

NORTHEASTERN UNIVERSITY INFORMATION SYSTEMS FALL '19



UNIVERSITY RANKING MODEL REPORT

APPLICATION ENGINEERING DEVELOPMENT
DESIGN ASSIGNMENT - 1

Created By:

Apurva Mathur (001088100)

Tilak Parmar (001382900)

Nisarg Patel (001371900)

PROBLEM DEFINITION

Creating a performance measurement solution to enable universities to measure the quality of the education they deliver to their students. Approach will be to look into how an educational system in terms of faculty and courses contribute to the growth of their graduates over a 5-year period. 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.

INTRODUCTION

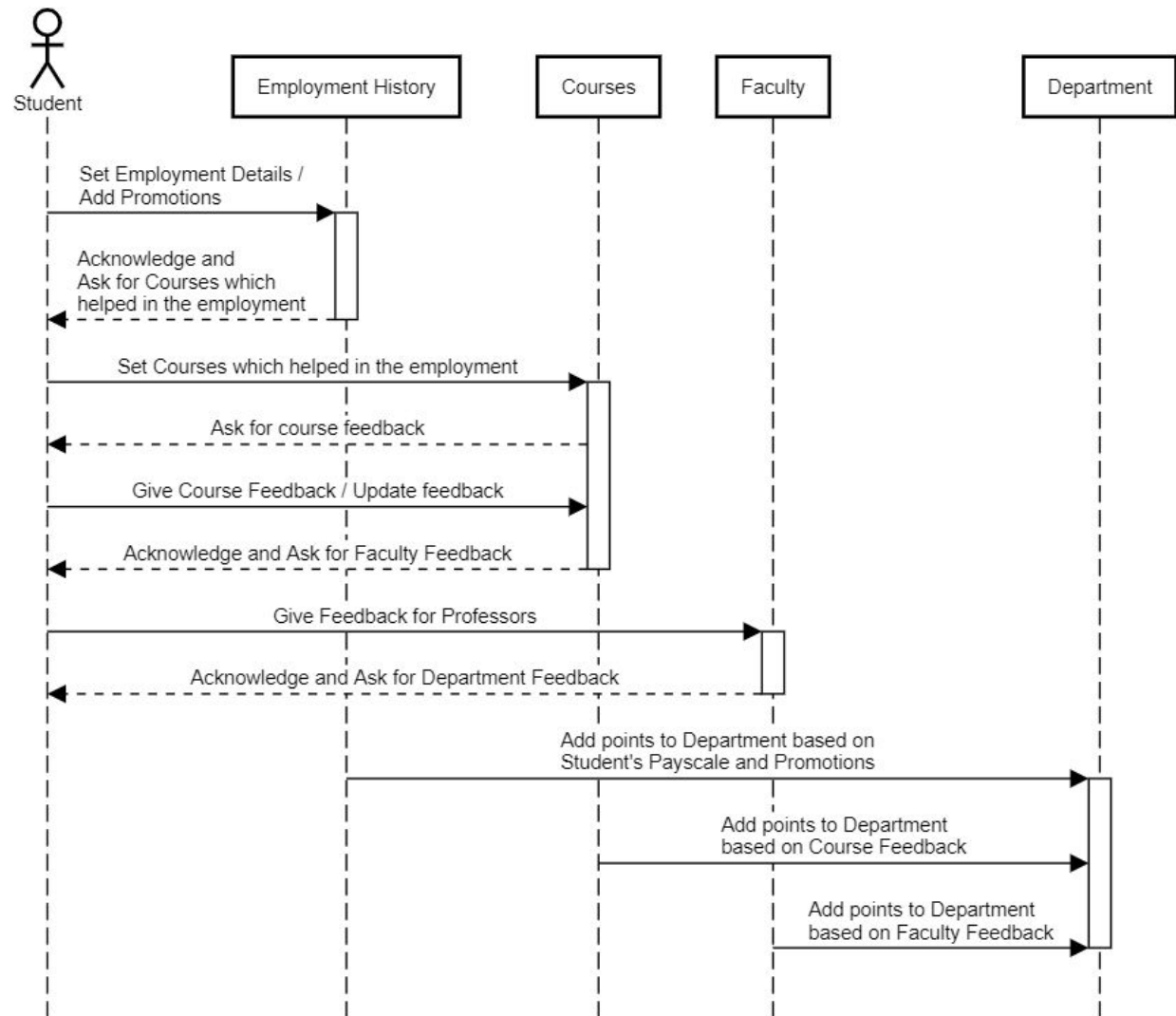
Every university needs to regularly assess themselves in terms of their performance and their relative positioning on the world stage. The challenge for universities is to determine what constitutes good performance in a university, whether it is even possible to measure performance meaningfully at all. Performance depends on how university is doing financially as well as on the quality of education given to their students.

As one of the metrics for performance measurement solution here we will take an approach which will look into how the education provided by the university has an impact on their students and their future employment. We will look into how an educational system in terms of faculty and courses contribute to the growth of their graduates over a 5-year period. We will track the jobs and promotions graduates get over time and assign rankings accordingly.

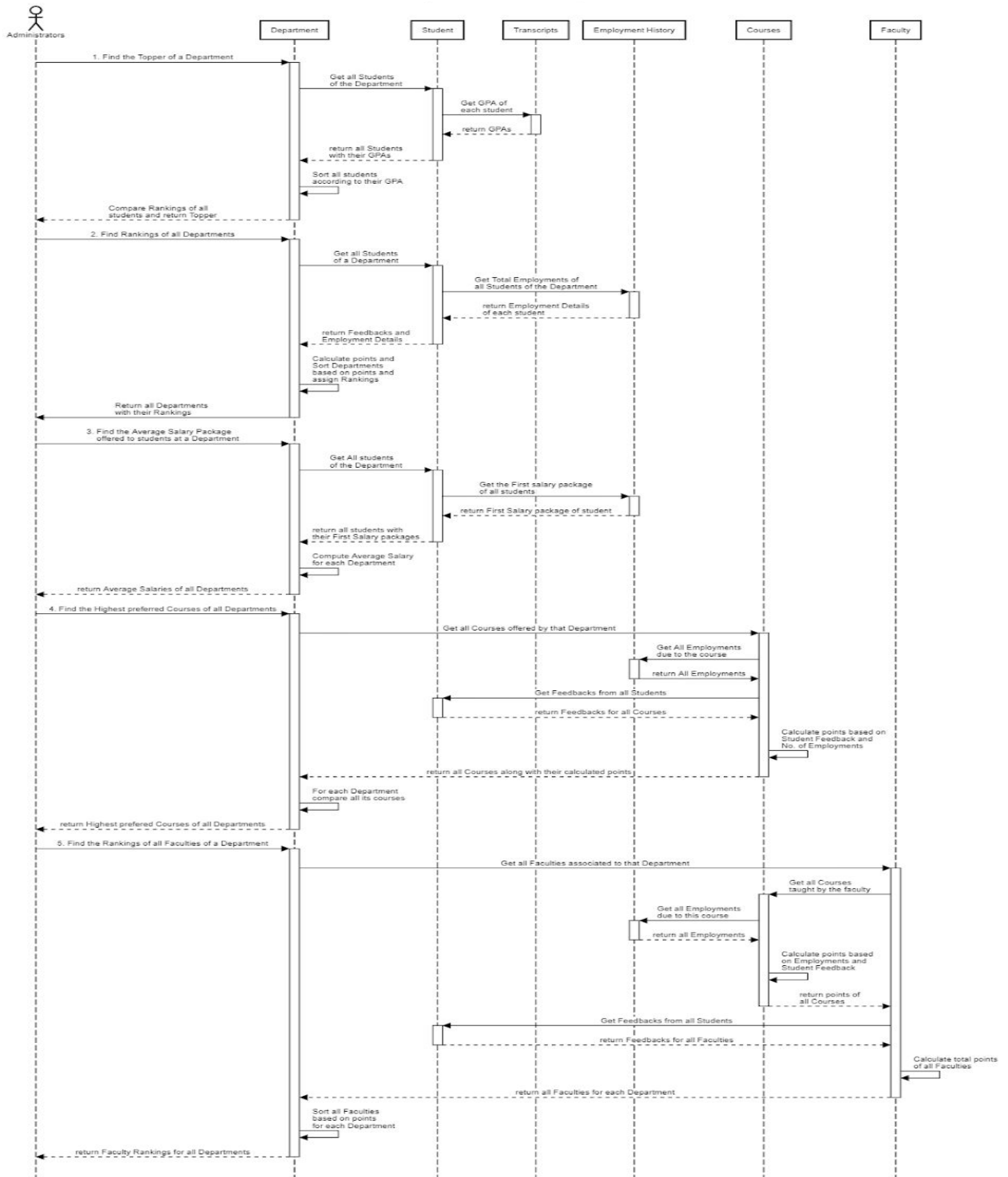
STAKEHOLDERS

- University Administrators
- Professors
- Students
- Alumni
- Employers
- Loan Providers

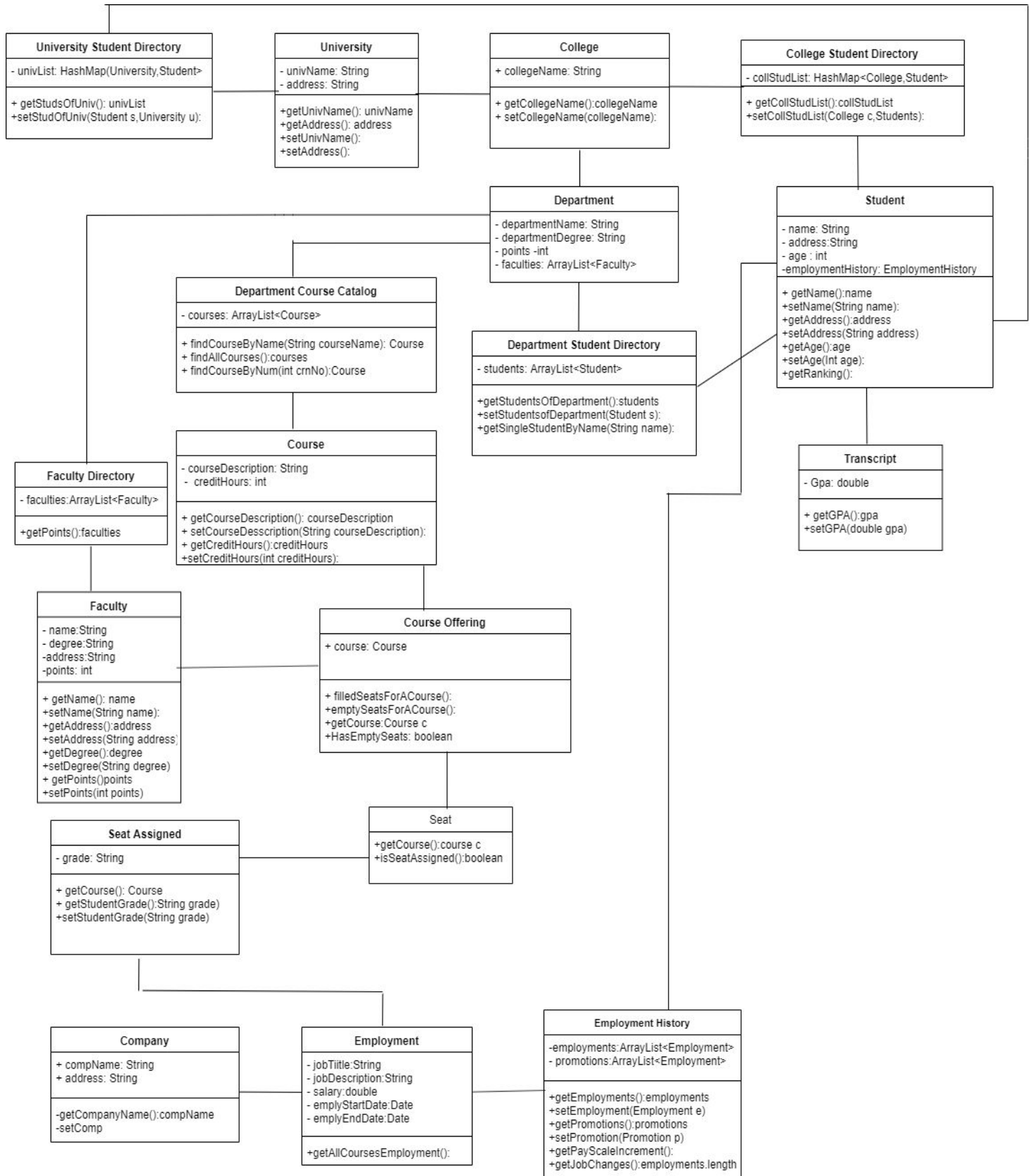
SEQUENCE DIAGRAM 1: STUDENT FEEDBACK



SEQUENCE DIAGRAM 2: ADMINISTRATOR OPERATIONS



OBJECT MODEL DIAGRAM



CLASSES/OBJECTS

University Student Directory:

Responsible for maintaining a list of all students assigned to a university.

University:

Responsible for academic matters such as colleges and students, business matters, takes the performance analysis from the respective colleges and departments in order to improve the quality of education.

Attributes:

String University Name

String address

A single university entity.

College:

Responsible for Academic matters such as departments and students, business matters such as staff, faculty, etc.

Attributes:

String collegeName

College Student Directory:

Responsible for maintaining a list of all students assigned to a college.

Department:

Responsible for Academic matters such as students and courses, Business matters such as staff, faculty, etc. Maintain course catalog and schedule courses and it contains course enrollment number.

Department Student Directory

Responsible for maintaining a list of all students assigned to a department.

Department Course catalog:

Responsible for maintaining a list of all courses offered by a department.

Methods:

CourseCatalog.findCourseByName(String courseName) : returns a specific course searched

CourseCatalog.findAllCourses: returns all courses by a department.

CourseCatalog.findCourseByNumber(int crnNo) : returns a course by crn no search.

Student:

Person authorized to get a college degree from the university.

Has a record of all courses taken during their stay

Has a grade point average (GPA)

Transcript:

It is a historical record of all the courses the student took during their stay at the university. The transcript class knows who the student is.

To find the GPA we retrieve all the seat assignments. Iterate over all the seat assignments. For a given seat assignment get the course grade. Total them up grades. Take the average.

Course :

It has a course Description.

It is core or elective.

Number of credit hours for the course.

Exists in course catalog.

Independent of teacher.

Course offering:

Represents a course that is being offered by a department.

Faculty:

Represents the faculty who teaches the particular course offered by the department

Faculty Directory:

Responsible for maintaining a list of all the professor teaching the courses offered.

Seat:

Represents an available seat in a class

Methods:

getCourse()

IsSeatAssigned()

Seat Assignment:

Represents the fact that a specific seat in a course is assigned to a student. The class is responsible for the student's status in the class as well as the student grades (midterm, final, homework solutions, etc.

Methods

SeatAssignment.getCourse() - Gives the course assigned to the student.

SeatAssignment.getStudent() - Gets the students details.

SeatAssignment.getStudentGrade(); - Get grades of the student

SeatAssignment.setStudentGrade();

Company:

Model a company with an address.

Attributes:

String companyName.

String address.

Methods:

Company.getCompanyName();

Company.setCompanyName(String companyName);

Company.getAddress();

Company.setAddress(Address a);

Employment:

Models each employment.

Attributes:

String JobTitle;

Double Salary;

int EmploymentStartDate;

Int EmploymentEndDate/Present;

String JobDescription;

Methods:

Employment.getAllCoursesEmployment(): All courses associated with an employment.

Employment History:

Models employment history over a period of 5 years for a student.

It maintains a list of all employments for a student.

Methods:

EmploymentHistory.getEmployments() : all employments a students has over a period of 5 years.

EmploymentHistory.setEmployment(Employment e): sets an employment for a student

EmploymentHistory.getPromotions(): Calculates no of promotions a student has over a period of 5 years

EmploymentHistory.setPromotions(Promotion p): sets a promotion for a student.

EmploymentHistory.getPayScaleIncrement(): calculates percentage Increment in salary of a student over a period of 5 years;

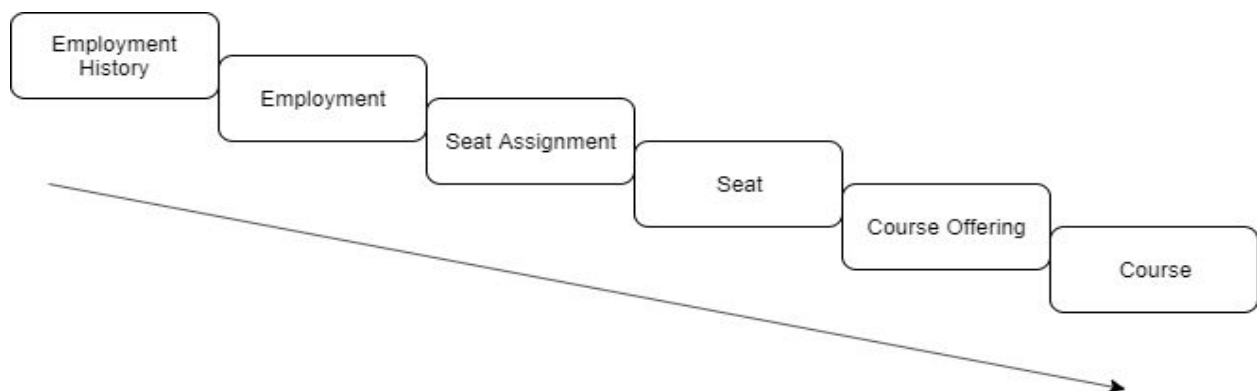
EmploymentHistory(): calculates job changes over a period of 5 years.

HOW THE MODEL ANSWERS QUESTIONS

1. How to determine which courses have an impact on the employment ?

We start with the student and get its respective employment history.

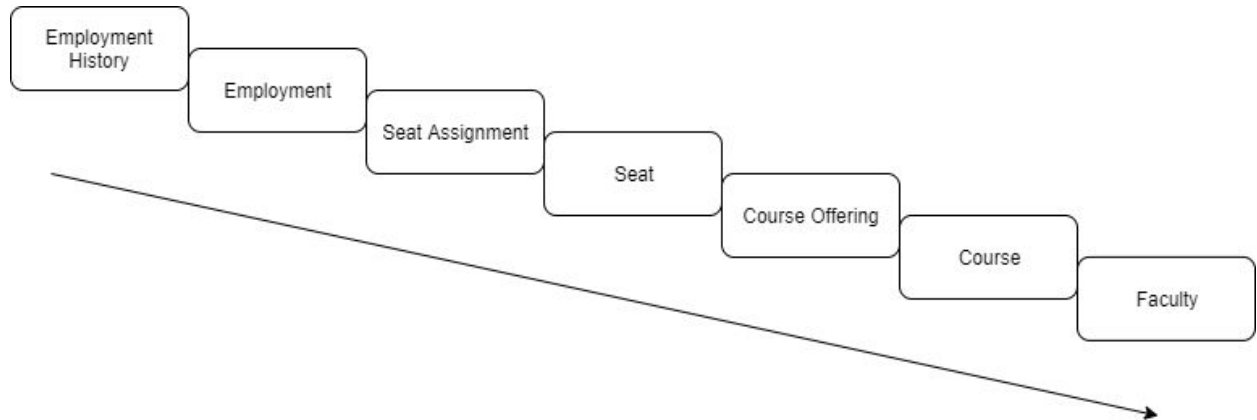
Then after getting the employment history we must iterate over all instances as shown in the below diagram. For example, the seat assignment class has a seat which has a list of all the courses that are offered over a number of semesters. Courses linked with high employment scores will get higher rankings.



2. How to determine which faculty who taught a particular had an impact on the employment?

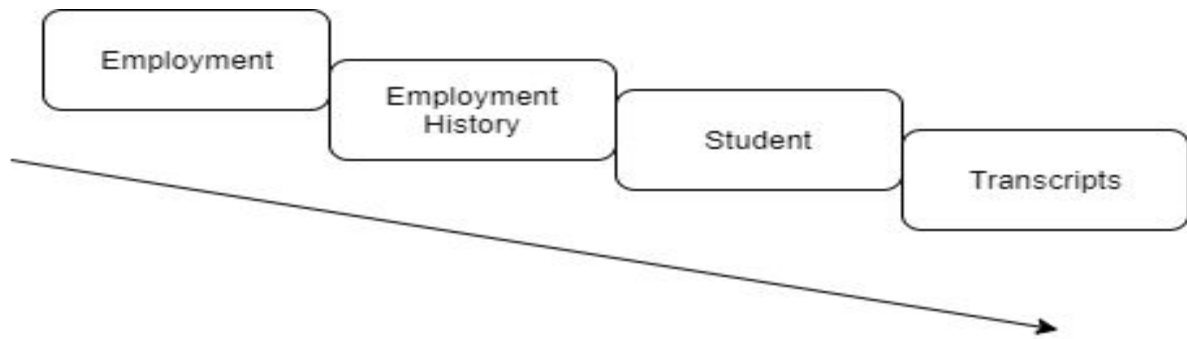
We start with the student and get its respective employment history.

Then after getting the employment history we must iterate over all instances as shown in the below diagram. For example, the seat assignment class has a seat which has a list of all the courses that are offered over a number of semesters. Each course has a faculty designated to it. Faculty linked with high employment scores will get higher rankings.



3. How to determine if grades have an impact on better employment?

We get the employment history of the student which has a list of all the jobs and promotions student had in a span of 5 years. We will iterate over the path given below and get the GPA of the student. After getting the GPA we can analyze if grades have an impact on employment or not.



SEQUENCE DIAGRAM OPERATIONS

1. How to find the Topper of a Department?

Admin can view topper of a Department.

1. An admin needs to login with valid credentials.
2. Admin clicks on Performance Matrix on home screen.
3. Admin clicks on topper of the department button.
4. Topper student is returned with its details.

Computation:

First Department Student Directory will be called. Here the method `getAllStudents()` will give all the students of a department. For each student see its Transcript and `getGPA()` method will give the GPA of all students. Sort all students based on their GPA and return student with the highest GPA.

2. How to find Rankings of all Departments?

Admin can view rankings of a department.

1. An admin needs to login with valid credentials.
2. Admin clicks on Performance Matrix on home screen.
3. Admin clicks on ranking of departments button.
4. Ranking of all Departments is returned with its details.

Computation:

First Department Student Directory will be called. Here the method `getAllStudents()` will give all students of a department. Then for each student its `EmploymentDetails` will be extracted and the feedback of departments calculated will be extracted and points will be assigned based on employment details and feedback i.e. more points will be given to department if for its each employment it has high payscale or promotions are more. Then sorting of departments will be done based on these points and rankings will be assigned. Department with highest points will get rank1 and so on. Then department details will be returned with their rankings.

3. How to find the Average Salary Package offered to students at a Department?

Admin can view Salary Packages of all Students.

1. An admin needs to login with valid credentials.
2. Admin clicks on Performance Matrix on home screen.
3. Admin clicks on Average Salary Package of Department button.
4. Ranking of all Departments based on Average package is returned with its details and average package.

Computation:

First Department Student Directory will be called. Here the method `getAllStudents()` will give all students of a department. Then for each student its `EmploymentDetails` will be extracted and the first package of all students will be summed up and returned for each department. Then the average of this sum of salaries of all students of a department will be calculated. Then all departments with their average salary package will be returned.

4. How to find the Highest preferred courses of each Department?

Admin can view highest preferred courses of all departments based on their rankings.

1. An admin needs to login with valid credentials.
2. Admin clicks on Performance Matrix on home screen.
3. Admin clicks on Highest Preferred Courses of each Department button.
4. A list of preferred courses for each department of the university is displayed.

Computation:

When admin clicks on the highest preferred courses button, Department class is called. Every Department object contains a list of all courses offered at that particular department. The courses call the employment history which returns the total number of students employed due to this course. Along with it, it calls the student feedback for that course. Each course calculates its points based on both of these scenarios. All the courses return their accumulated points to their individual Department object. The Department object sorts all of its courses based on their points and gives rankings. It then returns the course list along with their rankings to the user.

5. How to find the Rankings of all Faculties of a Department?

Admin can view the most preferred professor of a department based on their rankings.

1. An admin needs to login with valid credentials.
2. Admin clicks on Performance Matrix on home screen.
3. Admin clicks on Rankings of Faculties button.
4. Admin enters the Department name for which he wants to see the rankings.(Optional)
5. A list of faculty members of each department of the university is displayed according to their ranks.

Computation:

When admin clicks on the Rankings of all Faculties button, Department class is called. Every Department object contains a list of all its faculty members. Every faculty can teach more than one course. The department object calls the faculties, which in turn calls its courses offered. The courses call the employment history which returns the total number of students employed due to this course. Now, along with it, the faculty object also calls the student feedback. The faculty object calculates its own points based on these scenarios and returns it to the department object. The department object sorts the faculties based on their points and assigns ranks. It displays the final list to the user.