## ****ASSIGNMENT 3****

## ****UNIVERSITY MODEL****

**Team Members:**

DIKSHA BHATIA-002175782

SIMRAN PATIL- 001563069

VIRENDRA SINGH RATHORE- 001584432

## Deliverables:

1. Report outlining your proposed solution.

## Brief outline of our proposed solution:

According to our University Model, Student signs up for a particular course which can be determined by the faculty that is teaching that specific course.

Student is evaluated by the faculty based on assignments, laboratory work and projects.

Faculty gives feedback and allots grades for the same.

Based on the course chosen by the student, we can keep a record of the kind of job a student secures and simultaneously track his growth over a period of 5 years.

We have designed our own unique grading strategy to allocate grades which is a combination of grades assigned by the faculty and job performance evaluation.

## Faculty Grading Strategy:

< 2 F

2.1 - 3 B-

3 - 3.2 B

3.2 -3.5 B+

> 3.5 A

## Job Performance Evaluation:

Securing a course relevant Job: 5 Points Increment 1 Point for Appraisals over 5 years.

## Our Unique Grading System:

Securing a relevant job with a GPA > 3 : **10**

|  |  |
| --- | --- |
| Securing a relevant job with a GPA > 2: **8** |  |
| Not securing a relevant job with a GPA > 2: **6** |  |
| Not securing a relevant job with a GPA < 2: **4** |  |

## Overall Ranking System (Defined)

Overall Rank= College Grading system + Job Evaluation

## University Model - Class Diagram:

Diagram

Description automatically generated with low confidence

**Student Catalog:**

An official register owned by the University.

A track of all student records which consists of student information, courses opted for, courses concluded, late fee payment penalties etcetera are contained in the Student Catalog.

Methods:

setStudentRegister() :int

getStudentRegister() :int

## Student:

An individual admitted in the university program offered by a college.

Has a complete track history about the courses undertaken for the degree. Has a Cumulative Grade.

Methods:

getStudentID() :int setStudentID() :int getFirstName() :String setFirstName() :String getLastName() :String setLastName() :String

## Seat Allocation:

A particular seat in a course is allocated to a distinct student enrolled in the program and is represented by Seat Allocation.

Methods:

getSeatID() :int setSeatID() :int getSeatOccupied() :int setSeatOccupied() :int

## Seat:

All the available as well as vacant seats are depicted here.

Methods:

getNumberOfSeats() :int setNumberOfSeats() :int

## Domain:

A section of college where bifurcation of students is done based on their specialization.

Domain will have specific courses.

Methods:

getDomainID() :int setDomainID() :int getDomainName() :String setDomainName() :String

## Professor:

An individual who is responsible for holding lectures and assessing students in the University.

Assigned with the responsibility of specific courses.

Methods: getProfessorID() :int setProfessorID() :int getFirstName() :String setFirstName() :String getLastName() :String setLastName() :String

## Job:

A paid position of regular employment for students.

A student can enroll into only one job at a time. Methods:

getJobID() :int setJobID() :int

getOrganisationName() :String

setOrganisationName() :String

## Job Statistics:

The rating given by the employer to the student based on his/her performance is depicted here.

Methods:

getJobRating() :int setJobRating() :int

getDesignation() :String

setDesignation() :String

## College:

It oversees all students enrolled and all the different departments existing in the college. Responsible for managing information about faculty as well as students.

Methods:

getCollegeID() :int setCollegeID() :int

getCollegeName() :String setCollegeName() :String

## University:

It oversees the academic specifications about various colleges.

Responsible for maintaining a catalog of all programs and courses offered by each college.

Takes care of business-related matters such as course financing, digital marketing etcetera.

Supervises the resources of each college.

Annual revenue generated from all colleges is looked after by the University.

Methods:

getUniversityID() :int setUniversityID() :int getUniversityName() :String setUniversityName() :String

## Course:

A series of lessons in which a student is enrolled for completion of his/her degree.

Each course has a professor assigned who is responsible for grading the course.

Methods:

getCourseID() :int setCourseID() :int

getCourseName() :String setCourseName() :String

## Course Schedule:

It details the times of day and days of the week that each course will be offered.

Looks after the course timings to avoid time clashes between two courses offered for a program. Methods:

getCourseSchedule() :String setCourseSchedule() :String

## Course Offered:

Manages a directory of the courses offered by each department under a college for different programs.

Oversees interchange course offered list with other departments under the same college.

Methods:

getCourseCatalog() :String setCourseCatalog() :String

## Scores:

Depicts the rating assigned to each student based on his/her performance in a particular course.

An aggregate of all the tests organized such as assignments, lab work, midterms, projects, finals is calculated to provide the required score.

Methods:

getCourseScore() :String setCourseScore() :String

## Rank:

A cumulative aggregation of student list, scores obtained in the respective enrolled courses and job evaluation.

A total point score for each student in the University is calculated and a rank is assigned to every student.

A Student Ranking Directory for performance analysis and overall development of all colleges for different specializations is managed for cumulative years.

Methods:

getRankID() :int setRankID() :int getPoints() :Double setPoints() :Double getRankPosition() :int setRankPosition() :int

calculatePointsByStudentID(int studentID) :Double

The total point for a student is calculated based on the Job performance and the grades scored in their respective specializations in this method.

## Professor Rating System:

A rating structure for the faculty by students based on the performance and effectiveness of the teaching strategies adopted by the Professor during the length of the course.

An overall feedback score for each faculty member is calculated and compared to the prior academic year.

Performance statistics graphs for each professor in each semester and year is maintained.

Methods: getProfessorName() :String

setProfessorName() :String getPoints() :int

setPoints() :int

getRatingDetails() :String setRatingDetails() :String

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

1. **CGPA BASED RANKING**

**Diagram

Description automatically generated with low confidence**

A Student Rating System has the provision to provide rating for:

* Faculty Rating
* Course Rating

University/College Administrator has the rights to manage Student Information.

Once the ratings are registered, the Administrator has the authority to view ratings as per requirements.

Example of ratings that the Administrator can view:

* Course with high demand
* Course with low demand
* Course with more ratings
* Faculty with most ratings
* Faculty Overview (Average Ratings)
* Course Overview (Average Ratings)

# FACULTY FEEDBACK

Faculty has the provision to provide the ratings for Student.

They are responsible for supervising courses that require updated software/tools.

Once the Ratings are registered, the Administrator has the authority to view ratings as per requirements.

Example of ratings that can be viewed by the Administrator:

* Courses that require to be updated.
* Courses with syllabus change reference.
* Highest Rank Score
* Median of Scores
* Top 10 students with outstanding scores (GPA >3.2)

# 3. OVERALL RANKING SYSTEM

In the Overall Ranking System,

A student enrolls in a specific course under a specific faculty. Faculty is responsible for assessing the students’ academic performance and assigning a grade which is then used to calculate the overall ranking.

Job entity allows us to track the overall job performance of a student after the completion of his/her degree and keeps a record of his/her job appraisals over a period of 5 years which is then used to calculate the overall ranking system which is defined by us as follows.

## Faculty Grading Strategy:

< 2 F

2.1 - 3 B-

3 - 3.2 B

3.2 -3.5 B+

> 3.5 A

## Job Performance Evaluation:

Securing a course relevant Job: 5 Points Increment 1 Point for Appraisals over 5 years.

## College Grading System (Defined)

Securing a relevant job with a GPA > 3 : **10**

|  |  |
| --- | --- |
| Securing a relevant job with a GPA > 2: **8** |  |
| Not securing a relevant job with a GPA > 2: **6** |  |
| Not securing a relevant job with a GPA < 2: **4** |  |

## Overall Ranking System (Defined)

Overall Rank= College Grading system + Job Performance

**Example**:

If Diksha gets a 3.5 GPA by her faculty and secures a job based on the courses that she took in her semesters and has received a promotion already in her 2nd year, then her Overall Rank would be as follows:

Faculty Grade: 3.5 (A)

Job Rank: 6

College Grading: 10

Overall Rank: 10 + 6 = 16

**An object model showing the changes to the university model to support the new capabilities:**

**Diagram

Description automatically generated**

**­ Dashboard Login Screen**

**Graphical user interface, text, application

Description automatically generated**

**Performance Metrics Screen - Student Ranking**

**Graphical user interface, application, Teams

Description automatically generated**

**Performance Metrics Screen – Placement Summary**

**Table

Description automatically generated**

**Performance Metrics Screen – Accounting and Finance**

Table

Description automatically generated

**Performance Metrics Screen – Faculty Rating**

**Table

Description automatically generated**