Assignment 3

Group Members - Aditi Jain, Prerana Urs Praveen Kumar Urs, Suhani Shah

I. Problem Statement

To create a performance measurement solution to enable universities to measure the quality of the education they deliver to their students

II. Purpose

To learn the techniques for turning an object model into a machine for information gathering and data aggregation

III. Proposed Solution

We propose a solution that enables the university to measure the quality of education delivered by them, using the following performance metrics (considering the weightage of each factor)

a. Student Feedback - Coursework, faculty
b. Jobs bagged by the students in the current semester/year
c. General feedback - alumni and employers.

The above mentioned metrics are described as follows-

a. **Student feedback** - Students have to provide feedback for their respective courses and faculty. This will indirectly help the university to understand different problems students are facing and take necessary measures in the future.
b. **Jobs bagged by the students** - Provides the rate of students who could get a job as compared to the total students enrolled under the same department.
c. **General Feedback** - Department sends out invites to alumni and employers requesting for Feedback. Alumni and employers are to give feedback to the associated departments.

We identify the following people as responsible for providing good quality of education in our model :

1. **University** - Provides the right environment for the growth of its students keeping in mind the effective education system that can be implemented.
2. **Faculty** - Teaches various courses under different departments to the students.
3. **Students** - Studying in the university, responsible for their coursework, enrolling in courses, completing coursework and related activities. Applying for jobs and getting a job by the end of the course.

How to improve education and growth of their graduates?

The approach will be to look into how an educational system in terms of faculty and courses contribute to the growth of their graduates over a 5-year period.

1) Introducing different Modes of Teaching - *Online* and *In-person* classes. This can be implemented by introducing a connection between student and university objects.
2) Dividing the Faculty based on different departments in the university
3) Setting up a Course Schedule for each Department
4) Dividing Faculty based on the mode of teaching under the same Department
5) Providing workshops for online teaching faculty for smooth running of online classes
6) Providing workshops for Students who are in need of additional knowledge in a Subject
7) Managing Student/Faculty/Course records in the University portal
8) Administering a University portal to maintain different categories of search in-order to ease the process for both Students and Faculty.

Additional features that can be accommodated by the University to improve the education quality :

- Make up classes conducted by Professors
- Co-op opportunity to Students
- Workshops on soft-skills for Students
- Informative Speaker sessions by the Alumni

Our own ranking system for students to decide where they want to go for their studies or not will be based on the ratings we will calculate.

IV. Entities

1. **University :**

- List of departments
- View & Update functions accessed only by Faculty
- On view of each department, one can see the courses offered by this department
- Final Performance metrics is calculated by considering the weightage of each factor
    a. Student Feedback - On Course, faculty
    b. Jobs bagged by the students in the current semester/year
    c. General feedback - By Alumni and Employers on respective Department

2. **Student :**

- Class selection - list of classes available for students based on the semester
- Course Enroll - list of courses, type of course(online or in-person), their respective teaching faculty and feedback rating for the same
- Grades - Final GPA, Class schedule
- Feedback to Course and Faculty
- Co-op feedback
- Scholarship

3. **Faculty :**
    - View & Update Faculty profile
    - View & Update Course syllabus
    - View current teaching courses
    - View enrolled Students list
    - Class schedule for each Faculty

4. **Department :**
    - List of Degrees
    - Each degree has a list of offered courses
    - Rating of department based on feedback by students' and employers' rating
    - List of companies associated with the department - Employer Directory
    - Mandatory course enforced by individual department
    - Scholarship programs

5. **Employer :**
    - Linked to Department based on different job roles available for the students to apply
    - Availability of job role
    - Requirements for applying for the job
    - Student's possible Date of joining
    - Provides feedback on syllabus, co-op student and employed students from the same university for over 5 years

6. **Courses :**
    - Different of courses offered during the particular semester and course timing, teaching faculty, Name, credit hours, seats, minimum/max credits allowed per sem

7. **Domain :**
   (e.g Data scientist, software engineer roles)
   To track the connection of courses and their relevance to graduates' growth.
   - List of courses
   - Student’s preferred Domain name
   - Course wise priority
   - Number of students who got jobs in a specific domain

8. **Feedback**:
   This entity stores the feedback for different Courses and Faculty. This is used to calculate the total rating.

9. **Job Portal**:
   This dashboard displays different jobs available, from which students can view listings and apply from this portal. It updates the results of students who have got the job through the portal. This portal informs respective departments about the count of students who bagged jobs in various timelines.

10. **Alumni**:
    Alumni profiles are stored in this entity. Alumni can login to the system and update their current data related to their job profile, company they’re working for presently and other achievements.

11. **Curriculum**:
    Stores the list of Courses under a specific department based on the specialisation opted by the student.

12. **Academic**:
    This entity stores the academic information of students based on different courses opted by them. Here, transcripts are included to calculate the total GPA of each Student.

V. Design

Our application comprises the following buttons on the main screen:

University, Student, Faculty, Department & Employer

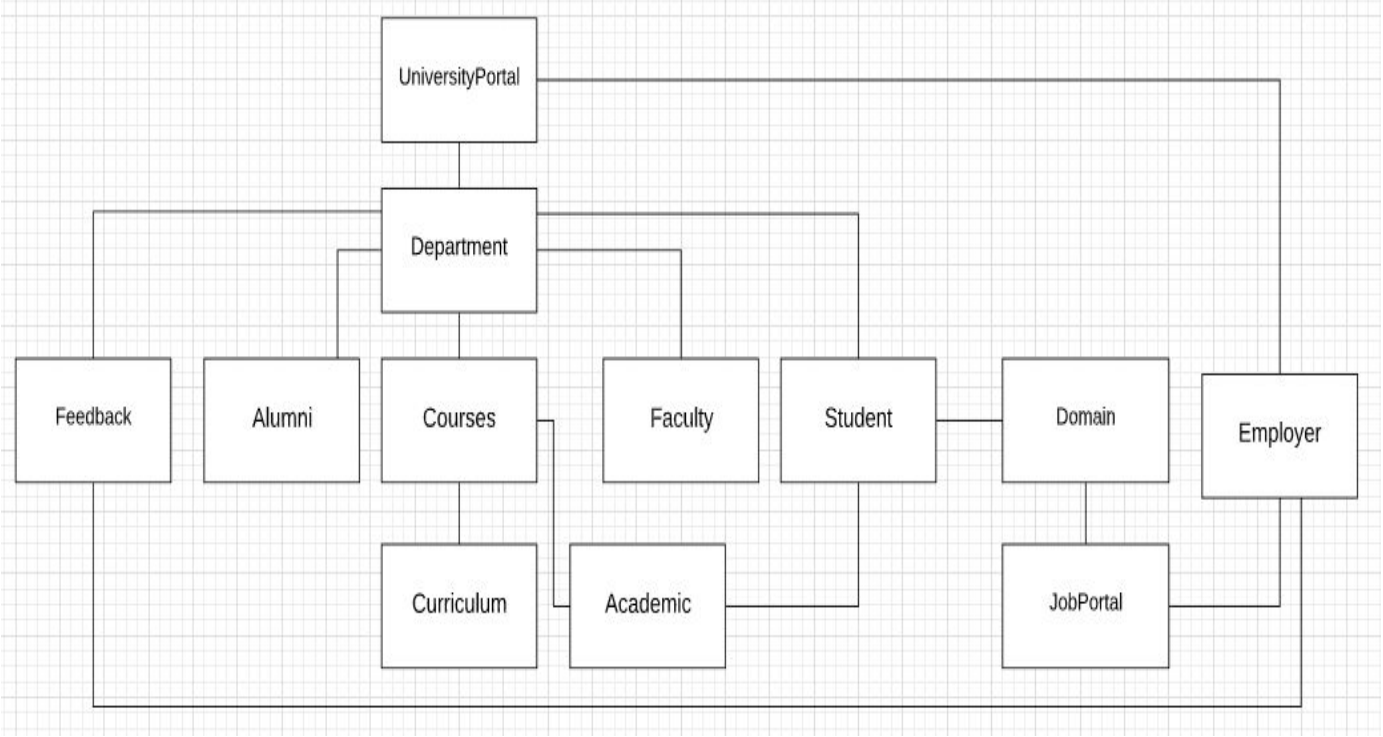
- On clicking the **University button**, university related functionalities as mentioned in the entities section, gets called. The Final Performance metrics button will be placed in the university dashboard to view the various performance measures taken into consideration to calculate the metrics. The University can manage different departments and finance information.
- On clicking the **Student button**, students can login to the system, register for courses, track courses, track progress & grades and also apply for jobs on the job portal. Students will have to give feedback for the courses opted and its corresponding professors. Based on these feedback, the performance ratings will be calculated.
- On clicking the **Faculty button**, faculty logs into the system, manages his/her courses, syllabus & faculty schedule per semester. They can also view the list of students in their class for that semester. Faculty can give feedback to students based on student's performance as well.
- On clicking the **Department button**, the department gets to manage students, faculty, and courses offered. Department sends invites to students, alumni and employers to get feedback.
- On clicking the **Employer button**, employers can login into the Job Portal panel and give relevant feedback to the department about the students and their performances in the interviews. Also, employers can manage a job posting page that appears on the job portal and is constantly updated based on the trend in Technology.

Basic methods() that can be implemented in our System are -

1. selectCourse()
2. selectFaculty()
3. giveFeedback()
4. getEnrolled()
5. calculateGPA()
6. applyJob()
7. getJobList()
8. getJobCount()
9. calculateTotalFeedback()
10. calculateJobCountMetrics()
11. calculateFeedbackMetrics()

VI. Object Model Diagram

The object model shows different objects interacting with each other. Various objects are connected with each other to get the desired functionality and to generate the required information. Please refer to the below diagram for our solution -



VII. Sequence Diagram

1. **Feedback gathered from Students, Alumni and Employers**
   Please refer Figure 1

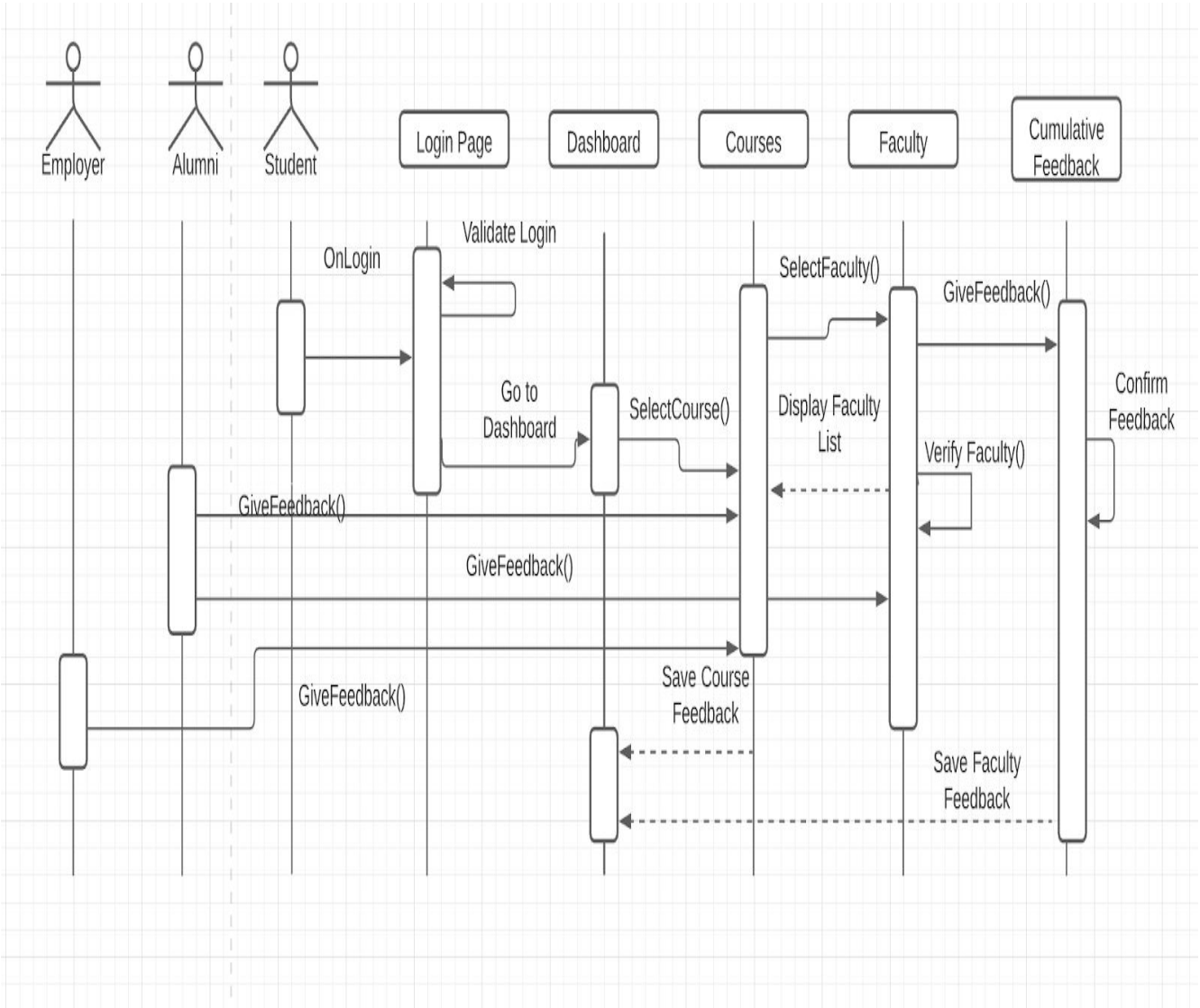


Figure 1- Sequence diagram for feedback gathering.

2. **Jobs bagged by the students**
   Please refer Figure 2

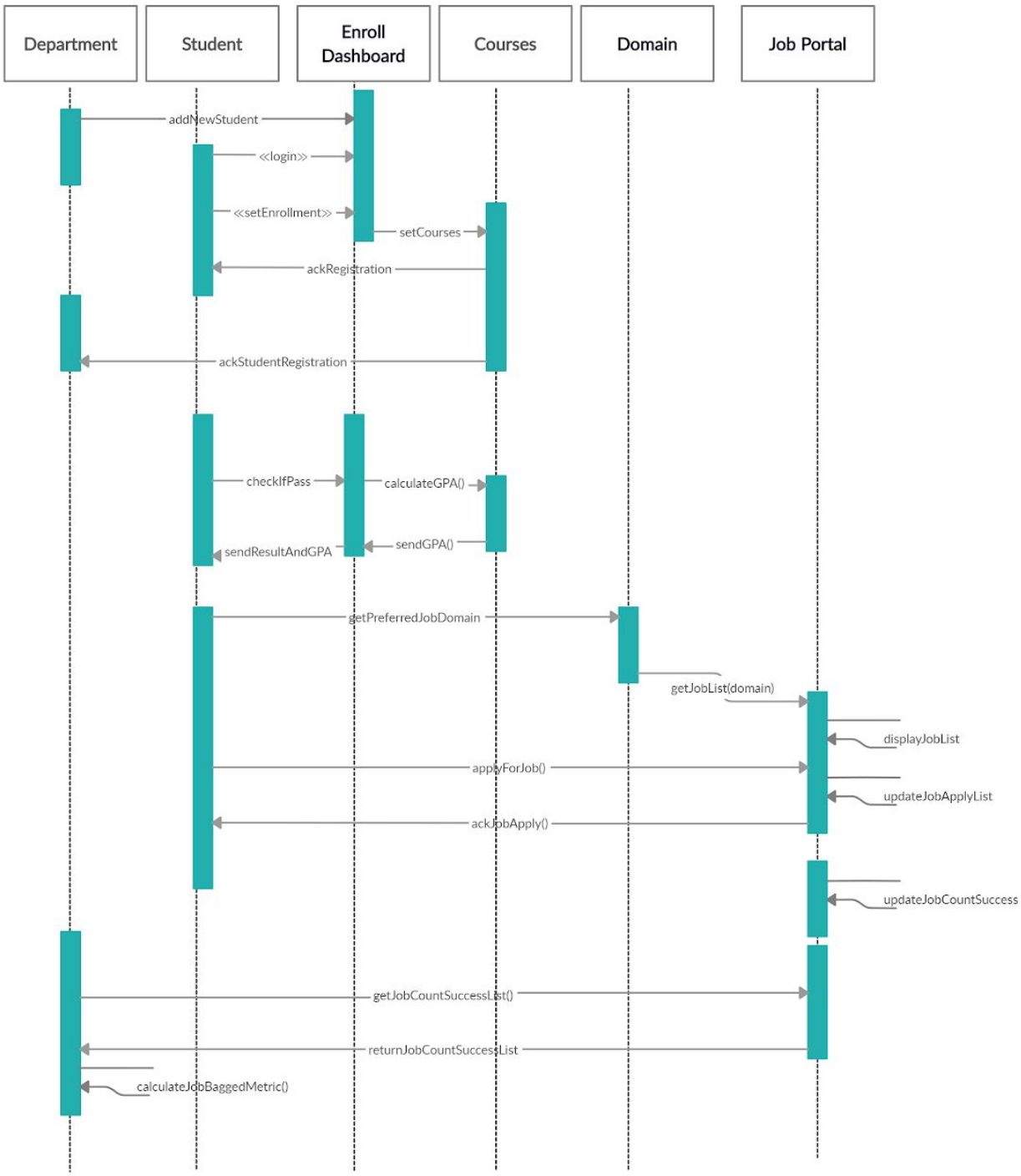
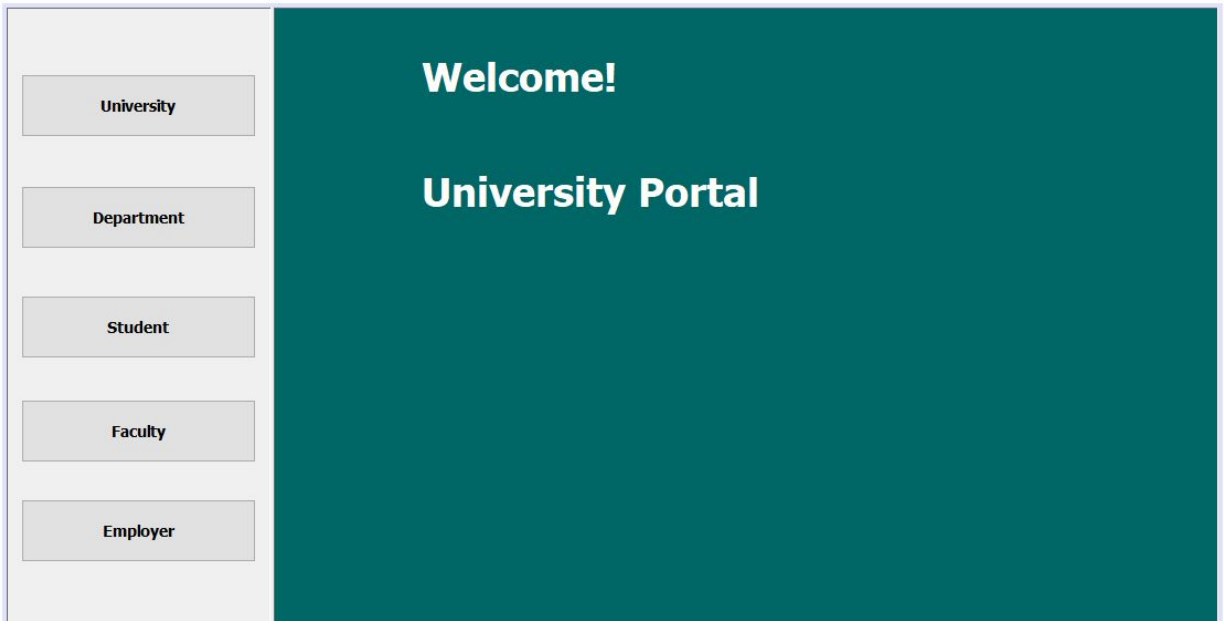


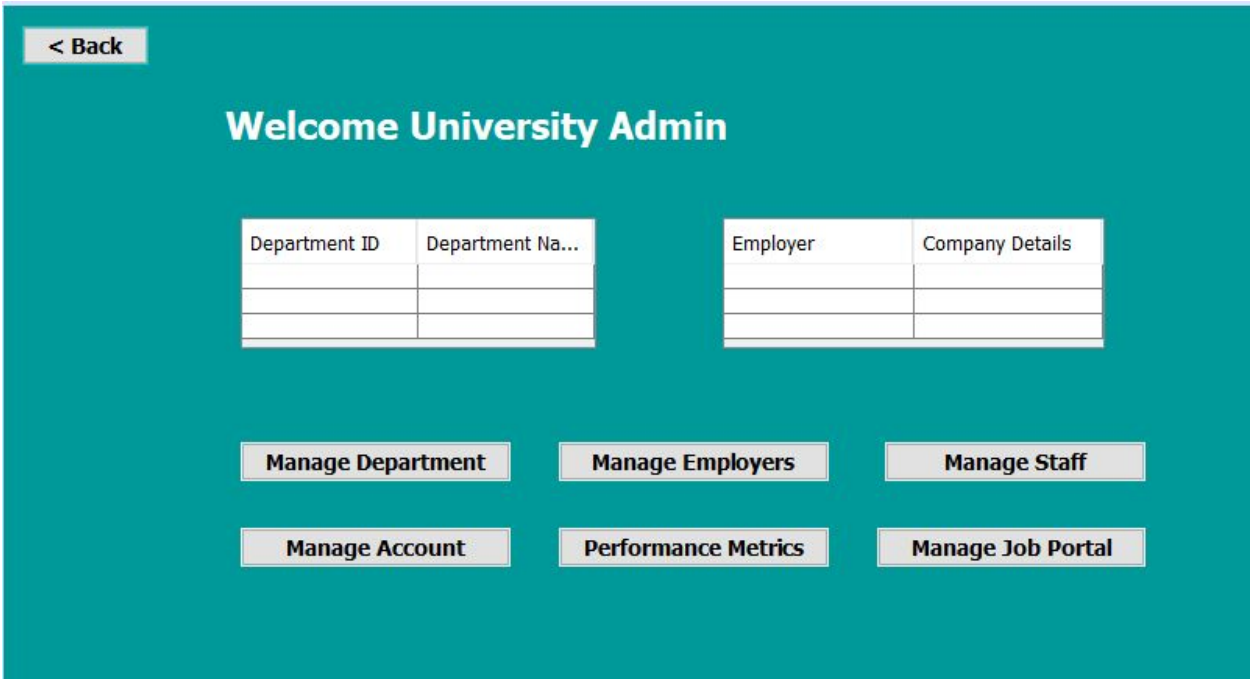
Figure 2- Sequence diagram for metric - job bagged by students

VIII. Dashboard Design (UI Basic Design)

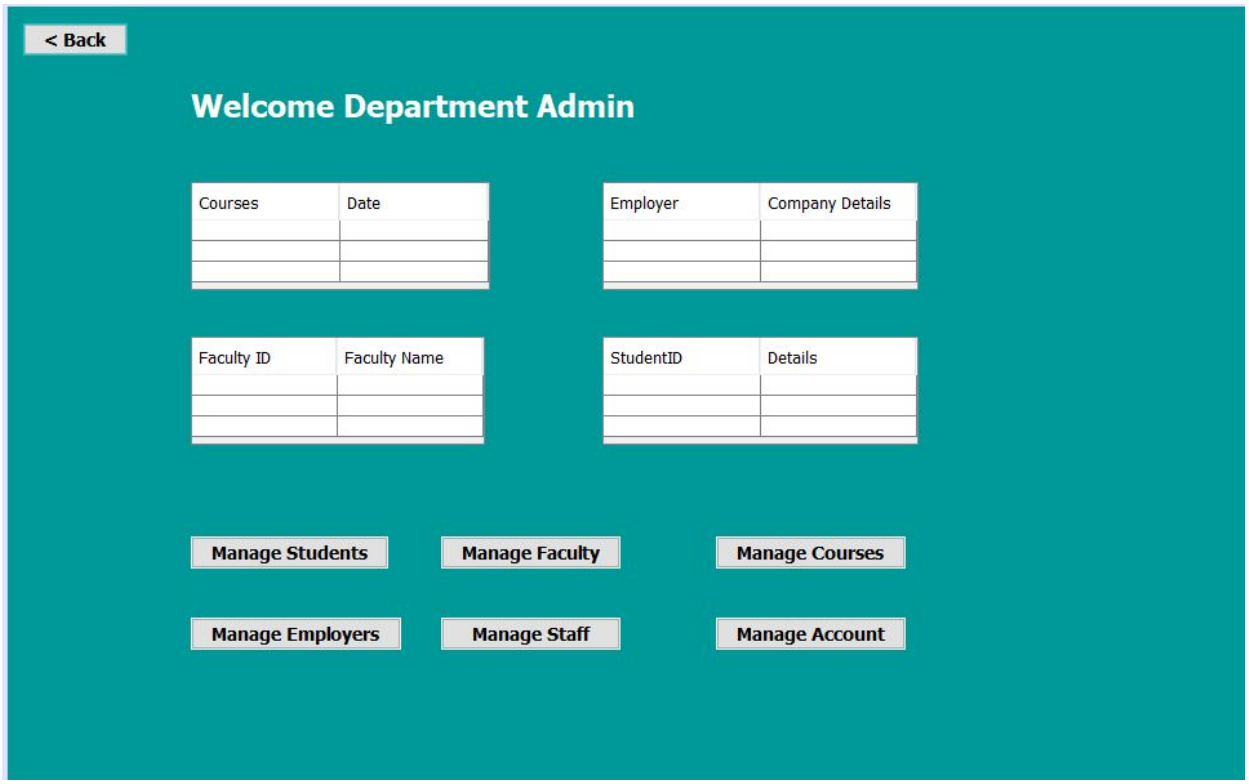
Main page :



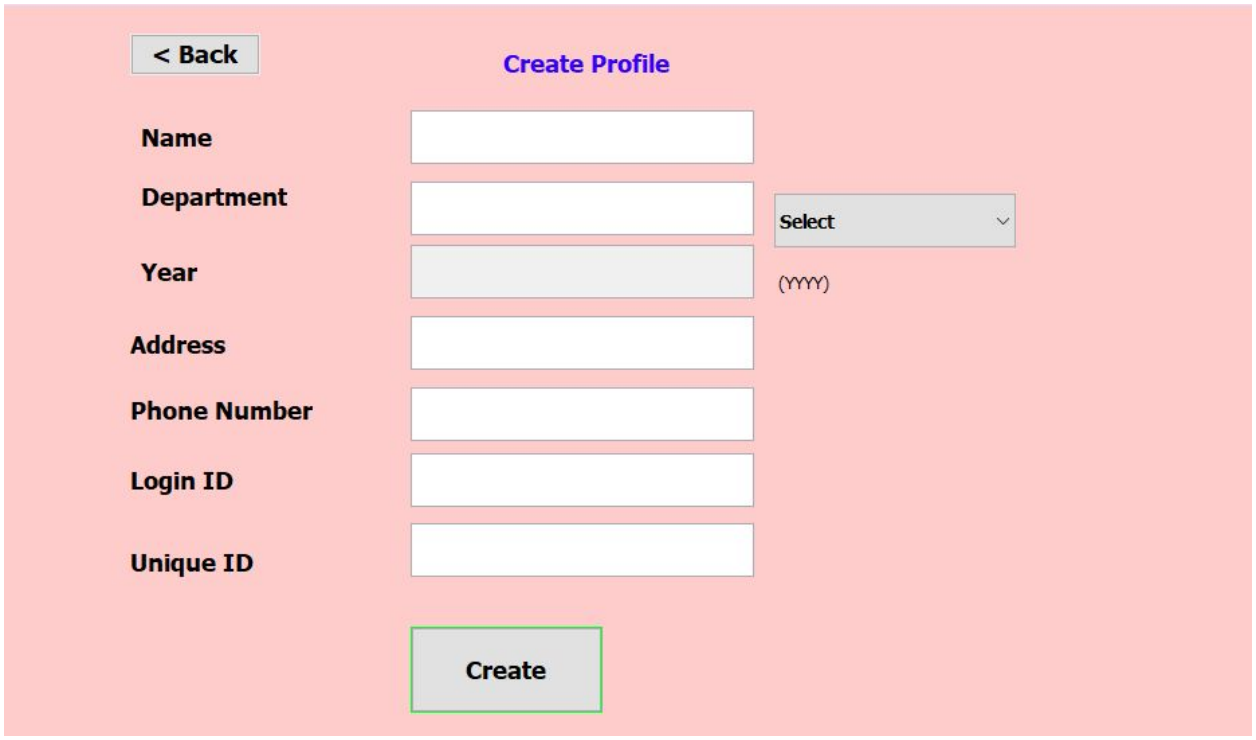
University Dashboard :



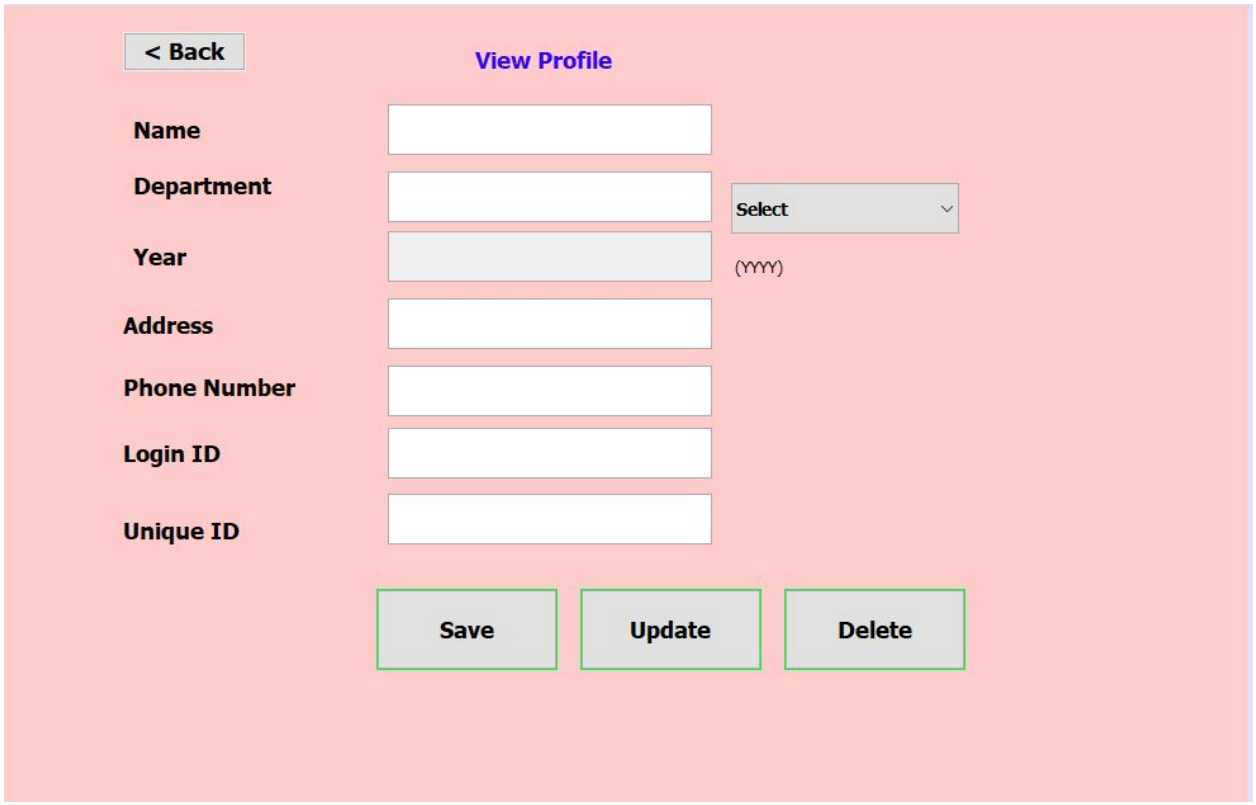
Department Dashboard :



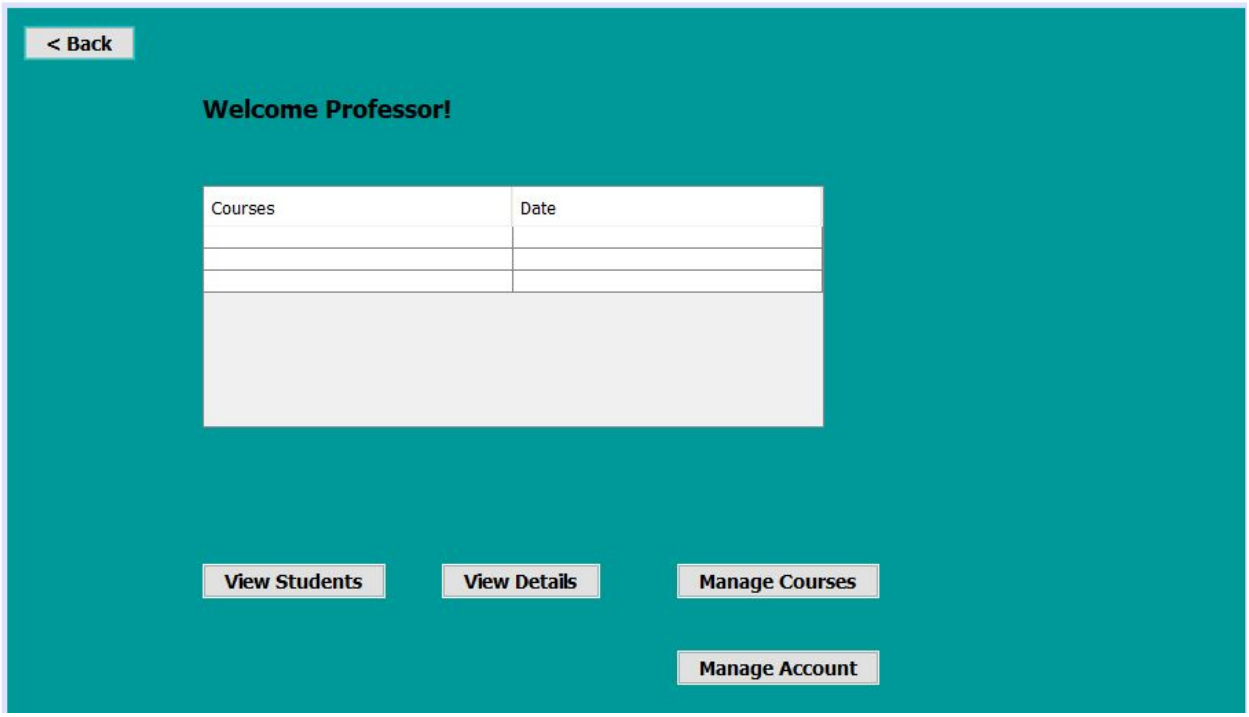
Profile Creation :



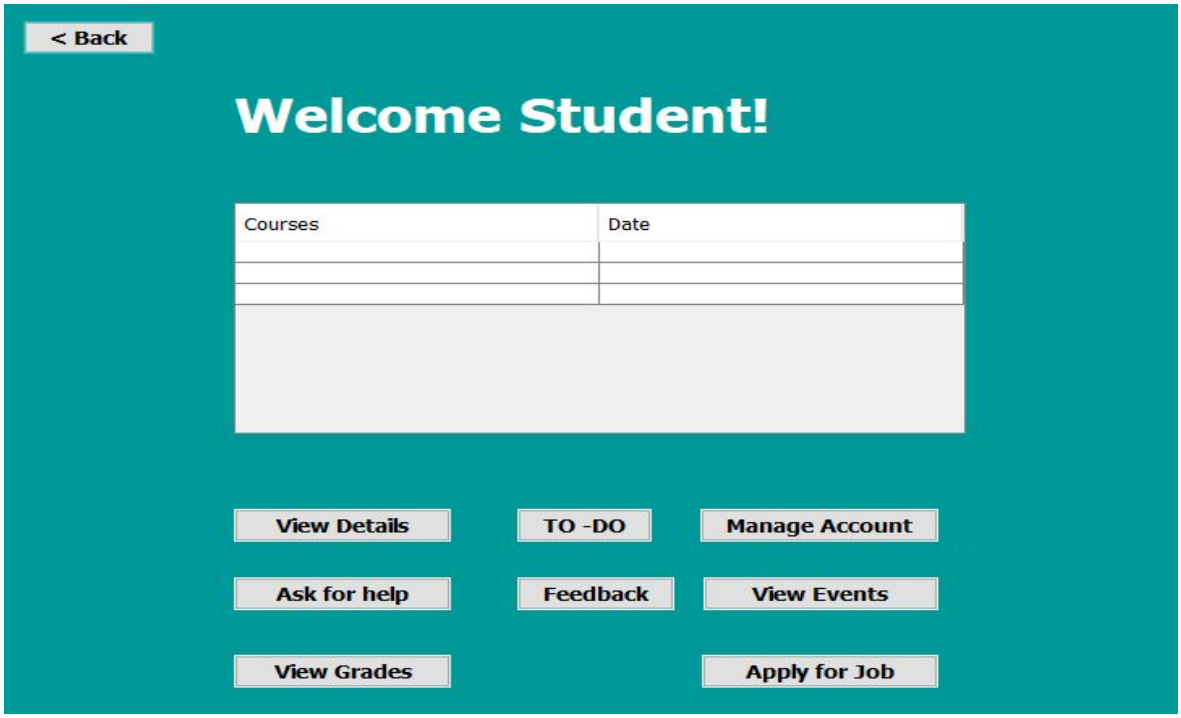
Manage Profile :



Professor Dashboard :

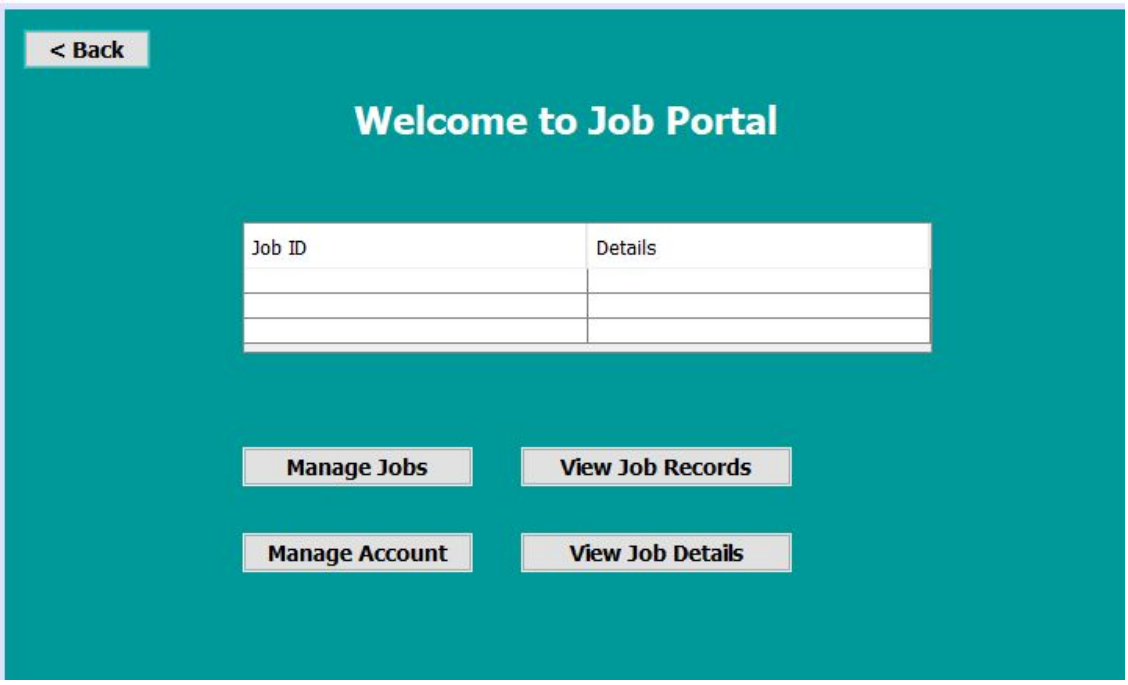


Student Dashboard :



Feedback :

Job Portal :



Employer Dashboard :