Application Engineering & Development

Assignment 3

Team Members	NUID
Akshata Nanjappa	**002100876**
Drishti Goda	**002199933**
Shivangee Nagar	**002191188**

Problem Statement: Performance measurement solution to enable universities to measure the quality of the education they deliver to their students.

Solution: Measure performance metrics at all stages of the student's journey at a university and after the student graduates from the university. The aim is to accomplish this by measuring multiple metrics viz.

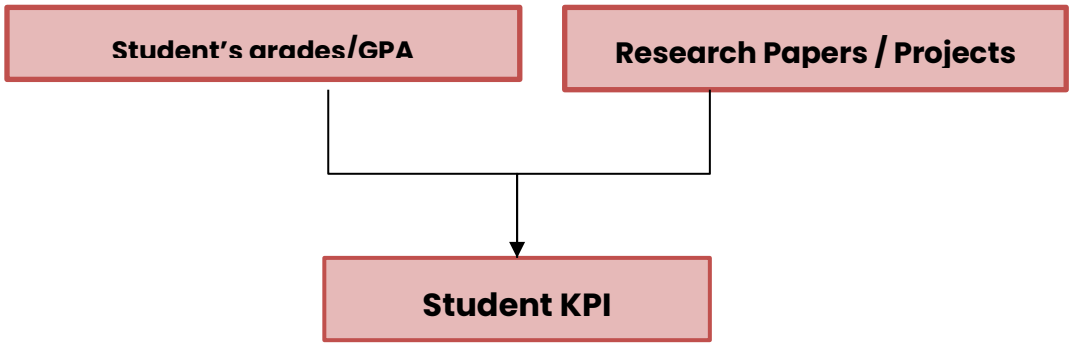
1. Student KPI
2. Professor KPI
3. Employer Student KPI
4. University KPI

Not only is the student's progress tracked in the university, but also after the student is employed. Continuous feedback from the employer is collected in terms of Employer Rating, which helps track the career progression of the University's students over a period. Additionally, this also helps decode the interplay between the availability of skills and the demand for skills. The most relevant courses can be tracked by the proposed solution, and this would help the university tweak existing course offerings or introduce new courses.

PERFORMANCE METRICES

Student KPI

The Student KPI is a direct measure of a student's performance at the university and the student's performance after they graduate from the university. The metrics that form the student KPI can only be awarded by a professor / instructor or university administrator.



During a student's time at university, one or more of the following can contribute towards the student's overall KPI:

1. Student's class performance: A student's grades in all courses taken during the duration of the program.
2. Research Papers / Projects: Academic research papers and projects If a student publishes one or more research papers, then additional points are awarded to the said student and this would eventually increase the student's score.

Calculation of Student KPI:

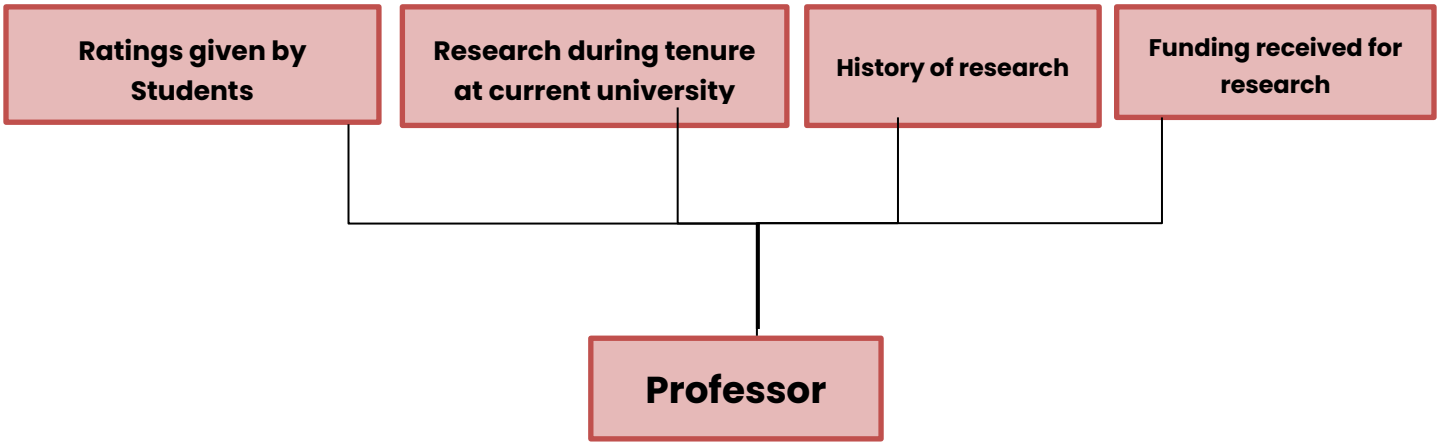
CalculateStudentKPI (Total Grades, ResearchWork, Projects)

The CalculateStudentKPI function is used to calculate the overall student KPI for a given student. The weightage is as follows:

1. Grades: (Student's Academic GPA * 2) + 2
2. Research Papers: Each research paper is rated by the professor under whose guidance it is being published. The average of all scores is taken and used in the overall KPI calculation.
3. Projects: Individual project ratings are averaged and added to the overall Student KPI rating.

Professor KPI

Professor KPI is a measure of a professor's performance. The following can contribute towards the professor's KPI:



1. Rating by Students: Students are allowed to rate the professor after completion of the course. The ratings are directly proportional to the professor KPI
2. Research projects at current university: If the professor has published one or more research papers at the current university, then additional points are awarded to the professor KPI.
3. History of research: A professor's contribution to research (in terms of research papers published etc.) should not be ignored while calculating the professor KPI. A professor's previously published papers / research programs is taken into consideration. A strong history of research will have a greater effect on the Professor KPI.
4. Funding received: Funding received public and private organizations indicates the quality of research being carried out by the professor(s) and the quality of research being done at the university. A weightage is given to the amount of funding received for research projects under a professor and is counted towards the professor KPI.

Calculation of Professor KPI:

calculateProfessorKPI (professor, funding, researchPapers, studentRating, getempStudentRating)

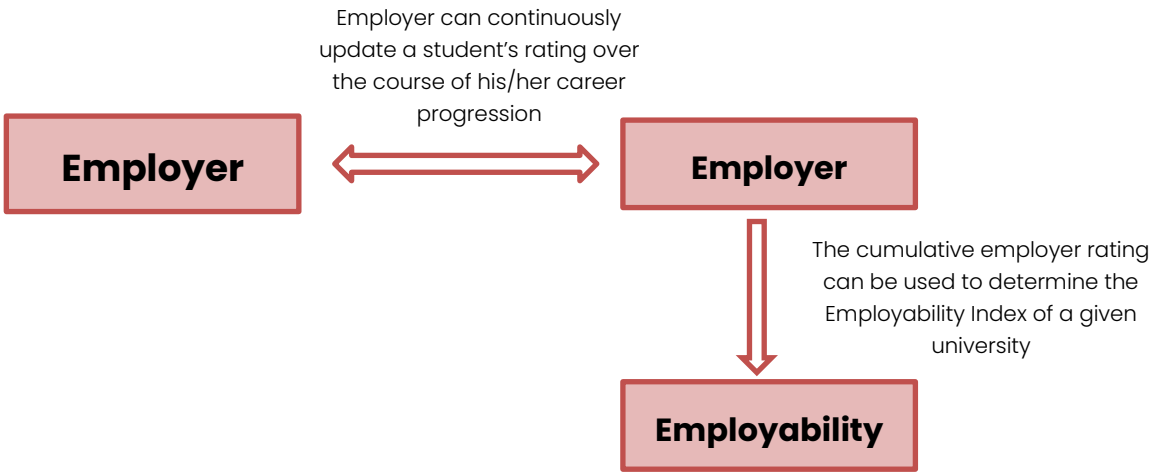
1. Student Ratings: Awarded by students after course completion. Can be between 0 - 10.
2. Research work at the university:

0-2 Projects	2.5 Points
3-5 Projects	5 Points
6-8 Projects	7.5 Points
Over 9 Projects	10 Points

3. History of research: A score is awarded by the department head based on the professor's history of research; score can be between 0-10.
4. Funding Received: The funding received is not always constant for all projects. It is determined by several factors including the domain of research, availability of funding sources etc. Hence, it is unfair to calculate a score based on the amount of funding received in $ value. The department head is allowed to grant a score out of 10 based on the kind and source of funding received.

Student Employment KP

The intent is not only to measure the student's performance in the university, but also to track the student career progression once the student is employed. The employee's performance at the workplace is directly proportional to the quality of education they received at the university. This can be continuously updated by the employer - it can be increased/decreased as per the employee's career progresses. For example: If an employee gets promoted and the employer wishes to increase the rating, they may choose to do so and vice versa.



The Employer Rating can also be used to calculate the **Employability Index** for a given university. This measure can be used to determine how well students transition from higher education to employment. This measure may be used by the university to tweak existing courses or add new courses that may be relevant to the industry and are currently not being taught at the university.

Calculation of Student Employment KPI:

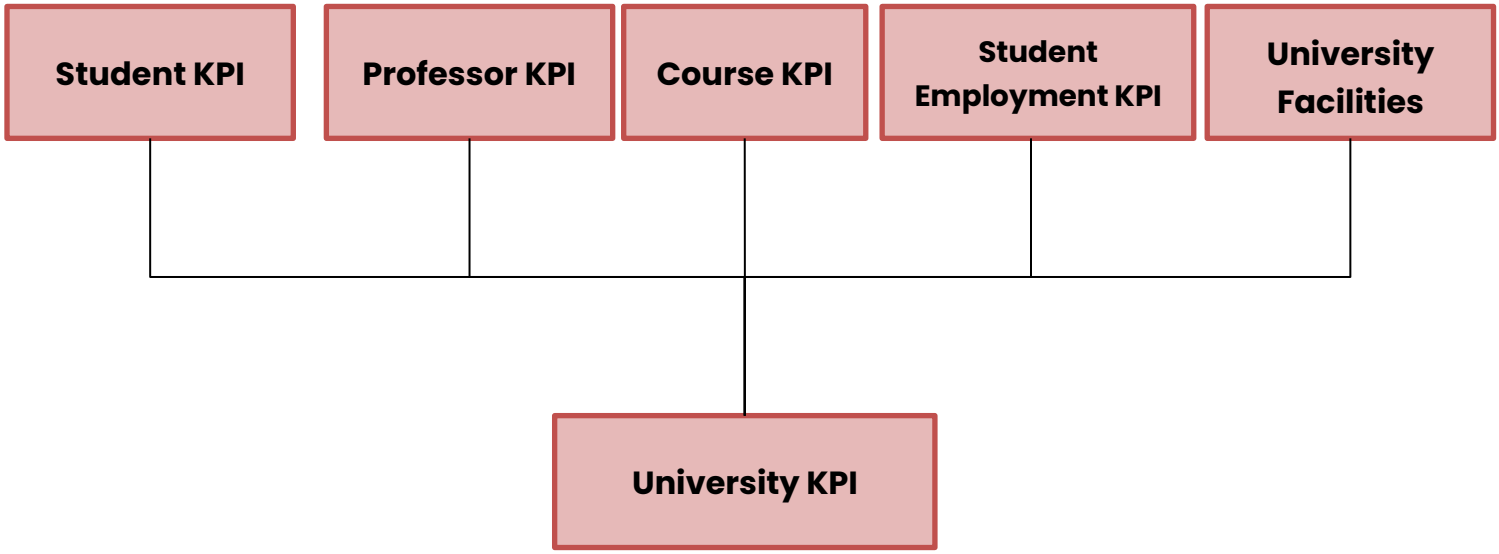
The student employment KPI is not a constant metric that is awarded once. It is a measure of how the student is performing in the real world after graduating from the university. The metric is constantly updated by the university's career department. For example, the career progression of a graduate student can be tracked in the following fashion:

Student ID	JobID	Job Position	Date	Student Employment KPI
00281961	8989989	Data Analyst Intern	Feb 15, 2020	8.0
00281961	8989990	Data Analyst	April 20, 2020	8.75
00281961	8989789	Senior Data Analyst	August 27, 2020	9.5

The Student Employment KPI is updated by the employer continuously whenever there is a change in the employee's position or there is a performance review. Using this system, not only is it possible to track an alumni student's career progression but also to view & examine timelines - in terms of how long it took the student to move out of an internship role to a permanent position at the company. In the above example, the student moved from an internship position to a permanent role in a span of just over 2 months.

University KPI

The university KPI is calculated based on the overall Student KPI, Professor KPI, Student Employer KPI, Course KPI and the University facilities.



1. Overall Student KPI: The average of Student KPIs of all students in the university.
2. Professor KPI: The average of all Professor KPIs currently teaching at the university.
3. Student Employer KPI: The ratings provided by employers to students graduated from the university contributes towards the overall University KPI.
4. University Facilities: On-Campus facilities like gymnasium, swimming pools, library size, campus accommodation facilities etc. play a factor in determining the University KPI. This can be a score between 0-10.
5. Course KPI: The Average of all Course KPIs in the University.

Determining Course Relevance - Job Relevance Factor

Improving the quality of education and offering more relevant courses that align with the needs of the fast-changing tech landscape is one of the many challenges that Universities face. In order for the university to determine and decode this information - the job domains in high demand can be tracked by the university. This can be done by:

1. Each job listing is linked to a job domain. For example: Data Scientist is linked to Data Science
2. The university maintains a list of all domains and the courses that are relevant to the domain. For ex: Data Science may have the following courses listed under it: Introduction to Data Science and Engineering Methods, Advanced Data Science, Machine Learning Algorithms & so on
3. Each job listing may be linked to one or more student IDs.
4. The student's career progression is tracked through the Student Employment KPI.
5. By analyzing the student career progression and the job domain it is linked to, it is possible to obtain a high-level picture of the domains where there is fast career progression. The university can use this information to offer more related courses or to continuously update the course curriculum (for domains, in demand) or remove courses altogether.

Conclusion

The proposed solution can address the problem statement of tracking the growth of graduates over a period, after graduating from the university. Unlike conventional university rankings, where metrics like student performance, acceptance rate etc. are taken into consideration - the proposed solution evaluates the outcome after graduation. By continuously tracking the student's career progression, the university gets insights into the job domains where there is fast paced career progression and the job domains where there is high demand for students. The university can then tweak its course curriculum to offer more relevant courses, that are in demand and may get its students hired. This is particularly useful in the developing world where there is a dearth of candidates with the right set of skills. The proposed solution helps give an overall Performance Indication of the university, that is not based on conventional metrics like pass percentage, acceptance rates etc. The Performance Indicator considers the performance of the student during their time at the university, the performance of the instructors/professors and the performance of the student at the workplace post-graduation. This, we believe is a more robust indicator to determine the University Rankings and the Performance.

Details

Student

Contains student details such as student ID, name, address, and other personal information. In addition to this, student's academic records like grades, research papers published, and academic projects are stored.

- getProjectsbyStudent ()
- getResearchWorkbyStudent ()
- getPersonalInfo ()

StudentDirectory

Contains student details of all Students

- getStudentbyID(studentID) - Returns student details by the Student ID
- getStudentbyName(studentName) - Returns student details by the Student Name

StudentCourseRegistry

Contains the list of all courses that a student has taken during the time at university. A job relevance is also present - which suggests the ideal domain the student may want to get into, based on the courses the student has completed / courses that the student is currently studying.

- getCourseInfo(course)
- getRequiredCourseKnowledgeforJob(job)
- calculateJobRelevanceFactor (course, job)
- getDegreeRequirements ()

StudentKPI

Used to calculate the Student KPI of a given student

- CalculateStudentKPI (Total Grades, ResearchWork, Projects) - Calculates the total student KPI based on a student's total GPA, research work and academic projects done at the university.

Professor

Contains professor details such as professorID, name, rating etc.

- getProfessorFundingDetails () - Get details about funding received for the professor's research projects.
- getProfessorResearchDetails () - Get details about research work being done by the professor
- getProfessorInfo ()

ProfessorDirectory

Contains details of all Professors at the university

- getProfessorbyID(professorID)
- getProfessorbyName (professorName)
- getAllProfessors ()

ProfessorKPI

Used to calculate the Professor KPI of a given professor using the professor rating given by students, the professors research activities at the university and the funding received for research projects.

- calculateProfessorKPI (Professor Rating,ResearchPoints, FundingPoints)

CourseList

Details of all courses:

- getCoursebyID(courseID)
- getCreditHourDetails()
- getCoursePrerequistes()
- getCourseSyllabus()
- getCourseMaterials()
- getCoursebyName(courseName)

CoursesTaken

Details of all courses taken by a given student in a semester

- getCoursebycourseID(courseID)
- getSemesterDetails()
- getMinimumCoursesforSemester()
- getStudentCourseCapacity()

SeatAssignment

Details of a single course taken by a student in a semester. A student may take one or more courses in a given semester.

- getStudentDetails(student);
- getCourseDetails(course);
- getProfessorDetails(professor);

- StudentCourseRating() - Rating given by the student after completion of the course
- StudentProfessorRating() - Rating that the student gives the professor teaching the course
- ProfessorStudentRating() - Rating given by Professor to Student at the end of the course.

CourseSchedule

A department may offer one or more courses to its students in a given semester.CouseSchedule contains all courses that are offered by a department in a given semester.

- getAllCoursesScheduled ()
- getCoursesbyDepartment (department)
- getCoursesbySemester (Semester)
- getDepartmentInfo (Department)

University

Contains details of the university like University Name, ID, Employability Index, Amenities etc

- getUniversityDetails ()
- getUniversitysEventsInfo ()
- getAmenitiesDetails ()
- getUniversitysEventsInfo ()

UniversityKPI

Contains the overall KPI of a university. Used to calculate the overall University KPI as well as obtain individual KPIs for students, professors, StudentEmloyement etc

- calculateUniversityKPI(ProfessorKPI,StudentKPI,StudentEmploymentKPI,AmentiesPoint) - Calculate overall University KPI using the average of Professor KPI, Student KPI, Student Employment KPI and University Facilities.
- getAllStudentKPIs () - Returns the average Student KPI of all students in the university
- getAllStudentEmploymentKPI () - Returns the average StudentEmployment KPI of all students in the university
- getAllCourseKPIs () - Returns the average Course KPI of all courses in the university

College

Contains details of colleges under a given university - like College ID, College Name etc

- getAllColleges ()
- getAllDepartmentInfo ()

Department

Departments can schedule courses and enable/restrict students to/from registering in one or more courses.

- scheduleCourse (semester)
- RegisterStudentforCourses ()
- getTotalStudentsinDepartement ()

Employer

Contains details of an employer which has job vacancies.

- getOpeningsDetails ()
- getEmployerPersonalInfo ()

EmployerDirectory

Contains details of all employers

- getEmployerbyID ()
- getEmployerbyName ()
- getEmployersbyOpening ()
- getAllEmployers ()

StudentEmploymentInfo

Contains details of all students and their employment information like employerID, empStudentRating, Job title etc.

- getStudentEmploymentRating () - Get the Rating provided by the employer to a student during his/her tenure at the company.

StudentEmployementKPI

Used to calculate the Student Employment KPI

- getStudentEmploymentAchievements ()
- StudentEmployerRating ()
- getStudentEmploymentAchievements ()
- getStudentPromotionDetails ()

Job

Used to store information of a Job listing and its attributes like Job Position, ID and Job type

- getJobSalary ()
- trackJobs ()
- getJobDescription ()

JobDomains

Contains list of all Domains and the required course knowledge required to work in these domains

- getJobByName ()
- getAllJobs ()
- getJobbyID ()

CareerDepartment

Used to manage the student employment details - the Career Department is tasked with updating the StudentEmployer Rating

- calculateStudentEmploymentKPI (empStudentRating)
- trackJobs ()
- manageStudents ()