**University Performance Evaluation Model**

**Team Members:**

-Pradyumna Reddy Maddi [NUID: 002197291]

-Rajasekhar Reddy Gosala [NUID: 002198135]

-Merwin Roy [NUID: 002959569]

**REPORT**

**MODEL OBJECTIVE –**

The proposed model in this project aims to provide institutions with a solution to quantify and evaluate various performance metric of university’s educational system through feedback fetched from its students. This information gathering system will help educational institutions to improve the quality of education they deliver to its students based on the feedback.

**ABSTRACT –**

The proposed system will measure the various performative aspects of the university through the use of feedbacks gathered from its students. The system will encompass various university characteristics such as courses, professors and faculty, the trend projected in the field of education and by taking the student details into consideration, the system will track the growth of the student by extrapolating their responses on the ranking system.

**APPROACH –**

The key components in the proposed system are as follows:

**Student:**

* Student will be an actor in this model which will enter their respective login credentials in order to proceed to fill and submit the feedback.
* On the feedback panel, students will rank courses and faculty along with their details which can be utilized by institutions to track the student growth based on the relevance of their response on the feedback form.

**Admin:**

* Admin will be an actor in this model which will have the required authorization access and control to add, modify or remove university details pertaining to the student feedback.
* Admin will have the access to view the dashboard (control panel) where the feedback information will be displayed in a coherent, comprehensible format

**Department:**

* Department entity fetches the feedback input from students as well as from other components (Transcripts to fetch student information, Course Information, Faculty details) in order to present it to the admin through the control panel (dashboard).

**SYSTEM WORKFLOW –**

* Student enters their respective login credentials(username and password) in order to login and access the feedback panel.
* Students will then be able to provide feedback on courses and faculty/professors through a rank-based format and submit their feedback.
* Admins can enter their respective login credentials(username and password) in order to login and access the control panel.
* Upon successful login, admins can then add, modify, or delete feedback details from control panel (dashboard).
* Admins can select department in order to survey the department wise rank of students, industry trend, faculty reviews, and industry trend in order to track student’s growth over a period of time though the university’s education system.

**SYSTEM ENTITIES –**

* **DEPARTMENT**

**Attributes:**

* DepartmentNumber
* DepartmentName

**Methods:**

* + **getStudentInfoByEmployStatus() –** This method sorts and returns student details on the basis of their employment status
  + **getStudentsByCoursesEnrolled()** – This method lists students on the basis of the courses they have enrolled themselves in

* + **calcProfRank()** – This method ranks faculty members and professors on the basis of the responses and feedback received from students.
  + **getHighRatedCourses()** – This method returns courses that are popular on the basis of the number of students enrolled in the course and the feedback received from students.
  + **getStudentDetails()** –This method fetches student details.
  + **getFacultyInfo()** – This method fetches faculty details
  + **getCourseInfo()** – This method fetches course details
  + **getGPA()** – This method fetches GPA of students on different courses to evaluate student performance
  + **calcFieldOfStudyTrend()** – This method projects field of study that the students can opt for based on their feedback, reviews and ranking.
  + **calcDeptRank()** – Method takes no of employed Students from Department Class and calculate the rank based on totalStudents/Employed Students ratio and average GPA.

* **COLLEGE**

**Attributes:**

* Name
* Location
* contactInfo
* deptList

**Method:**

* + **getDeptList() –** Method returns a list of department

* **COURSE CATLOG**

**Attributes:**

* courseName
* courseDetails
* courseStructure

**Method:**

* + **getCourseInfo() –** Method takes course name and returns details about the corresponding course

* **TRANSCRIPTS**

**Attributes:**

* coursesEnrolled
* GPA
* Major
* yearOfPassing

**Method:**

* + **calcGPA() –** Method calculates students GPA

* **STUDENT**

**Attributes:**

* studentId
* name
* employmentStatus
* careerlevel
* organization
* username
* password
* enrolledCourses

**Method:**

* + **rankProf() –** Method ranks professors based on student feedback
  + **rankCourse() –** Method ranks courses based on student feedback
  + **getCourseList()** – Method returns a list of courses

* **FACULTY**

**Attributes:**

* facultyName
* courseTaught
* facultyExperience

* **LOGIN**

**Attributes:**

* Username
* Password

**Methods:**

* + **validateCredentials() –** Method takes username and password from user (Student/Admin) and validates their respective credentials for successful login

* **ADMIN Portal**

**Attributes:**

* Username
* Password

**Methods:**

* + **viewControlPanel() –** Method to display Control Panel (Dashboard) to the admin
  + **updateCourseCatalog() –** Method to update course details
  + **addProfessor() –** Method to add professors
  + **addCourse() –** Method to add courses
  + **deleteCourse() –** Method to remove a course
  + **deleteProfessor() –** Method to remove professor
  + **updateProfessor()** – Method to modify professor details
  + **updateCourse()** – Method to modify course details

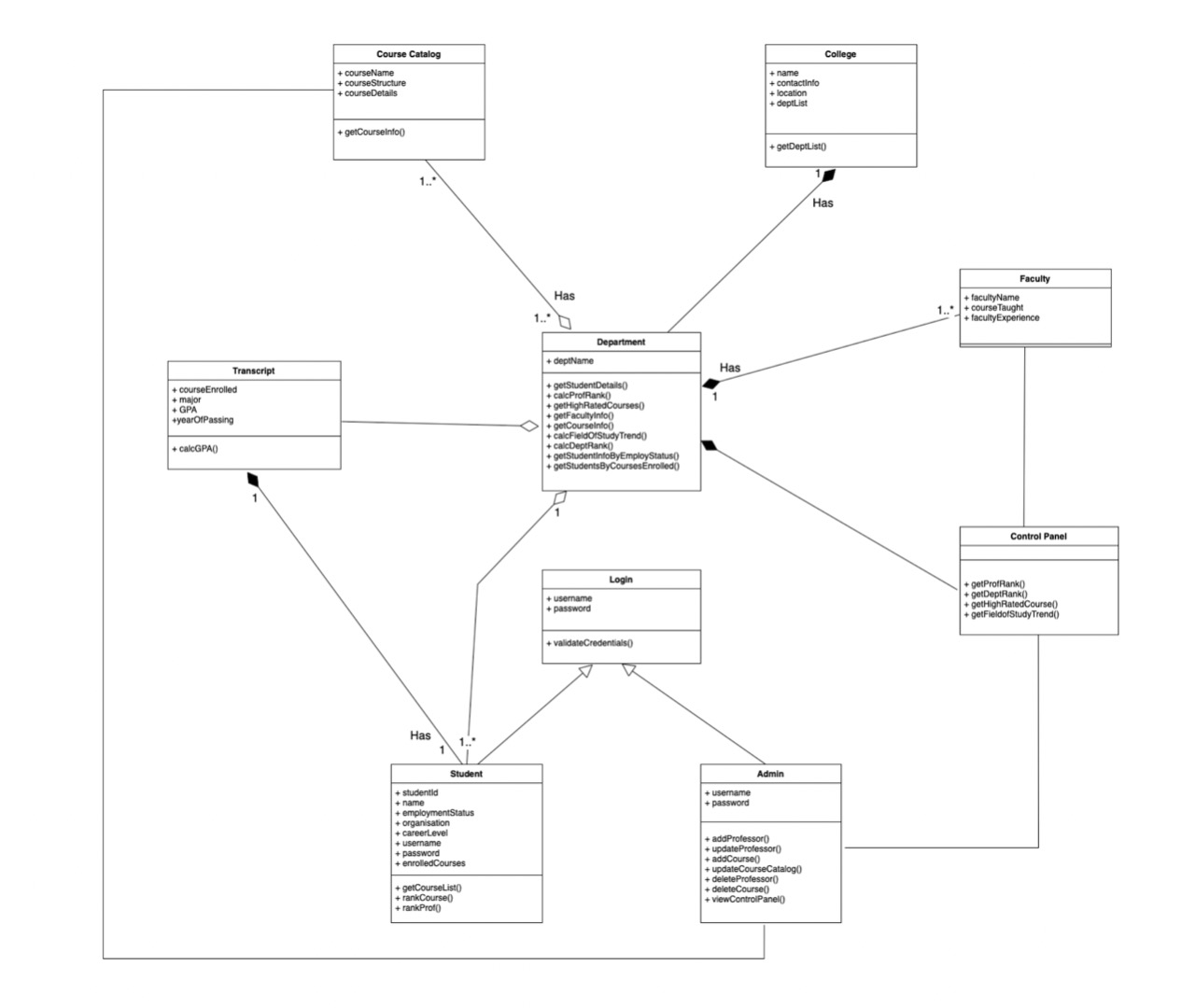
* **Control Panel**

**Methods:**

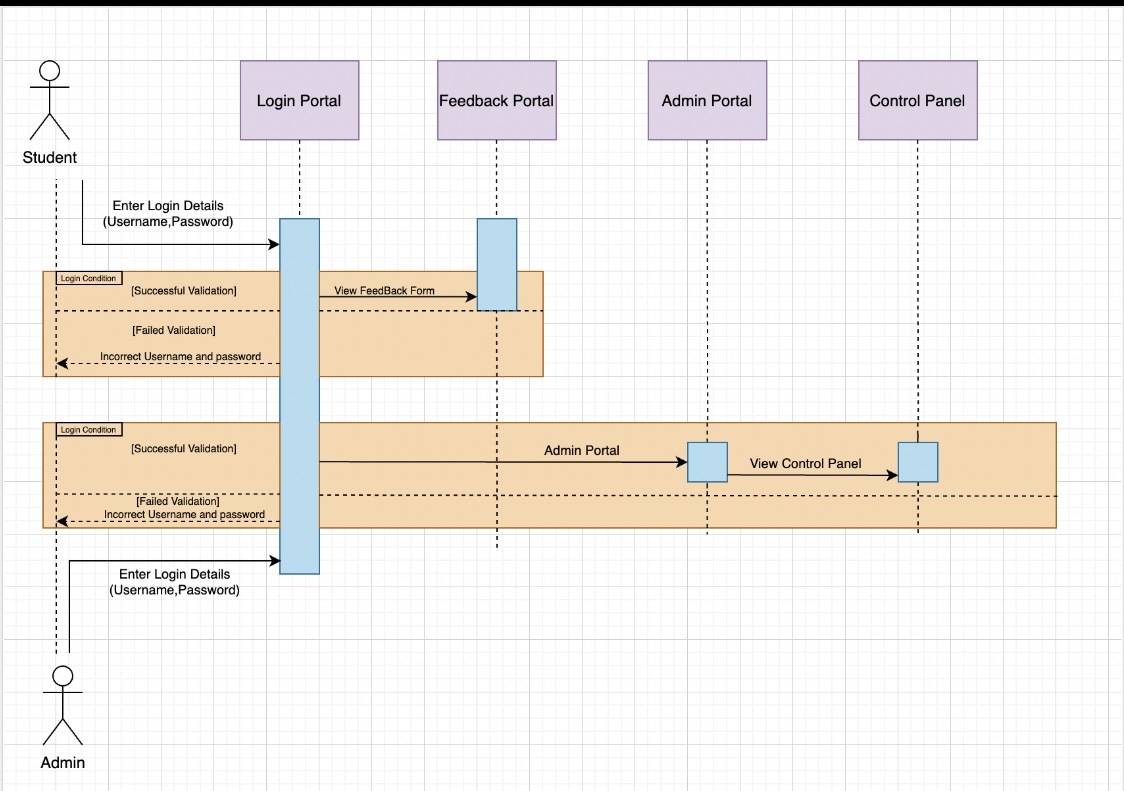
* **getDeptRank() -** return from department. calcDeptRank()
* **getProfRank()** - return from department.calcProfRank()
* **getHighRatedCourses()** - return from department.getHighRatedCourses()
* **getFieldOfStudyTrend()** – return from department.calcFieldOfStudyTrend()

**DIAGRAMS:**

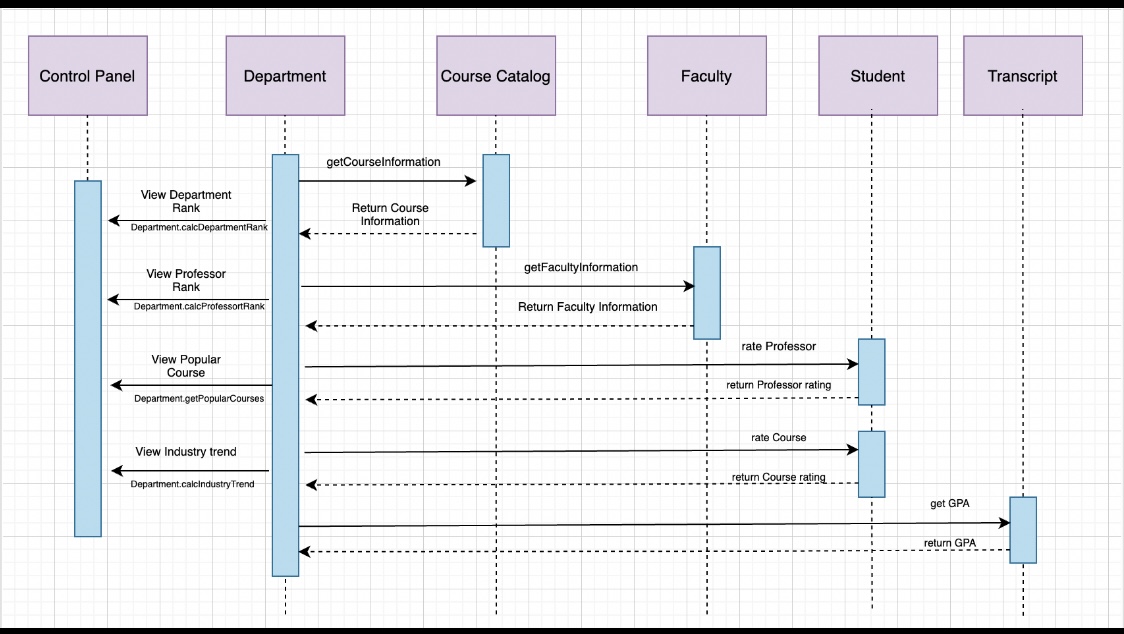
1. **Class Diagram**



1. **Sequence Diagram 1**

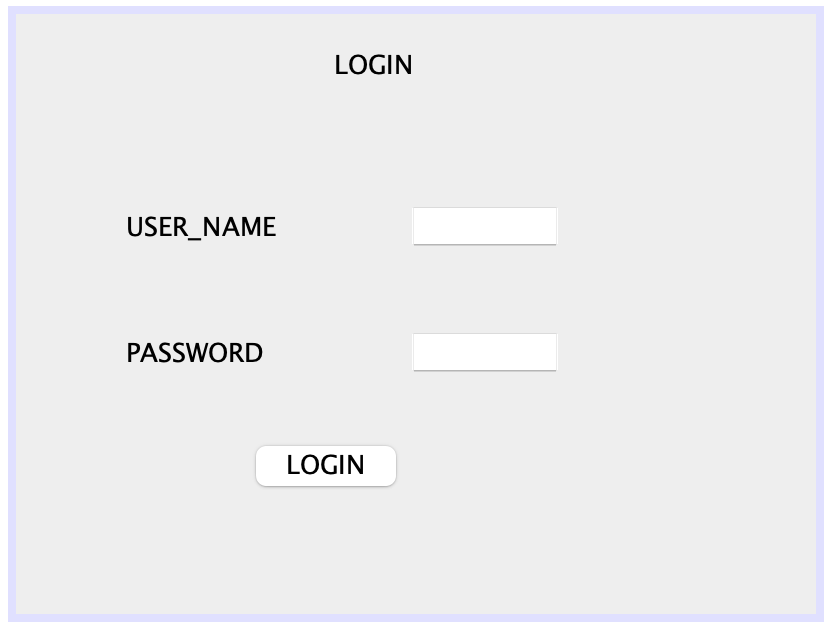


**3) Sequence Diagram 2**

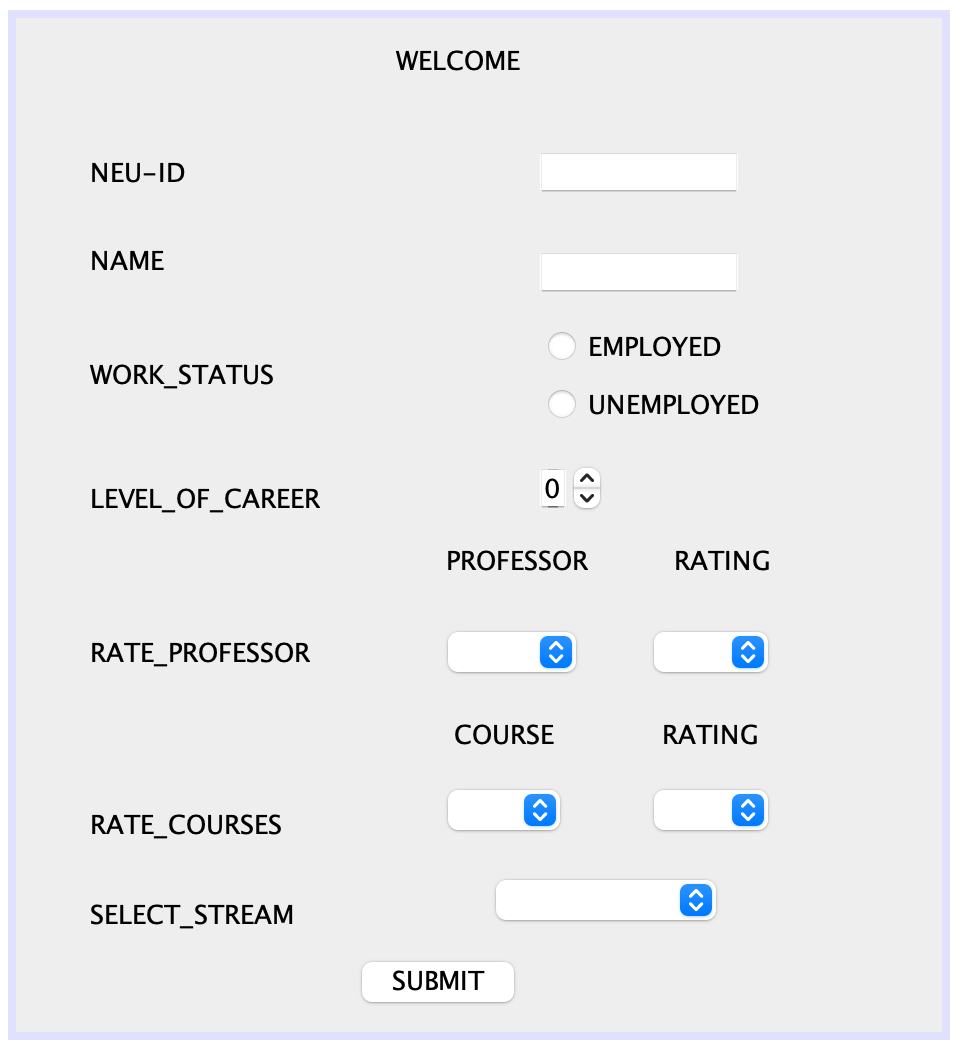


**INTERFACE:**

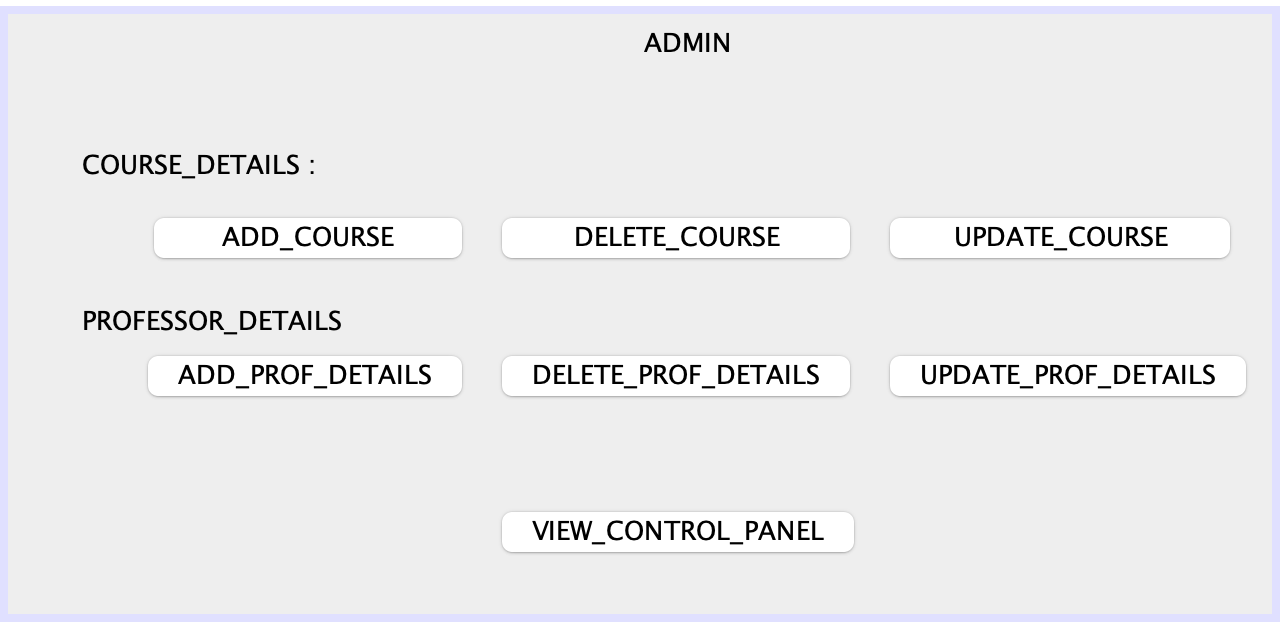
1. Login Page



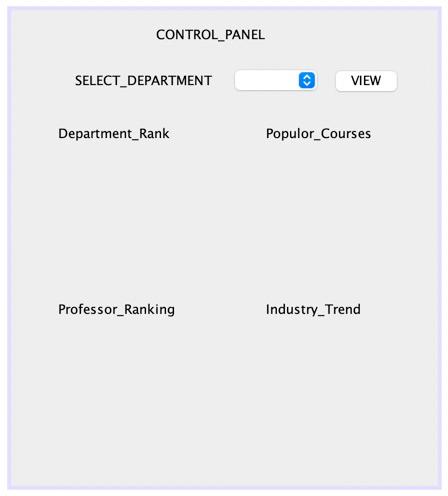
1. Feedback Form



1. Admin Portal

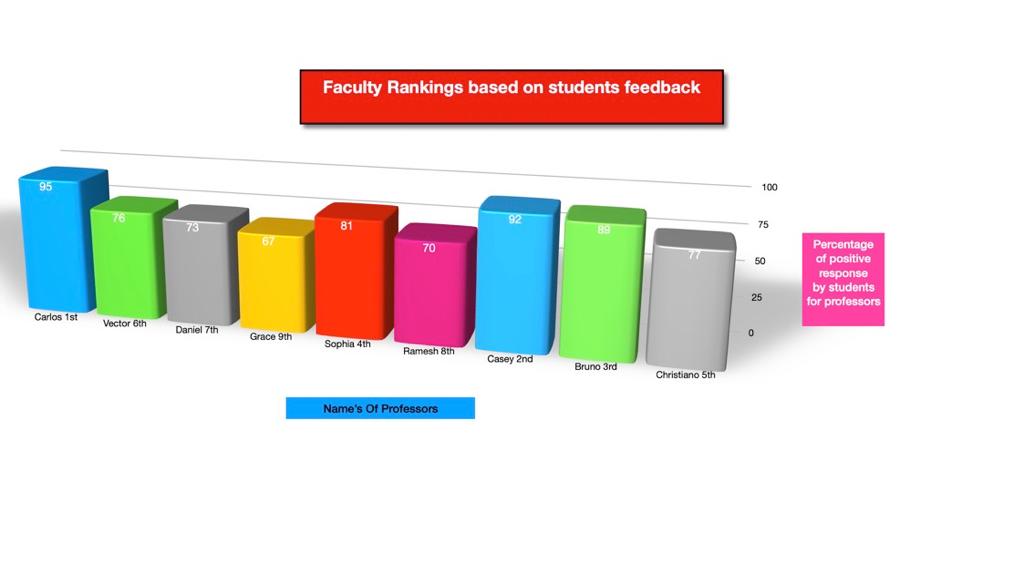


1. Control Panel

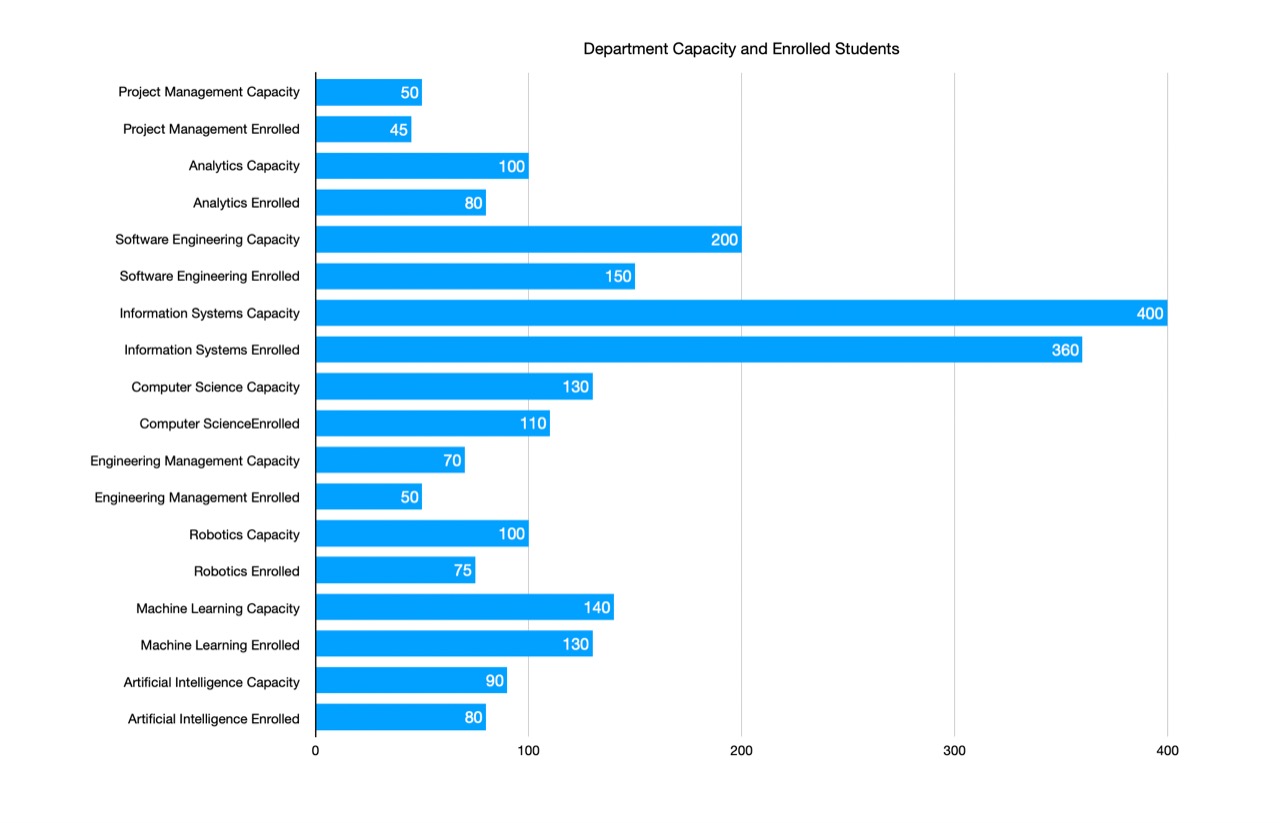


**CONTROL\_PANEL GRAPHS:**

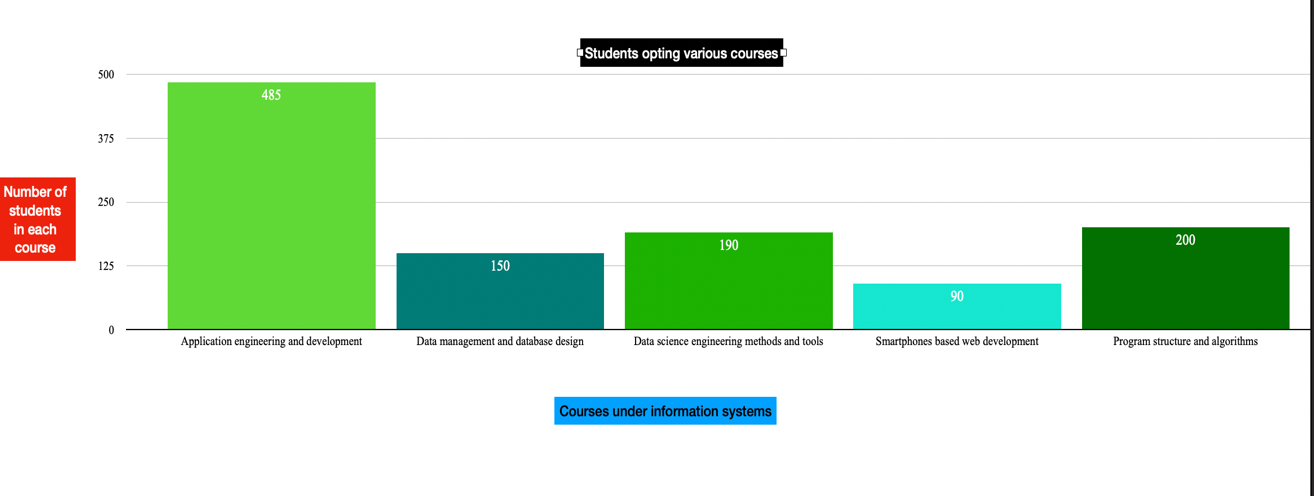
1. **Department Rank**



1. **Professor Ranking**



1. **Popular Courses**



1. **Industry Trend**

