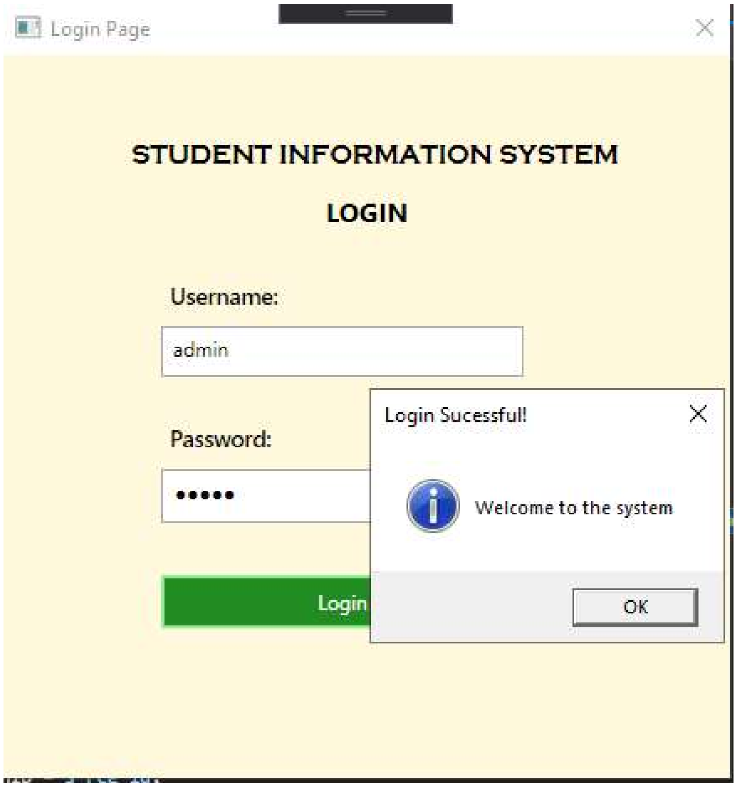
Figure 13: Testing 1.2

5.2 Testing 2

Objective: To display success message and navigate to home page when the username and password matches.

Output: Displays successful message and navigates to home page.

Result: Successful.



Login Page

STUDENT INFORMATION SYSTEM

LOGIN

Username: admin

Password: •••••

Login Sucessful!

Welcome to the system

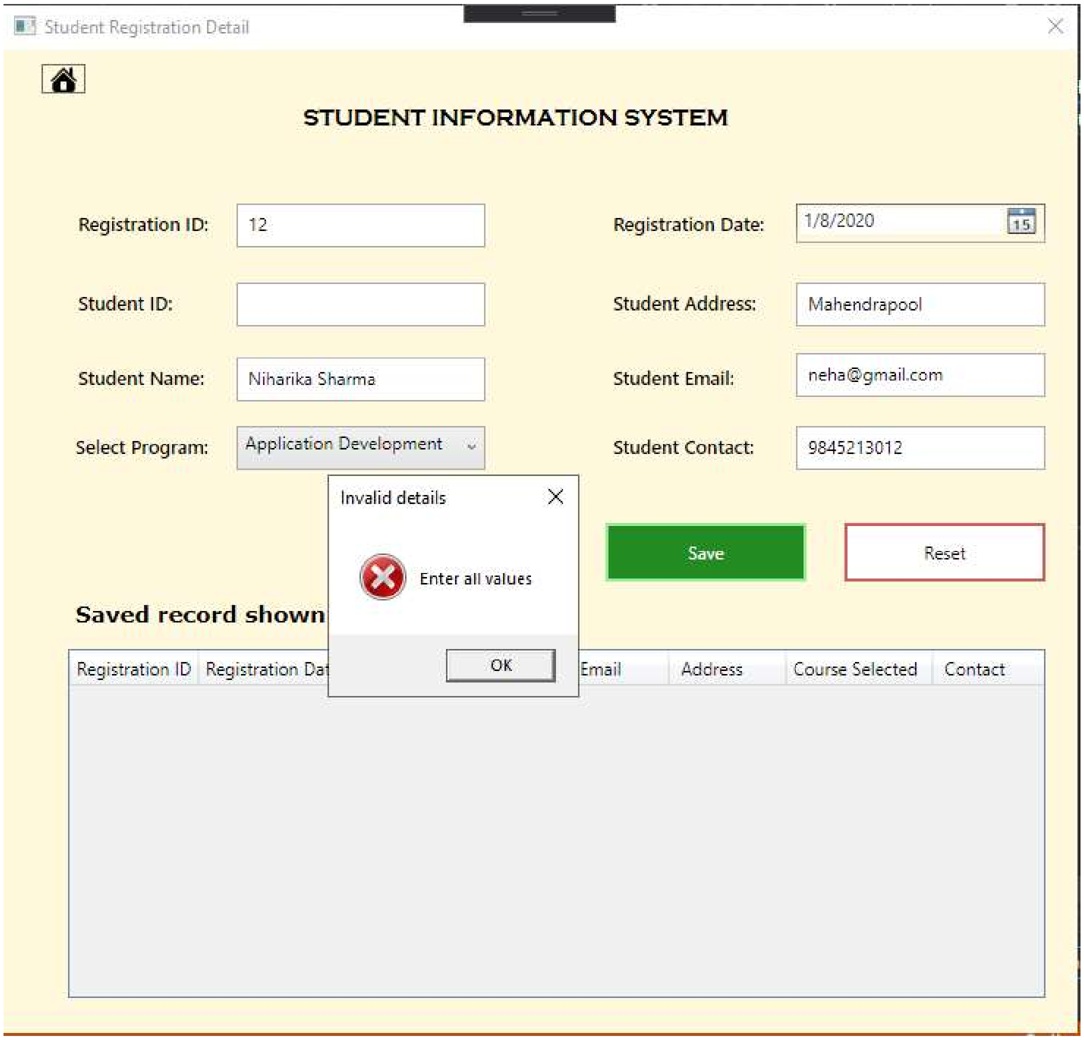
Figure 14: Testing 2

5.3 Testing 3

Objective: To display error message when the user leaves text fields empty in student enroll page.

Output: Displays error message for the invalid input.

Result: Successful.



Student Registration Detail

STUDENT INFORMATION SYSTEM

Registration ID:	12	Registration Date:	1/8/2020
Student ID:		Student Address:	Mahendrapool
Student Name:	Niharika Sharma	Student Email:	neha@gmail.com
Select Program:	Application Development	Student Contact:	9845213012

Invalid details

Enter all values

Saved record shown

Registration ID	Registration Dat	Email	Address	Course Selected	Contact

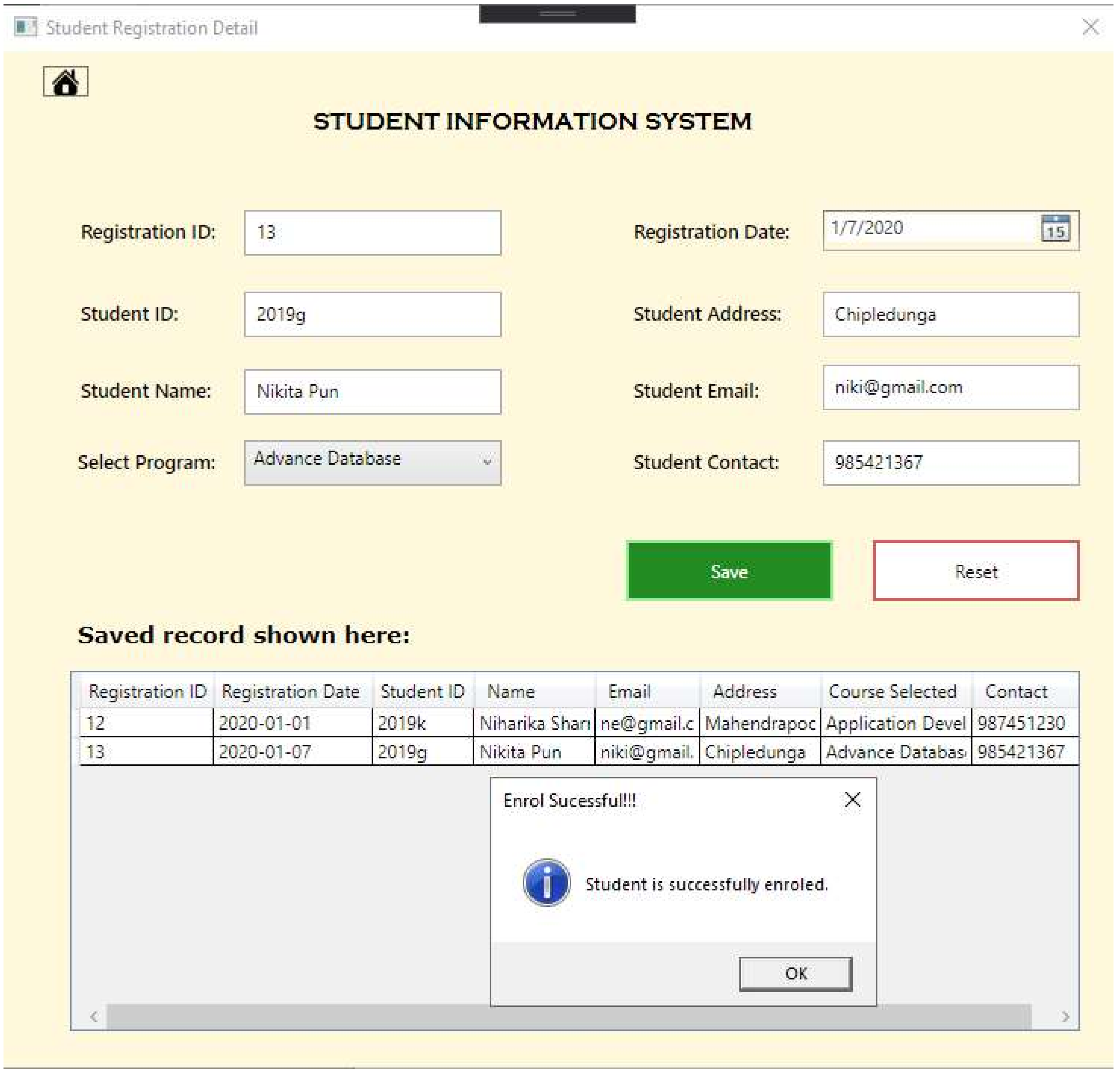
Figure 15: Testing 3

5.4 Testing 4

Objective: To check whether XML file consisting data of student’s details or not.

Output: XML file has been created with the details provided by user.

Result: Successful.



Registration ID:	13	Registration Date:	1/7/2020
Student ID:	2019g	Student Address:	Chipledunga
Student Name:	Nikita Pun	Student Email:	niki@gmail.com
Select Program:	Advance Database	Student Contact:	985421367

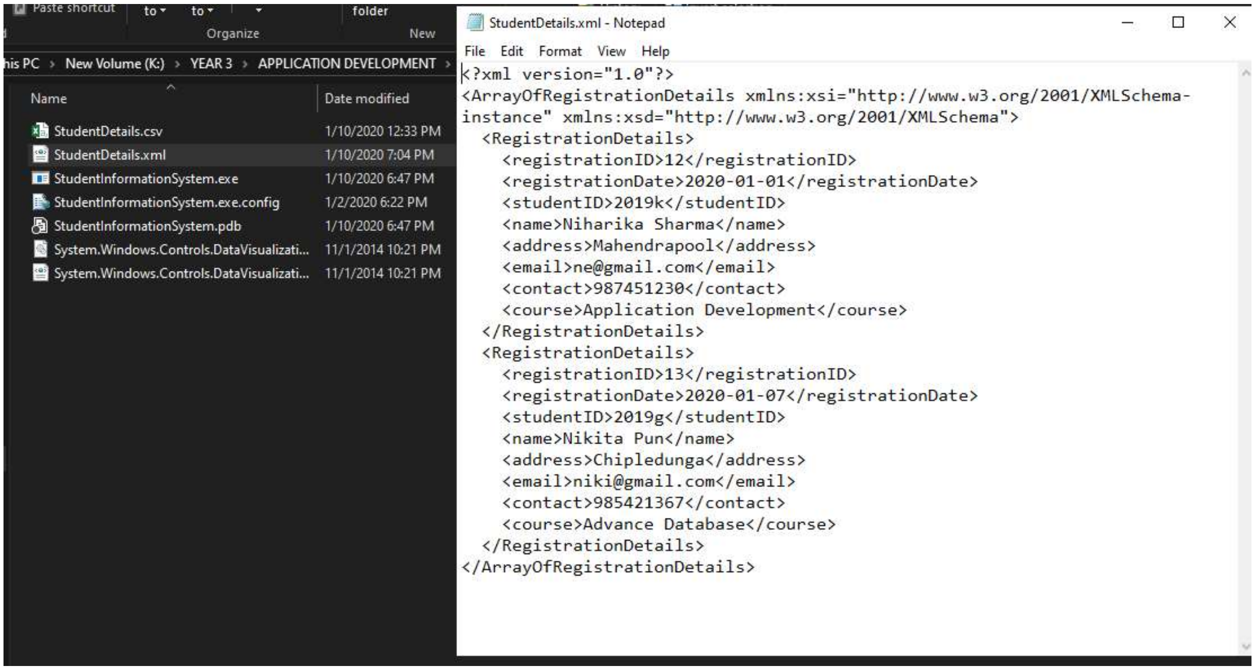
Saved record shown here:

Registration ID	Registration Date	Student ID	Name	Email	Address	Course Selected	Contact
12	2020-01-01	2019k	Niharika Shari	ne@gmail.c	Mahendrapoc	Application Devel	987451230
13	2020-01-07	2019g	Nikita Pun	niki@gmail.	Chipledunga	Advance Databas	985421367

Enrol Sucessful!!!

Student is successfully enroled.

Figure 16: Testing 4


Paste shortcut
to ▾
to ▾
folder
Organize
New
his PC › New Volume (K:) › YEAR 3 › APPLICATION DEVELOPMENT ›
StudentDetails.xml - Notepad
File Edit Format View Help

Name	Date modified
StudentDetails.csv	1/10/2020 12:33 PM
StudentDetails.xml	1/10/2020 7:04 PM
StudentInformationSystem.exe	1/10/2020 6:47 PM
StudentInformationSystem.exe.config	1/2/2020 6:22 PM
StudentInformationSystem.pdb	1/10/2020 6:47 PM
System.Windows.Controls.DataVisualizati...	11/1/2014 10:21 PM
System.Windows.Controls.DataVisualizati...	11/1/2014 10:21 PM

```
<?xml version="1.0"?>
<ArrayOfRegistrationDetails xmlns:xsi="http://www.w3.org/2001/XMLSchema-
instance" xmlns:xsd="http://www.w3.org/2001/XMLSchema">
  <RegistrationDetails>
    <registrationID>12</registrationID>
    <registrationDate>2020-01-01</registrationDate>
    <studentID>2019k</studentID>
    <name>Niharika Sharma</name>
    <address>Mahendrapool</address>
    <email>ne@gmail.com</email>
    <contact>987451230</contact>
    <course>Application Development</course>
  </RegistrationDetails>
  <RegistrationDetails>
    <registrationID>13</registrationID>
    <registrationDate>2020-01-07</registrationDate>
    <studentID>2019g</studentID>
    <name>Nikita Pun</name>
    <address>Chipledunga</address>
    <email>niki@gmail.com</email>
    <contact>985421367</contact>
    <course>Advance Database</course>
  </RegistrationDetails>
</ArrayOfRegistrationDetails>
```

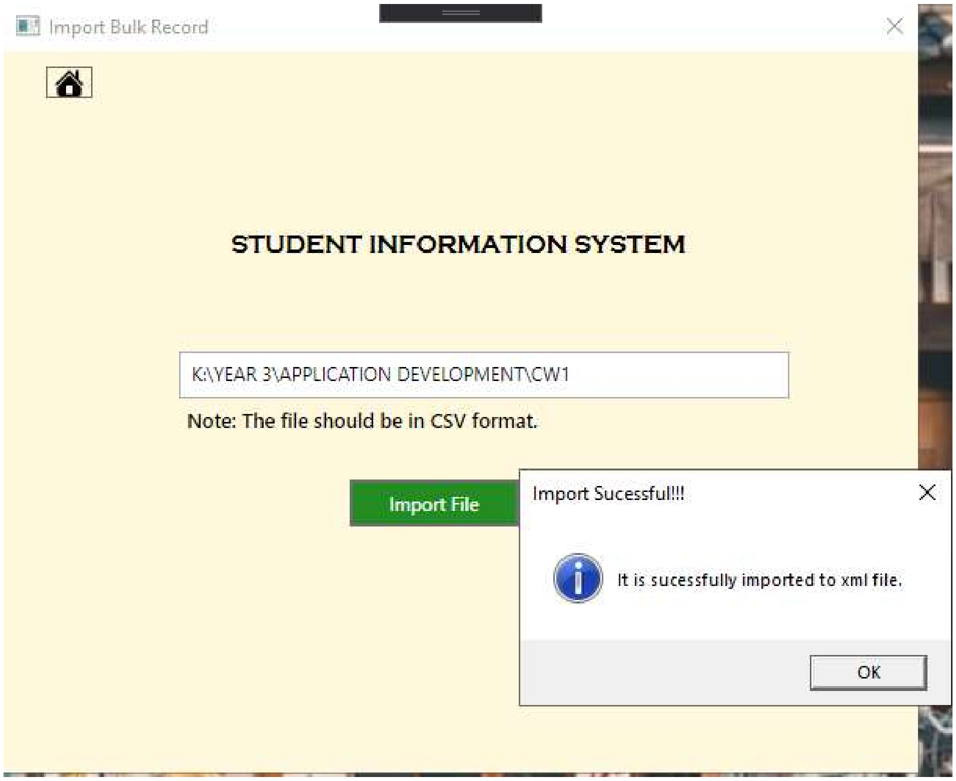
Figure 17: Testing 4.1

5.5 Testing 5

Objective: Check the bulk data of csv file is added to existing XML.

Output: The bulk data is added to XML successfully.

Result: Successful.



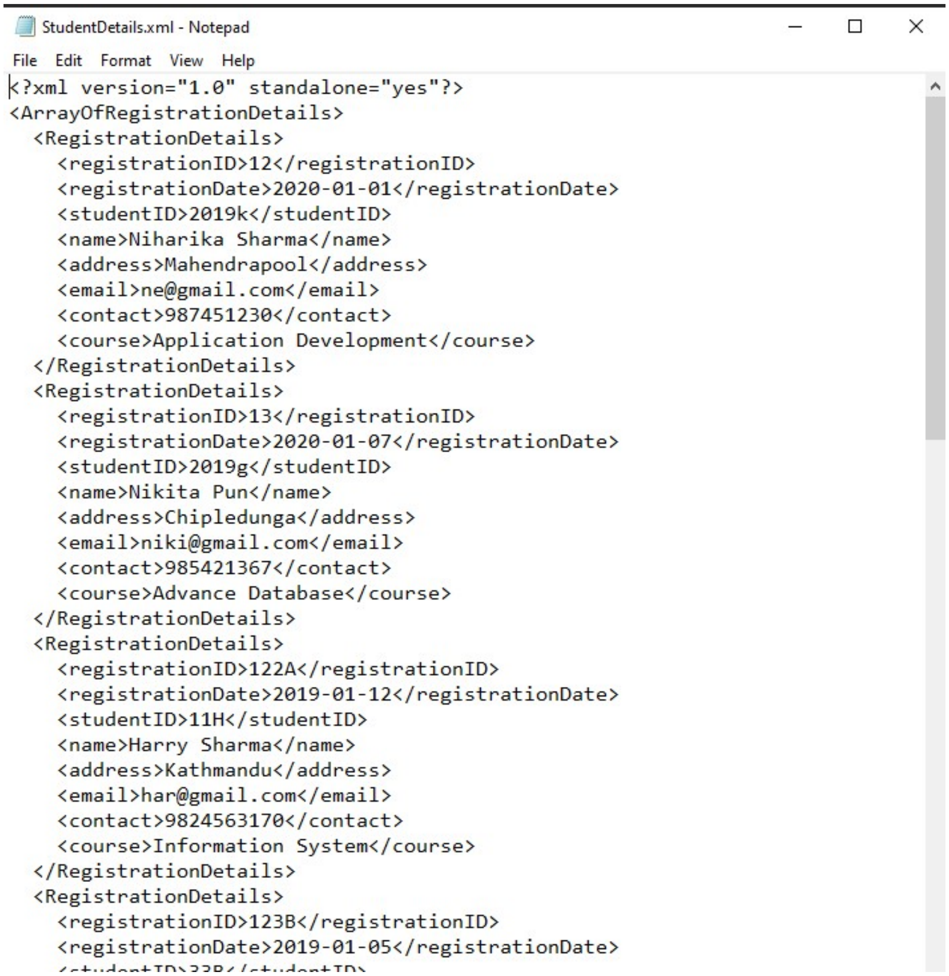
K:\YEAR 3\APPLICATION DEVELOPMENT\CW1

Note: The file should be in CSV format.

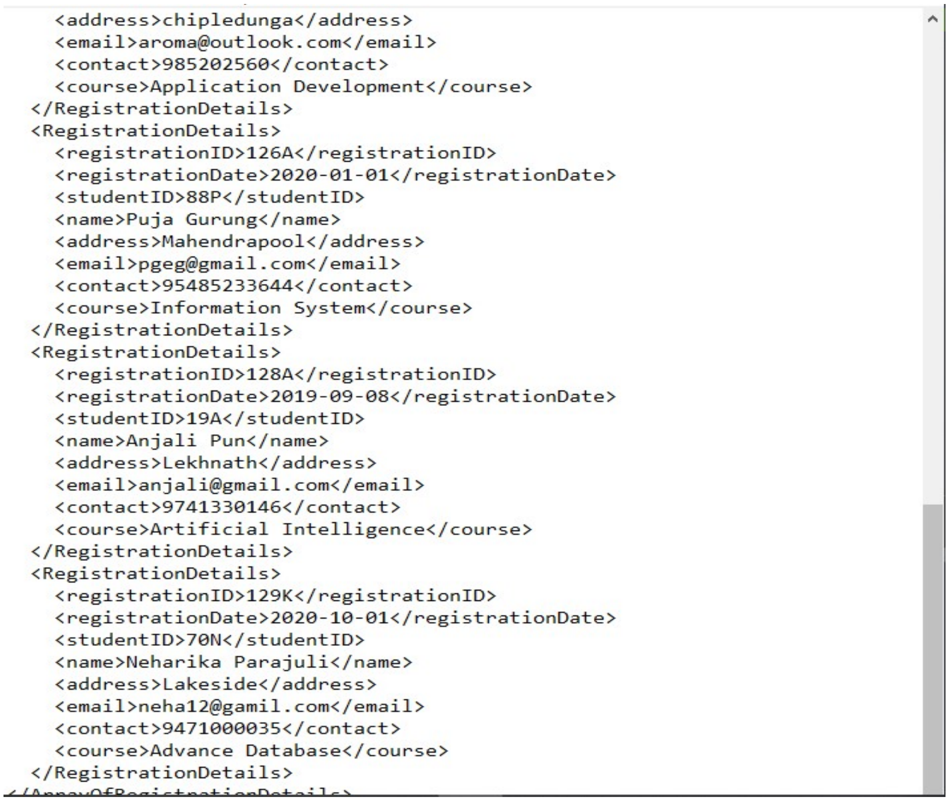
Import Sucessful!!!

It is sucessfully imported to xml file.

Figure 18: Testing 5

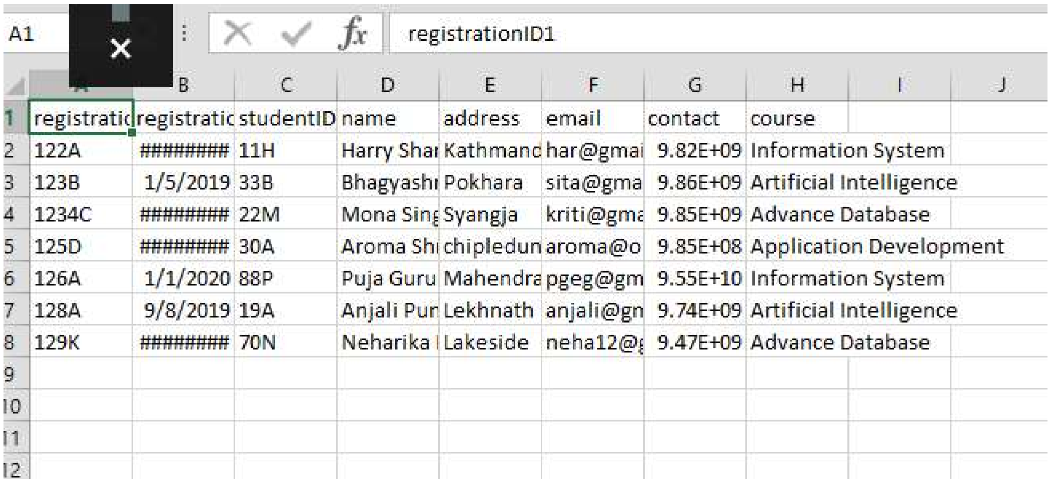


```xml
<?xml version="1.0" standalone="yes"?>
<ArrayOfRegistrationDetails>
  <RegistrationDetails>
    <registrationID>12</registrationID>
    <registrationDate>2020-01-01</registrationDate>
    <studentID>2019k</studentID>
    <name>Niharika Sharma</name>
    <address>Mahendrapool</address>
    <email>ne@gmail.com</email>
    <contact>987451230</contact>
    <course>Application Development</course>
  </RegistrationDetails>
  <RegistrationDetails>
    <registrationID>13</registrationID>
    <registrationDate>2020-01-07</registrationDate>
    <studentID>2019g</studentID>
    <name>Nikita Pun</name>
    <address>Chipledunga</address>
    <email>niki@gmail.com</email>
    <contact>985421367</contact>
    <course>Advance Database</course>
  </RegistrationDetails>
  <RegistrationDetails>
    <registrationID>122A</registrationID>
    <registrationDate>2019-01-12</registrationDate>
    <studentID>11H</studentID>
    <name>Harry Sharma</name>
    <address>Kathmandu</address>
    <email>har@gmail.com</email>
    <contact>9824563170</contact>
    <course>Information System</course>
  </RegistrationDetails>
  <RegistrationDetails>
    <registrationID>123B</registrationID>
    <registrationDate>2019-01-05</registrationDate>
    <studentID>33B</studentID>
```



```
    <address>chipledunga</address>
    <email>aroma@outlook.com</email>
    <contact>985202560</contact>
    <course>Application Development</course>
  </RegistrationDetails>
  <RegistrationDetails>
    <registrationID>126A</registrationID>
    <registrationDate>2020-01-01</registrationDate>
    <studentID>88P</studentID>
    <name>Puja Gurung</name>
    <address>Mahendrapool</address>
    <email>pgeg@gmail.com</email>
    <contact>95485233644</contact>
    <course>Information System</course>
  </RegistrationDetails>
  <RegistrationDetails>
    <registrationID>128A</registrationID>
    <registrationDate>2019-09-08</registrationDate>
    <studentID>19A</studentID>
    <name>Anjali Pun</name>
    <address>Lekhnath</address>
    <email>anjali@gmail.com</email>
    <contact>9741330146</contact>
    <course>Artificial Intelligence</course>
  </RegistrationDetails>
  <RegistrationDetails>
    <registrationID>129K</registrationID>
    <registrationDate>2020-10-01</registrationDate>
    <studentID>70N</studentID>
    <name>Neharika Parajuli</name>
    <address>Lakeside</address>
    <email>neha12@gamil.com</email>
    <contact>9471000035</contact>
    <course>Advance Database</course>
  </RegistrationDetails>
</ArrayOfRegistrationDetails>
```

Figure 19: Testing 5.1



	A	B	C	D	E	F	G	H	I	J
1	registratic	registratic	studentID	name	address	email	contact	course		
2	122A	########	11H	Harry Shar	Kathmanc	har@gmai	9.82E+09	Information System		
3	123B	1/5/2019	33B	Bhagyashı	Pokhara	sita@gma	9.86E+09	Artificial Intelligence		
4	1234C	########	22M	Mona Sing	Syangja	kriti@gma	9.85E+09	Advance Database		
5	125D	########	30A	Aroma Shı	chipledun	aroma@o	9.85E+08	Application Development		
6	126A	1/1/2020	88P	Puja Guru	Mahendra	pgeg@gm	9.55E+10	Information System		
7	128A	9/8/2019	19A	Anjali Pur	Lekhnath	anjali@gn	9.74E+09	Artificial Intelligence		
8	129K	########	70N	Neharika I	Lakeside	neha12@	9.47E+09	Advance Database		
9										
10										
11										
12										

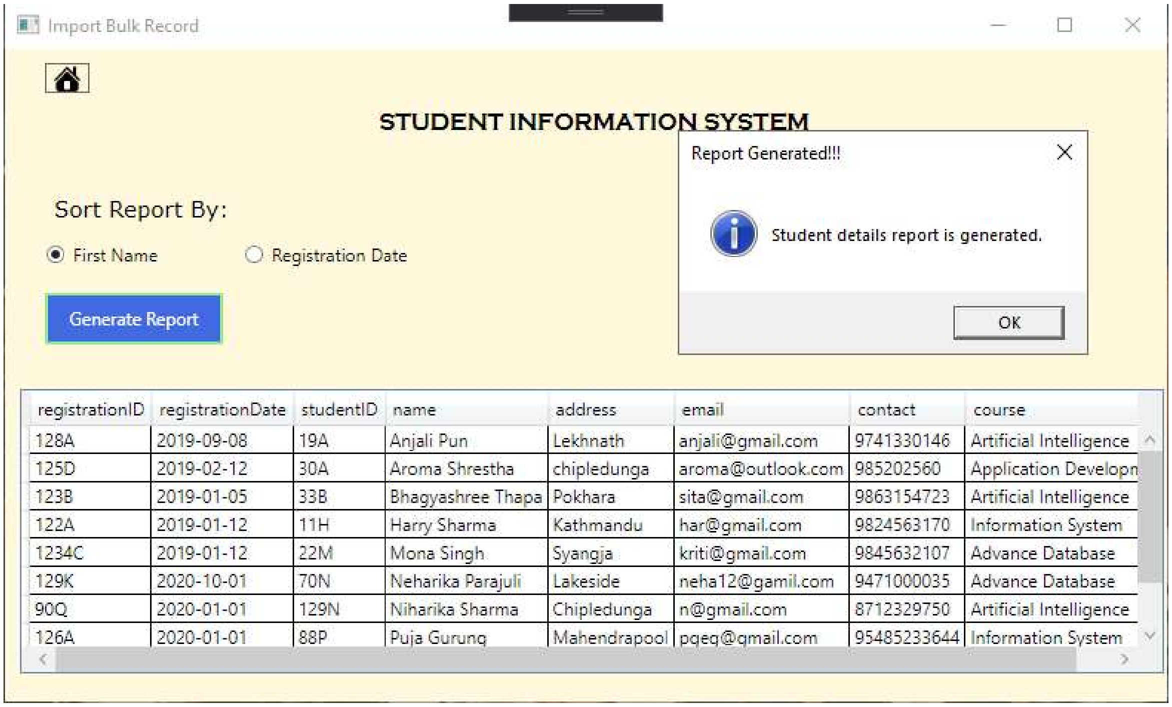
Figure 20: Testing 5.2

5.6 Testing 6

Objective: Check if the report is sorted by first name and displayed to grid.

Output: Displays the data with the first name sorting.

Result: Successful.



Report Generated!!!

Student details report is generated.

registrationID	registrationDate	studentID	name	address	email	contact	course
128A	2019-09-08	19A	Anjali Pun	Lekhnath	anjali@gmail.com	9741330146	Artificial Intelligence
125D	2019-02-12	30A	Aroma Shrestha	chipledunga	aroma@outlook.com	985202560	Application Developn
123B	2019-01-05	33B	Bhagyashree Thapa	Pokhara	sita@gmail.com	9863154723	Artificial Intelligence
122A	2019-01-12	11H	Harry Sharma	Kathmandu	har@gmail.com	9824563170	Information System
1234C	2019-01-12	22M	Mona Singh	Syangja	kriti@gmail.com	9845632107	Advance Database
129K	2020-10-01	70N	Neharika Parajuli	Lakeside	neha12@gamil.com	9471000035	Advance Database
90Q	2020-01-01	129N	Niharika Sharma	Chipledunga	n@gmail.com	8712329750	Artificial Intelligence
126A	2020-01-01	88P	Puja Gurung	Mahendrapool	pqeq@gmail.com	95485233644	Information System

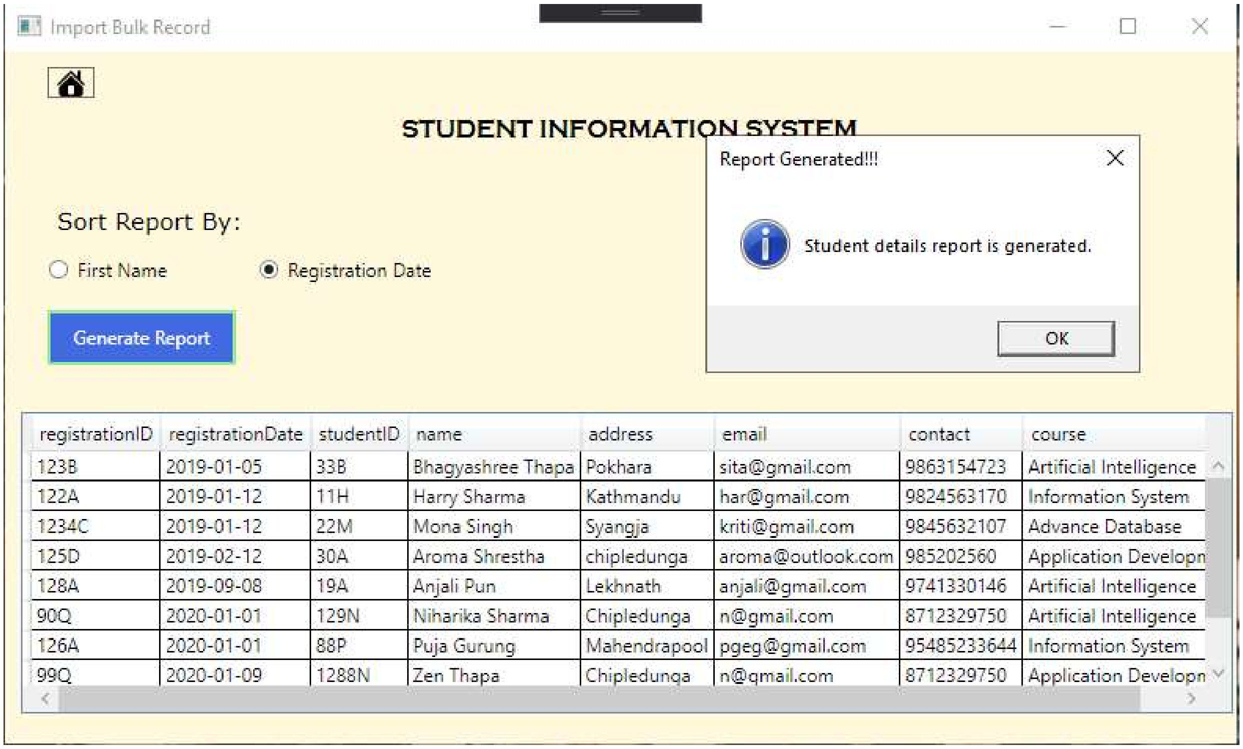
Figure 21: Testing 6

5.7 Testing 7

Objective: Check if the report is sorted by date and displayed to grid.

Output: Displays the data with the date sorting.

Result: Successful.



Report Generated!!!

Student details report is generated.

registrationID	registrationDate	studentID	name	address	email	contact	course
123B	2019-01-05	33B	Bhagyashree Thapa	Pokhara	sita@gmail.com	9863154723	Artificial Intelligence
122A	2019-01-12	11H	Harry Sharma	Kathmandu	har@gmail.com	9824563170	Information System
1234C	2019-01-12	22M	Mona Singh	Syangja	kriti@gmail.com	9845632107	Advance Database
125D	2019-02-12	30A	Aroma Shrestha	chipledunga	aroma@outlook.com	985202560	Application Developn
128A	2019-09-08	19A	Anjali Pun	Lekhnath	anjali@gmail.com	9741330146	Artificial Intelligence
90Q	2020-01-01	129N	Niharika Sharma	Chipledunga	n@gmail.com	8712329750	Artificial Intelligence
126A	2020-01-01	88P	Puja Gurung	Mahendrapool	pgeg@gmail.com	95485233644	Information System
99Q	2020-01-09	1288N	Zen Thapa	Chipledunga	n@gmail.com	8712329750	Application Developn

Figure 22: Testing 7

5.8 Testing 8

Objective: Check if pie chart is created with the xml data.

Output: Displays pie chart.

Result: Successful.

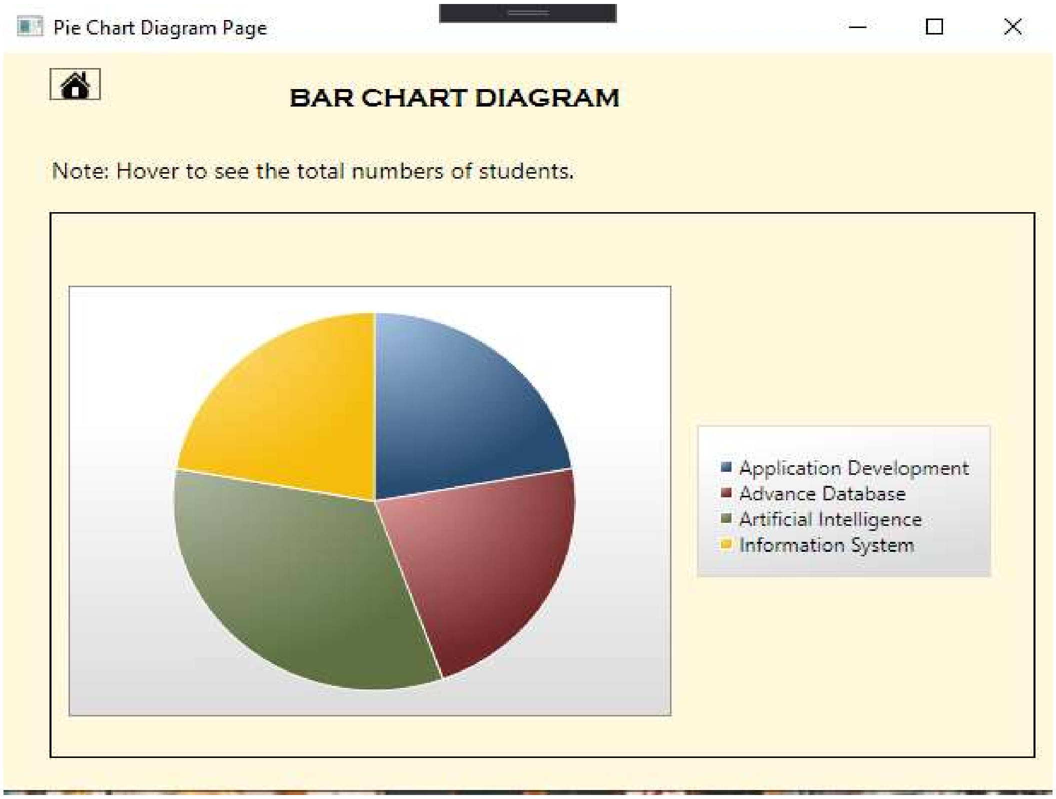


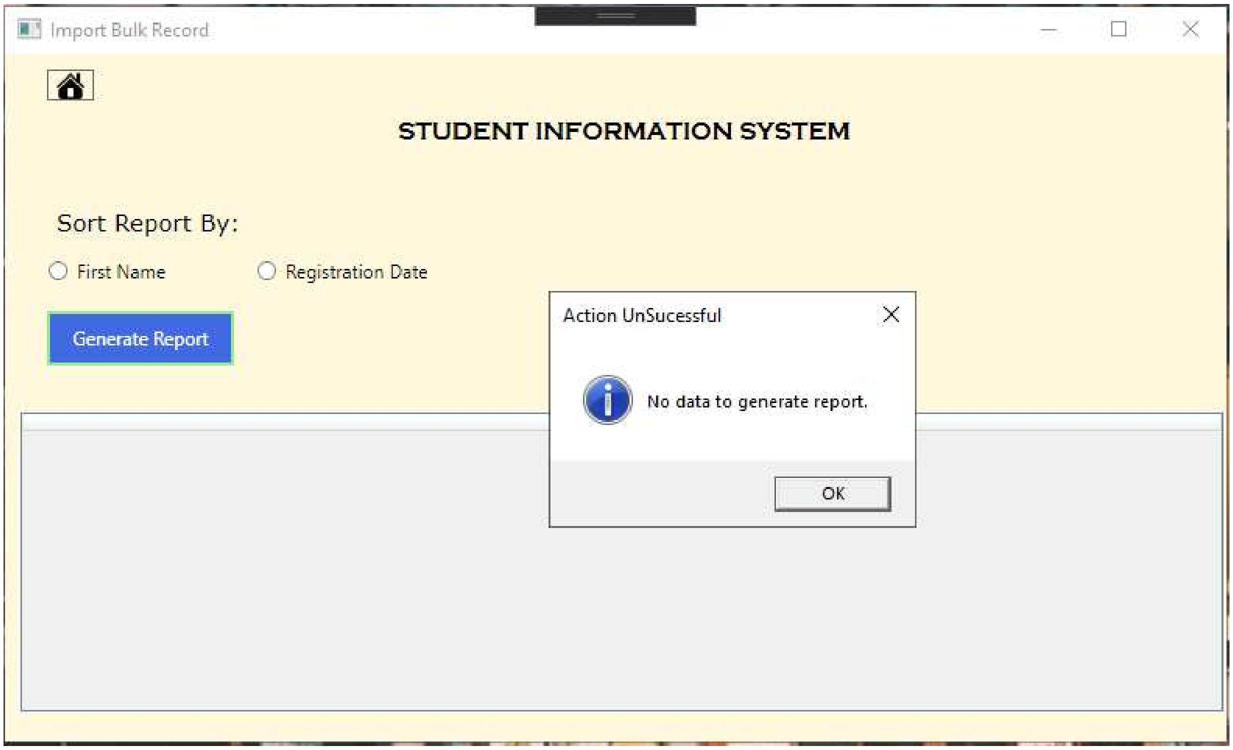
Figure 23: Testing 8

5.9 Testing 9

Objective: Display error message when there is no xml file for generating report.

Output: Displays error message.

Result: Successful.



Action UnSucessful

No data to generate report.

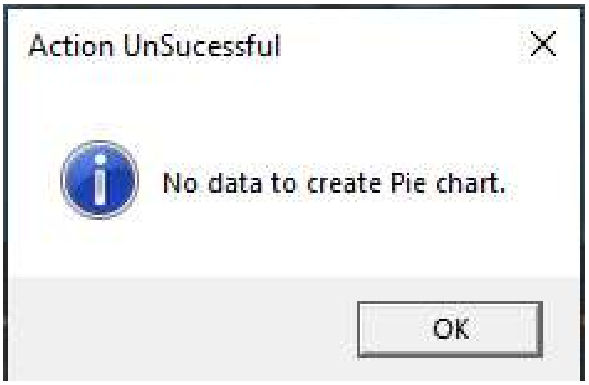
Figure 24: Testing 9

5.10 Testing 10

Objective: Display error message when there is no xml file for generating pie chart.

Output: Displays error message.

Result: Successful.



Action UnSucessful

No data to create Pie chart.

Figure 25: Testing 10

6. Conclusion

7. Appendix