AED DESIGNING ASSIGNMENT

NEERAJ PREMANI | 001306812 HEMANT JAIN | 001305974 JAYSHIL JAIN | 001300508

Deliverables

1. Report outlining your proposed solution.

A brief outlining of our proposed solution:

Student enrolls into a particular course which according to our Object model we can determine which faculty is teaching that particular course.
Faculty assesses the student w.r.t to Assignments, Lab work, project and gives feedback as well as assigns grades for the same

Based on the course that a particular student takes we can track which job a particular student lands upon and track the growth over 5 year period.

In order to assign grades, we have created our own grading scheme which combines both faculty grading and job performance review

Faculty Grading Scheme:

< 2	F
2.1 - 3	B-
3 - 3.2	B
3.2 -3.5	B+
> 3.5	A

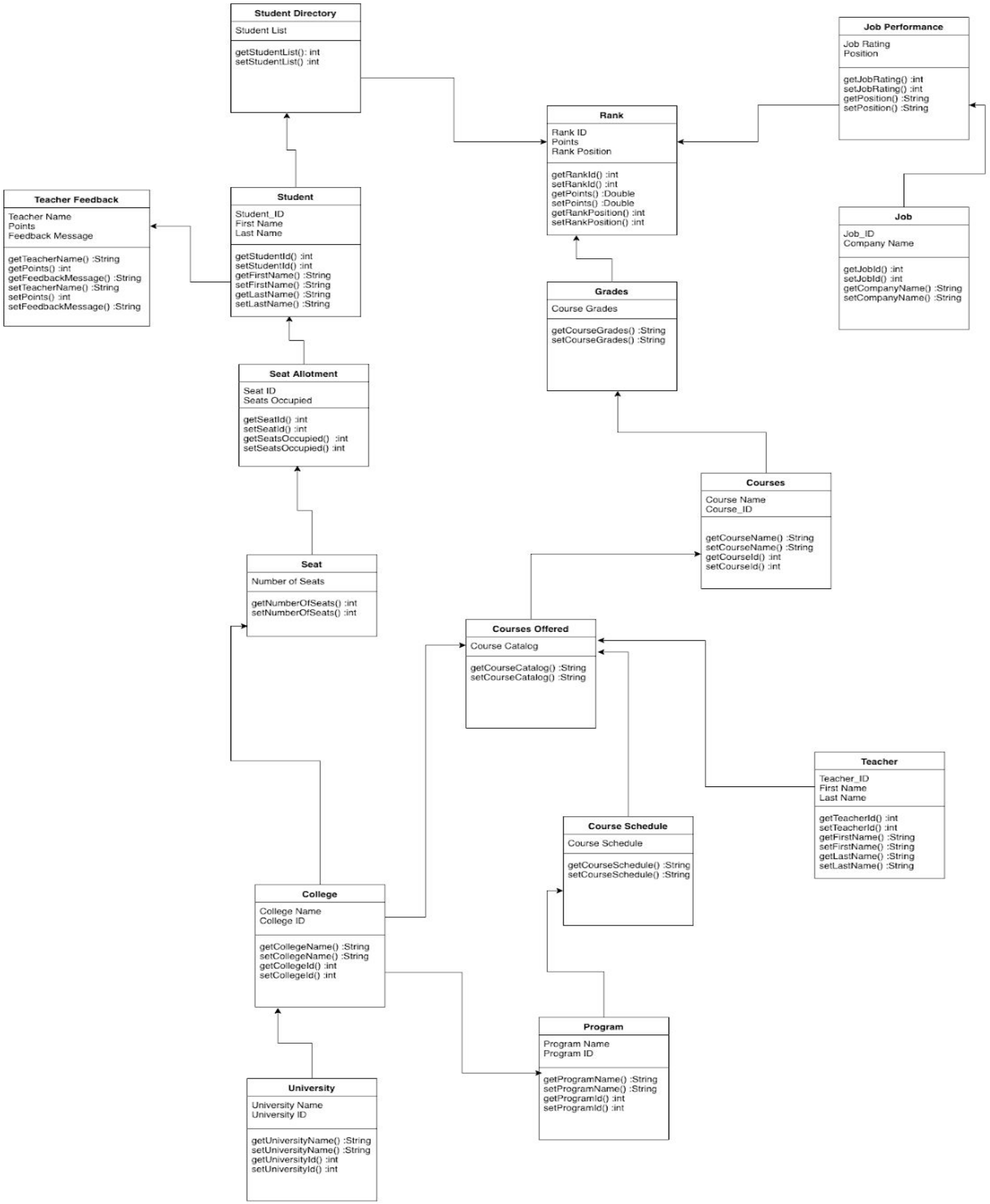
Job Performance review:
Getting a job relevant to courses taken: 5 Points
Increment 1 Point for Promotions over 5 years.

Our Grading System:

If getting a relevant job with a GPA of greater than 3 :	10
If getting a relevant job with a GPA of greater than 2 :	8
If not getting a relevant job with a GPA of greater than 2:	6
If not getting a relevant job with a GPA of less than 2:	4

Overall Ranking System (Defined)
Overall Rank= College Grading system + Job Performance

University Object Model Diagram



Student Directory:
An official document owned by the University.
It contains all the student records enlisting the student details, grades, courses,fees payment dues ,course completed etc.
Methods:
setStudentList()
getStudentList()

Student
Person enrolled in the university program offered by a college.
Has a complete track record about the course undertaken for the Degree.
Has a Cumulative Grade.
Methods:
getStudentId() :int
setStudentId() :int
getFirstName() :String
setFirstName() :String
getLastName() :String
setLastName() :String

Seat Allotment:
Depicts that the a specific seat in a course is assigned to a specific student enrolled in the program.
Methods:
getSeatId() :int
setSeatId() :int
getSeatsOccupied() :int
setSeatsOccupied() :int

Seat:
It represents an available as well as the empty seats in a class.
Methods:
getNumberOfSeats() :int
setNumberOfSeats() :int

Program
It is a part of college that will be segregating students based on their specialization.
Program will have specific courses.
Methods:
getProgramName() :String
setProgramName() :String
getProgramId() :int
setProgramId() :int

Teacher
Person who is part of the University responsible for conducting classes and evaluating students.
Has particular subjects assigned.
Methods:

getTeacherId() :int
setTeacherId() :int
getFirstName() :String
setFirstName() :String
getLastName() :String
setLastName() :String

Job
Is defined as a body that employees students.
The student can enroll into only one job at a time.
Methods:
getJobId() :int
setJobId() :int
getCompanyName() :String
setCompanyName() :String

Job Performance
Defined as rating given by the employer to the student based on his/her performance at the job.
Methods:
getJobRating() :int
setJobRating() :int
getPosition() :String
setPosition() :String

College:
It responsible for the different departments and the students enrolled in the college.
Manage all the details and updates of the faculty as well as students.
Methods:
getCollegeName() :String
setCollegeName() :String
getCollegeId() :int
setCollegeId() :int

University:
It is responsible for the academic details about the various colleges and maintain a list about the programs and the courses offered by the specific colleges.
Also responsible for business matters such as course financing,marketing the colleges,etc
Managing the resources for the colleges.
Maintaining a track about the revenue generation from each college for each year

Methods:
getUniversityName() :String
setUniversityName() :String
getUniversityId() :int
setUniversityId() :int

Course Schedule:
Preparing the course schedules for the courses offered by the colleges.
Managing the courses timing avoiding the time clashes between the two courses offered for a program.
Methods:
getCourseSchedule() :String
setCourseSchedule() :String

Courses Offered:
Manage a list of the courses offered by specific department under a college for the offered programs.
Handle the interchange course offered list with the other departments under the same college.
Methods:
getCourseCatalog() :String
setCourseCatalog() :String

Rank:
Cumulating collection of the students list,grades from the enrolled courses and the job performance.
Manage an overall point score for each students at the University level and calculate and assign a rank to the students.
Manage a Student Ranking List for the performance measurement and the effective growth of the colleges for the various programs for cumulative years.
Methods:
getRankId() :int
setRankId() :int
getPoints() :Double
setPoints() :Double
getRankPosition() :int
setRankPosition() :int
calculatePointsByStudentId(int studentId) :Double
This method calculates the overall points for the student based on the Job performance and the Student Grades from the Program enrolled.

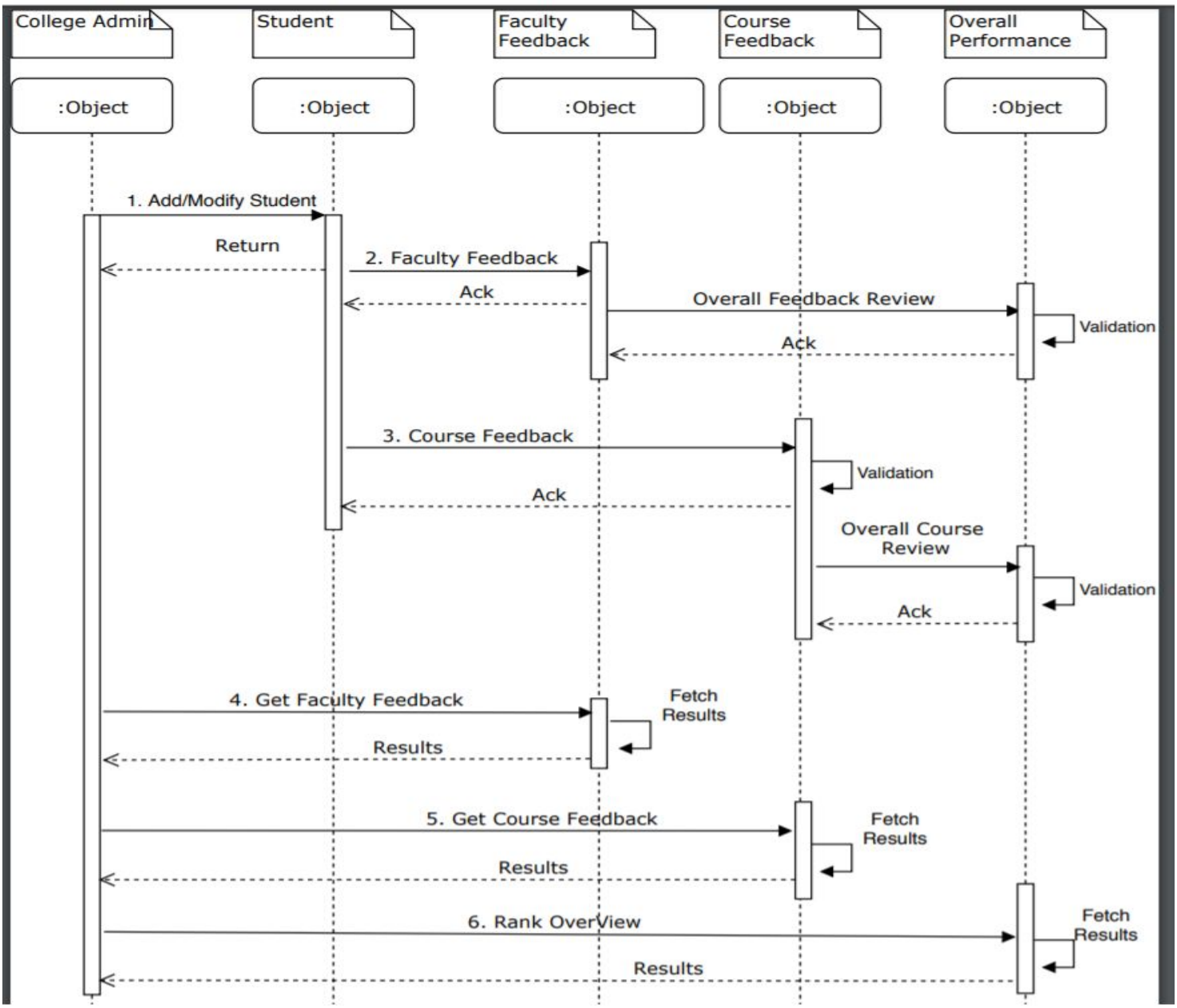
Teacher Feedback System:
Maintaining a feedback system for the faculty evaluation by the students according to their performance and the teaching strategies and course length and breadth offered.
Get the overall feedback score for each faculty and evaluate the faculty compared to the previous year.
Maintain the performance graphs for each professor for each semester and each year.
Methods:
getTeacherName() :String
getPoints() :int
getFeedbackMessage() :String
setTeacherName() :String
setPoints() :int
setFeedbackMessage() :String

Grades
The rating allocated to each student based on his/her performance in a particular course.
The grade is an aggregate of all the tests conducted(like midterms, assignments, projects, etc)
Methods:
getCourseGrades() :String
setCourseGrades() :String

Courses
Courses is where a student needs to enroll in order to complete his degree.
Each course has a Teacher allocated to it who looks after the grading.
Methods:
getCourseName() :String
setCourseName() :String
getCourseId() :int
setCourseId() :int

Sequence diagrams showing how to navigate the university object model to deliver performance metrics needed for performance and feedback.

1. STUDENT FEEDBACK

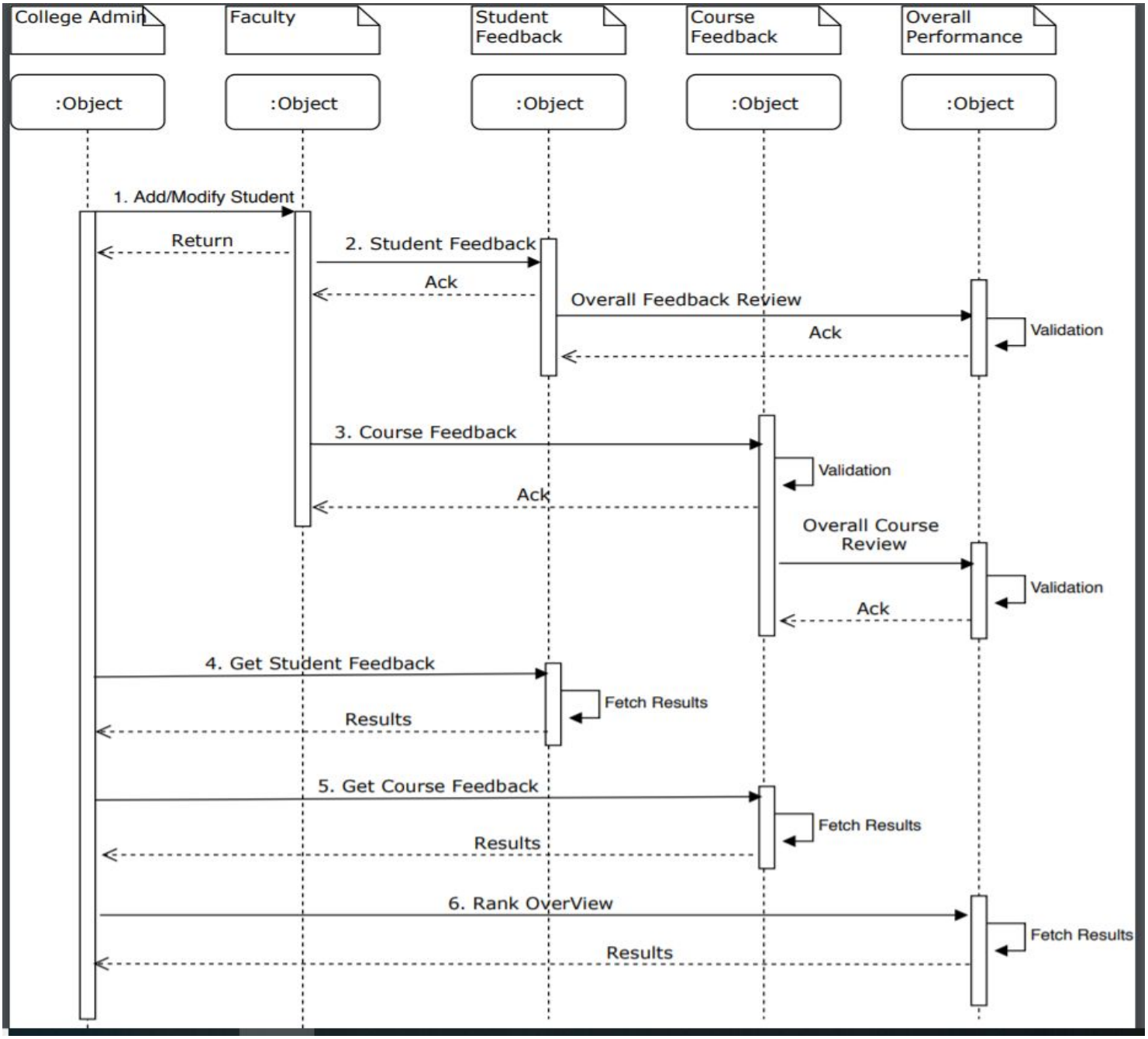


A Student Feedback System has the provision to provide feedback for:
Faculty Feedback & Course Feedback
University/College Admin has the rights to manage Student Information
Once the Feedbacks are stored the admin has the rights to view feedback as per the requirements

Example of feedbacks that can be viewed by the admin:
1. Courses with more ratings

2. Course with high demand
3. Course with least demand
4. Faculty with most ratings
5. Faculty Overview (Average Ratings)
6. Course Overview (Average Ratings)

2. FACULTY FEEDBACK



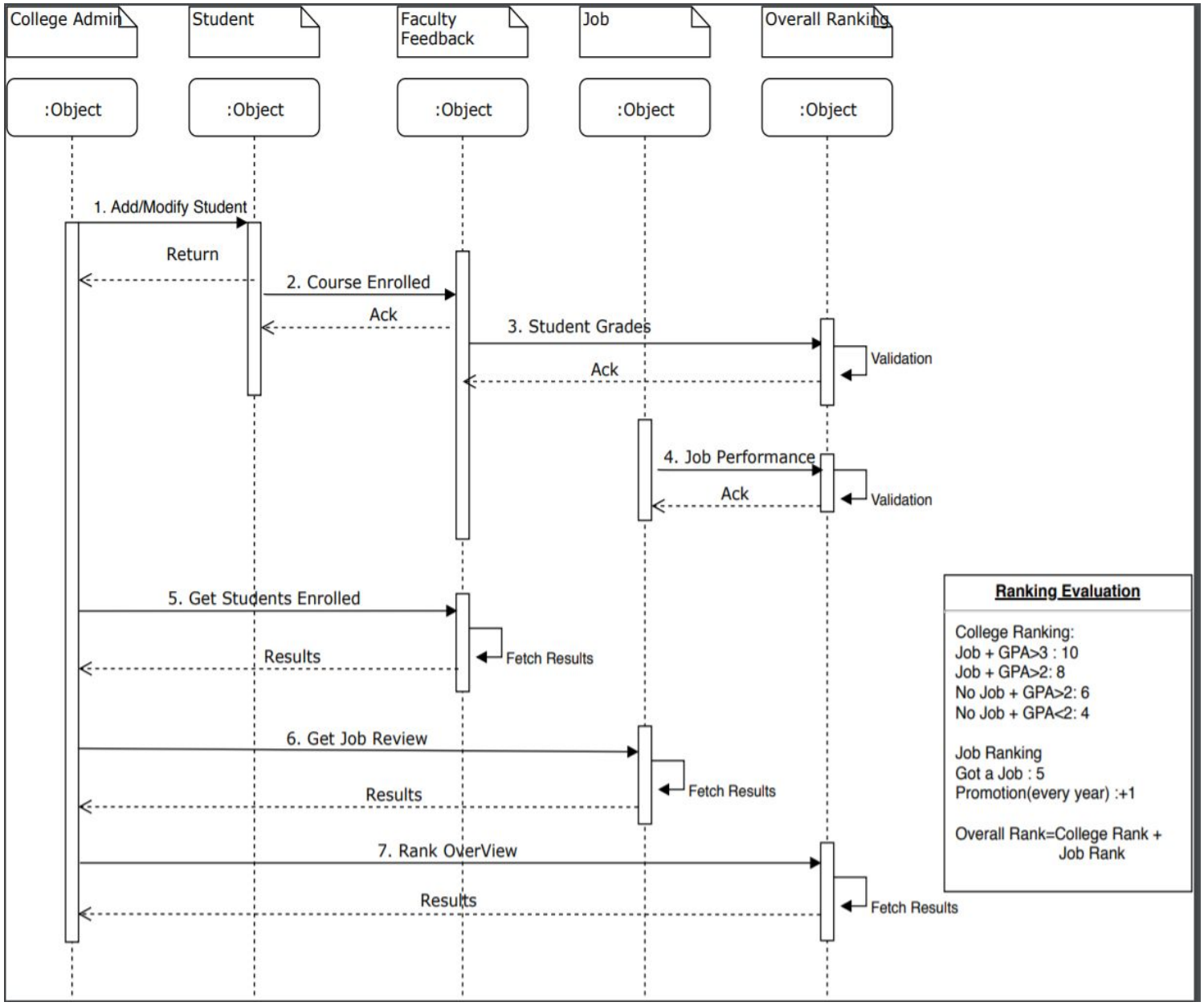
Faculty has the provision to provide the feedback for Student as well as the courses that need updation w.r.t to software/tools used.

Once the Feedbacks are stored the admin has the rights to view feedback as per the requirements

Example of feedbacks that can be viewed by the admin:

1. Top 10 students with good grades (GPA >3.2)
2. Highest Rank Grade
3. Median of Grades
4. Courses with updations
5. Courses with syllabus change reference

3. **OVERALL RANKING SYSTEM**



In the Overall Ranking System,

A student initially enrolls into a particular course under a particular faculty. Faculty is in charge of assessing the students academic performance and will assign a grade which is used to calculate overall ranking which we have defined

Job entity allows us to track the overall job performance of a student after his academic life and to keep a track of his promotions and increments over 5 years which is used to calculate overall ranking system which we have defined

Faculty Grading Scheme:

< 2	F
2.1 - 3	B-
3 - 3.2	B
3.2 -3.5	B+
> 3.5	A

Job Performance review:

Getting a job relevant to courses taken: 5 Points
Increment 1 Point for Promotions over 5 years.

College Grading System (Defined)

If getting a relevant job with a GPA of greater than 3 : 10
If getting a relevant job with a GPA of greater than 2 : 8
If not getting a relevant job with a GPA of greater than 2: 6
If not getting a relevant job with a GPA of less than 2: 4

Overall Ranking System (Defined)

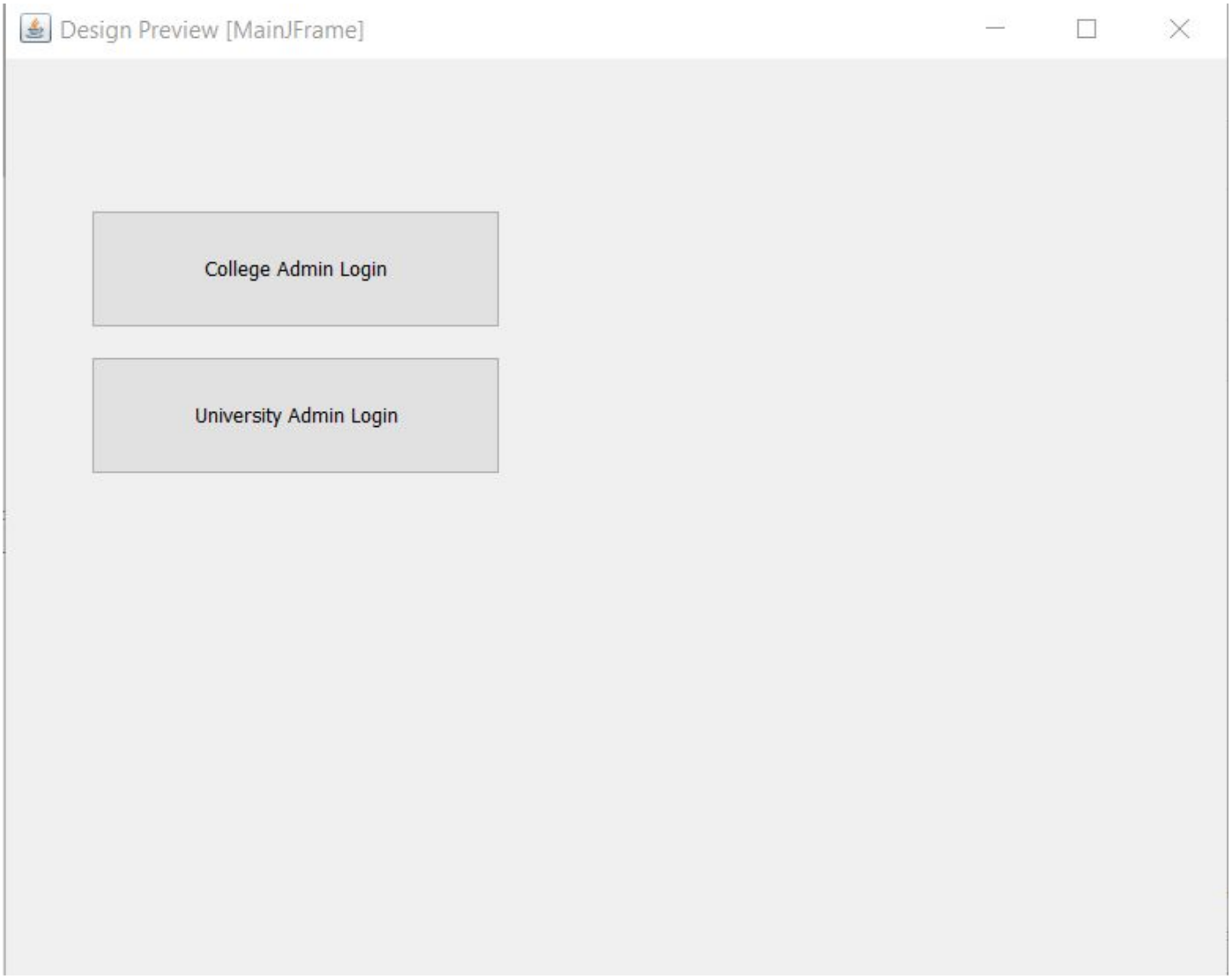
Overall Rank= College Grading system + Job Performance

So for **Example**:
If Robert gets a 3.5 GPA by his faculty and he lands upon a job based on the courses that he took in his academics And has received a promotion already in his 2nd year than his Overall Rank would be as follows:

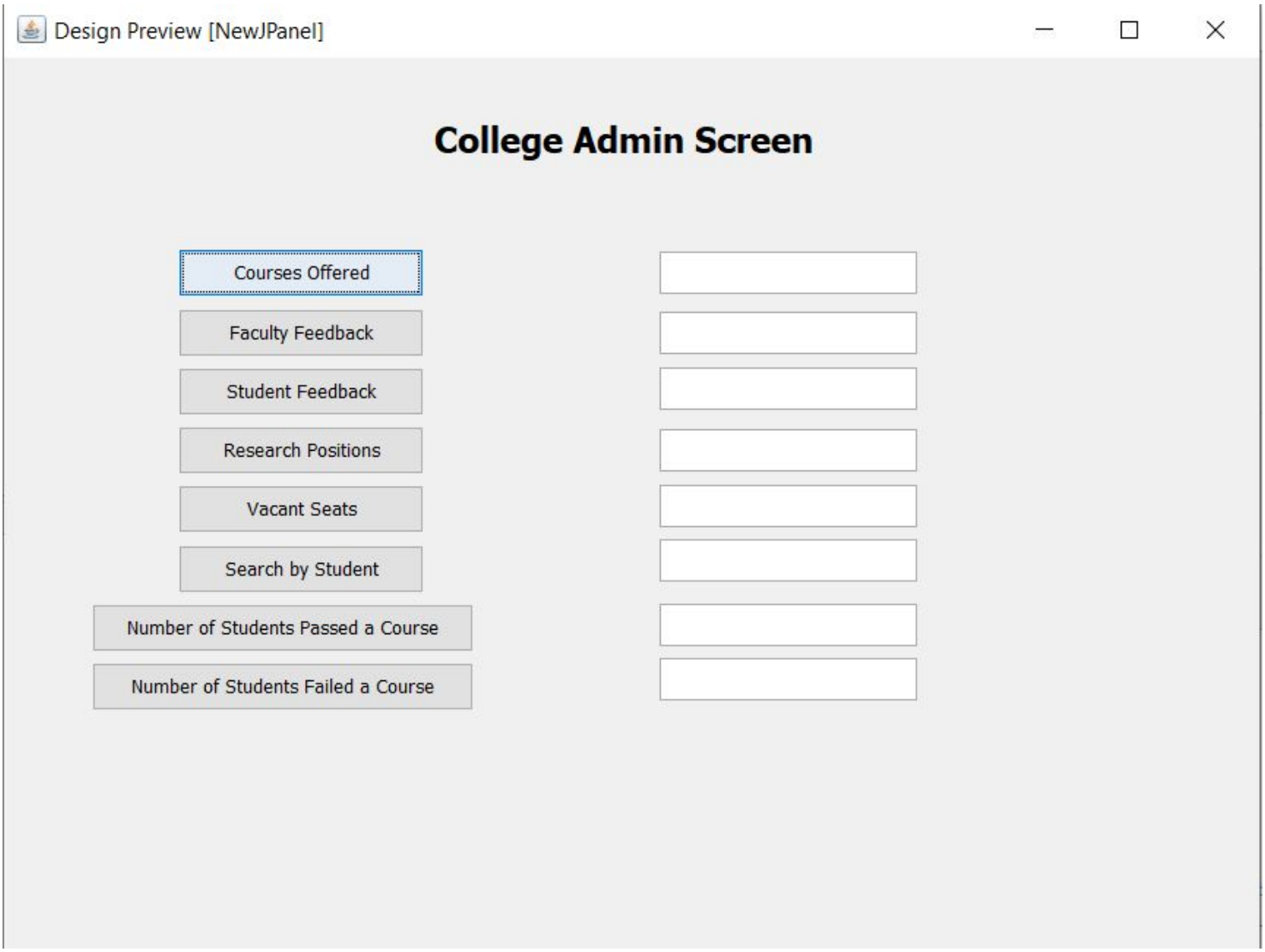
Faculty Grade: 3.5 (A)
Job Rank : 6
College Grading: 10
Overall Rank : 10 + 6 = 16

3. An object model showing the changes to the university model to support the new capabilities. This must include the additional methods and attributes required to deliver the results.

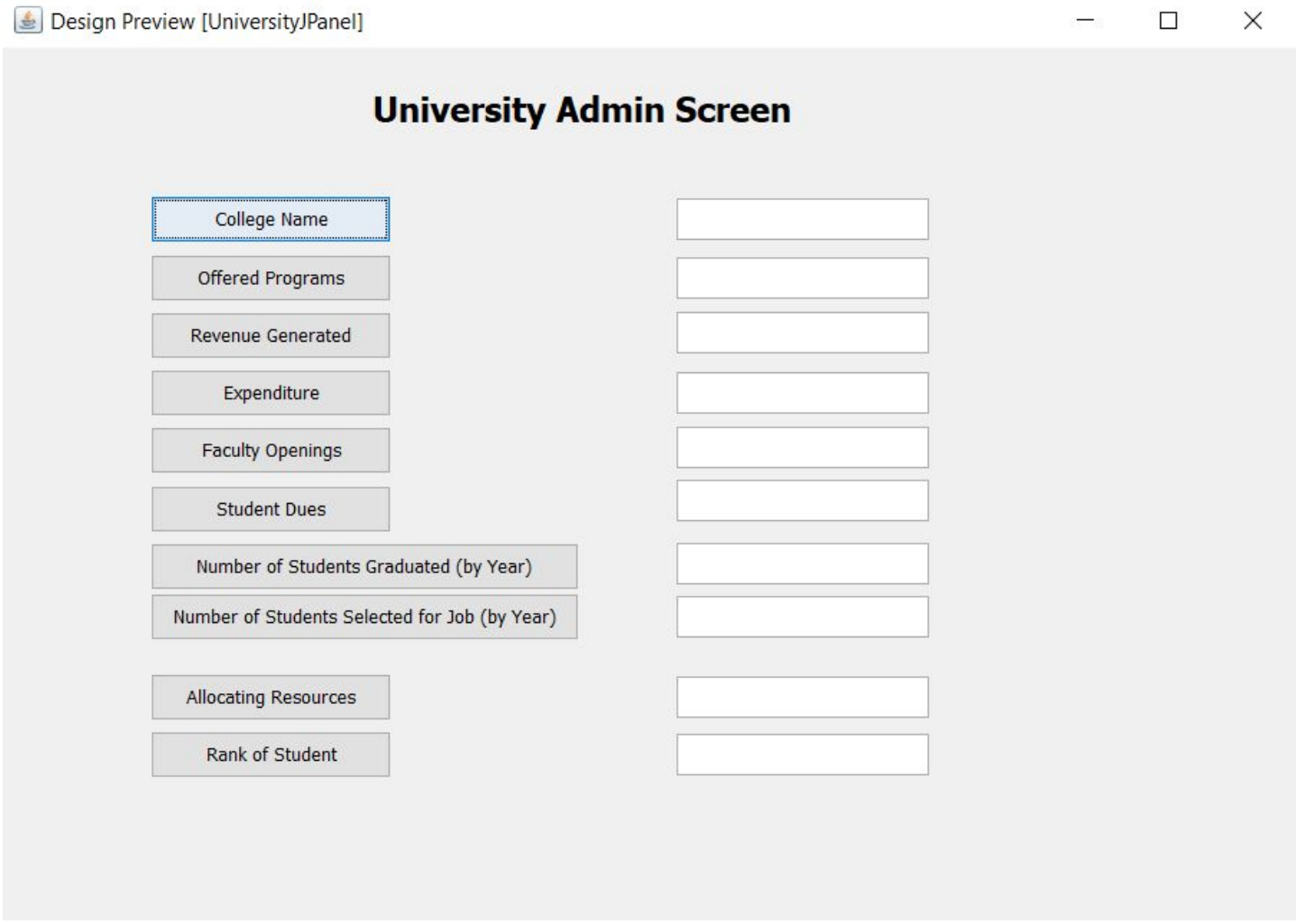
MainJFrame Dashboard Screen



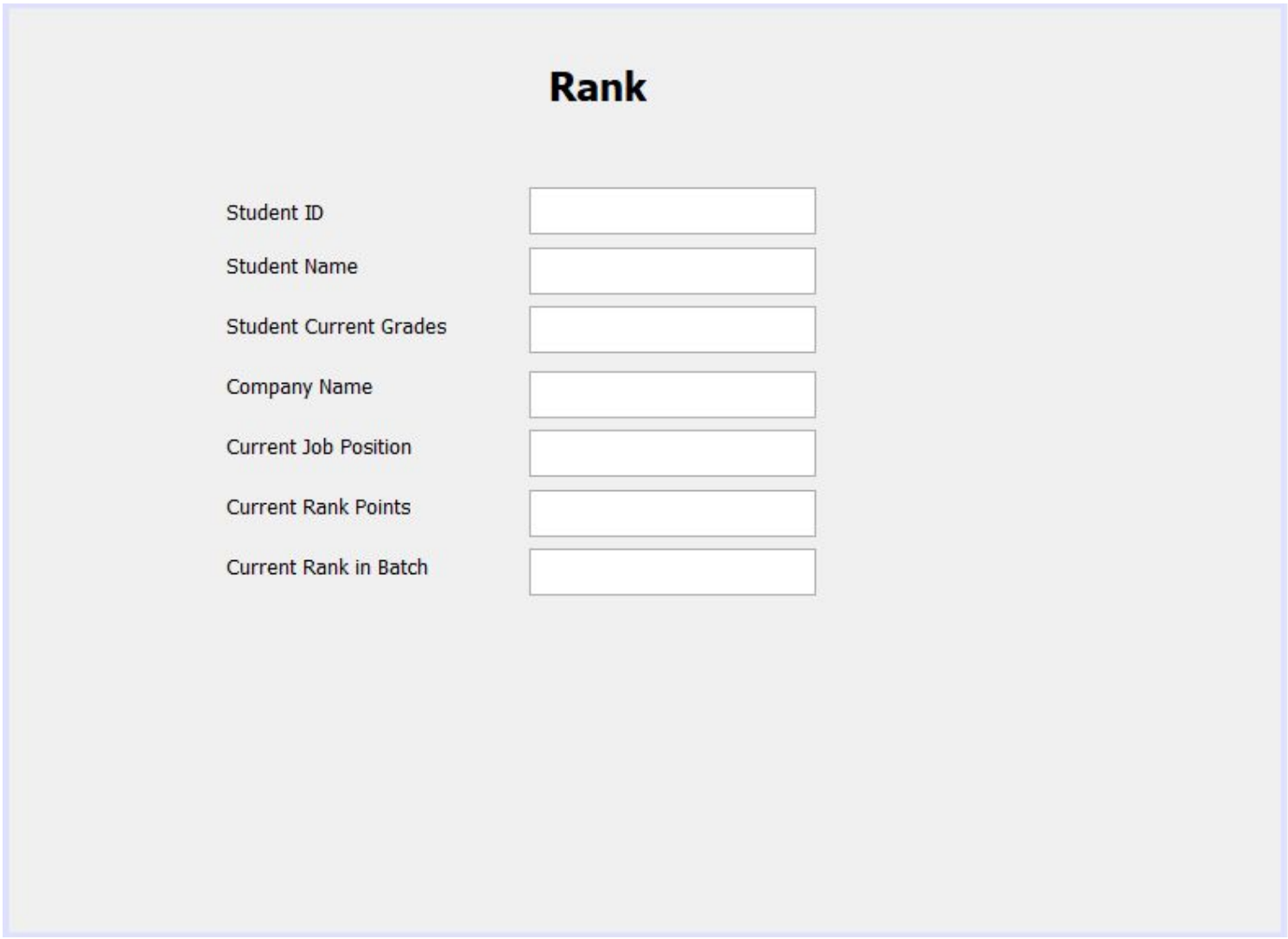
College Admin Screen for Performance View:



University Admin Screen for Performance View:

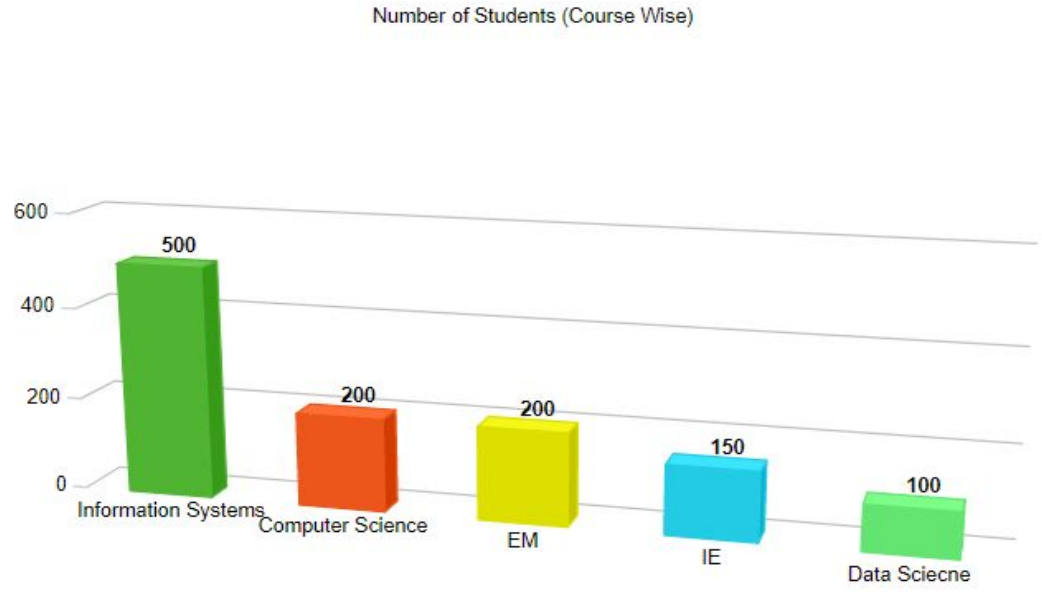


Overall Rank View Screen:

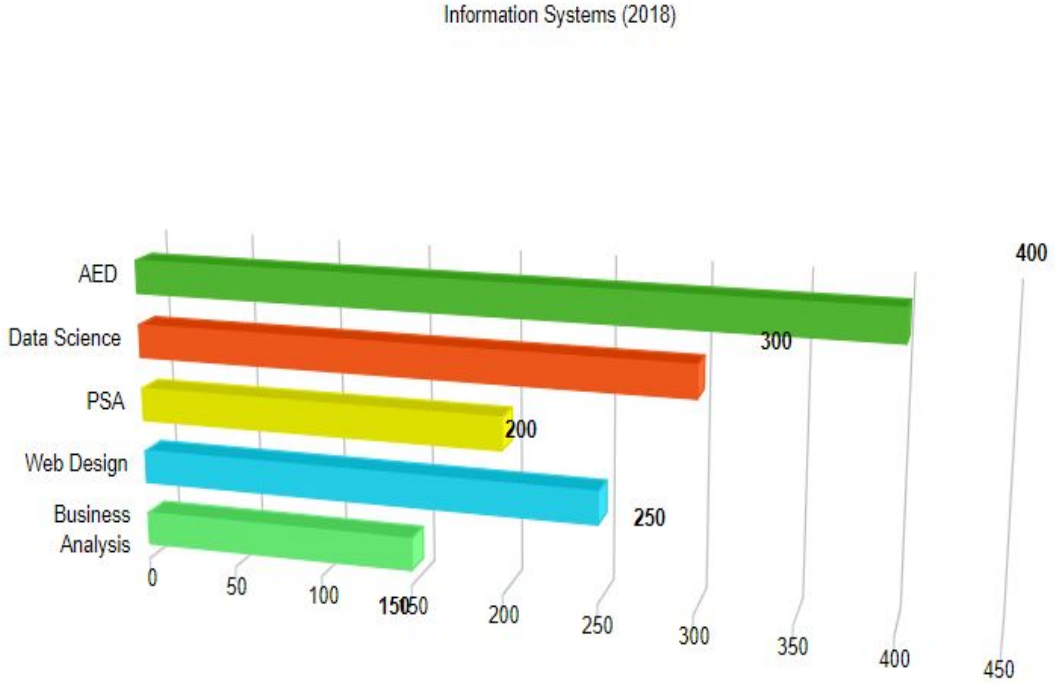


Dashboard allows the admin user to get statistic in the form of graphs and charts on a single button action

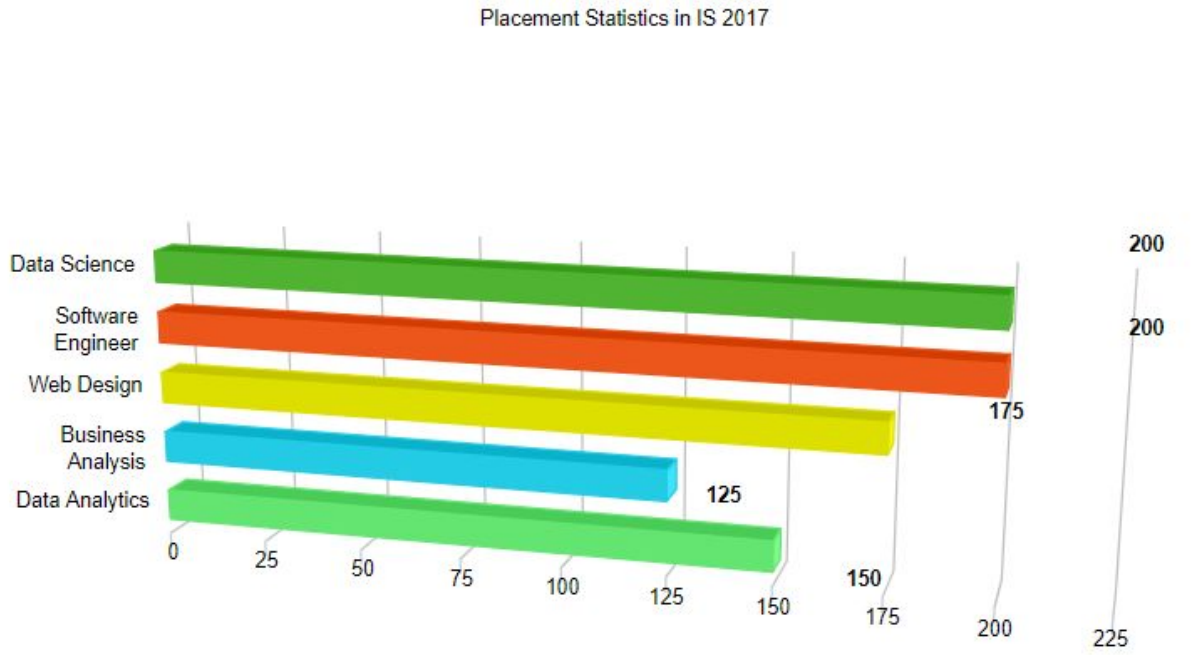
1. **NUMBER OF STUDENTS ENROLLED IN A PARTICULAR COURSE**



2. **COURSE STATISTICS IN INFORMATION SYSTEMS (2018)**



3. PLACEMENT STATISTICS IN THE YEAR 2017



4. PLACEMENT STATISTICS OF LAST 5 YEARS