Report

**Performance**

**Measurement**

**Model for University**

Assignment No. 3

Participants:

1. Aarti Mehta
2. Siddhartha Savant
3. Tejas Bawankar

Table of Content

|  |  |  |
| --- | --- | --- |
| **Sr No** | **Content** | **Page Number** |
| 1. | Objective | 3 |
| 2. | Assignment Goals | 4 |
| 3. | Class Tables and Attributes | 5 |
| 4. | UML Diagram |  |
| 5. | Sequence Diagram |  |
| 6. | Input Forms |  |
| 7. | Ranking Criteria |  |
| 8. | Dashboard |  |
| 9. | Conclusion |  |

1. Objective

The objective of this assignment is to instill in you the techniques for turning an object model into a machine for information gathering and data aggregation. We want to use software engineering techniques to improve the quality of education anywhere and hold people accountable for improving the quality of life through education, learning to learn, and feedback.

1. Goals

* To create a performance measurement solution to enable universities to measure the quality of the education they deliver to their students.
* To track the jobs and promotions graduates get over time and assign rankings accordingly.
* To track the connection of courses and their relevance to graduates' growth.
* To design a dashboard that enables college and university administrators to compare the performance of their academic units.
* To consider ways to define your own ranking system for students to decide where they want to go for their studies.

1. Classes used in performance measurement model

|  |  |  |
| --- | --- | --- |
| **CLASS NAME** | **CLASS ATTRIBUTES** | **CLASS FUNCTIONS** |
| Students | String studentID  String firstName  String lastName  String emailID  College college  Department department  Date yearOfGraduation  Transcript transcript  sernam internship | getCollegeName()  getCourseList()  getGpa()  getDepartmentName()  isInternship()  getYearOfGraduation()  getTranscript()  getAggregateScore() |
| Employed | int salary  String CompanyName  String title  String jobDomain  int companyRanking  int companyAppreciation | getRating() |
| SelfEmployed | long sername  int employeeCount  long funding  int awards  String startupDomain  int numberOfClients | getRating() |
| HigherEducation | String degreeType  int universityRanking  int awards  int numberOfPublications  int numberOfPatents  String educationDomain | getRating() |
| CurrentStudents | String degreeType  int awards  int NumberOfPublications  Int NumberOfPatients  String educationDomain | getRating() |

|  |  |  |
| --- | --- | --- |
| Course | String courseName  int sernam  int courseCredits  int registrationCount  int courseFees  StudentDirectory studentDirectory  FacultyDirectory facultyDirectory  Course courseTaught | getCourseName()  getCourseID()  getCourseCredits()  getRegistrationCount()  getCourseFees() |
| Faculty | String facultyName  String facultyType  String facultyQualification  String facultyAchievements | getFacultyName()  getFacultyType()  getFacultyQualification()  getFacultyAchievements()  getCourse() |
| Department | String departmentName  CourseDirectory courseDirectory  StudentDirectory studentDirectory  FacultyDirectory facultyDirectory | getDepartmentName()  getCourseDirectory()  getStudentDirectory()  getFacultyDirectory() |
| FacultyDirectory | List facultyList  Department department | getFacultyList()  addFaculty()  updateFaculty()  deleteFaculty()  searchFaculty() |
| StudentsDirectory | List studentList  Department department | getStudentName()  addStudent()  updateStudent()  deleteStudent()  searchStudent()  getStudentRating() |
| Transcript | Float gpa  List courseList  Int totalCredits | calculateGPA()  getCourseList()  getTotalCredits() |
| CustomRanking | List studentList  List CollegeList | getRankingByDepartment()  getRankingByFaculty()  getRankingByStudent()  getRankingByCollege() |
| CourseDirectory | List courseList  String lastUpdated  Department department | getCourse()  addCourse()  updateCourse()  deleteCourse()  newCourse() |
| Admin | String username  String password | login() |

1. Object Model Diagram

* The model diagram proposed has multiple units like student, college, department, course, faculty, etc.
* Custom ranking is based on various parameters and acts as a filter while getting all the records of the student in the university.
* In the object model, the student class is inherited by the HigherEducation, CurrentStudents, Employed and SelfEmployed classes.

1. Sequence Diagram

* In the sequence diagram, the admin acts as the main actor which is external to the system, and which interacts with the entities of the system.
* Rating of the students depends on the type of class the student is extending. The factors considered are number of publications, number of patents, salary, company ranking, number of clients, awards, etc.

1. Student Profile

* Information of every student who has graduated and current student will be taken as input. There are four types of students: Employed Student, Self Employed Student, Student pursuing higher education and current student at university.
  1. **Employed Student Input Form**
  2. **Self Employed Student Input Form**
  3. **Higher Education Student Input Form**
  4. **Current Student Input Form**

1. Ranking Criteria

**Ranking by Employed Students -**

|  |  |  |
| --- | --- | --- |
| **Ranking Criteria** | **Ranking Scale** | **Ranking Points** |
| Company Ranking | 1 - 10  11- 50  51 - 100  101 - 200  201 - 500 | 10  9  8  7  6 |
| Employee Salary | 200k+  150k - 200k  100k - 150k  80k - 100k  Below 80k | 10  9  8  7  6 |
| Number of Achievements | 5+  4  3  2 | 10  9  8  7 |
| Relevance to Degree | 100%  50 - 90%  30 - 50%  None | 10  7  5  0 |
| GPA | 4  3.8 - 4  3.5 - 3.7  3 - 3.5  <3 | 10  8  7  6  5 |
| Internship | Yes  No | 10  0 |

**Ranking by Self Employed Students -**

|  |  |  |
| --- | --- | --- |
| **Ranking Criteria** | **Ranking Scale** | **Ranking Points** |
| Employee Count | 200+  151-200  101-150  51-100  Below 50 | 10  9  8  7  6 |
| Turnover | 10M+  1M-10M  500K - 1M  100K - 500K  Below 100K | 10  9  8  7  6 |
| Funding | 10M+  1M-10M  500K - 1M  100K - 500K  Below 100K | 10  9  8  7  6 |
| Number of Awards | 5+  4  3  2 | 10  9  8  7 |
| Number of Clients | 100+  81-100  61-80 51-60  41-50  Below 40 | 10  9  8  7  6  5 |
| Relevance to Degree | 100%  50 - 90%  30 - 50%  None | 10  7  5  0 |
| GPA | 4  3.8 - 4  3.5 - 3.7  3 - 3.5  <3 | 10  8  7  6  5 |
| Internship | Yes  No | 10  0 |

**Ranking by Higher Education Students -**

|  |  |  |
| --- | --- | --- |
| **Ranking Criteria** | **Ranking Scale** | **Ranking Points** |
| University Ranking | 1 - 10  11- 50  51 - 100  101 - 200  201 - 500 | 10  9  8  7  6 |
| Number of Patients | 1 or more  none | 10  0 |
| Number of Publications | 5  3 or 4  1 or 2  None | 10  8  6  0 |
| Number of Achievements | 5+  4  3  2  None | 10  9  8  7  0 |
| Relevance to Degree | 100%  50 - 90%  30 - 50%  None | 10  7  5  0 |
| GPA | 4  3.8 - 4  3.5 - 3.7  3 - 3.5  <3 | 10  8  7  6  5 |
| Internship | Yes  No | 10  0 |

1. Performance Dashboard
   1. **Performance by Student**

* The table will have the information of every student at the University, the score would be calculated based on the parameters in the above page, and the list would be in order of Rank of student.
  1. **Performance by College and Department**
* Every College has been given a score and Rank. The score is the aggregate of the score of each student who has studied in that College. Student Record specific to a college can be searched. It will fetch you the details of student in that particular college.
  1. **Performance by Faculty**
* Every faculty has been given a score and rank. The score is the aggregate of the score of each student who has taken a course of the faculty. Alumni Record specific to a faculty member can be searched. It will fetch you the details of student’s who has taken courses under that particular faculty.