

INFO5100
APPLICATION ENGINEERING
DEVELOPMENT

Assignment 3

University Model Design

Vraj Mashruwala (001004658)
Shashwat Shrey (002128122)
Abhishek Jaiswal (002191150)

Table of Contents

	<u>Page</u>
I. Revision History	3
II. Problem Statement	4
III. Class Diagram	5
IV. Important Functions	6
V. Sequence Diagram	9
VI. Dashboard	10
VII. Graphical Analysis	12

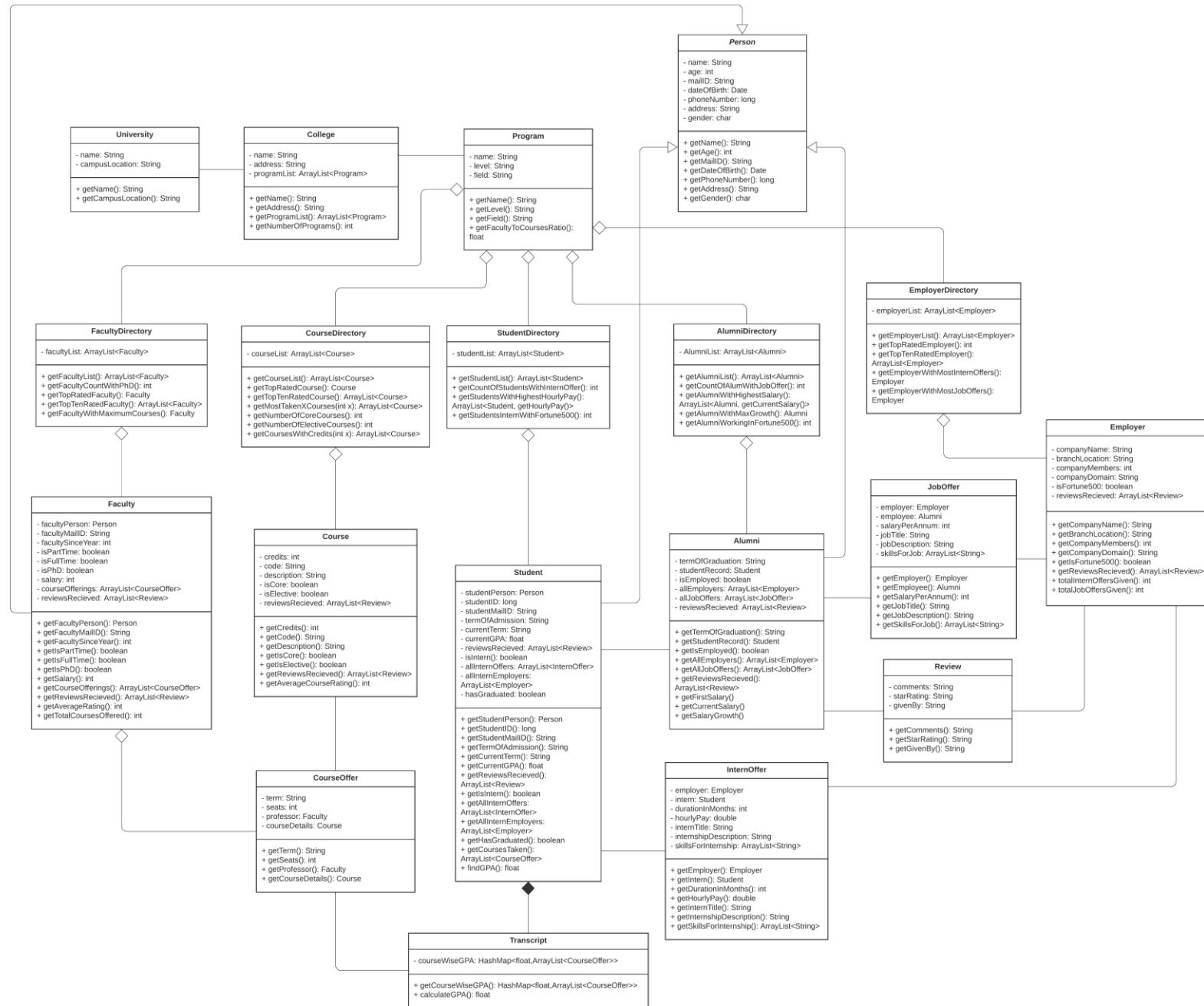
REVISION HISTORY

S. No	Version	Details of changes	Date
1	1.0	Class Diagram, Sequence Diagram, Analytics	10/24/2021

PROBLEM STATEMENT

The University Model Design is a depiction of how components within a university interact with each other to provide some valuable operations that not only makes the university function efficiently but also help us derive some extremely vital insights into our university. Our goal is to design a system that makes it possible to analyze the performance of students not only during their studies as well as after graduating in their respective fields. This would mean gathering various courses, internship and employment statistics. We then make comparisons to understand what factors such as GPA, faculty, choice of courses helped the students to perform better and land their respective positions in their respective fields. In this model, we should include employer details which would provide prospective viewers of our system with a detailed overview of where students from our university get employed. The last component that is required is gathering reviews of various important actors of our system viz. faculty, courses and employers. This would help us in the analytics that we can present using our University Model.

CLASS DIAGRAM



IMPORTANT FUNCTIONS

Here we have described the functions that are defined in our University Model apart from all the getters and setters which we have defined in every class for each attribute of that class.

College

getNumberOfPrograms() - finds us the total number of programs in the college

Program

getFacultyToCourseRatio() - ratio of number of faculty members to the number of courses offered

FacultyDirectory

getFacultyCountWithPhd() - number of faculty members who have a PhD. This lets our rank the quality of faculty in universities.

getTopRatedFaculty() - returns the faculty with the maximum star rating

getFacultyTenRatedFaculty() - returns the top ten faculty based on star rating

getFacultyWithMaximumCourses() - returns the faculty who has the maximum number of Course Offerings

CourseDirectory

getTopRatedCourse() - returns the course which has the highest star rating

getTopTenRatedCourse() - returns the top 10 highest rated courses

getMostXTakenCourses(int x) - returns the top X courses which are taken by students

getNumberOfCoreCourses() - returns the number of courses that were mandatory

getNumberOfElectiveCourses() - returns the number of courses selected from the electives part by the student

getCoursesWithCredits() - returns course along with their respective credits

Student

getCoursesTaken() - returns the list of courses taken by the student

findGPA() - returns the GPA of the student in a float value

Transcript

calculateGPA() - returns the final GPA of the student

Student Directory

getStudentList() - returns the entire list of students

getCountOfStudentsWithInternOffer() - returns the number of students with an intern offer

getStudentWithHourlyPay() - returns the hourly pay the students are receiving

`getStudentInternWithFortune500()` - returns the number of student with an offer from a fortune500 company

AlumniDirectory

`getCountOfAlumWithJobOffer()` - returns number of alumni with a job offer

`getAlumniWithHighestSalary()` - returns the list of alumni in decreasing order of salary

`getCurrentSalary()` - returns the current salary of an alumni

`getAlumniWithMaxGrowth()` - returns the alumni that has had the max growth in his career

`getAlumniWorkingInFortune500()` - returns the alumni working with a fortune500 company

EmployeeDirectory

`getTopRatedEmployer()` - returns the top rated employee

`getTopTenRatedEmployer()` - returns the list of top ten rated employees

`getEmployerWithMostInternOffers()` - returns employer object with maximum intern offers

`getEmployerWithMostJobOffers()` - return employer object with maximum job offers

Intern Offer

`getEmployer()` - returns employer object

`getIntern()` - returns Student object that has got an intern offer

`getDurationMonths()` - returns the number of months in integer

`getHourlyPay()` - returns the hourly pay of the student

`getInternTitle()` - returns the job title in String format

`getInternshipDescription()` - returns details about the internship tasks

Employer

`isFortune500()` - returns boolean value if the company is a fortune500. This is good metric to use to see how popular our students are among the Fortune 500 companies

`totalInternOffersGiven()` - returns integer providing total intern offers. This would help us find which employer hires maximum interns.

`totalJobOffersGiven()` - returns integer providing total job offers. This would help us find which employer hires maximum full time student.

Alumni

`getFirstSalary()` - returns the first salary of an alumni after graduation

`getCurrentSalary()` - returns the current salary of the alumni. This could be different from the first salary as time passes. This is used to calculate the growth of our alumni

`getSalaryGrowth()` - returns the growth the alumni has had since graduation in terms of salary rise. This is good metric to use to see success achieved by our alumni

Student

getCoursesTaken() - returns an Array List of Course containing all the courses the student has taken
findGPA() - calculates GPA based on transcript details which contains a hash map of all courses taken and course grade the student received

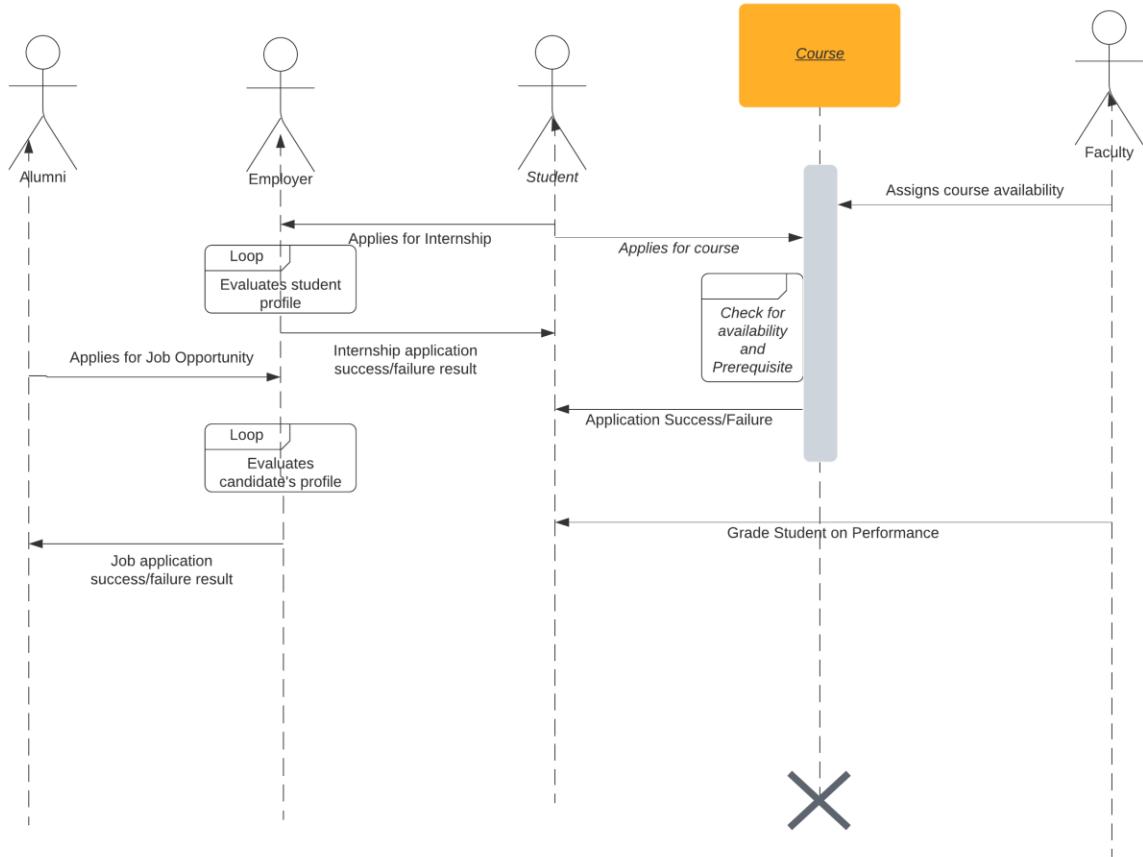
Course

getAverageCourseRating() - returns the average rating of this course based on reviews given

Faculty

getAverageRating() - returns the average star rating of the faculty from all the reviews received
getTotalCoursesOffered() - returns the number of Course Offerings that this faculty has had

SEQUENCE DIAGRAM



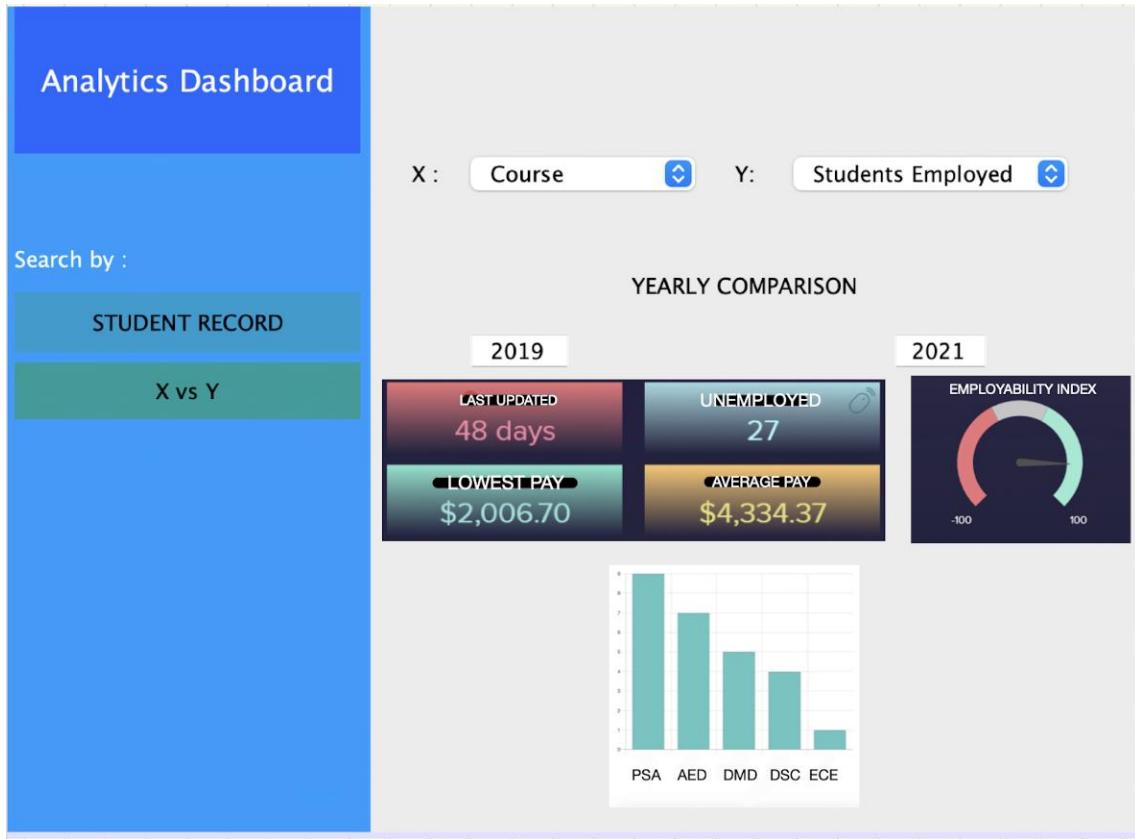
The above UML demonstrates the usual activity flow among Students, Employers, Faculty and Alumni inside the University Model. How all the actors interact with each other in the system to synchronize the data flow and other validations that exist within the system.

DASHBOARD

1. Displaying the available records of a student fetched using NUID as the key.

The screenshot shows a user interface for an 'Analytics Dashboard'. On the left, there's a sidebar with 'Analytics Dashboard' at the top, followed by 'Search by' and two options: 'STUDENT RECORD' and 'X vs Y'. The main area has a light gray background. At the top right, there's a search bar labeled 'NUID :' with a text input field containing 'ABCD' and a blue 'GO' button. Below it are four input fields with labels: 'NAME :' (containing 'ABCD'), 'EMPLOYMENT STATUS :' (with three radio buttons for 'FULL TIME', 'INTERN', and 'UNEMPLOYED'), 'HOURLY PAY :', and 'COMPANY :'. There's also a field labeled 'BATCH OF :'. In the center, there's a large square placeholder icon of a person's head and shoulders.

2. Allows the user to get data based on what the user wants to compare. Various options available to select what to see on X and Y axis to get data.



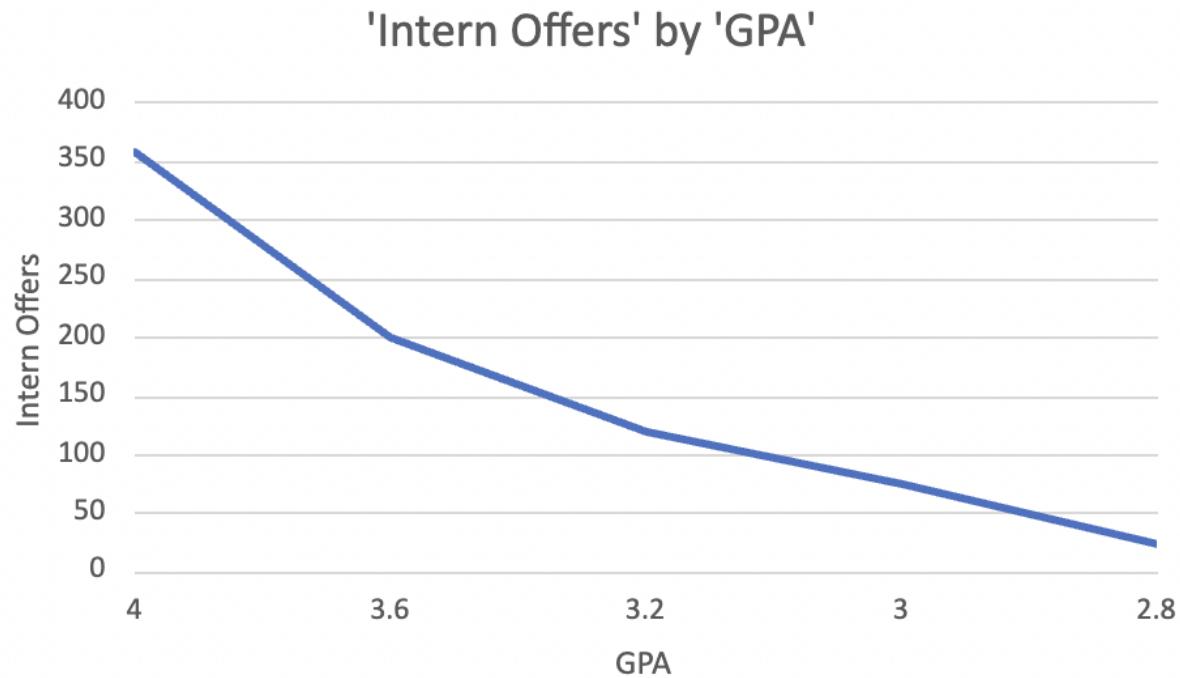
After the user selects the attributes, the class diagram will explain to us how to write a function that can fetch the information we want because of the modularity in the design.

Let's say someone wants the comparison of GPA and Employability, the student object individually is capable of providing the list of GPAs and also fetching the boolean values for employment status. Which can be used to create a list and pass as a parameter to get the graph we want to display, for analysis.

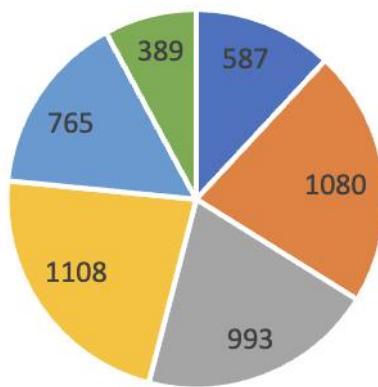
GRAPHICAL ANALYSIS

Here we are displaying a few graph results that our University Model is able to calculate and generate. We have captured numerous important characteristics where needed, for example GPA for students, intern offer details for students, employment offer details for alumnus, for employers we are also capturing if the employer company is a Fortune 500 company or not. For the intern offer we also capture hourly pay, similarly for employment we capture start salary, current salary. We also have reviews and ratings for courses, faculty and employers. All these attributes can easily be used to derive a number of helpful statistics that will be shown on our dashboard.

As examples, below we have shown sample graphs for a few of the many statistics we can find from our University Model.



Number of Alumni



Chart

■ Project Managers ■ Software Engineers ■ Data Scientists
■ Data Analyst ■ Web Designer ■ Business Analyst

'Number of Students' by 'Per Hour Pay'

