UNIVERSITY MODEL

Design Project

SHWETA WAKALE
RAMYA KANGURI
TANAYA DESHMUKH

Table of Contents

Problem Statement

- To improve the quality of educational institutes by providing their students with optimum facilities and make them industry ready.
- To create a performance measurement solution to enable universities to measure the quality of the education they deliver to their students.

Deliverables

- Ways to create a performance measurement solution to enable universities to measure the quality of the education they deliver to their students.
- Report outlining your proposed solution.
- Sequence diagrams showing how to navigate the university object model to deliver performance metrics needed for performance and feedback.
- A class diagram showing the changes to the university model to support the new capabilities. This diagram must include the additional methods and attributes required to deliver the results.
- Designing a dashboard that enables college and university administrators to compare the performance of their academic units.
- Ways to define your own ranking system for students to decide where they want to go for their studies.
- Ways to track the jobs and promotions graduates get over time and assign rankings accordingly and track the connection of courses and their relevance to graduates' growth.

Proposed Solution

- The majority of factors that contribute towards the quality of education at an institution and ranking are:
    - Courses
    - Feedback
    - Faculty - Student Ratio
    - GPA
    - Placement
    - Promotions
- Placement or Employment is one of the key factors for deciding the quality of education. This is achieved by the graduation rate, better the graduation rate, better the placement/employment rate of the university.
- The Semester wise and cumulative grade points average is a deciding factor for which all employers can a student apply to.
- A faculty student ratio tells how many students are taught by one faculty member. Higher is the faculty: student ratio, better is the educational institute as lesser the number of students per faculty, more time can be devoted per student individually by that particular faculty member.
- The employee rating can also be considered as a contributing factor. The grades are directly proportional to the employee rating, more the grade point scored by the student at the college, more is their employee rating at the company.
- The number of students in each college placed in the top 50 companies can also add weightage towards ranking.
- The connection of the courses and their relevance to graduates' growth can be determined by relating the industry requirements to the course and predicting the growth and placement of the student.

Actors

1. Faculty
2. Student
3. Admin
4. University

Entities

1. Student

    - Students can register for courses.
    - Student is responsible for giving feedback for the professor for the course that he has registered for. As well as for the department.
    - Attributes: Student Id, Name, Email Id, Address, Major, Job Status, Student type.
    - Methods: newStudentDetail() returns the students information like name,Id and courses.
      findStudent() returns student information using studentId.
      registerACourse() will enable a student to register for a course.
      updateInfo() will update the necessary student info.
      calculateGPA() calculates the GPA for the course based on the Grading Criteria

2. Faculty
    - A professor has can teach multiple courses.
    - A professor is responsible for giving feedback for the department.
    - Attributes: Name, Email, Professor ID, Qualification, Experience, Research.
    - Methods: getProfInfo() returns the professor information.
      giveFeedback() submits the department feedback by the professor.

3. Admin

    - Administrator can view and compare the performance information of every University and its Departments.
    - Attributes: Username
    - Methods: getStudentPerformance() returns the student performance as a group or an individual.
      getProfessorPerformance() returns the ratings and feedback of a selected professor.
      comparePerformance() returns a comparison between the selected department professors and campuses.

4. Performance
   The performance entity compares the different departments and professors based on the feedbacks received.
   Attributes: Rank
   Methods:
   calculateProfessorsPerformance() calculates the professor ratings based on the given feedbacks.
   calculateStudentPerformance() calculates the department ratings based on the given feedbacks.
   calculateStudentPlacementPerformance() calculates the student placement ratings based on the feedbacks of student.

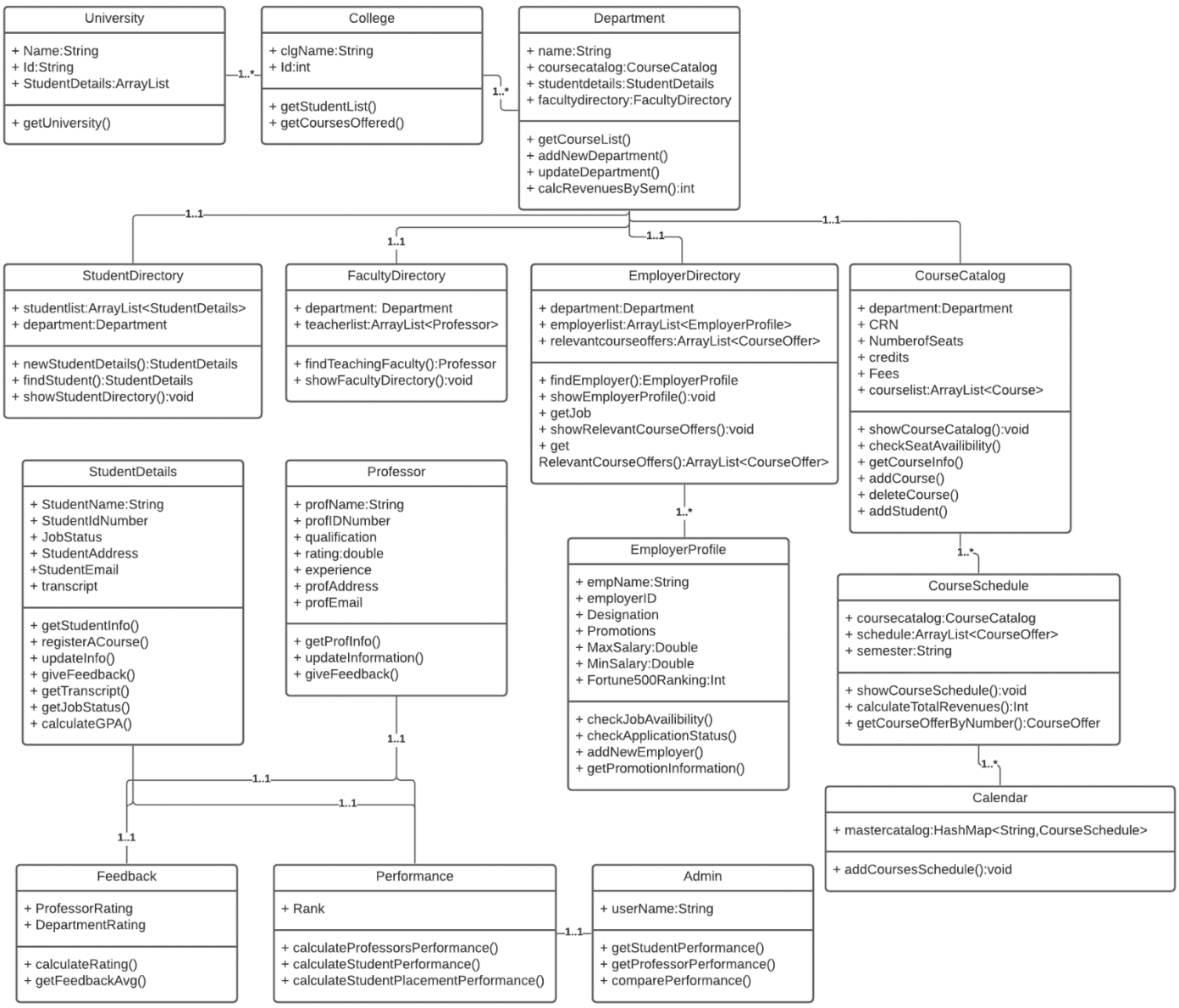
5. Course
   The course entity contains the information necessary to identity the course like unique CRN.
   Attributes: CRN, Number of Seats, Credit Hours, Department, Fees
   Methods: checkSeatAvailability() returns the availability about a course to a student.
   getCourseInfo() returns the professor teaching the course, available seats, credit hours, department and fees.
   addCourse() helps to add new course.

6. Department
   Department includes the various courses available in the course catalog.
   Attributes: Name, Course Catalog
   Methods: getCourseList() returns the available courses under the department.
   calcRevenuesBySem() calculates the revenue of courses selected in that semester.

7. Feedback
   This entity has professors rating and departmental rating and departmental rating as its attributes. These attributes will be used later to calculate the total rating for the professor and the department to evaluate the performance of the respective entities.
   Functions: getFeedbackAvge() returns the average of the feedbacks for the entity.
   calculateRating() returns the total rating for a particular professor or department.

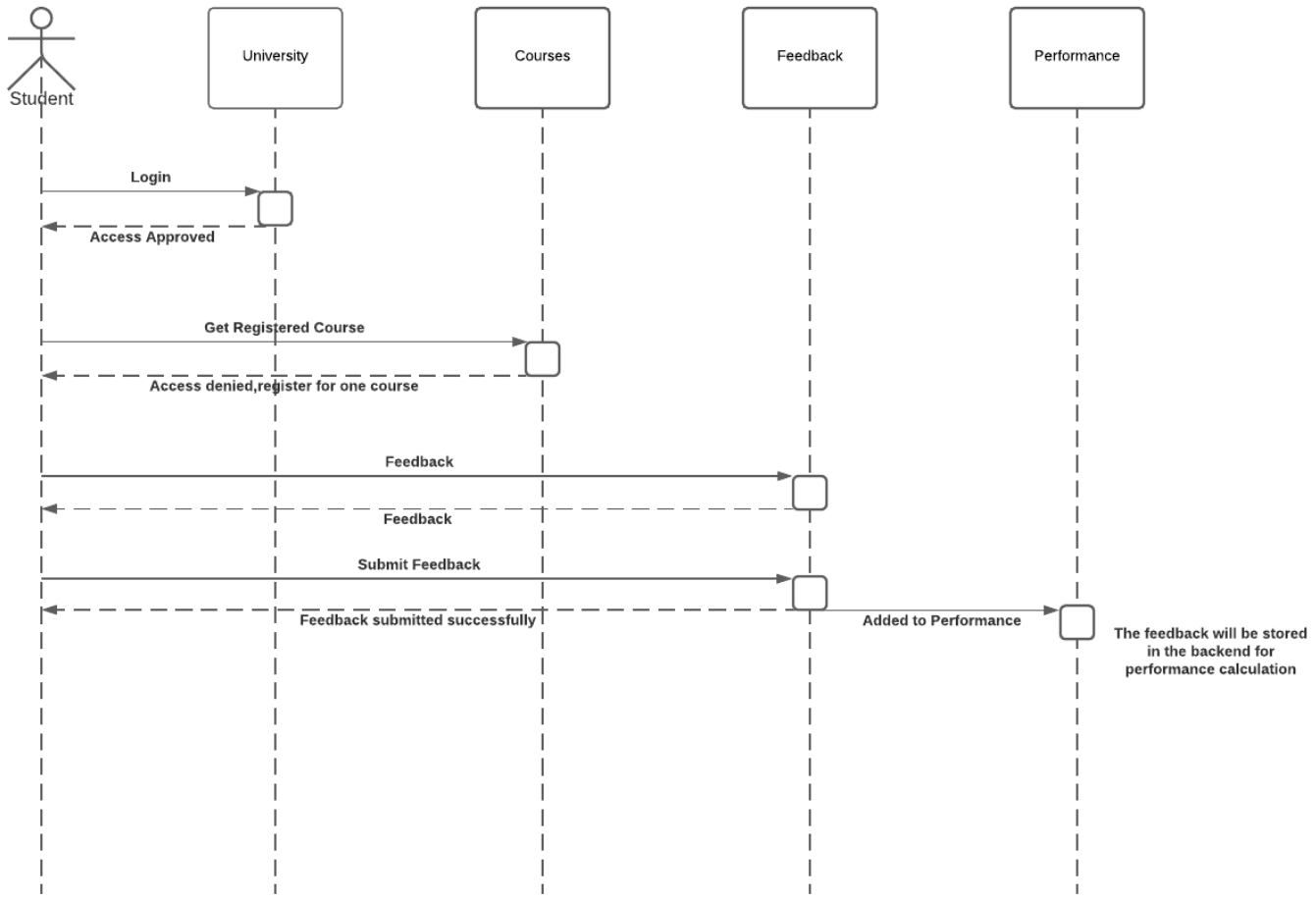
8. Employee
   This entity has Job Description, Job Type, Promotion, Designation and Salary as its attributes.
   Methods: checkJobAvailability() returns the available jobs
   checkApplicationStatus() returns the status for the jobs a student has applied.
   getPromotionInformation() returns the information related to the promotion of an employee who is a former student of the university.

Class Diagram

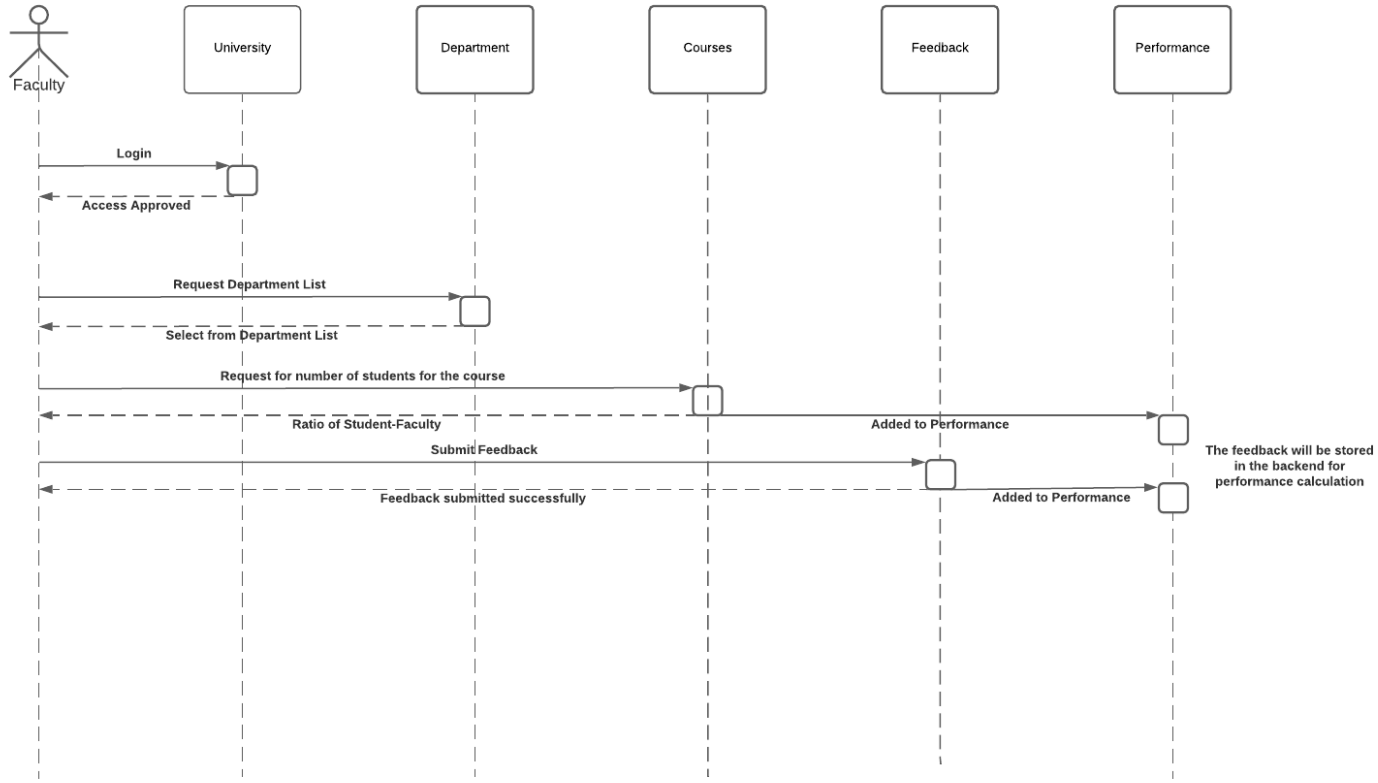


Sequence Diagram

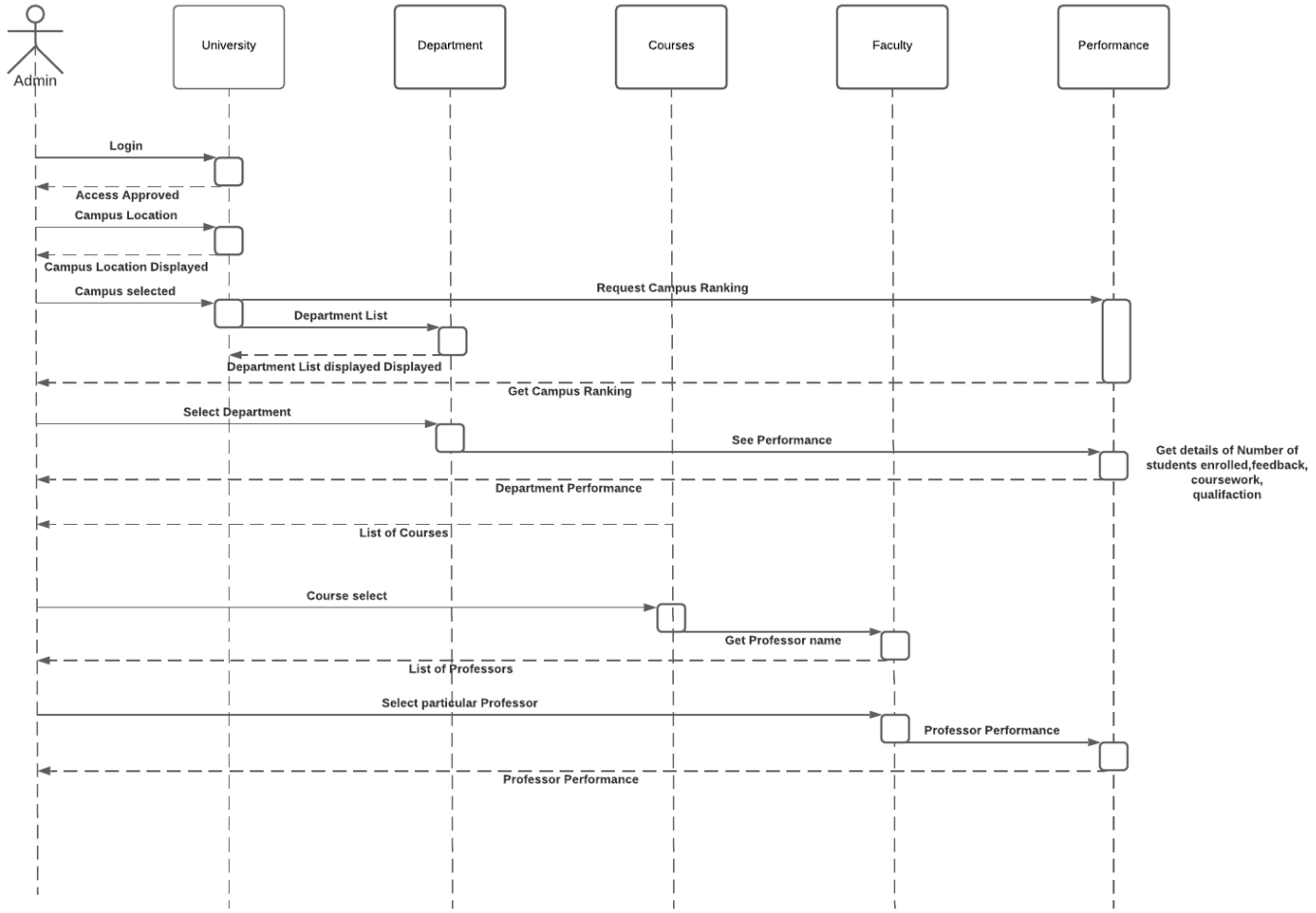
STUDENT:



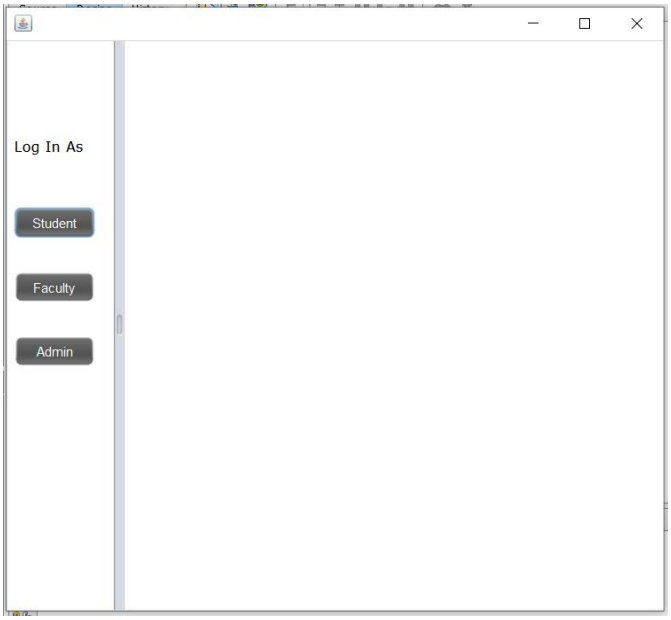
FACULTY :

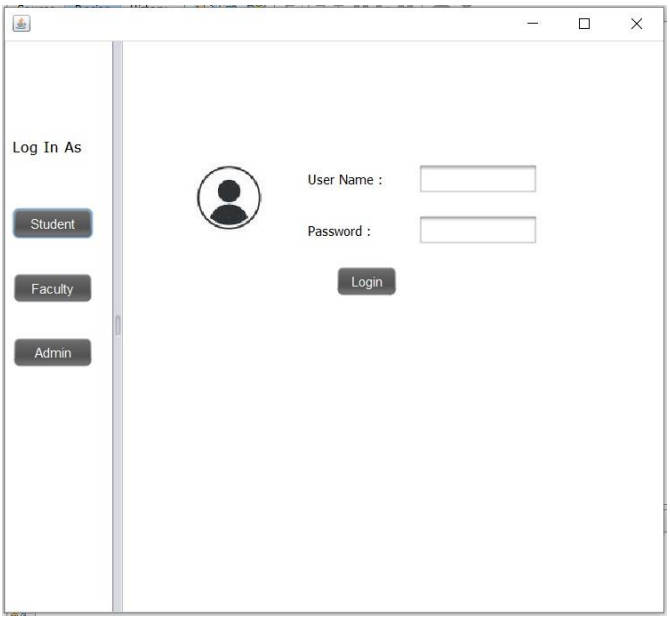

Faculty
University
Department
Courses
Feedback
Performance
Login
Access Approved
Request Department List
Select from Department List
Request for number of students for the course
Ratio of Student-Faculty
Added to Performance
The feedback will be stored in the backend for performance calculation
Submit Feedback
Feedback submitted successfully
Added to Performance

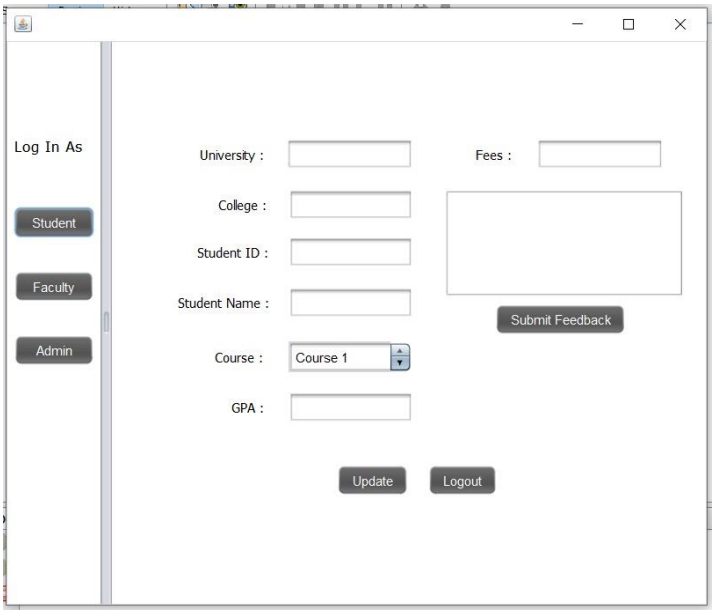
ADMIN:

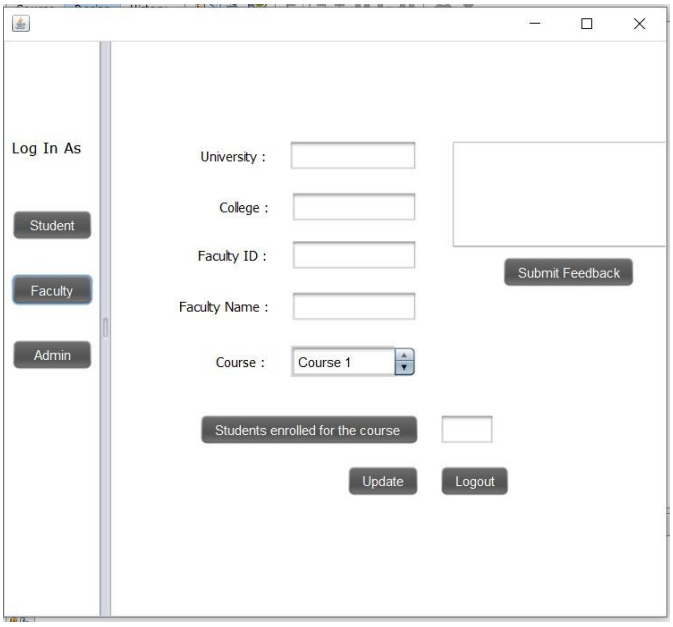

Admin
University
Department
Courses
Faculty
Performance
Login
Access Approved
Campus Location
Campus Location Displayed
Campus selected
Request Campus Ranking
Department List
Department List displayed Displayed
Get Campus Ranking
Select Department
See Performance
Get details of Number of students enrolled,feedback, coursework, qualifaction
Department Performance
List of Courses
Course select
Get Professor name
List of Professors
Select particular Professor
Professor Performance
Professor Performance

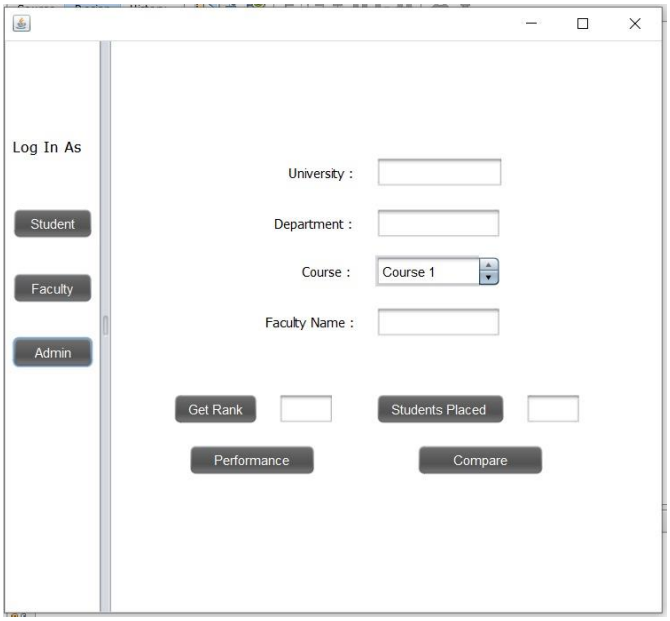
User Interface










Log In As
Student
Faculty
Admin
University :
Department :
Course :
Course 1
Faculty Name :
Get Rank
Students Placed
Performance
Compare