University Model Design Document

Objective:

1. To explore techniques of turning an object model into an application for information gathering and data aggregation by using software engineering techniques.
2. To study ways to create a performance measurement solution to enable universities to measure the quality of the education they deliver to their students to improve the quality of education through learning to learn and feedback.
3. To measure the educational system of a university in terms of faculty and courses in terms of growth of their graduates over a 5-year period.
4. Design a dashboard that enables college and university administrators to compare the performance of their academic units.

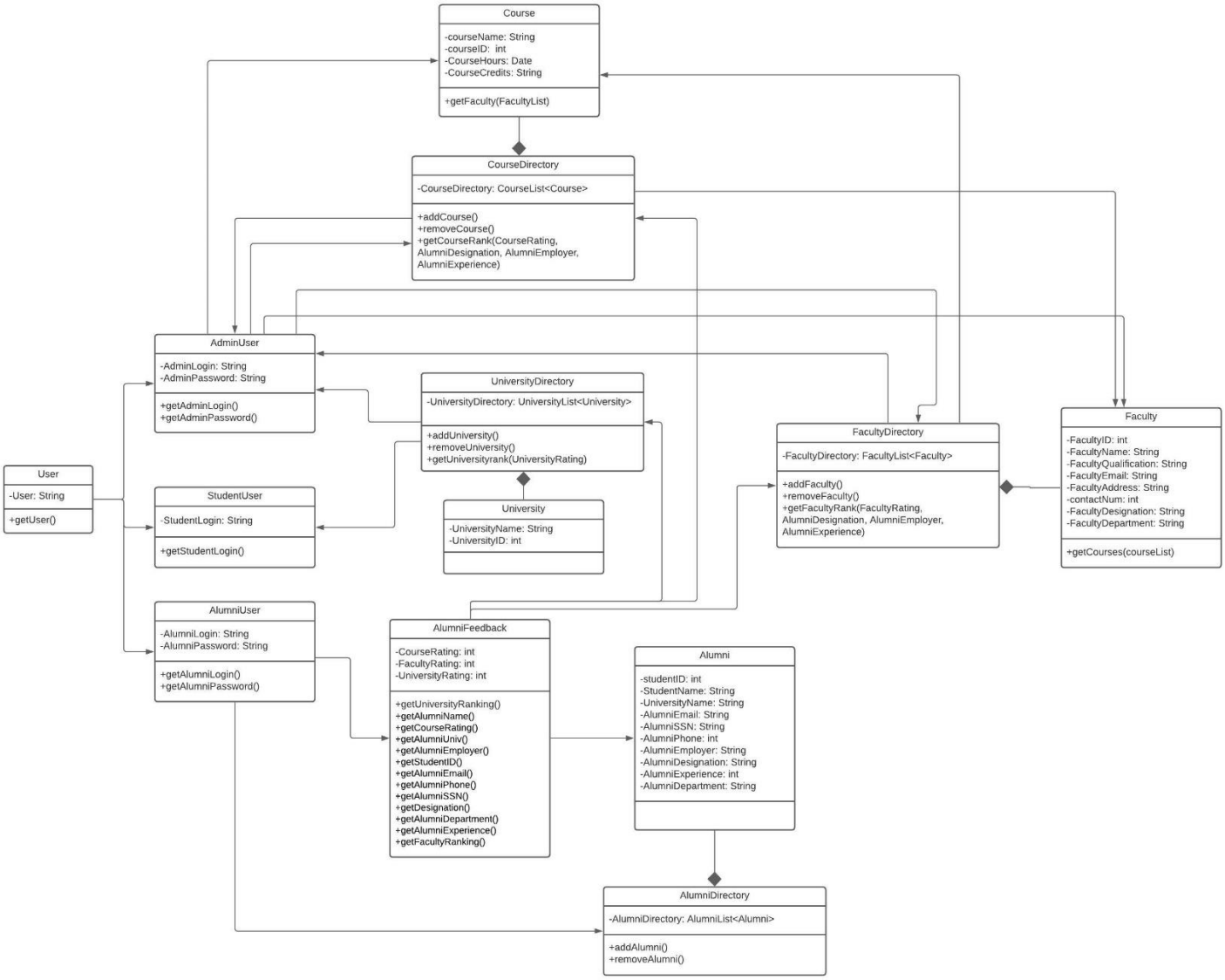
Deliverables:

1. Develop a Java program to show the implementation of the complete system.
2. Presentation showing outline and implementation.
3. Sequence diagrams showing how to navigate the university object model to deliver performance metrics needed for performance and feedback.
4. An object model to develop university model with relevant attributes to evaluate and improve the education system to deliver appropriate results
5. Your application must enable the creation and update functions for any of the attributes of concern.

Abstract:

Unified Modeling Language is used for this application to develop a standardized way of visualizing and evaluating the current education system for data aggregation and information gathering. Based on the analysis and comparison of different courses and faculty, a rank is assigned based on the growth of Alumni, of any program with a set of courses and faculty, over a period of five years. Based on this information and the feedback given by Alumni, decisions are made on an administrator level to modify the course structure and add improvements. Additionally, this can be used to showcase how the courses and faculty have helped graduate students achieve a successful career over a period of five years.

Class Diagram:



Class Description:

User:

1. Defines the user on the dashboard. There are 3 types of users:
    a. Admin
    b. Student
    c. Alumni

2. getUser(): This function is invoked on the dashboard to assign the type of user to display the relevant page

AdminUser:

AdminUser is the Administrator. He/She is responsible for updating all information about the university, courses offered and the respective faculty.

1. getAdminLogin(): Gets the login details of the Administrator
2. getAdminPassword(): Gets the password of the Administrator.

StudentUser:

Refers to prospective students looking for relevant academic programs to apply for.

1. getStudentLogin(): Gets login details of the student.

AlumniUser:

Refers to the Alumni who have graduated from a university. They are eligible to give feedback of the course offerings and faculty.

1. getAlumniLogin(): Gets the login details of the Alumni.
2. getAlumniPassword(): Gets password details of the Alumni.

Course:

Refers to the courses offered. Includes course name, ID, credits offered and number of hours.

1. getFaculty(FacultyList): Gets the relevant faculty list of a particular program/course.

CourseDirectory:

Refers to the list of courses offered. One can add, delete and update courses.

1. addCourse(): Add courses to the list.
2. removeCourse(): Removes courses from the list.
3. getCourseRank(CourseRating, AlumniDesignation, AlumniEmployer, AlumniExperience): Gets rank of the Course based on four factors – Course rating given by the Alumni, Designation of the Alumni over the 5-year period, Employer of the Alumni and the number of years of experience.

FacultyDirectory:

Refers to the list of faculty teaching the courses.

1. addFaculty(): Add faculty to the list.

2. removeFaculty(): Removes faculty from the list.
3. getFacultyRank(FacultyRating, AlumniDesignation, AlumniEmployer, AlumniExperience): Gets rank of the Faculty based on four factors – Faculty rating given by the Alumni, Designation of the Alumni over the 5-year period, Employer of the Alumni and the number of years of experience.

Faculty:

Refers to the profile of each individual Faculty.

1. getCourses(CourseList): Gets the relevant course list of a particular faculty.

Alumni:

Refers to the profile of each individual Alumni.

AlumniDirectory:

Refers to the list of Alumni from the University.

1. addAlumni(): Add alumni to the list.
2. removeAlumni(): Removes alumni from the list.

AlumniFeedback:

1. getUniversityRanking(): Alumni ranks the the university based on his experience and career growth

2. getAlumniName(): Alumni Name

3. getCourseRating(): Alumni rates the course.

4. getAlumniUniv(): University of the Alumni.

5. getAlumniEmployer(): The company where the Alumni is employed.

6. getStudentID(): Student ID of the Alumni.

7. getAlumniEmail(): Email of the Alumni.

8. getAlumniPhone(): Phone number of the Alumni.

9. getAlumniSSN(): SSN ID of the Alumni

10. getDesignation(): Designation of the Alumni at the time of login.

11. getAlumniDepartment(): Department where the Alumni studied.

12. getAlumniExperience(): Number of years of work experience at the time of Login.

13. getFacultyRanking(): Rank of the faculty given by the Alumni.

University:

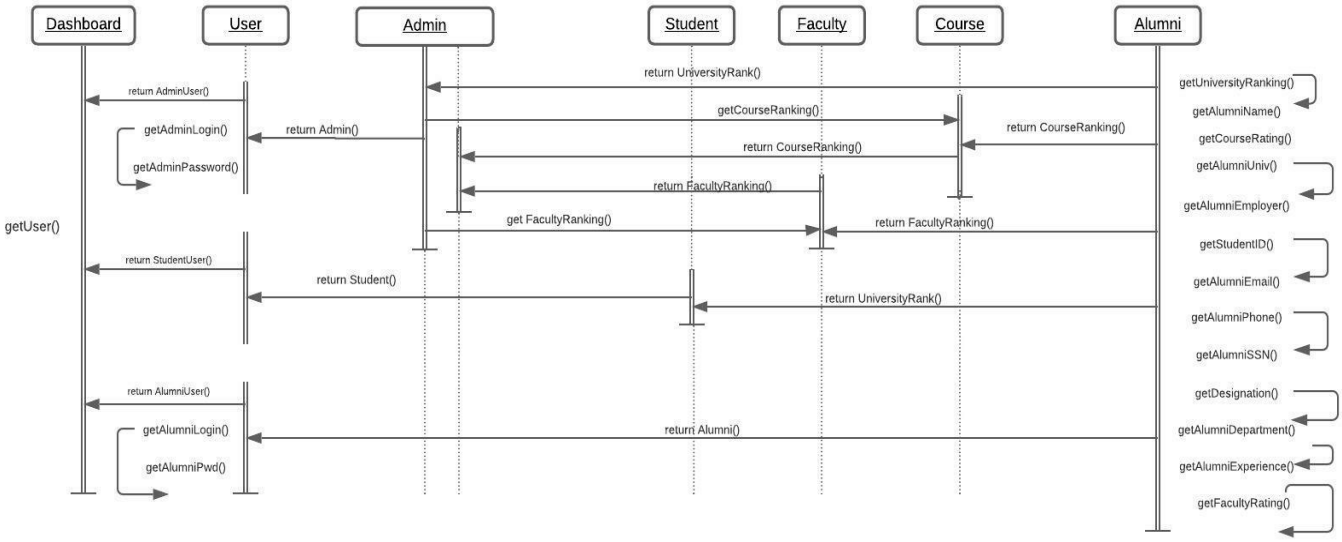
Refers to the University attended by the Alumni.

University Directory:

Refers to the list of Universities. One can add, delete and update universities.

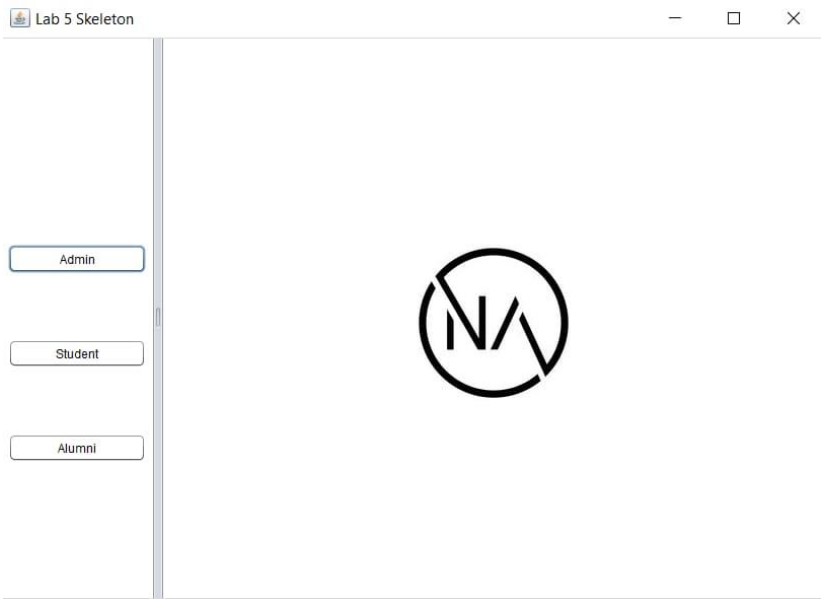
1. addUniversity(): Add courses to the list.
2. removeUniversity(): Removes courses from the list.
3. getUniversityRank(UniversityRating): Gets rank of the University based on rating given by the Alumni.

Sequence Diagram:

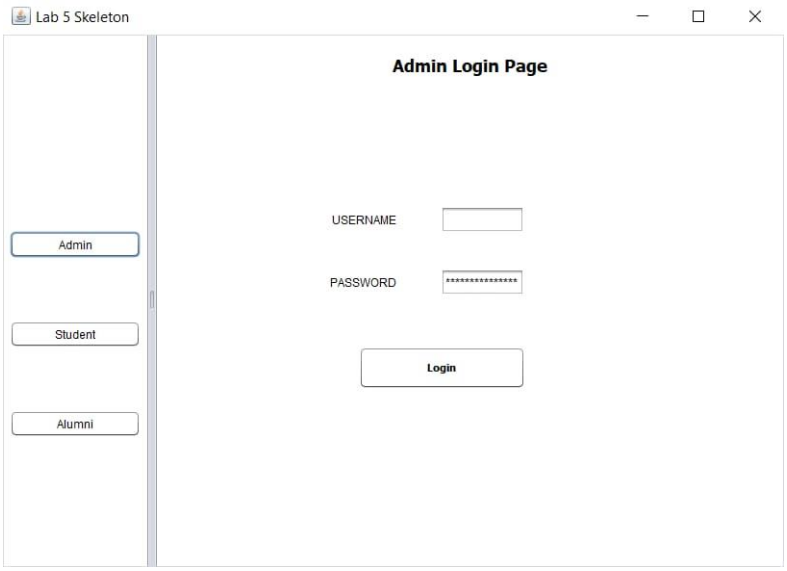


UI Diagrams:

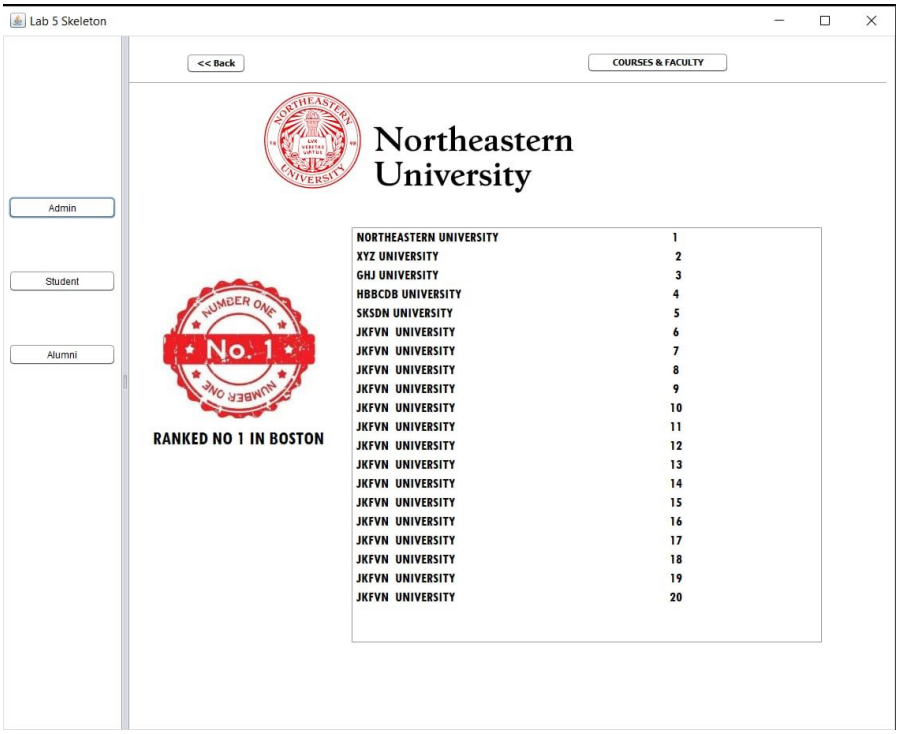
1. User page:



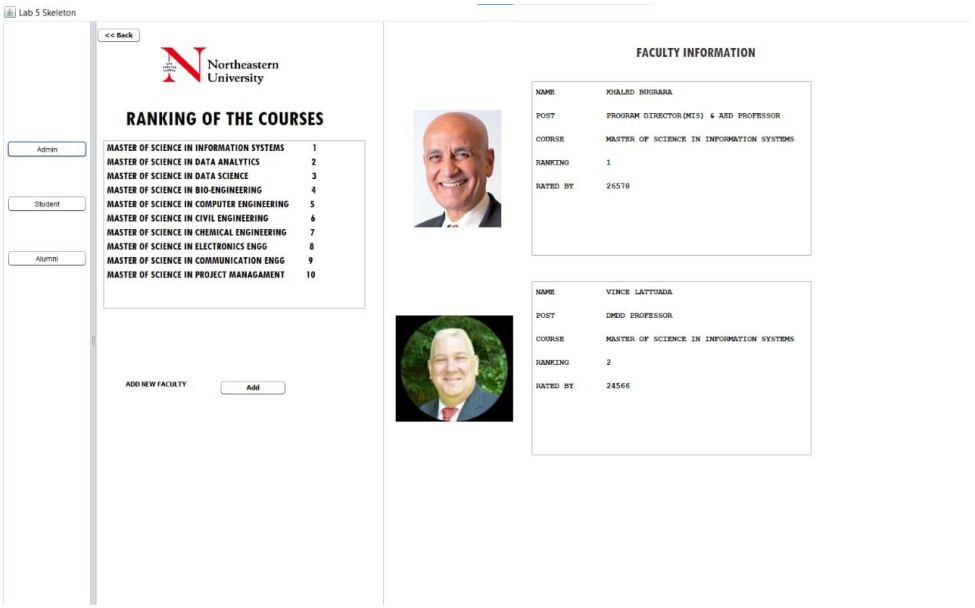
2. Admin Login



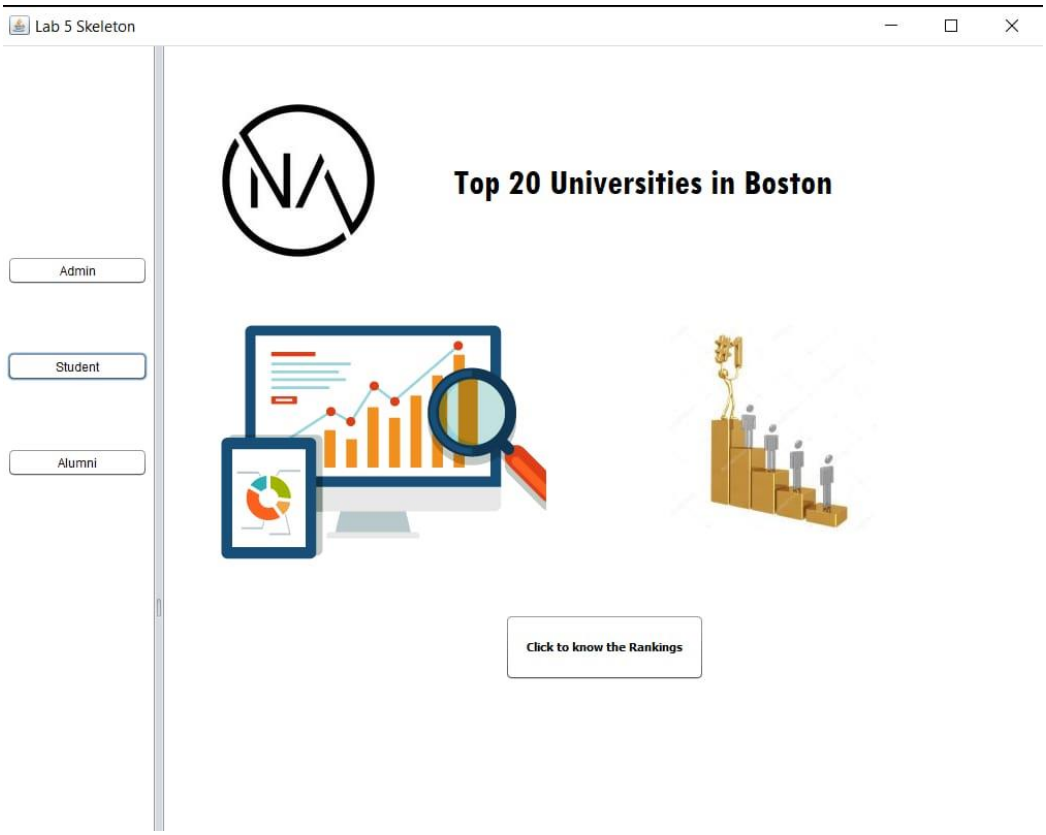
3. Admin Page



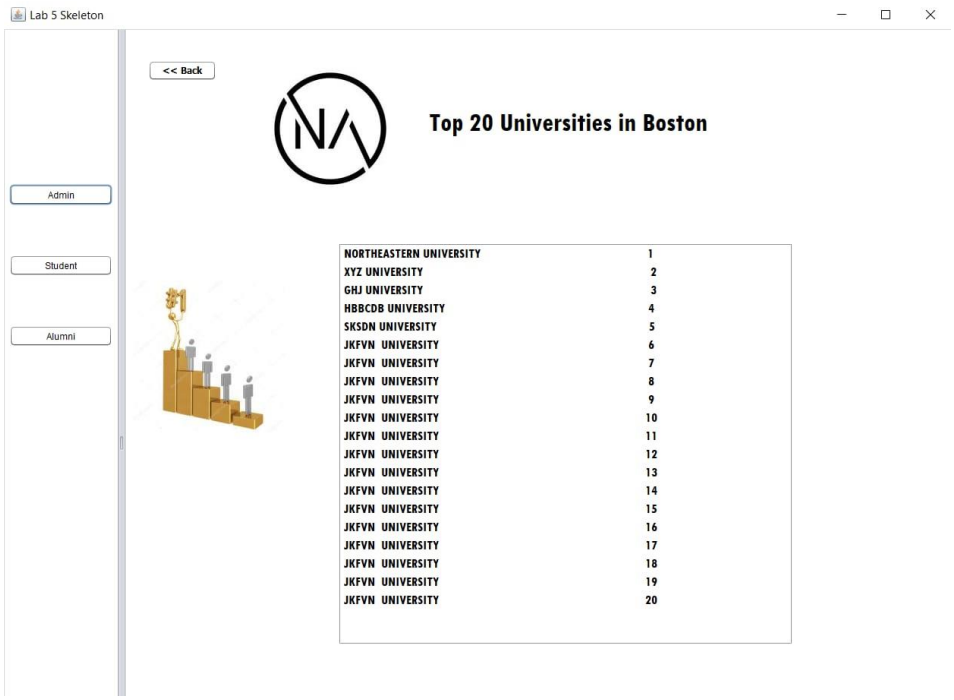
4. Course/Faculty Ranking: Viewed by Admins as well as Prospective students



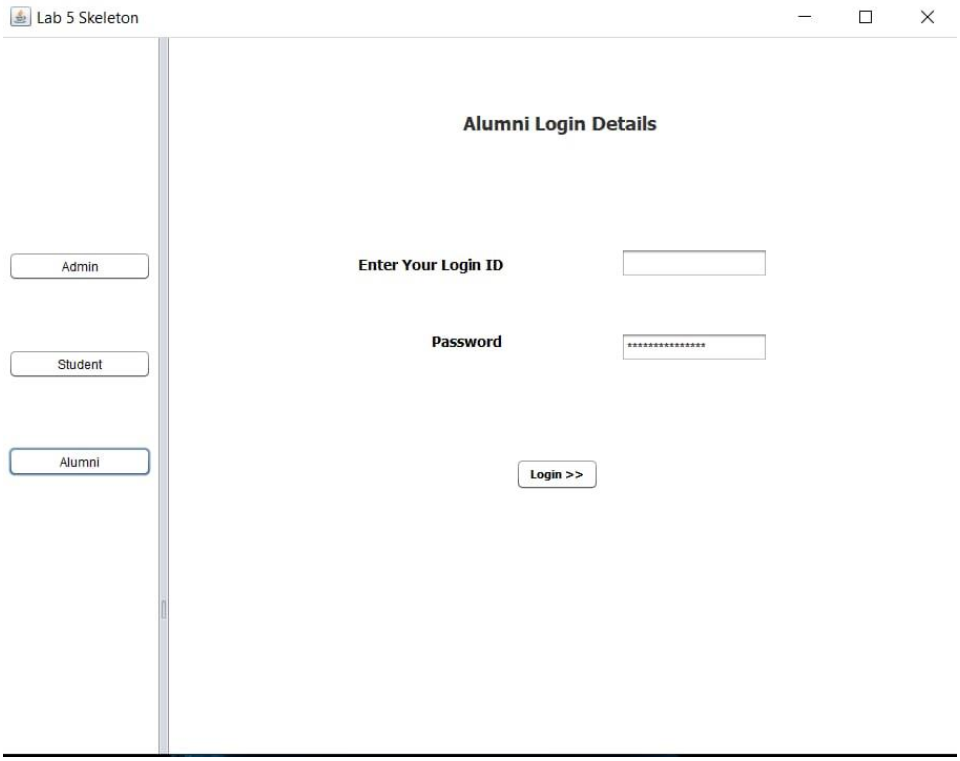
5. Top 20 Universities – Viewed by student:



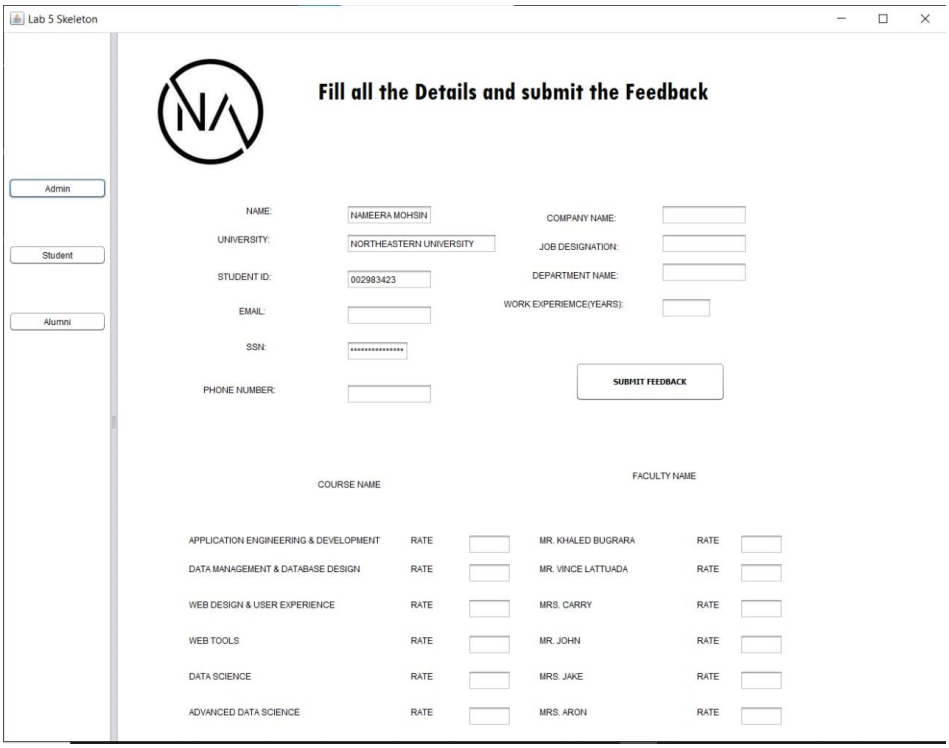
6. University Rankings viewed by student.



7. Alumni Login Page:



8. Alumni Feedback Pages:



Conclusion:

The object model of the university is created along with the sequence diagram to improve the education system of Universities.