STUDENTS GRADING SYSTEM

DECEMBER, 2018

AIN SHAMS UNIVERSITY – FACULTY OF ENGINEERING
1 Al-Sarayyat st, Abbassiya, Cairo 11517, Egypt

FACULTY OF ENGINEERING

AIN SHAMS UNIVERSITY

CSE221: Object-Oriented Analysis and Design

Students Grading System

Submitted By:

Abdelrahman Ibrahim ELGhamry

Ghamry98@hotmail.com
16P3043@eng.asu.edu.eg

Hossam ELDin Khaled Mohmed

hossampen97@gmail.com
16P3025@eng.asu.edu.eg

Abdelrahman Amr Issawi

aid-issawi@hotmail.com
16P6001@eng.asu.edu.eg

Submitted to:

Prof. Dr. Islam El-Maddah

DECEMBER, 2018

A REPORT FOR OBJECT-ORIENTED ANALYSIS AND DESIGN COURSE CODDED CSE221 WITH THE REQUIREMENTS OF AIN SHAMS UNIVERSITY

Table of Contents

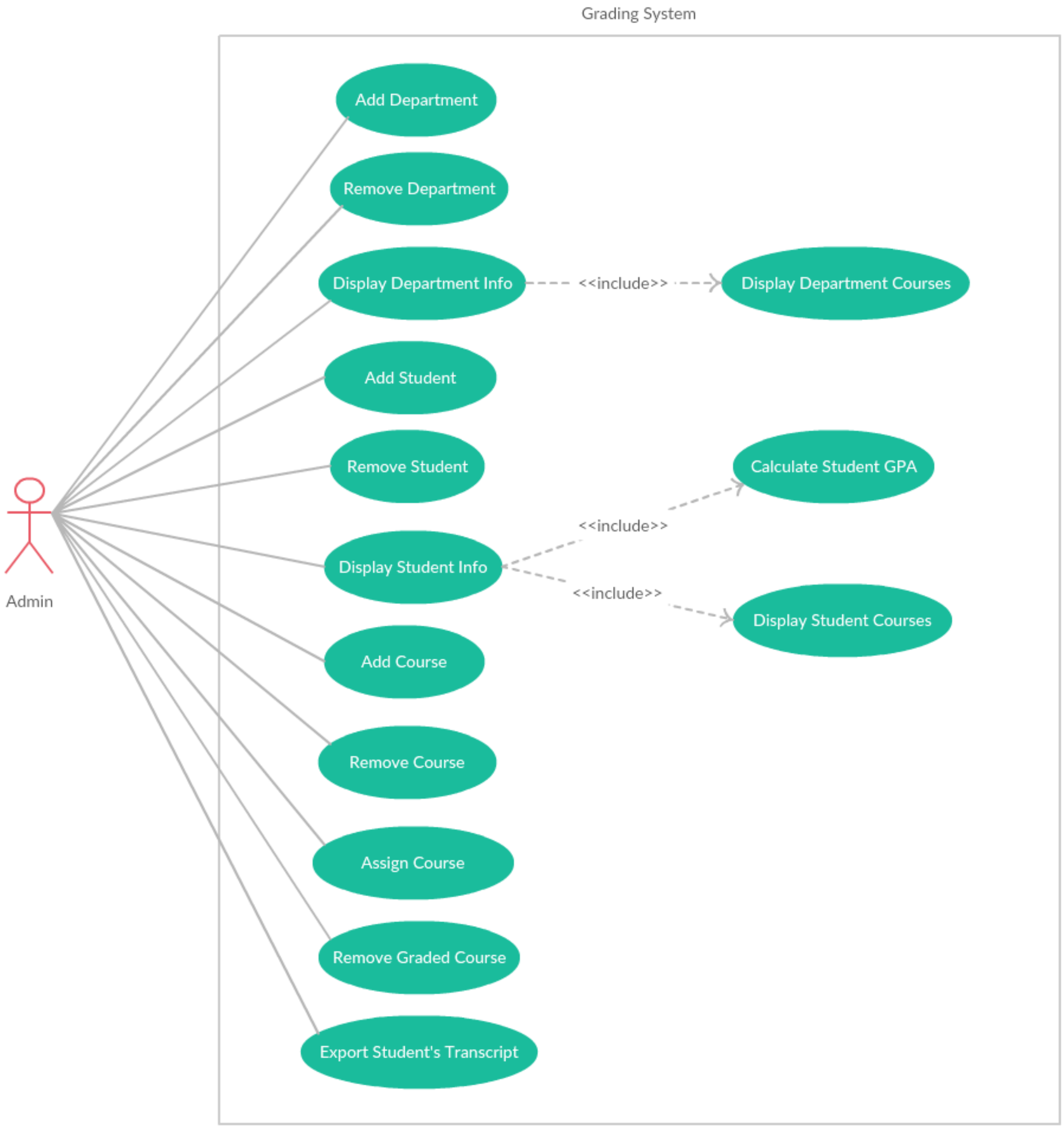
1.0 PROBLEM STATEMENT

Develop a system to include university departments and students' information. Each department offer some courses and each student must choose a department. The system should consider the courses taken by a student and their grades.

The system calculates the cumulative GPA for each student and can provide a transcript that contains all courses taken by the student.

2.0 USE CASE

2.1 Use Case Diagram



2.2 Use Case Description

Use Case name:	Add Department.
Goal in context:	To add a new department to the system.
Pre-Conditions:	None.
Successful end condition:	Department is added successfully to the system.
Failure end condition:	Department is not added to the system.
Primary Actor:	Admin.
Secondary Actor:	None.
Trigger:	Admin clicked on add department button.
Input:	Department name and description.
Output:	None.
Main flow:	1. Admin clicks “Add New Department” button. 2. Add department form is opened. 3. Admin enters the department name and description. 4. Admin clicks “Ok” button.
Extensions:	4.1 Admin clicks “Cancel” button, so the system will ignore the written data in the text fields if exist. 4.2 Admin clicks “Ok” button without entering either the department name or the department description, system will display a message informing the user to re-enter the missing data.

Use Case name:	Remove Department.
Goal in context:	Remove a department from the system.
Pre-Conditions:	Department exists in the system.
Successful end condition:	Department is removed successfully from the system.
Failure end condition:	Department is not removed from the system.
Primary Actor:	Admin.
Secondary Actor:	None.
Trigger:	Admin clicked on remove department button.
Input:	Department index.
Output:	None.
Main flow:	1. Admin selects a department from the system departments. 2. Admin clicks on “Remove Department” button.
Extensions:	2.1 Admin clicks on “Remove Department” button without selecting a department at first, system will display a message telling the user to select a department.

Use Case name:	Add Student.
Goal in context:	Add a student to the system.
Pre-Conditions:	Department exists in the system.
Successful end condition:	Student is added successfully to the system.
Failure end condition:	Student is not added to the system.
Primary Actor:	Admin.
Secondary Actor:	None.
Trigger:	Admin clicked on add new student button.
Input:	Student Name, Email, Phone number, and department.
Output:	None.
Main flow:	1. Admin clicks on ”Add New Student” button. 2. Add student form is opened. 3. Admin enters the student’s data (Name, Email, Phone number, and department). 4. Admin clicks on “OK” button.
Extensions:	4.1 Admin clicks “Cancel” button, so the system will ignore the written data in the text fields if exist. 4.2 Admin clicks “Ok” button without entering any of the data needed to add the student, system will display a message informing the user to re-enter the missing data.

Use Case name:	Remove Student.
Goal in context:	Remove a student from the system.
Pre-Conditions:	Student exists in the system.
Successful end condition:	Student is removed successfully from the system.
Failure end condition:	Student is not removed from the system.
Primary Actor:	Admin.
Secondary Actor:	None.
Trigger:	Admin clicked on remove student button.
Input:	Student index.
Output:	None.
Main flow:	1. Admin selects a student from the system students. 2. Admin clicks on “Remove Student” button.
Extensions:	2.1 Admin clicks on “Remove Student” button without selecting a student at first, system will display a message telling the user to select a student.

Use Case name:	Add Course.
Goal in context:	Add a new course to the system.
Pre-Conditions:	Department exists in the system.
Successful end condition:	Course is added successfully to the system.
Failure end condition:	Course is not added to the system.
Primary Actor:	Admin.
Secondary Actor:	None.
Trigger:	Admin clicked on add course button.
Input:	Course name.
Output:	None.
Main flow:	1. Admin clicks on “Add New Course” button. 2. Add new course form is opened. 3. Admin enters the course name. 4. Admin clicks on “Ok” button.
Extensions:	4.1 Admin clicks “Cancel” button, so the system will ignore the written data in the text fields if exist. 4.2 Admin clicks “Ok” button without entering any of the data needed to add the course, system will display a message informing the user to re-enter the missing data.

Use Case name:	Remove Course.
Goal in context:	Remove a new course to the system.
Pre-Conditions:	Course exists in the system.
Successful end condition:	Course is removed successfully from the system.
Failure end condition:	Course is not removed from the system.
Primary Actor:	Admin.
Secondary Actor:	None.
Trigger:	Admin clicked on remove course button.
Input:	Course index.
Output:	None.
Main flow:	1. Admin selects a course from the system courses. 2. Admin clicks on “Remove course” button.
Extensions:	2.1 Admin clicks on “Remove Course” button without selecting a course at first, system will display a message telling the user to select a course.

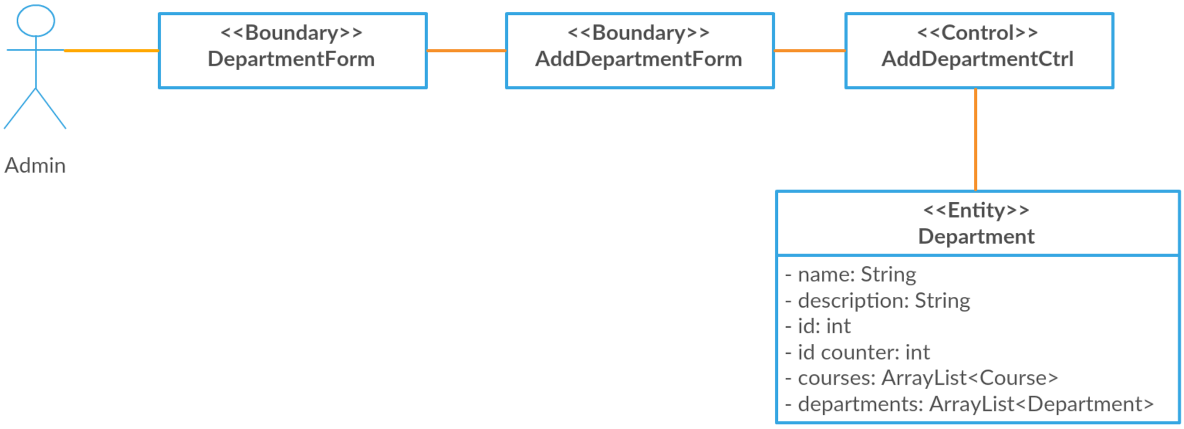
Use Case name:	Assign Course.
Goal in context:	Assign a graded course to the student.
Pre-Conditions:	Student exists in the system.
Successful end condition:	Course is assigned to the student successfully.
Failure end condition:	Course is not assigned to the student.
Primary Actor:	Admin.
Secondary Actor:	None.
Trigger:	Admin clicked on assign new course button.
Input:	Course name and grade.
Output:	None.
Main flow:	1. Admin selects a student from the system students. 2. Admin clicks on “Assign New Course” button. 3. Assign new course form is opened. 4. Admin chooses the course name and enter the course grade. 5. Admin clicks on “Ok” button.
Extensions:	5.1 Admin enters a GPA not in range [0,4], the system displays an error message with the specified range.

Use Case name:	Remove Graded Course.
Goal in context:	Remove a graded course from the student’s courses.
Pre-Conditions:	Graded course exists in the system.
Successful end condition:	Course is removed from the student’s courses successfully.
Failure end condition:	Course is not removed from the student’s courses.
Primary Actor:	Admin.
Secondary Actor:	None.
Trigger:	Admin clicked on remove course button.
Input:	Course index.
Output:	None.
Main flow:	1. Admin selects a course from the student courses. 2. Admin clicks on the course name. 3. Admin clicks on “Remove Course” button.
Extensions:	3.1 Admin clicked on “Remove Course” button without selecting a course. System will display a message telling the user to select a course.

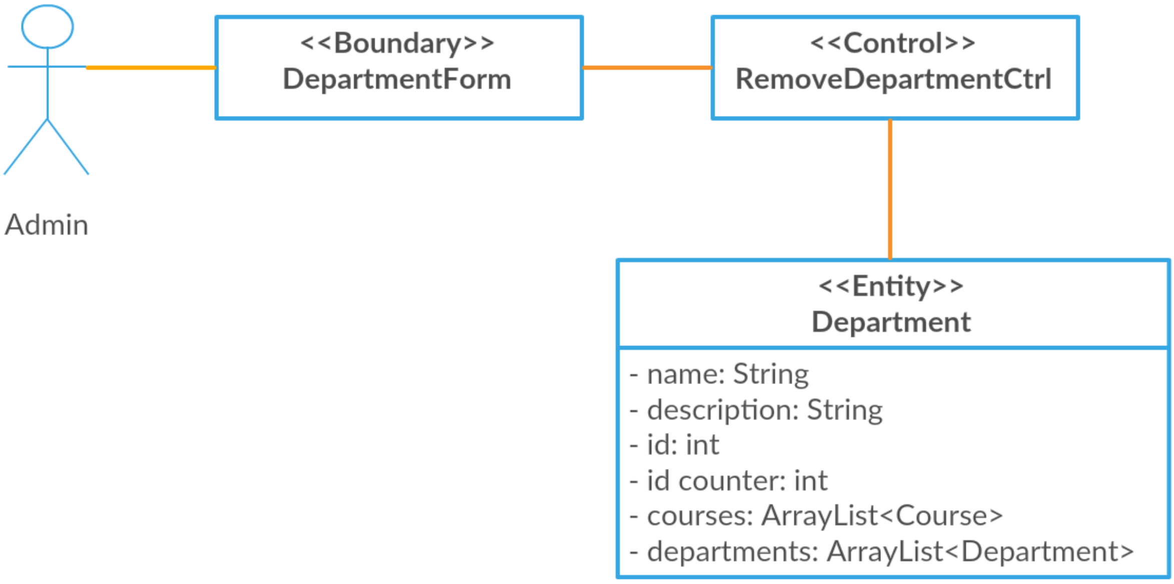
Use Case name:	Export Student's Transcript.
Goal in context:	Export the student's transcript.
Pre-Conditions:	Student is selected.
Successful end condition:	Transcript is exported successfully.
Failure end condition:	Transcript is not exported.
Primary Actor:	Admin.
Secondary Actor:	None.
Trigger:	Admin clicked on export transcript button.
Input:	Students courses and grades.
Output:	Detailed Student transcript.
Main flow:	1. Admin selects a student from the system students. 2. Admin clicks on "Export Transcript" button. 3. System displays a message if the transcript is exported successfully.
Extensions:	2.1 Admin clicks on "Export Transcript" button without selecting a student at first, system will display a message telling the user to select a student.

3.0 ANALYSIS CLASSES

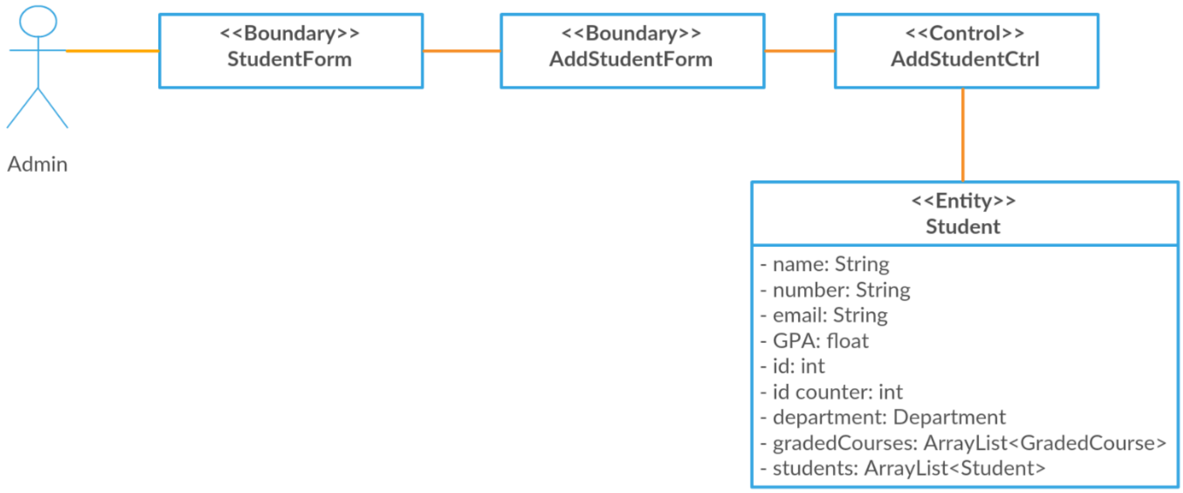
3.1 Add Department



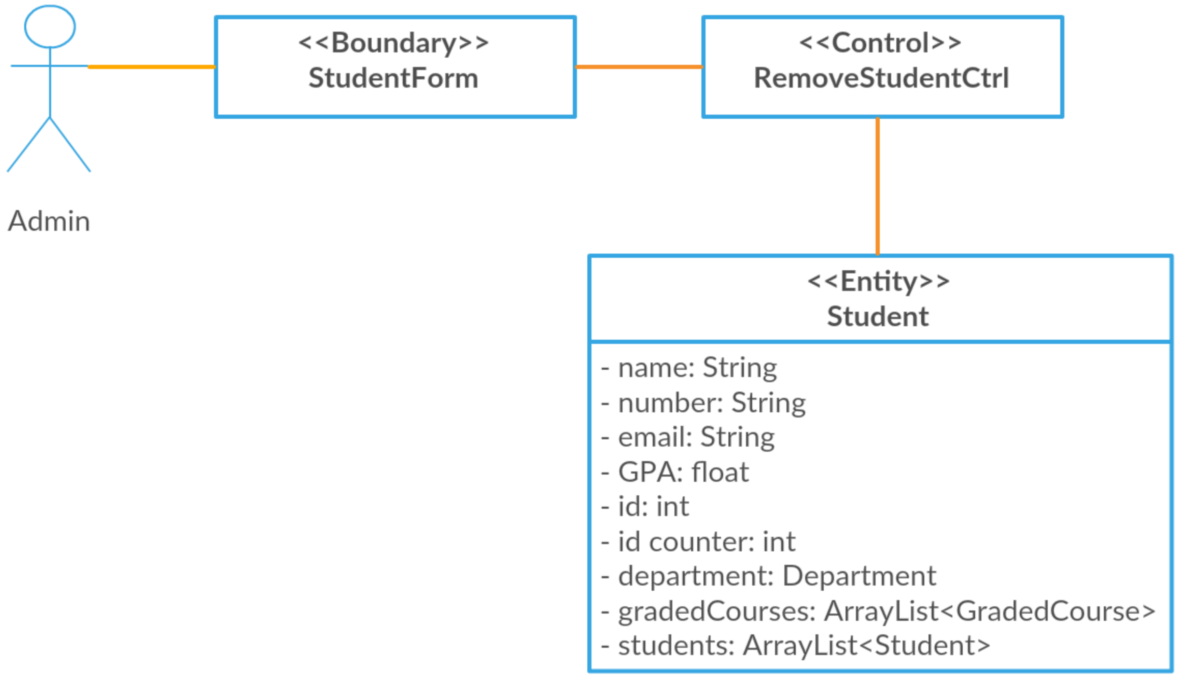
3.2 Remove Department



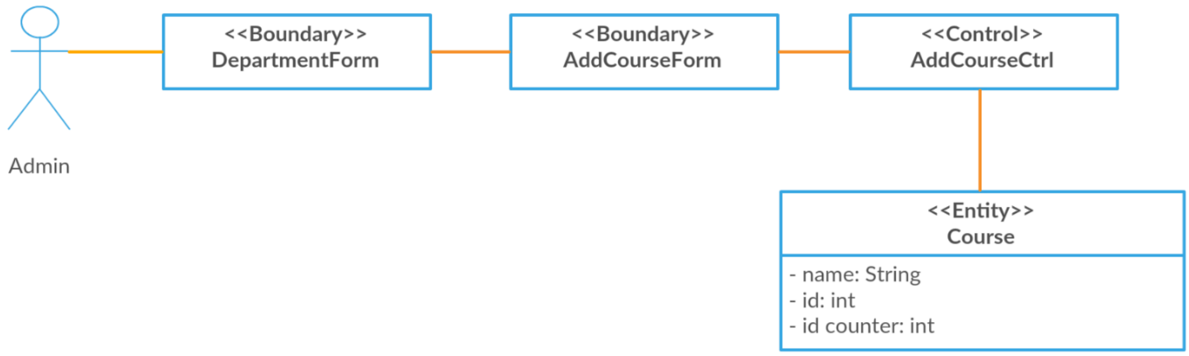
3.3 Add Student



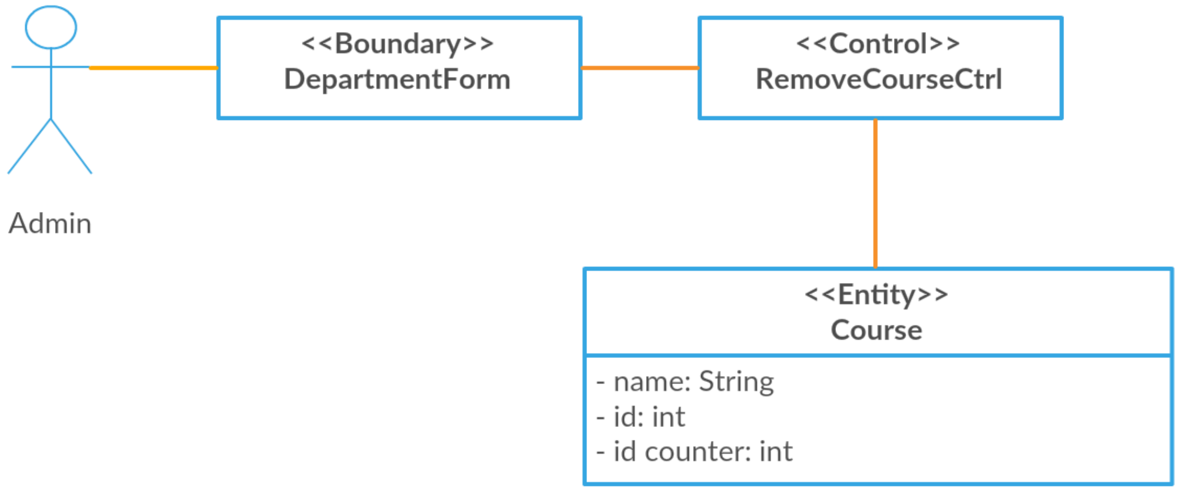
3.4 Remove Student



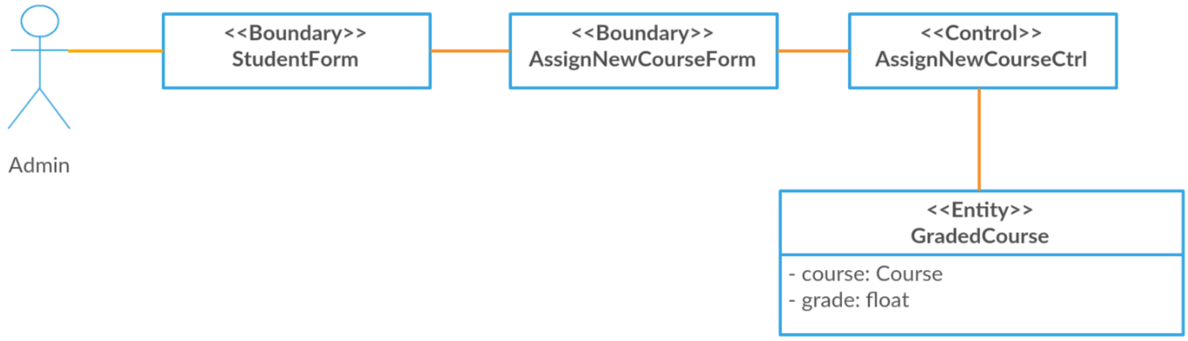
3.5 Add Course



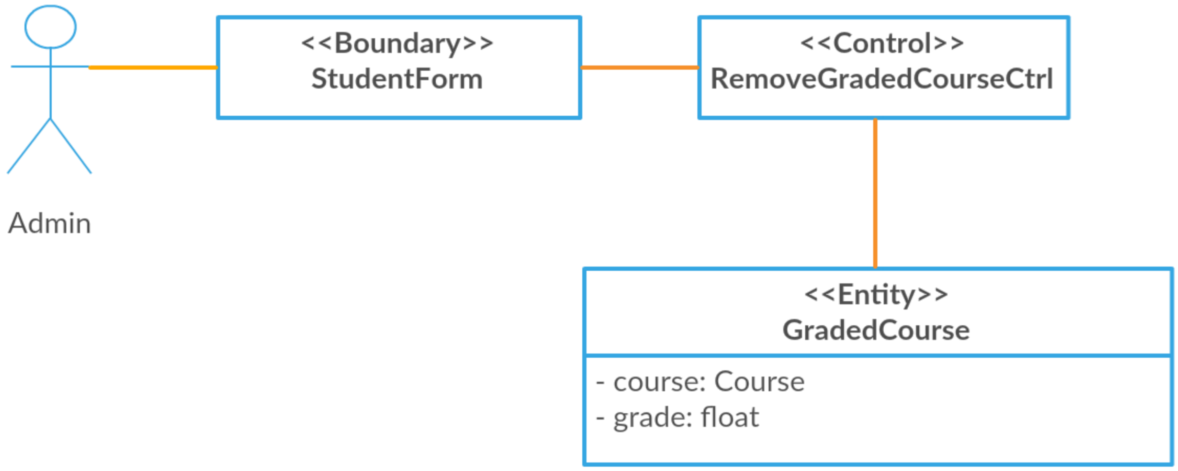
3.6 Remove Course



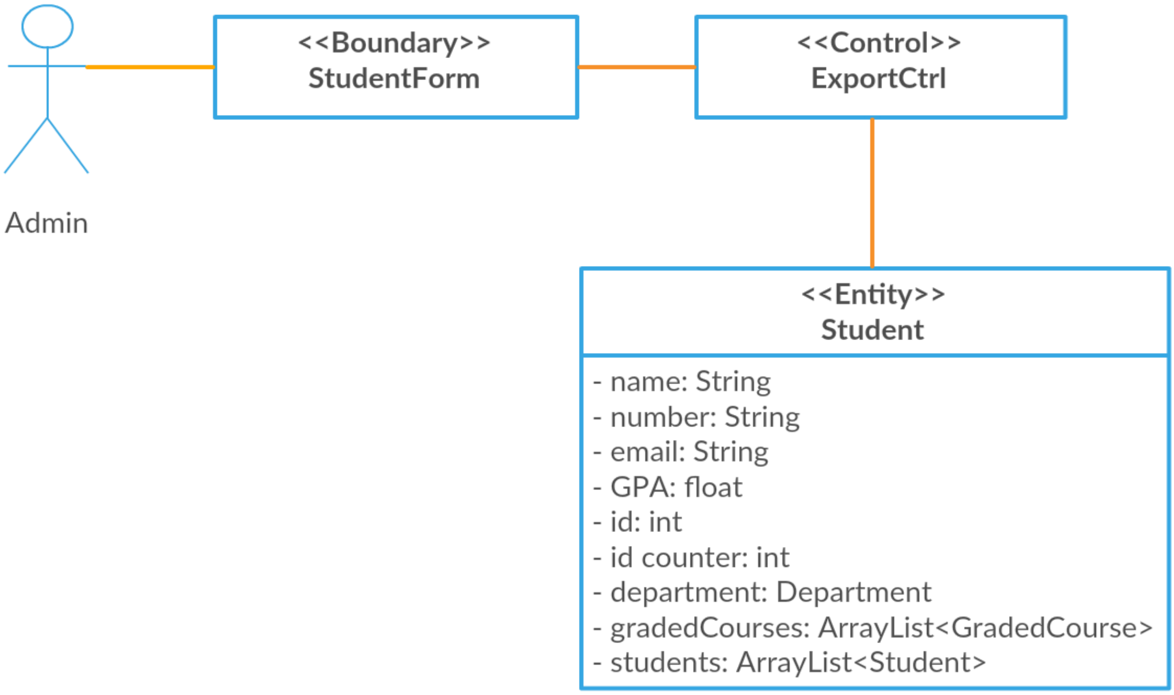
3.7 Assign Course



3.8 Remove Graded Course

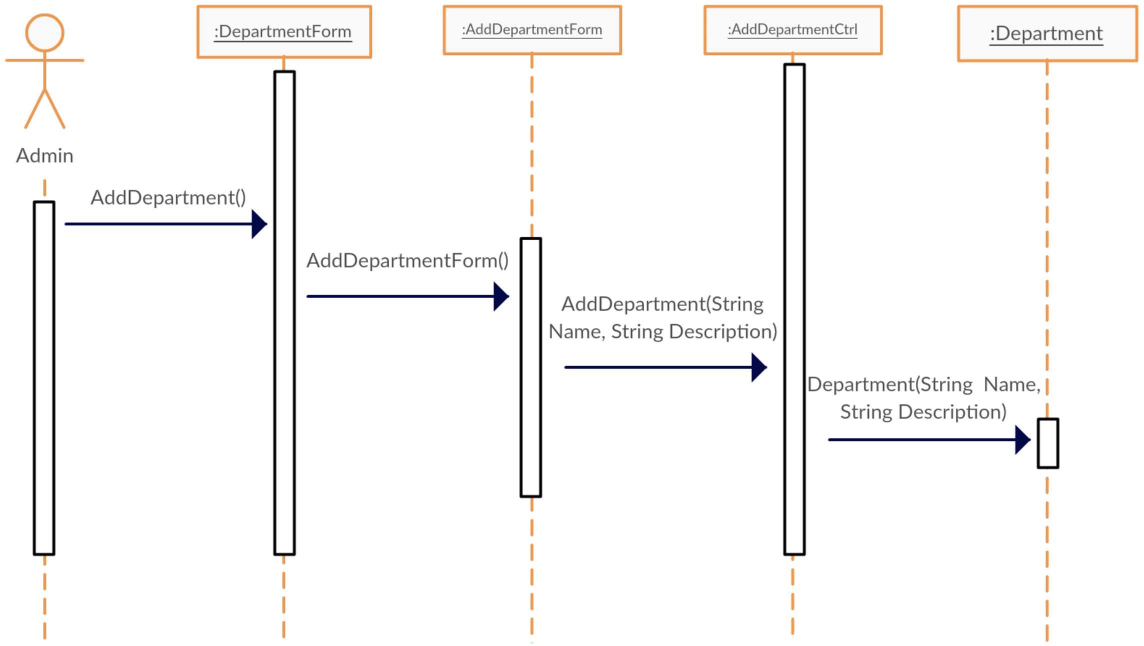


3.9 Export Student’s Transcript

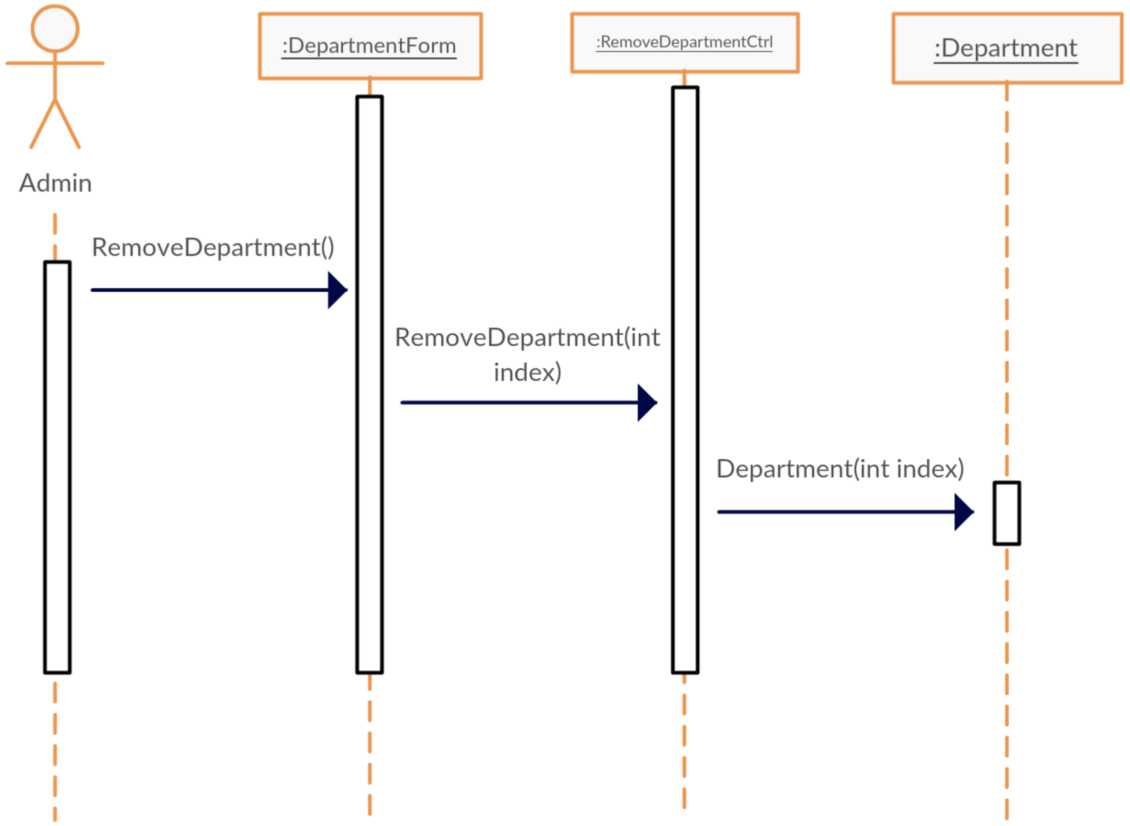


4.0 SEQUENCE DIAGRAM

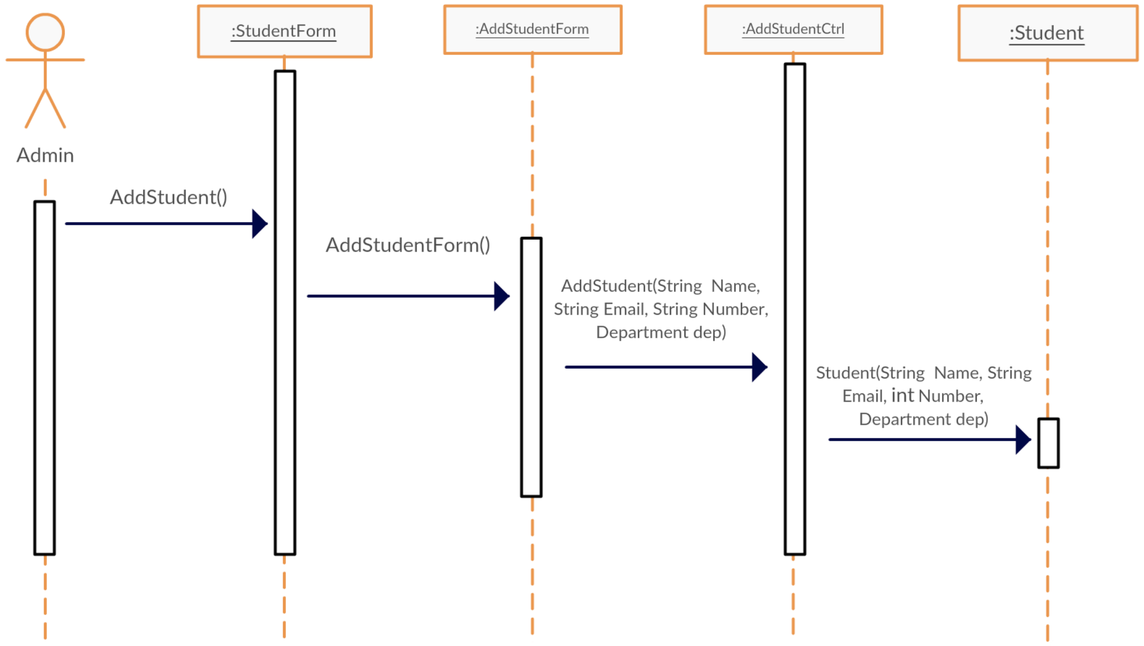
4.1 Add Department



4.2 Remove Department

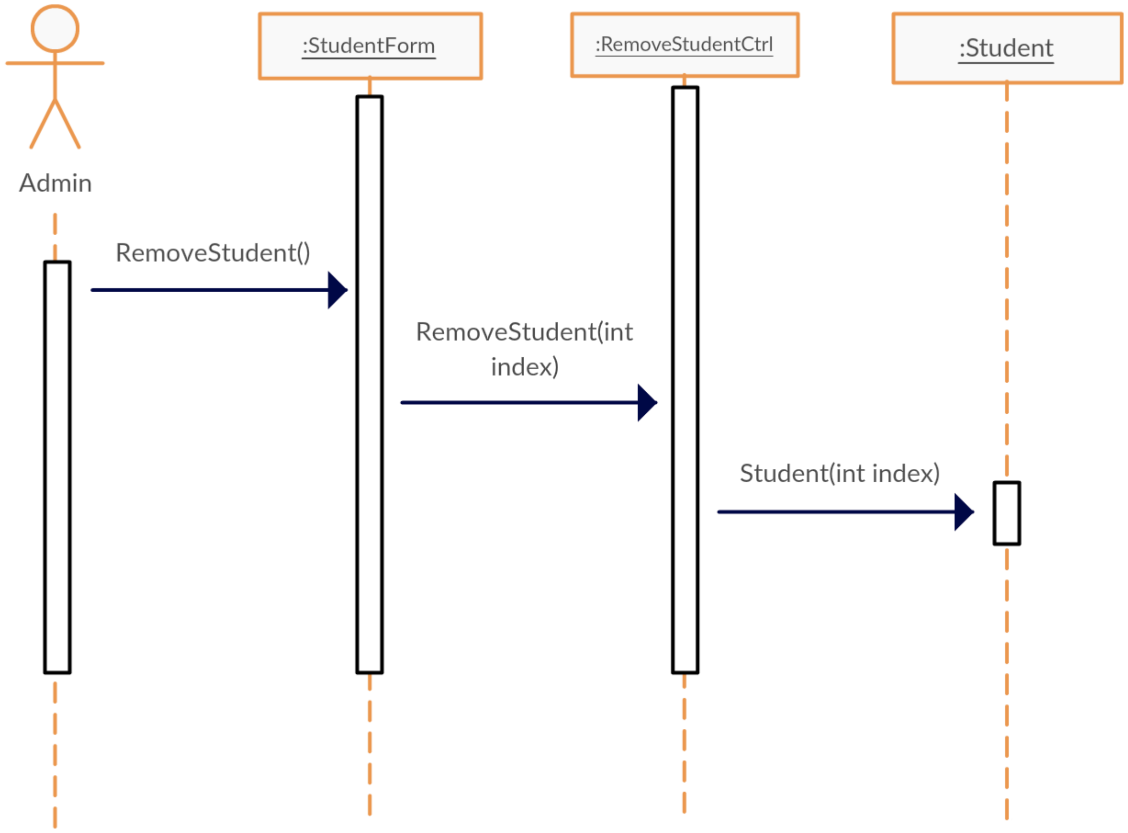


4.3 Add Student

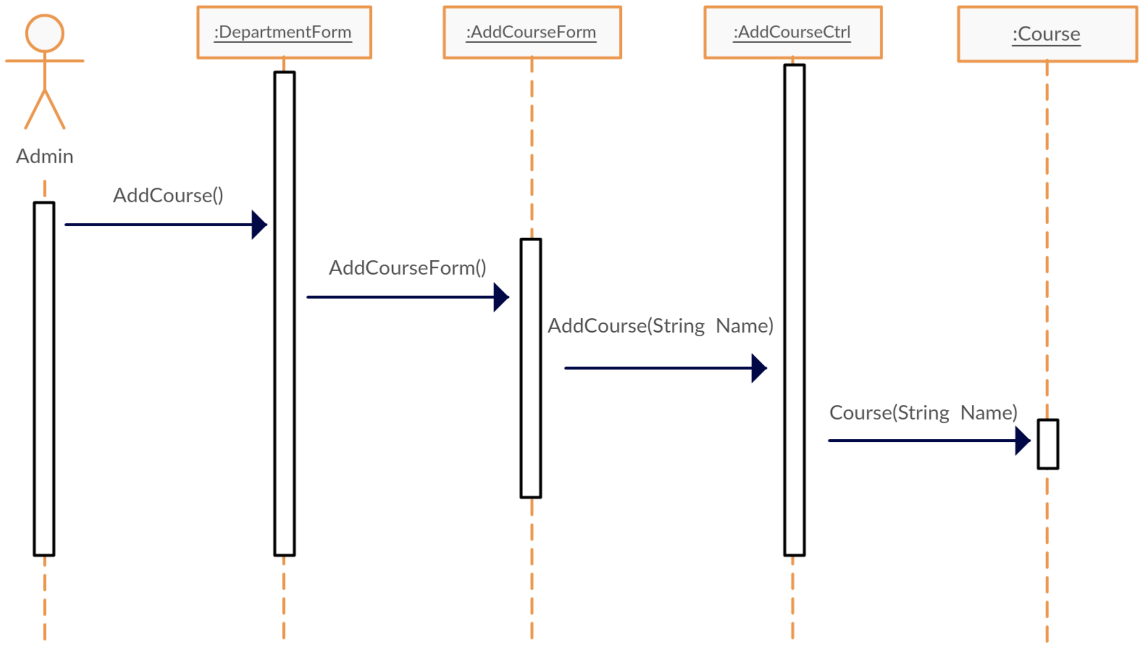


Admin
:StudentForm
:AddStudentForm
:AddStudentCtrl
:Student
AddStudent()
AddStudentForm()
AddStudent(String Name, String Email, String Number, Department dep)
Student(String Name, String Email, int Number, Department dep)

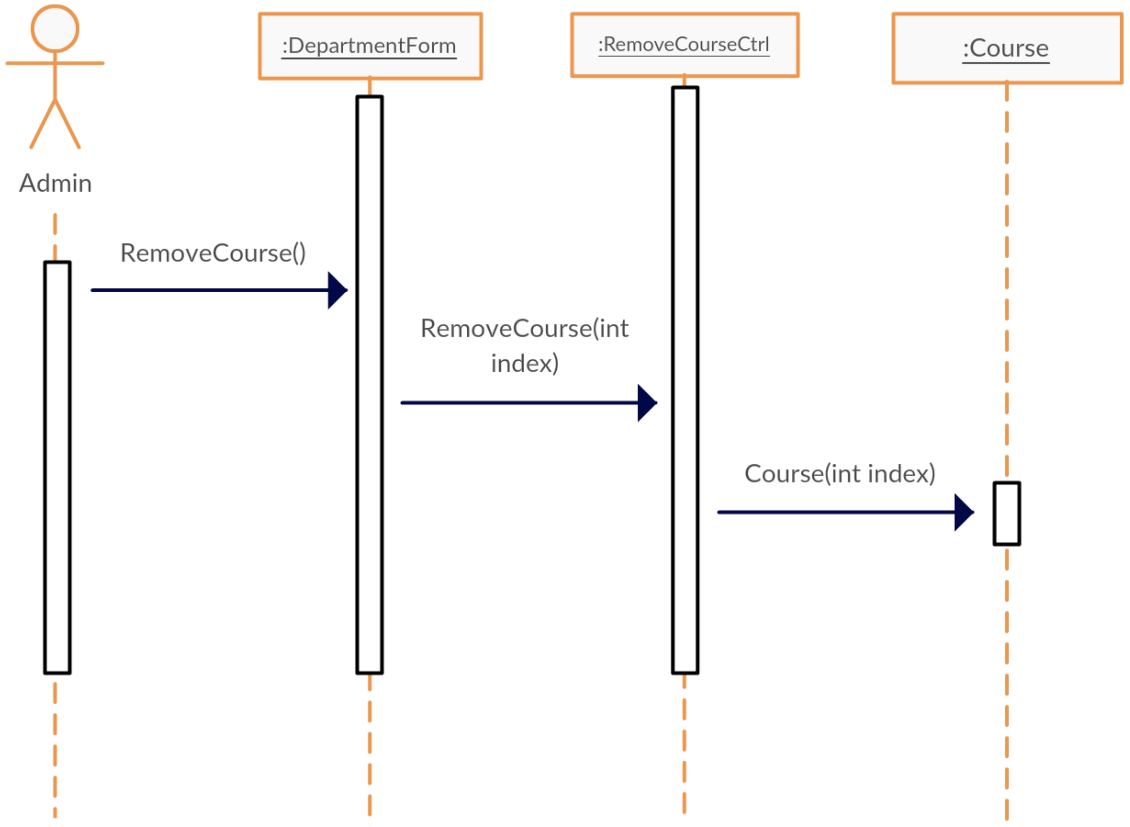
4.4 Remove Student



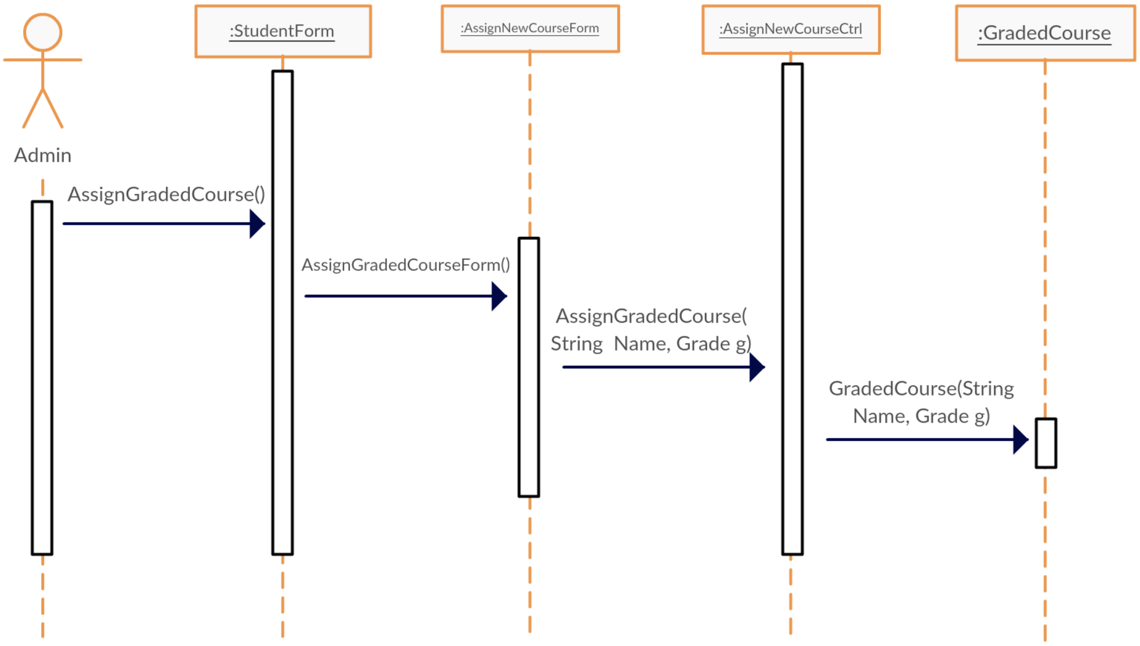
4.5 Add Course



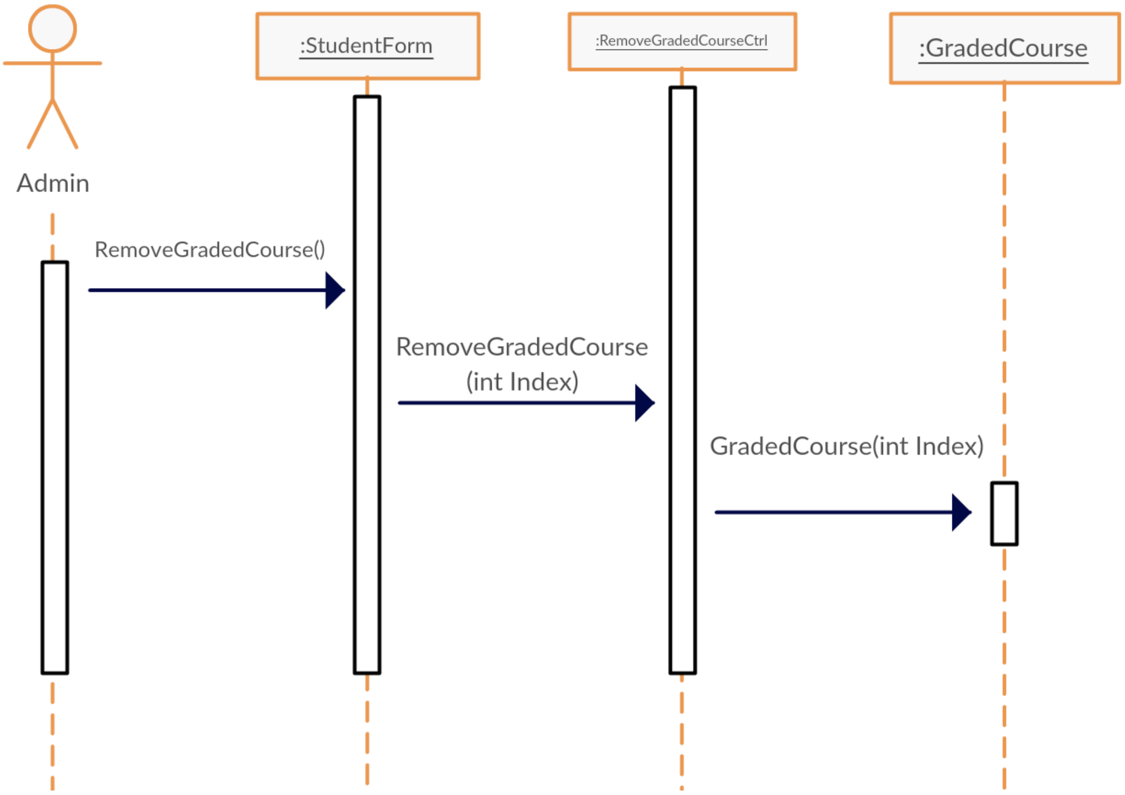
4.6 Remove Course



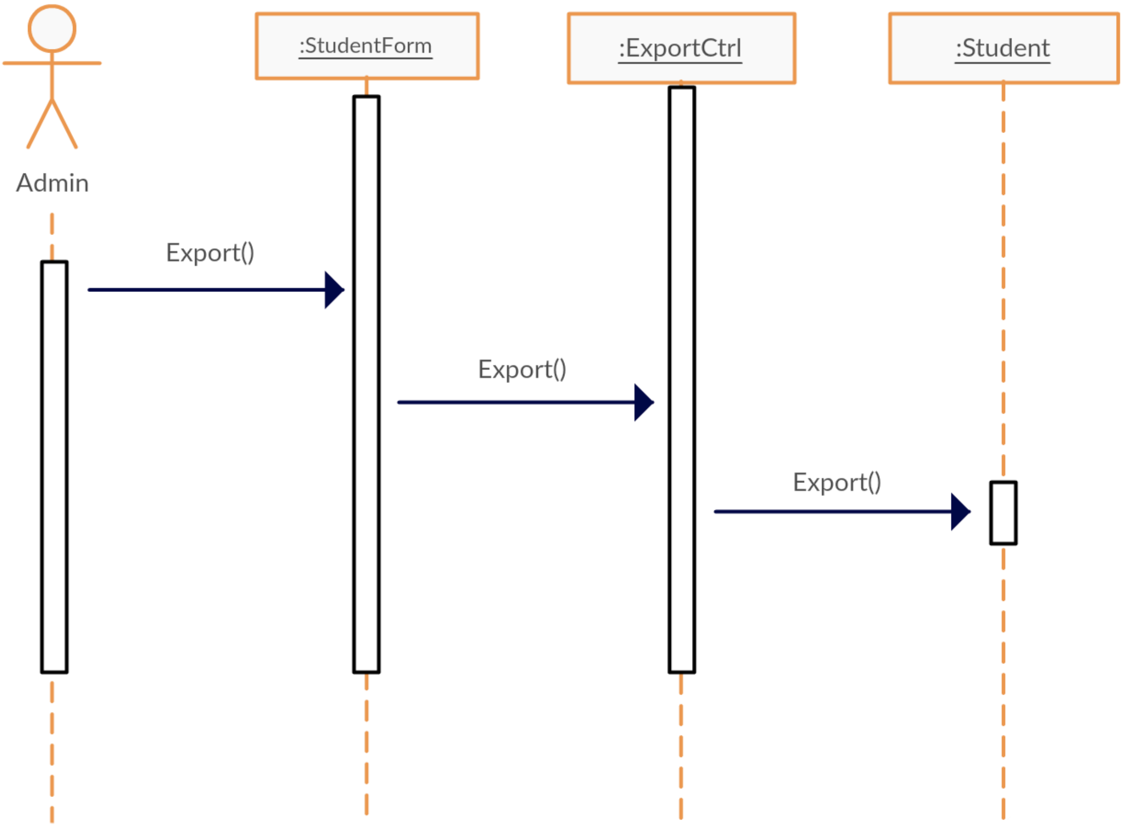
4.7 Assign Course



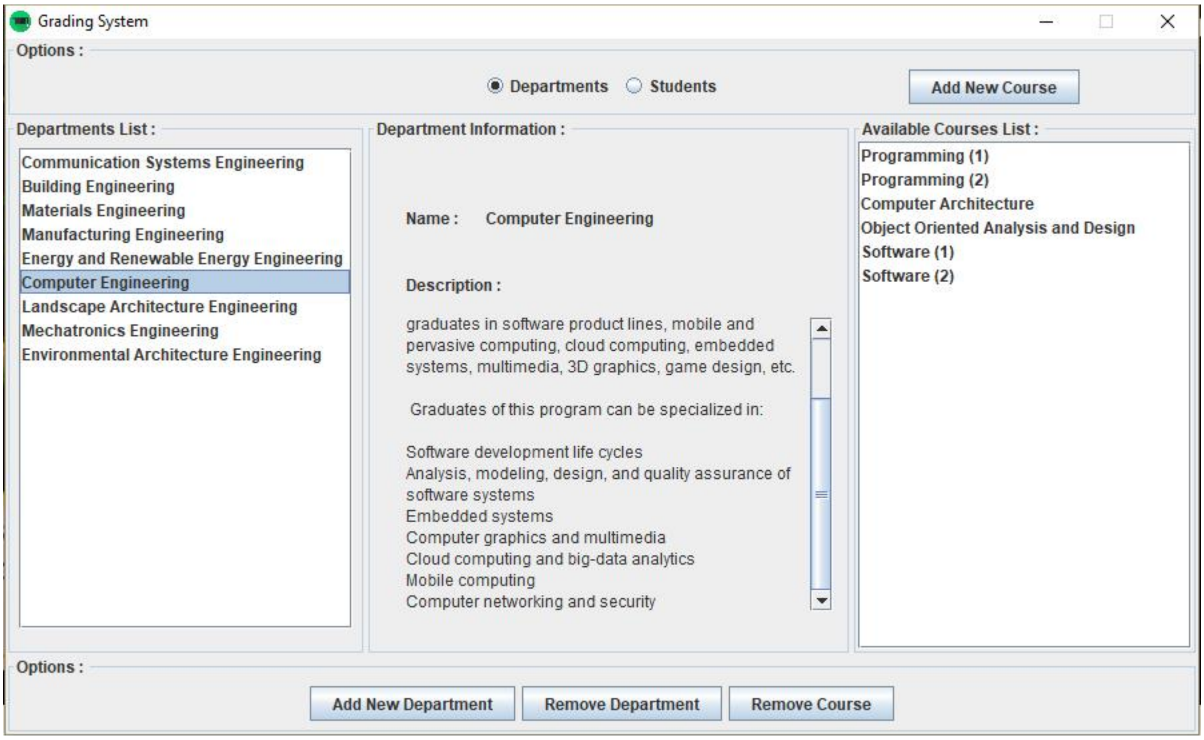
4.8 Remove Graded Course



4.9 Export Student's Transcript



5.0 THE RUNNING OF THE APPLICATION



Departments List:

- Communication Systems Engineering
- Building Engineering
- Materials Engineering
- Manufacturing Engineering
- Energy and Renewable Energy Engineering
- Computer Engineering
- Landscape Architecture Engineering
- Mechatronics Engineering
- Environmental Architecture Engineering

Name : Computer Engineering

Description :

graduates in software product lines, mobile and pervasive computing, cloud computing, embedded systems, multimedia, 3D graphics, game design, etc.

Graduates of this program can be specialized in:

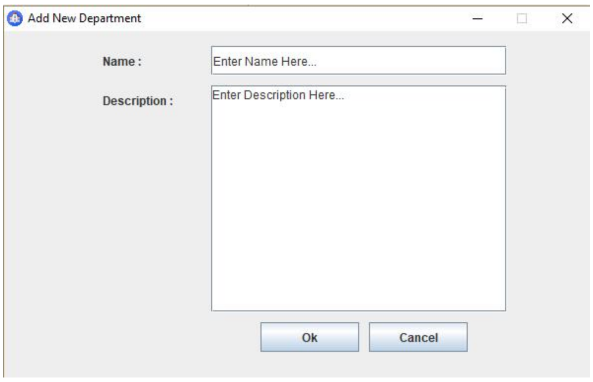
Software development life cycles
Analysis, modeling, design, and quality assurance of software systems
Embedded systems
Computer graphics and multimedia
Cloud computing and big-data analytics
Mobile computing
Computer networking and security

Available Courses List:

- Programming (1)
- Programming (2)
- Computer Architecture
- Object Oriented Analysis and Design
- Software (1)
- Software (2)

Department Form is the first view of the system, the admin may add a department or remove a selected one, also may add a course to the department or remove a selected course from the specified department. Whenever the admin selects a department name, a brief description about the department appears in the "Department Information" space.

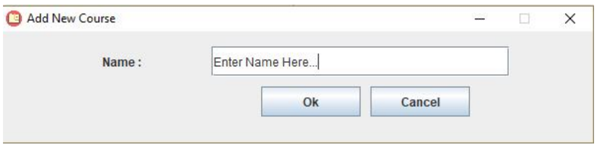
Add new department form appears once the admin clicks on "Add Department" button, the admin then in supposed to enter the department name and it's description.



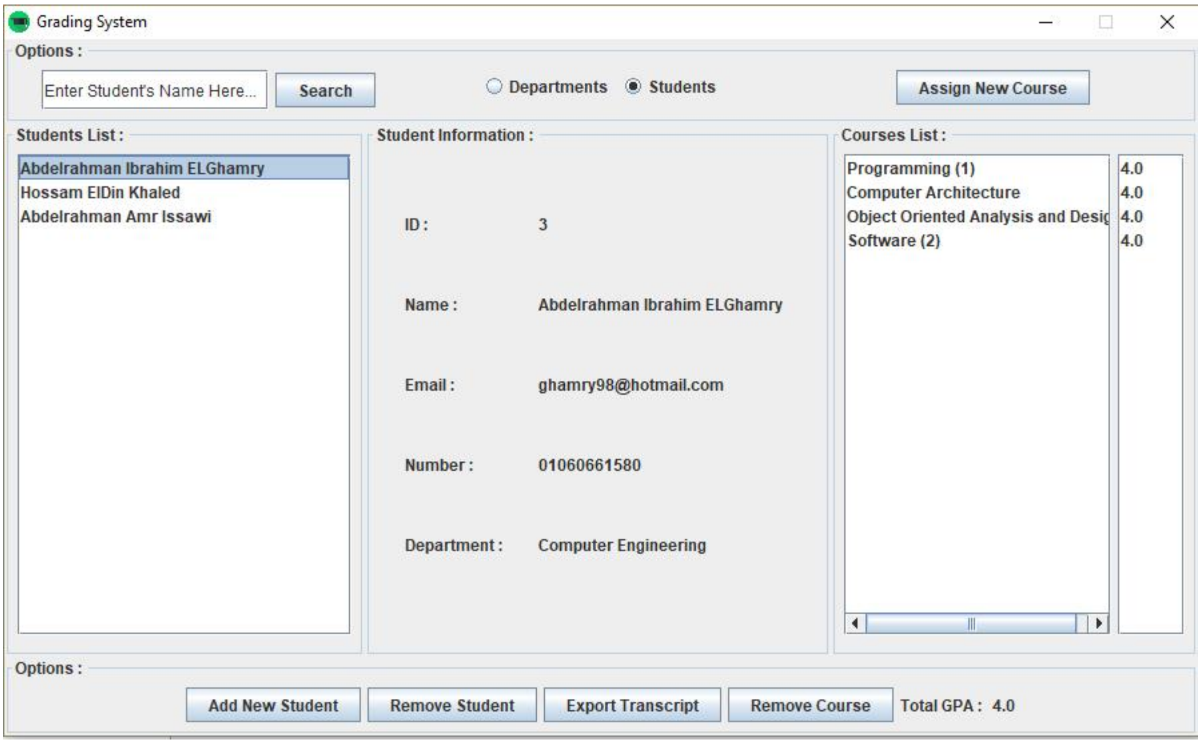
Name : Enter Name Here...

Description : Enter Description Here...

Add new course form appears once the admin clicks on "Add New Course" button, the admin then in supposed to enter the course name.



Name : Enter Name Here...



Abdelrahman Ibrahim ELGhamry
Hossam ElDin Khaled
Abdelrahman Amr Issawi

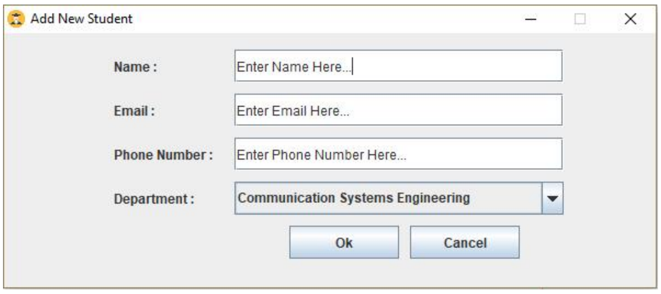
ID :	3
Name :	Abdelrahman Ibrahim ELGhamry
Email :	ghamry98@hotmail.com
Number :	01060661580
Department :	Computer Engineering

Programming (1)	4.0
Computer Architecture	4.0
Object Oriented Analysis and Desig	4.0
Software (2)	4.0

Total GPA : 4.0

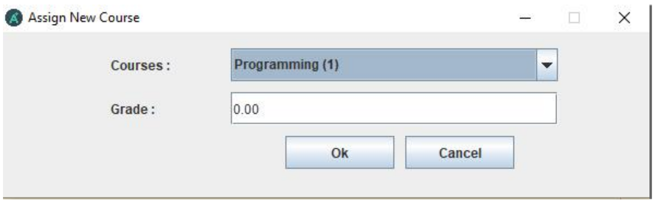
Admin can switch from the department form to the student form where he can add a student to the system or remove a selected student, assign a graded course to a selected student, remove a selected graded course from the specified student courses, and also the admin can export student's transcript for a selected student. Whenever the admin selects any student the student's data appear in the "Student Information" space, also the student graded courses appears in the "Course List" space.

Add new student appears once the admin clicks on "Add New Student" button, the admin then in supposed to enter the student data and choose the student's department.

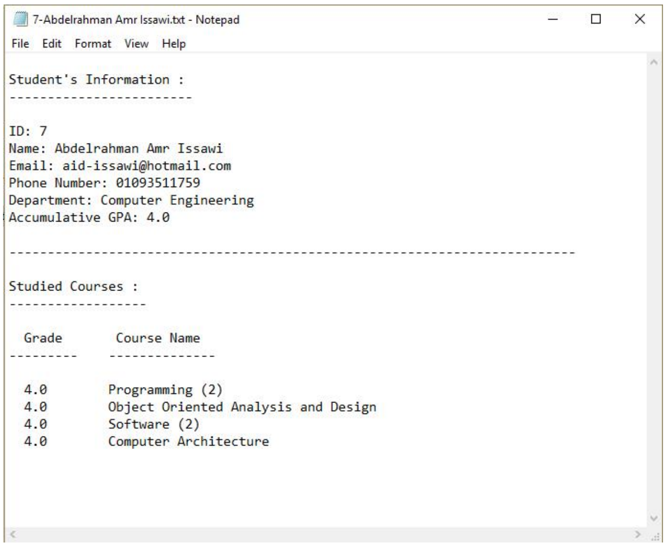


Name :	Enter Name Here...
Email :	Enter Email Here...
Phone Number :	Enter Phone Number Here...
Department :	Communication Systems Engineering

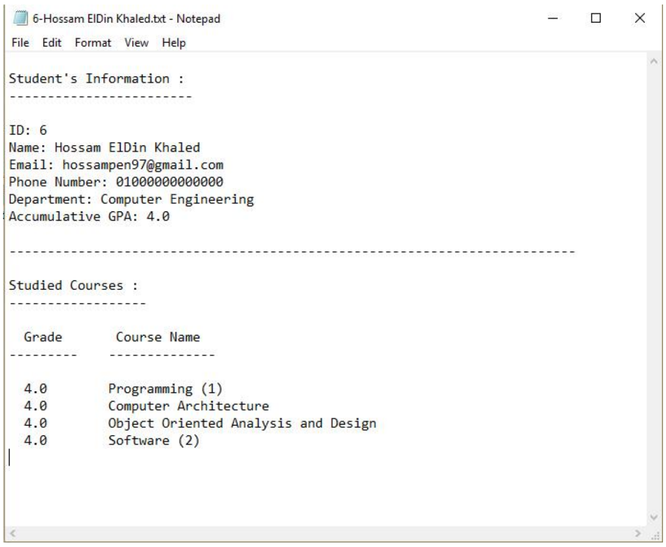
Assign new course appears once the admin clicks on "Assign New Course" button, the admin then in supposed to choose a course and enter it's GPA within range [0,4].



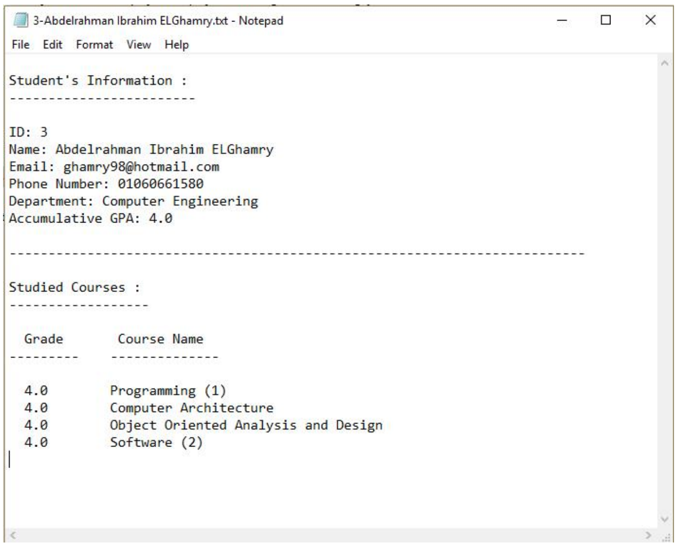
Courses :	Programming (1)
Grade :	0.00


7-Abdelrahman Amr Issawi.txt - Notepad
File Edit Format View Help

```
Student's Information :
-----------------------

ID: 7
Name: Abdelrahman Amr Issawi
Email: aid-issawi@hotmail.com
Phone Number: 01093511759
Department: Computer Engineering
Accumulative GPA: 4.0

-------------------------------------------------------------------------

Studied Courses :
-----------------

  Grade       Course Name
---------    -------------

  4.0        Programming (2)
  4.0        Object Oriented Analysis and Design
  4.0        Software (2)
  4.0        Computer Architecture
```


6-Hossam ElDin Khaled.txt - Notepad
File Edit Format View Help

```
Student's Information :
-----------------------

ID: 6
Name: Hossam ElDin Khaled
Email: hossampen97@gmail.com
Phone Number: 01000000000000
Department: Computer Engineering
Accumulative GPA: 4.0

-------------------------------------------------------------------------

Studied Courses :
-----------------

  Grade       Course Name
---------    -------------

  4.0        Programming (1)
  4.0        Computer Architecture
  4.0        Object Oriented Analysis and Design
  4.0        Software (2)
```


3-Abdelrahman Ibrahim ELGhamry.txt - Notepad
File Edit Format View Help

```
Student's Information :
-----------------------

ID: 3
Name: Abdelrahman Ibrahim ELGhamry
Email: ghamry98@hotmail.com
Phone Number: 01060661580
Department: Computer Engineering
Accumulative GPA: 4.0

-------------------------------------------------------------------------

Studied Courses :
-----------------

  Grade       Course Name
---------    -------------

  4.0        Programming (1)
  4.0        Computer Architecture
  4.0        Object Oriented Analysis and Design
  4.0        Software (2)
```

Whenever Admin clicks on “Export Transcript” button after selecting a student, a transcript is saved to the disk contains the student information (i.e. Name, email, …etc.), the student’s accumulative GPA calculated, the student studied courses and each course grade.