Faculty of Engineering & Technology
Electrical & Computer Engineering Department

Linux Laboratory-ENCS3310

First semester 2021-2022

Project Report

Students Records Management System

Prepared by:

Eyab Ghifari-1190999

Kareem Afaneh-1190359

Instructor's Name: Dr.Khader Mohammad **&** Dr. Mohammad Jubran
Assistant's Name: Eng.Saleem Hamo **&** Eng.Katy Sadi

Section: 1 & 2
Date: 07-01-2022

Objectives:

- Provide operations related to students record system like :
  - Add a new record
  - Add new semester with student course and grades
  - Update Student Record
  - Show Student statistics
  - Show Global statistics
- Use OOP concepts.
- Use data structures.
- Clean and Well Indented Code.

Contents

Results and Screen Shots:

System:

Here is the main menu of the Program, which shows the available options for the user:

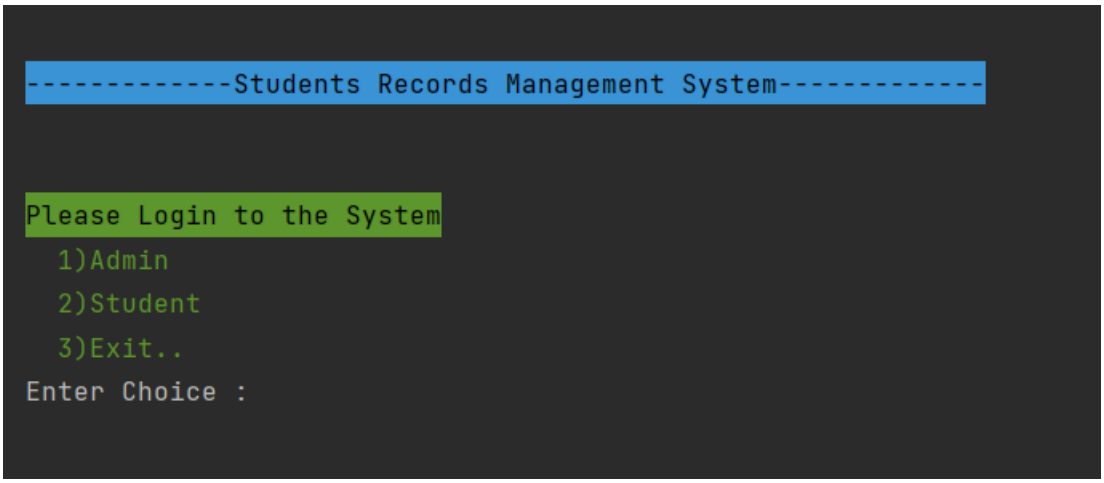

Fig. 1 Students Records Management System Menu

The next figure represents inserting a wrong input:

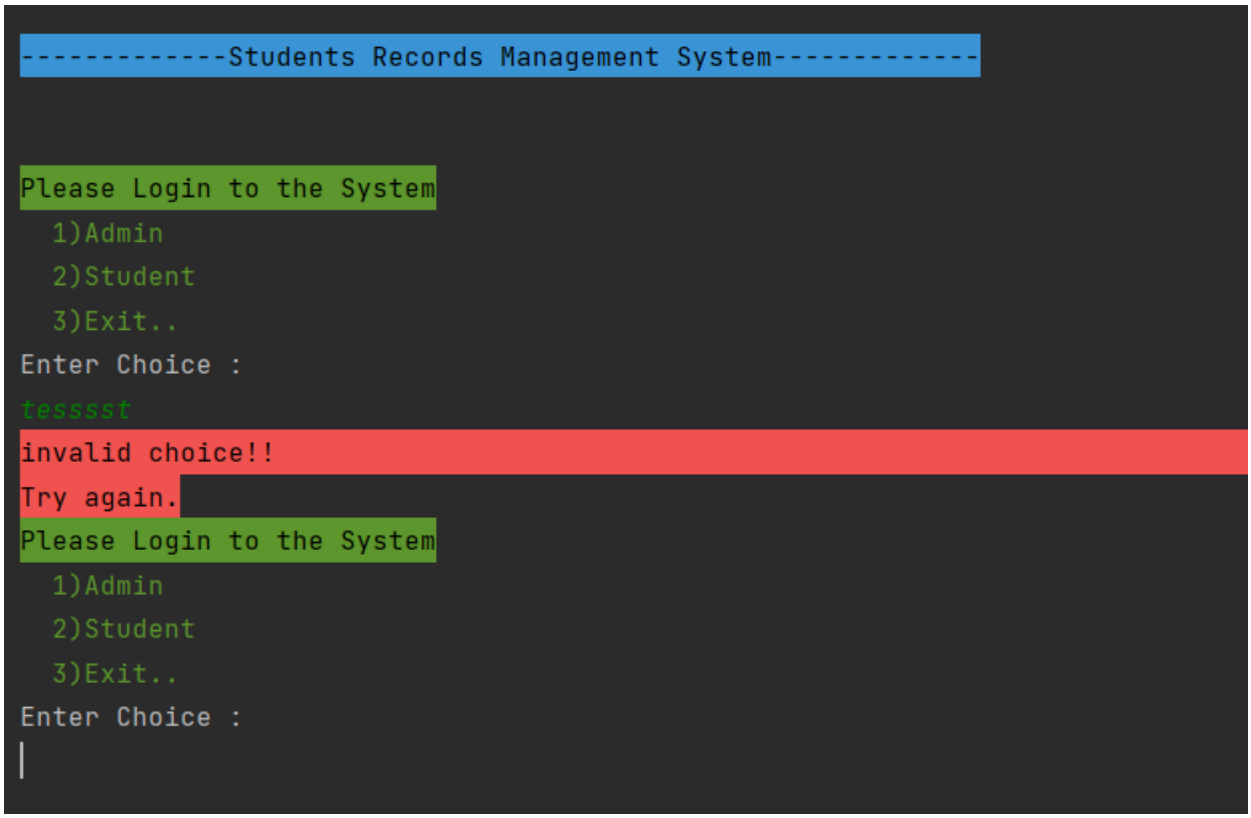

Fig. 2 Inserting Wrong Input

Admin:

If we choose 1, the Admin menu will appear:

```
Please Login to the System
  1)Admin
  2)Student
  3)Exit..
Enter Choice :
1
----------------------Admin Menu ----------------------
1)Add a new record
2)Add new semester with student course and grades
3)Update Student Record
4)Show Student statistics
5)Show Global statistics
6)Searching for ID's Student based on average or taken hours.
7)Back to main menu
8)Exit..
Enter Choice :
```

Fig. 3 Admin menu

Then if we enter an existed number student, an error message will appear:

```
----------------------Admin Menu ----------------------
1)Add a new record
2)Add new semester with student course and grades
3)Update Student Record
4)Show Student statistics
5)Show Global statistics
6)Searching for ID's Student based on average or taken hours.
7)Back to main menu
8)Exit..
Enter Choice :
1
Please enter the new ID to add its own record:
1191000
Error , This ID already exists !!
----------------------Admin Menu ----------------------
```

Fig. 4 Insert Existed ID

Inserting a new valid number student:

```
----------------------Admin Menu ----------------------
1)Add a new record
2)Add new semester with student course and grades
3)Update Student Record
4)Show Student statistics
5)Show Global statistics
6)Searching for ID's Student based on average or taken hours.
7)Back to main menu
8)Exit..
Enter Choice :
1
Please enter the new ID to add its own record:
1207766
do you want to add any semester to this student ?
1)yes
2)no
2
This student has been successfully added
```

Fig. 5 Inserting a new Valid number student

Adding new semester to an existed student:

```
----------------------Admin Menu ----------------------
1)Add a new record
2)Add new semester with student course and grades
3)Update Student Record
4)Show Student statistics
5)Show Global statistics
6)Searching for ID's Student based on average or taken hours.
7)Back to main menu
8)Exit..
Enter Choice :
2
Please enter the new ID to add its own record:
000000
Error , this id not found !!
```

Fig. 6 Inserting wrong ID to add a new Semester

Adding a wrong Year Format:

```
1)Add a new record
2)Add new semester with student course and grades
3)Update Student Record
4)Show Student statistics
5)Show Global statistics
6)Searching for ID's Student based on average or taken hours.
7)Back to main menu
8)Exit..
Enter Choice :
2
Please enter the new ID to add its own record:
1207766
How many semester do want to add ?
1
Please Enter Year (example: 2020-2021) :
20000-2222
Error , wrong year format !!
```

Fig. 7 Wrong Year Format

Inserting a new Semester with valid Courses name:

```
Enter Choice :
2
Please enter the new ID to add its own record:
1207766
How many semester do want to add ?
1
Please Enter Year (example: 2020-2021) :
2021-2022
Please Enter the semester (1 represents first semester 2 for second semester, 3 for summer semester)
1
How many course do want to add ?
2
Please Enter # 1 Course Name :
ENCS3310
Please Enter # 1  Course Grade :
99
Please Enter # 2 Course Name :
ENEE2103
Please Enter # 2  Course Grade :
90
```

Fig. 8 Inserting a new Semester with valid Courses name

Trying to add wrong course name (Or unavailable course):

```
Enter Choice :
2
Please enter the new ID to add its own record:
1207766
How many semester do want to add ?
1
Please Enter Year (example: 2020-2021) :
2020-2021
Please Enter the semester (1 represents first semester 2 for second semester, 3 for summer semester)
1
How many course do want to add ?
1
Please Enter # 1 Course Name :
ARAB133
Error , wrong course name (ENCS or ENEE only) !!
```

Fig. 9 Insert unavailable Course

Updating a course grade for specific student:

```
Enter Choice :
3
Please enter the new ID to change its record:
1207766
Please Enter the Course name to change its grade:
ENCS3310
Please Enter the new grade:
95
>>The record has been successfully modified
```

Fig. 10 Updating a course grade

Shows the statistical student information's:

```
Enter Choice :
4
Please enter the new ID to add its own record:
1191000
1 ) ENCS2340
2 ) ENCS2380
3 ) ENCS3130
4 ) ENCS3310
5 ) ENCS3320
6 ) ENCS3330
7 ) ENCS3340
8 ) ENCS3390
9 ) ENCS4110
10 ) ENCS4130
11 ) ENCS4210
12 ) ENCS4300
13 ) ENCS4310
14 ) ENCS4320
15 ) ENCS4330
16 ) ENCS4370
17 ) ENCS4380
18 ) ENCS5140
19 ) ENCS5150
20 ) ENCS5200
21 ) ENCS5300
22 ) ENEE2103
23 ) ENEE2304
24 ) ENEE2307
25 ) ENEE2312
26 ) ENEE2360
27 ) ENEE3309
28 ) ENEE4113
taken hours : 16
average per semester :
2020-2021/1  :  58.89
2020-2021/2  :  55.00
2020-2021/3  :  55.00
overall average : 57.19
```

Fig. 11 statistical student information's

Shows the Global statistical students information's:

```
Enter Choice :
5
overall students average :  80.52
2020-2021/1  :  10.76
2020-2021/2  :  5.00
2020-2021/3  :  2.00
2021-2022/1  :  10.67
2055-2051/1  :  1.00
2090-2091/2  :  3.00
```

Fig. 12 Global statistical students information's

Plotting a histogram that shows distribution of student's grades:

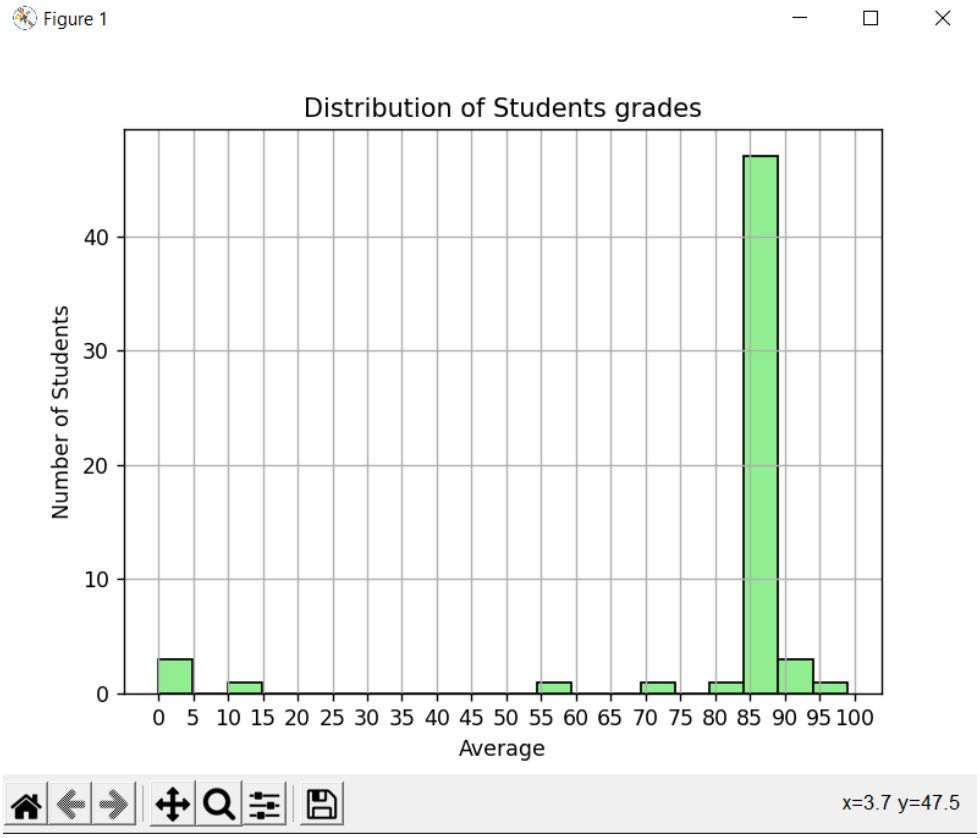

Fig. 13 Plot that shows distribution of student's grades

Make a search based on Average:

```
Enter Choice :
6
Please Enter the Search Method based on:
1)Average
2)Taken hours
1
choose the criteria you want to search based on
1)equal (e.g 70=)
2)Larger Than (e.g 70>)
3)Smaller (e.g 70<)
2
Please enter the value you want to search based on:88
1 ) ID : 1191001  GPA: 99.00
2 ) ID : 1207766  GPA: 93.75
3 ) ID : 1210000  GPA: 90.00
4 ) ID : 1230011  GPA: 90.00
```

Fig. 14 Search based on Average

Make a search based on Taken Hours:

```
Please Enter the Search Method based on:
1)Average
2)Taken hours
2
choose the criteria you want to search based on
1)equal (e.g 70=)
2)Larger Than (e.g 70>)
3)Smaller (e.g 70<)
2
Please enter the value you want to search based on:12
1 ) ID: 1191000  Total hours: 16
2 ) ID: 1191001  Total hours: 38
```

Fig. 15 Search based on taken hours

Student:

If we choose 2 from the system menu, the student menu will appear:

```
------------Students Records Management System------------


Please Login to the System
  1)Admin
  2)Student
  3)Exit..
Enter Choice :
2
Please enter Your ID first:
1191000
----------------------Student Menu ----------------------
1)Show Student statistics
2)Show Global statistics
3)Back to main menu
4)Exit..
Enter Choice :
1
1 ) ENCS2340
2 ) ENCS2380
3 ) ENCS3130
4 ) ENCS3310
5 ) ENCS3320
6 ) ENCS3330
7 ) ENCS3340
8 ) ENCS3390
```

Fig. 16 Student menu

But before the student menu appears, we he/she should insert his/her ID.

So then if he/she choose 1 from the menu ➔ the statistical student information's will appear:

```
14 ) ENCS4320
15 ) ENCS4330
16 ) ENCS4370
17 ) ENCS4380
18 ) ENCS5140
19 ) ENCS5150
20 ) ENCS5200
21 ) ENCS5300
22 ) ENEE2103
23 ) ENEE2304
24 ) ENEE2307
25 ) ENEE2312
26 ) ENEE2360
27 ) ENEE3309
28 ) ENEE4113
taken hours : 16
average per semester :
2020-2021/1  :  58.89
2020-2021/2  :  55.00
2020-2021/3  :  55.00
overall average : 57.19
```

Fig. 17 statistical student information's

And if he/she choose 2 ➔ the global statistical student's information's will appear:

```
overall students average :  80.52
2020-2021/1  :  10.76
2020-2021/2  :  5.00
2020-2021/3  :  2.00
2021-2022/1  :  10.67
2055-2051/1  :  1.00
2090-2091/2  :  3.00
```

Fig. 18 global statistical student's information's

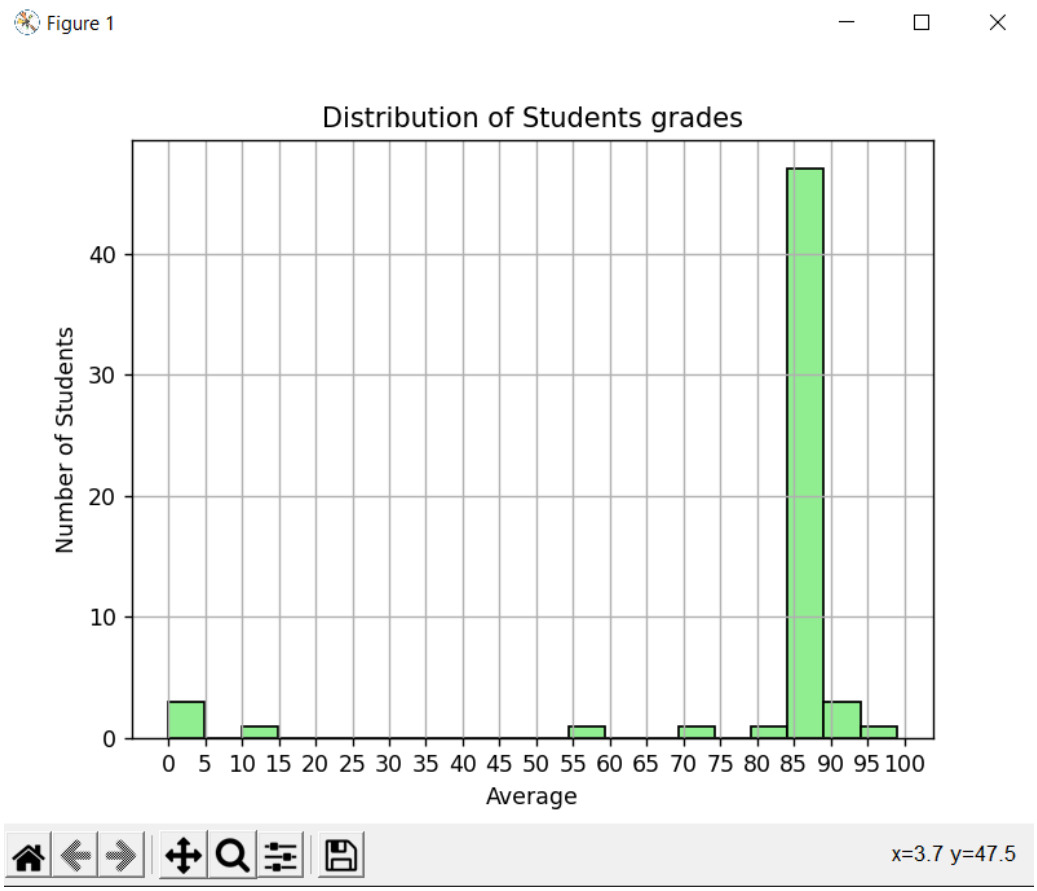

Fig. 19 The plot of global statistical student's information's

Descriptions of the used data structures:

In Main class:

We used a list to save all courses of ENEE and ENCS:

```
coursesList = []
```

Why?

Because we want a data structure such as an array, to save all the available courses in the computer engineering major.

We used a Dictionary to save the Students Objects:

```
studentsFiles = {}
```

Why?

Because the dictionary is a good data structure that consist of keys and values, so that each key point to one object which is in our case the student object. So that we can back to this object by just put the key (ID number).

We used a Set to save all Semesters:

```
semesters = set()
```

Why?

Because we don't want to repeat the same semester many times; and the set doesn't contains repeated elements, so using it to save semesters is the best choice.

In student class:

We used a dictionary to save the semesters objects for each student:

```
self.semesters = {}
```

Why?

Because the dictionary is a good data structure that consist of keys and values, so that each key point to one object which is in our case the Semester object. So that we can back to this object by just put the key (Semester String).

We used another dictionary to save all courses that the student takes:

```
self.courses = {}
```

Why?

Because we want data structure that consist of a key (course name) that points to a value (grade). So that we can back to this value by just put the key (Course name String). And if we want to update a course grade, we just call the courses [course name String].

In Semester class:

We used dictionary to save all courses in the semester object that the students takes in that semester:

```
self.courses = {}
```

Why?

Because we want a data structure that consist of a key (course name) that points to value (grade). So any update on this course, we can just call the courses [course name String]

Using OOP in our Project:

In our project we used The OOP concepts; so we make 2 classes which are student class, and semester class.

In student class; we add all the related attributes to the student, for example the courses he takes, the taken hours, the registered Semesters. In this class we make a function that calculates the Average for a specific semester, and another function that calculates the cumulative Average for the student.

Note: that in student class we makes objects of semester class.

In Semester class; we add all the related attributes to the Semester, for example the courses registered in this semester, and the taken hours in this semester, also in this class we make a function that calculates the Semester Average for a specific student.

Appendix:

Attached:

- Full Code (three seprated .py files: main, Student, Semester)
- Students File
- Courses File
- ShellScript To Generate Student Files