**Report**

**Overview**

The objective of this design is to rate a University based on its performance using Software Engineering Techniques like Object Modelling and Sequential Diagrams using UML. This design enables an End User to measure the quality of Education they deliver by measuring factors such as Student Performance, Placements, Career Growth and how Faculty and Courses contribute to the student’s performance metrics.

**Object Model Diagram**

The Object Model Diagram contains the model of the University with the following classes.

**Seat:**

In class Seat, we have defined the attribute ‘*name*’ which is of type String and ‘*id*’ which if of type Integer.

**SeatAssignment:**

In *SeatAssignment*, we have defined the method *getStudentCourseDetails()* which takes *id* as input and returns the course that the student has taken.

**CourseLoad:**

In *CourseLoad*, *getListOfCourses()* method is defined with *id* as an input parameter which returns the list of courses that student has taken in a particular semester.

**Transcripts:**

In *Transcripts*, *getStudentCourseDetailsPerSemester()* method is defined with student *id* as input which returns the list of courses that student has taken for all semesters.

**Students:**

In *Student* class, we have defined *getTranscripts()* method in order to get results of student transcripts details for all the semesters. *StudentId* is passed as input parameter.

**StudentAccount:**

In the *StudentAccount* class, *getStudentAccountStatus()* method is defined with the Student *id* as input to get the current status of the student (if student is an alumni or not).

**StudentRole:**

In *StudentRole* we have defined the attributes ‘*currentRole*’ of type String to get the current position of a student as an employee, ‘*organizationName*’ of type string to get the organization where the student is working currently and ‘*joiningPosition*’ of type string to get the initial position the student is hired as an employee. The method *getStudentJobDetails()* is defined to fetch the list of student’s present and past job details and the method getStudentPromotionDetails() is defined to get the number of students who got promotions as an employee.

**CourseOffering:**

In class *CourseOffering*, *getCourseSchedule()* method is used to fetch the course schedule based on the *courseId*. The method *getTotalNumberOfCourses()* is defined to get the list of courses for a particular department based on *departmentId*.

**DepartmentCourseOffering:**

In class *DepartmentCourseOffering*, fields *deptId*, *courseId* of type int and *deptName*, *courseName* of type Strings are defined and the method *getCourseCredits()* is defined which returns the number of credits assigned to each course based on the *courseId* input.

**College:**

In class *College*, attributes *collegeId* of type int and *collegeName* of type String are defined. The methods *getDepartmentDetails()* is defined in order to get the results of program details of a college based on the *collegeId*. The method *getNumberOfStudentsPerCollege()* is defined to fetch the total number of students in a college.

**University:**

In the *University* class, *universityId* of type int and *universityName* of type string fields are defined.

**City:**

In class *City*, *cityName* of type String is defined.

**Faculty:**

In class *Faculty*, fields *facultyId* of type int and *facultyName* of type string are defined. Methods *getFacultyPerCourse()* is defined with courseId as input parameter and returns the details of faculty for that course. Method *getFacultyPosition()* is defined to get the faculty position details based on faculty id passed. Method *getFacultyFeedback()* is defined to get the feedback details given by the students for that particular faculty.

**UniversityRating:**

In the class *UniversityRating*, *calculateUniversityRating()* and *calculateCollegeRanking()* are defined to calculate the rankings.

we are calculating the ranks of the colleges as following:

**Percentage Criteria** -

JobDetails : 20%

PromotionDetails : 20%

Faculty Feedback : 30%

Course Ranking : 30%

After getting the details of each section separately we are calculating a score for each college based on the above assigned percentage criteria. Based on the score calculated, rankings are assigned to each college separately and the average of all the colleges is calculated to determine the university performance.