INFO 5100 Application Engineering and Development

Assignment 3

University Model Implementation

Team No 19

Created by:

1. Rahul Tiwari (001568172)
2. Anurag Shrivastava (001568861)
3. Shreya Kichloo (001306118)

Content

Problem Statement

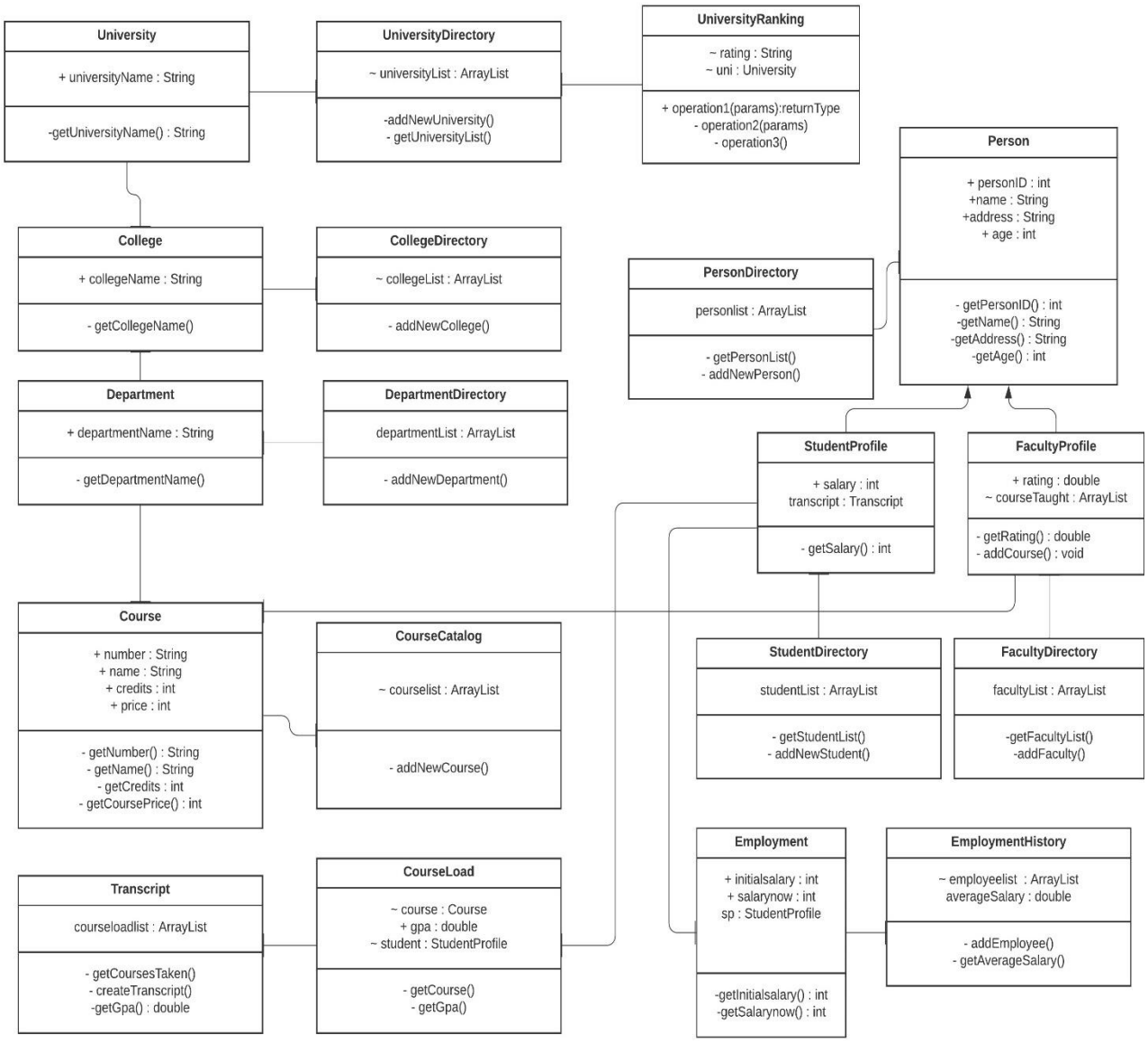
The objective of this assignment is to instill in us the techniques for turning a design into a digital system for data collection, insight, as well as social impact. We want to use software engineering design and programming techniques to improve the quality of education university setting anywhere in the world and hold people accountable for improving the quality of life for people through education, learning to learn, and feedback.

The task is to study ways to create a performance measurement solution to enable universities to measure the quality of the education they deliver to their students. Quality here means keeping courses fresh and aligned with industry trends. The approach will be to look into how an educational system involving faculty, courses, and employers contribute to the professional growth of their graduates over a 5-year period. You must figure out ways to track the jobs and promotions graduates get over time and assign rankings accordingly. In addition, track the connection of courses and their relevance to graduates growth.

Deliverables

1. Running java code showing the implementation of a complete system at the department and college levels.
2. Presentation outlining the solution and implementation.
3. Sequence diagrams showing how to navigate the university object model to deliver performance metrics needed for performance and feedback.
4. An object model showing the changes to the university model to support the new capabilities.
5. This must include the additional methods and attributes required to deliver the results.
6. Investigate how we create digital education systems assuming a university is an intermediary (broker) between students and employers. Their brand is about credibility. How will the implementation code change based on this assumption?
7. The application must enable the creation and update functions for any of the attributes of concern.

UML Diagram


University
+ universityName : String
-getUniversityName() : String
UniversityDirectory
~ universityList : ArrayList
-addNewUniversity()
- getUniversityList()
UniversityRanking
~ rating : String
~ uni : University
+ operation1(params):returnType
- operation2(params)
- operation3()
Person
+ personID : int
+name : String
+address : String
+ age : int
- getPersonID() : int
-getName() : String
-getAddress() : String
-getAge() : int
College
+ collegeName : String
- getCollegeName()
CollegeDirectory
~ collegeList : ArrayList
- addNewCollege()
PersonDirectory
personlist : ArrayList
- getPersonList()
- addNewPerson()
Department
+ departmentName : String
- getDepartmentName()
DepartmentDirectory
departmentList : ArrayList
- addNewDepartment()
StudentProfile
+ salary : int
transcript : Transcript
- getSalary() : int
FacultyProfile
+ rating : double
~ courseTaught : ArrayList
- getRating() : double
- addCourse() : void
Course
+ number : String
+ name : String
+ credits : int
+ price : int
- getNumber() : String
- getName() : String
- getCredits : int
- getCoursePrice() : int
CourseCatalog
~ courselist : ArrayList
- addNewCourse()
StudentDirectory
studentList : ArrayList
- getStudentList()
- addNewStudent()
FacultyDirectory
facultyList : ArrayList
-getFacultyList()
-addFaculty()
Transcript
courseloadlist : ArrayList
- getCoursesTaken()
- createTranscript()
-getGpa() : double
CourseLoad
~ course : Course
+ gpa : double
~ student : StudentProfile
- getCourse()
- getGpa()
Employment
+ initialsalary : int
+ salarynow : int
sp : StudentProfile
-getInitialsalary() : int
-getSalarynow() : int
EmploymentHistory
~ employeelist : ArrayList
averageSalary : double
- addEmployee()
- getAverageSalary()

Classes Involved

1. University

Attribute	Data Type	Description
universityName	String	Name of the University.

Methods: getUniversityName().

2. UniversityDirectory

Attribute	Data Type	Description
universityList	ArrayList	List of University names.

Methods: addNewUniversity(), getUniversityList().

3.College

Attribute	Data Type	Description
collegeName	String	Name of the College in the University.

Methods: getcollegeName()

4. CollegeDirectory

Attribute	Data Type	Description
collegeList	ArrayList	List of all College in the University

Methods: addnewCollege()

5.Department

Attribute	Data Type	Description
departmentName	String	Name of the department

Method: getDepartmentName();

6 .DepartmentDirectory

Attribute	Data Type	Description
departmentlist	ArrayList	List of department

Method: addNewDepartment();

7. Course

Attribute	Data Type	Description
Number	String	Contains course number
name	String	Contains course name
credits	int	Contains course credits
price	int	Contains price for each course per credit.

Methods: getNumber(),getName(),getCredits(),getPrice(),getCoursePrice().

8. CourseCatalog

Attribute	Data Type	Description
courselist	ArrayList	Contains list of courses.

Methods: addNewCourse().

9. CourseLoad

Attribute	Data Type	Description
course	Course	Contains reference to the object of the course class
gpa	double	Contains GPA of the student
student	StudentProfile	Contains reference to the object of the StudentProfile class.

Methods: getCourse(), getgpa().

10. Transcript

Attribute	Data Type	Description
courseloadlist()	ArrayList	Contains objects of course load class.

Methods: getCoursesTaken(),createTranscript(),getgpa().

11. Person

Attribute	Data Type	Description
personID	int	Contains person id
name	String	Contains name of person
address	String	Contains address of person
age	int	Contains age of person

12. PersonDirectory

Attribute	Data Type	Description
personlist	ArrayList	Contains list of person

Methods: addNewPerson().

13. StudentProfile

StudentProfile extends Person class

Attribute	Data Type	Description
transcripts	Transcript	Contains reference to object of transcript class

14. StudentDirectory

Attribute	Data Type	Description
studentList	ArrayList	Contains list of students

Method: addNewStudent().

15. FacultyProfile

FacultyProfile extends Person

Attribute	Data Type	Description
rating	double	Contains faculty rating
courseTaught	ArrayList	Contains list of courses under the faculty

Method: getRating(), addCourse().

16. FacultyDirectory

Attribute	Data Type	Description

facultyList	ArrayList	Contains list of faculty

Method: getFacultyList(), addFaculty().

17.Employment

Attribute	Data Type	Description
initialsalary	int	initial salary of students
salarynow	int	Salary after 5 years
sp	StudentProfile	Reference to StudentProfile object

Method: getIntialsalary(), getSalarynow().

18.EmploymentHistory

Attribute	Data Type	Description
employeelist	ArrayList	List of employees
averageSalary	double	Average salary of all employees

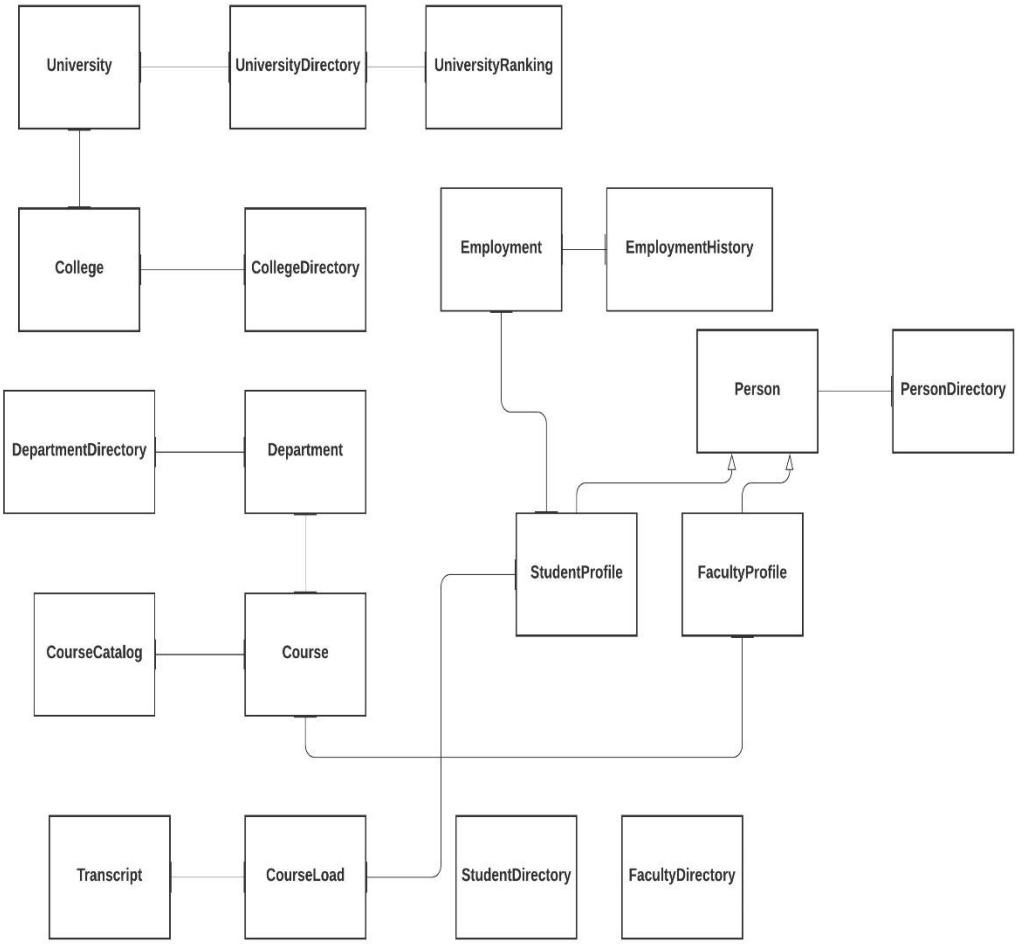
Method: addEmployee(), getAverageSalary().

19.UniversityRanking

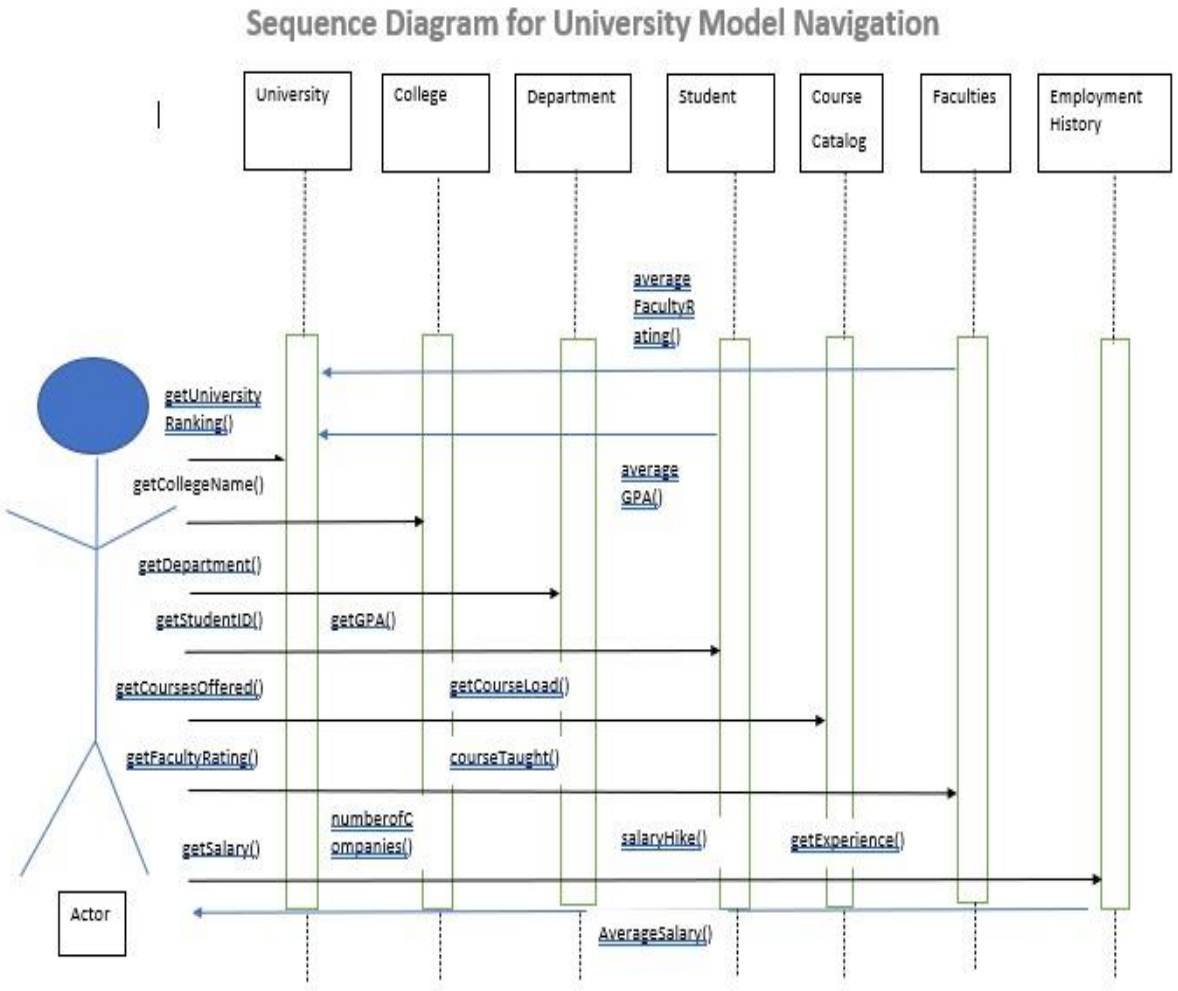
Attribute	Data Type	Description
rating	String	University rating
uni	University	Reference object to university

Analysis

Object Model



Sequence Diagram



Proposed Solution for the University Ranking Model:

University Ranking is a criteria setup to define the overall university standings among a list of various universities determining various factors such as quality of education, overall strength of students, faculty ratings and overall GPA of students.

Coding Implementation of Ranking Model:

We have created a getUniversityRanking() method, which would take into consideration below mentioned methods:

- getAverageGPA() – This method would return the average GPA of all the students in the University.
- averageFacultyRating() – This method would return the average Faculty ratings currently teaching in the University. This would determine the quality offered is top notch.
- averageSalary() – This method returns the average Salaries of students hired in various companies after graduating from the University. Establishing a smooth character buildup and transition of students into the corporate world.
- totalStudents() – This method would return the total number of students currently enrolled within the University curriculum, determining the overall popularity of the University.

$$\text{getUniversityRanking()} = \frac{\text{getAverageGPA()} + \text{averageFacultyRating()} + \text{averageSalary()} + \text{totalStudents()}}{1000}$$

Logic Behind above mentioned methods:

```
public String getUniversityRating(double a, double b, double c, int size){

    double rating =0;
    rating = (a+b+c)/3;
    rating= rating/10000;

    if(0<size && size<=100){
       rating+=1;
    }
    else if(100<size & size<=200){
       rating+=2;
    }
    else if(200<size & size<=500){
       rating+=3;
    }
    else if(500<size & size<=1000){
       rating+=4;
    }
    else
```

```
        rating+=5;

      if (rating>10){
        rating=10;
      }

      finalRating = rating;
      String s = String.format("%.2f", rating);
      return s;
   }
```

getAverageSalary() Implementation:

```
 public double getAverageSalary(){

      double totalSalary=0;

      for(Employement emp : employeelist){

        totalSalary+=(emp.getInitialsalary() + emp.getSalarynow())/2;
      }
      return totalSalary/employeelist.size();
   }
```

getAverageRating() Implementation:

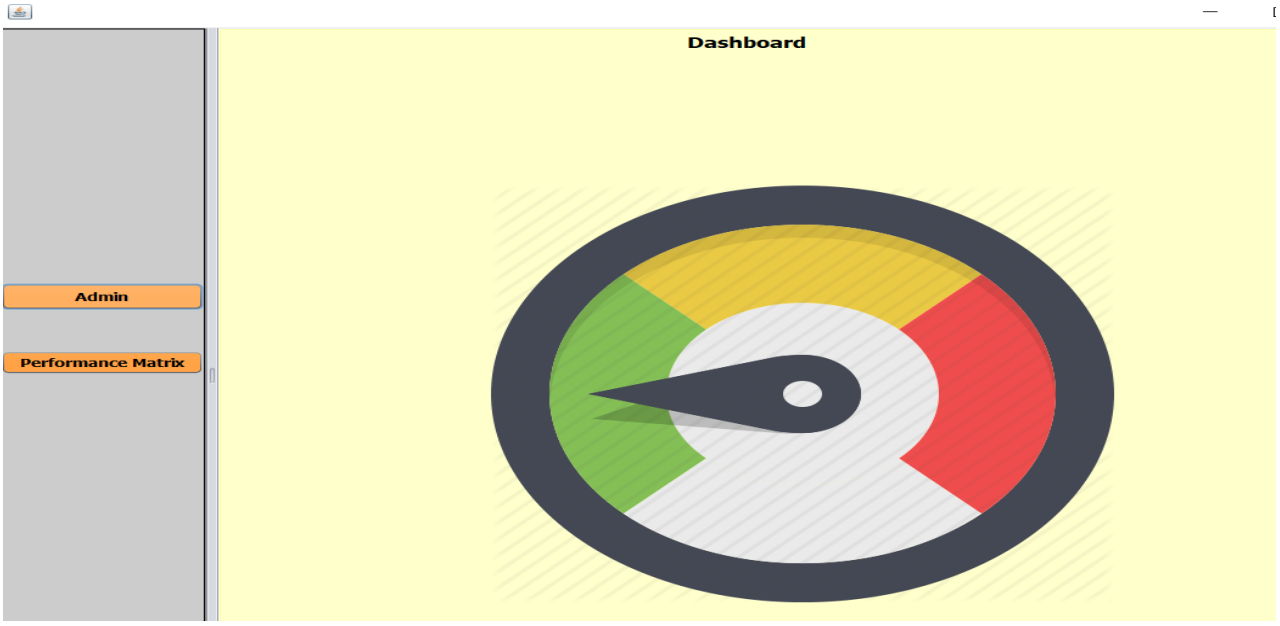
```
public double getAverageRating(){
      double avgRating=0;
      for(FacultyProfile fp : facultyList){
        avgRating+= fp.getRating();
      }
      int totalFaculties = facultyList.size();
      return avgRating/totalFaculties;
   }
```

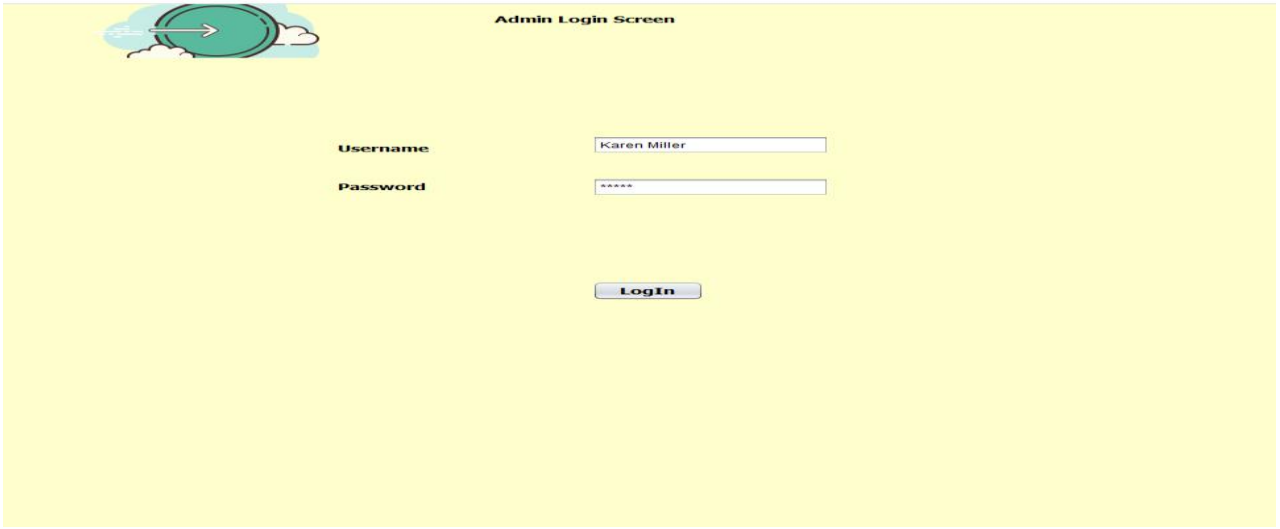
getAverageGPA() Implementation:

```
public double getAverageGPA(){
      List<CourseLoad> courseGPAList = getCoursesTaken();
      int courses = courseGPAList.size();
      double totalGPA = 0;
      for(CourseLoad c : courseGPAList){
        totalGPA+= c.getGpa();
      }

      return totalGPA/courses;   }
```

Dashboard


Dashboard
Admin
Performance Matrix


Admin Login Screen
Username
Karen Miller
Password
*****
LogIn

Welcome Admin Screen

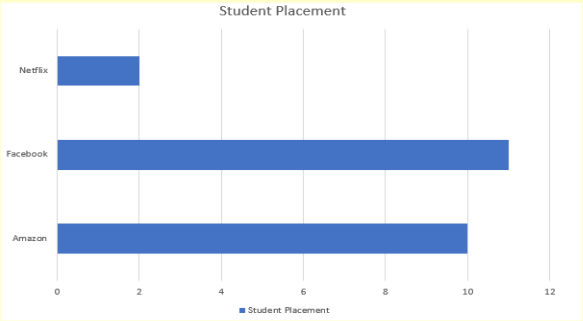
University Ranking 20

List of Colleges and Department

College	Department
College of Engineering	Information Systems
Information Systems	CSYE
Khoury	Computer Science
NUSL	Evidence Law

Courses	Industry Domain	Companies	Total Students Placed
Application Engineering D...	Java	Amazon	10
Data Science	Python	Facebook	4
Data Warehousing	BigData	Netflix	2
MachineLearning	AI	Facebook	7

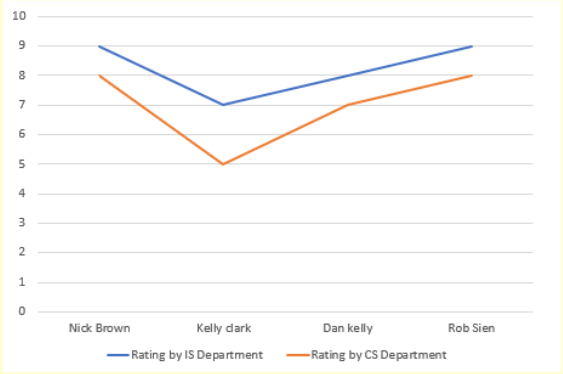
<< Student



Faculty Details

Faculty ID	Name	Courses	Department	Rating
NU101	Nick Brown	AI	Information Syste...	9
NU003	Kelly clark	Energy Systems	Science	7
NU130	Dan kelly	DDMD	Khoury	8
NU111	Rob Sien	AED	Information Syste...	5

<<



Student Details

Student ID 00310744

Name Mark Sloan

College llege of Engineering

Final GPA A-

Start Term 9/1/2012

Graduation Year 9/1/2014

Courses	Grade	Faculty
AED	A+	Karl Bugara
OOD	B-	Peter Daniels
Data Science	A-	Nick Brown
AI	B	Nick Brown
DDMD	A-	Jessica Fisher

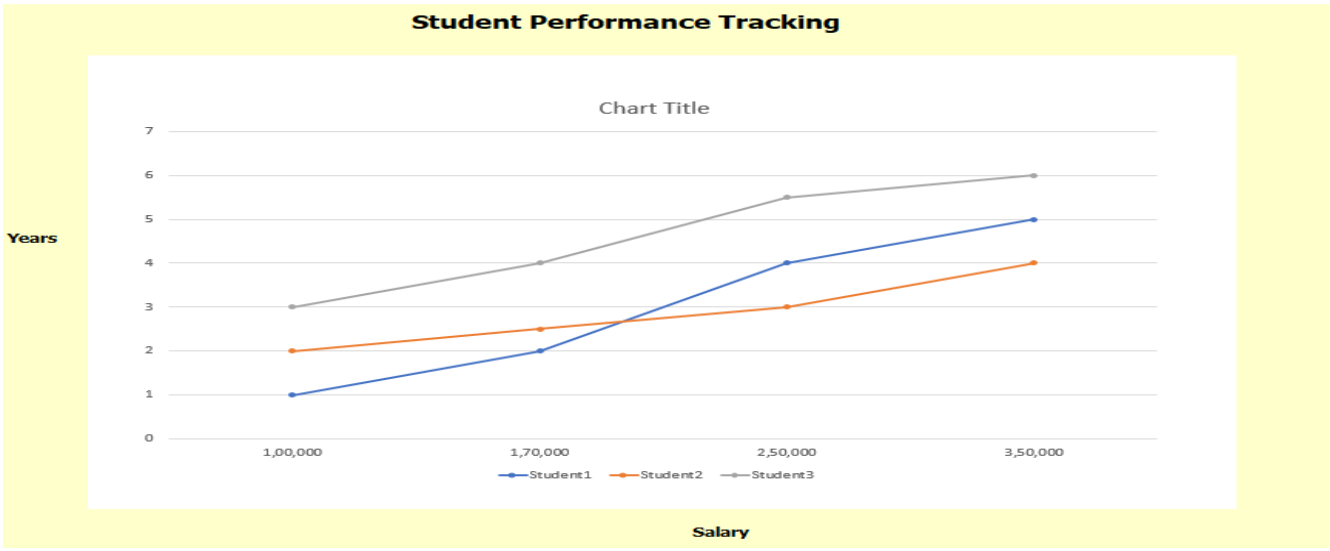
Company	Designation	Salary	Start Date	End Date
Amazon	Software De...	100,000	20/12/2014	1/1/2018
Facebook	Senior Devel...	250,900	2/2/2018	2/2/2019
Netflix	Team Lead	390,560	2/4/2019'	2/4/2020
Amazon	CTO	400,000	2/12/2020	

Faculty Details

Welcome to Student Screen

Student ID 00310744

Search


Student Performance Tracking
Chart Title
Years
7
6
5
4
3
2
1
0
1,00,000
1,70,000
2,50,000
3,50,000
Student1
Student2
Student3
Salary