
Team K

Authors: John Badji | Fherdinand J. Corcuera | Kelvin Wu

CUNY Zero
PHASE II: Design Report
For a Graduate Program Management System

I. INTRODUCTION

Team K is developing a web-based application for **CUNY Zero**. An online platform that provides several tools for managing academic aspects for a university graduate program. CUNY Zero allows students to manage their academics through searching, enrolling, and dropping classes. The software provides instructors/ faculties a powerful tool for managing the courses they are teaching, and the students enrolled in their classes. It gives total control to the Registrar's office to control all sorts of activities that ensure a smooth transition to a school term and throughout the semester. Among other things, users can find basic information related to courses, students, faculty as well as enrollment options.

Team K is using Python's **Django** to satisfy the requirements of the system. We chose Django because it is a high-level, free, and open-source web framework that provides a clean and pragmatic design. And, because it offers a built-in back end that is super important when starting a project.

The overall picture of the system is depicted in **Figure 1** through a Collaboration Class diagram. The top-level users' interactions of what is intended from the software are summarized in the diagram.

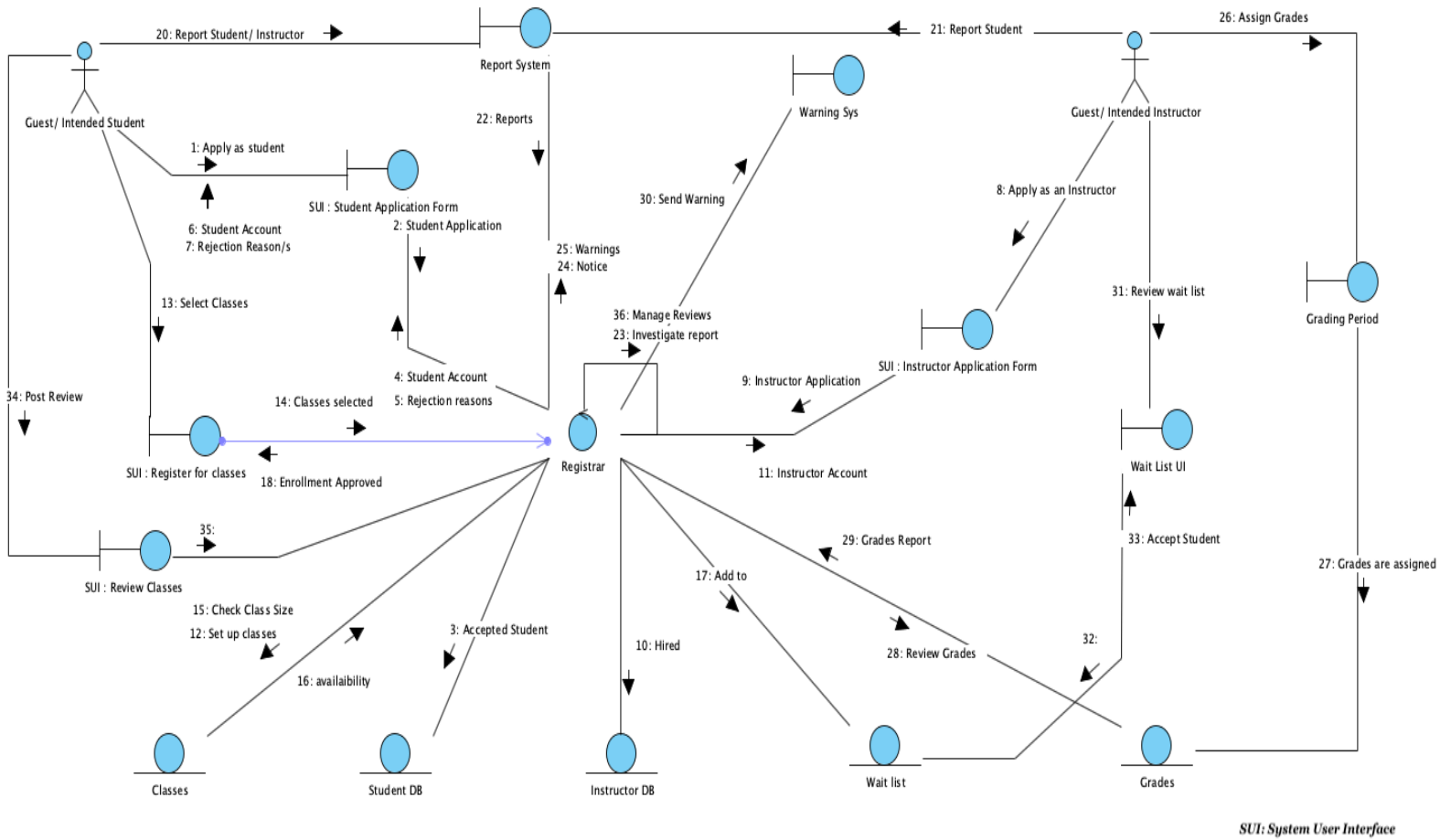
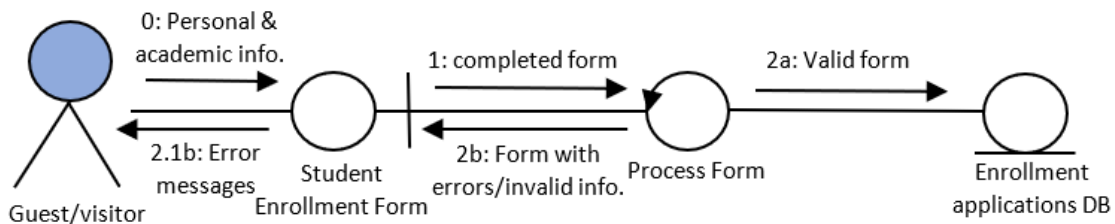


Figure 1: Overall picture of the system using Collaboration Class Diagram

II. DIAGRAMS FOR EACH USE CASE

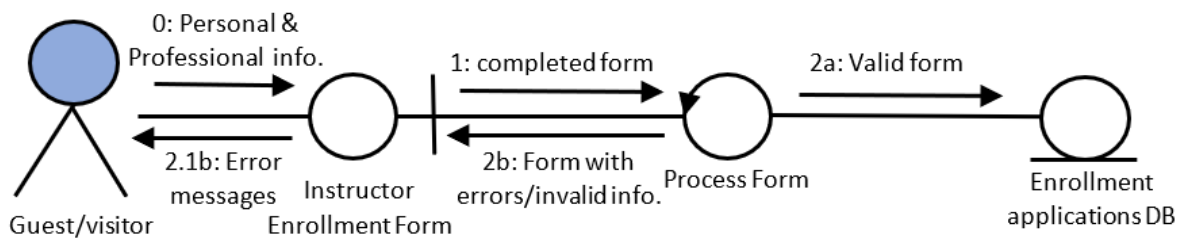
Student Application:

- Normal scenario(s):
 - Visitor fills out form with correct/valid information and it is submitted, stored, and awaiting decision.
- Exception(s):
 - User tries to submit form with invalid information (e.g., phone number containing letters, incorrect email format, date of birth in future, missing info., etc.)



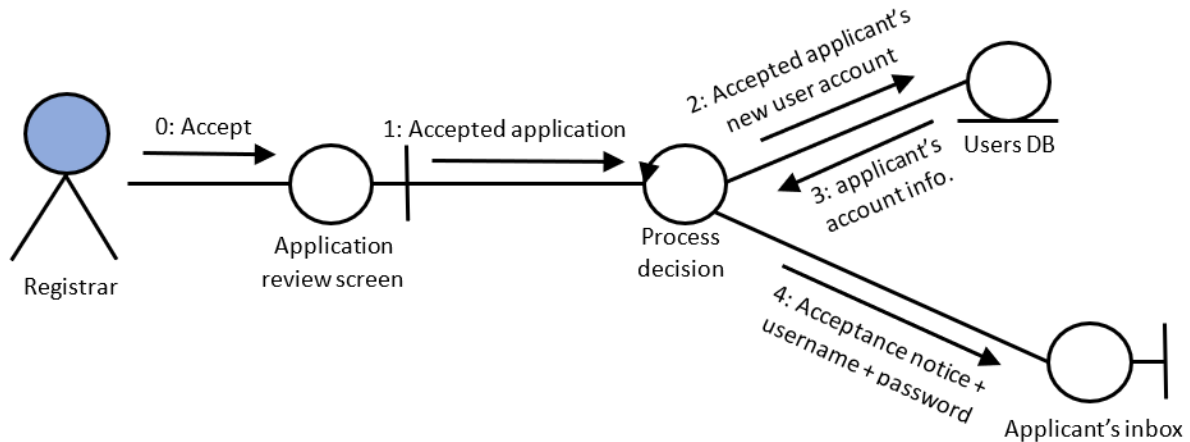
Instructor Application:

- Normal scenario(s):
 - Visitor fills out form with correct/valid information and it is submitted, stored, and awaiting decision.
- Exception(s):
 - User tries to submit form with invalid information (e.g., phone number containing letters, incorrect email format, date of birth in future, missing info., etc.)



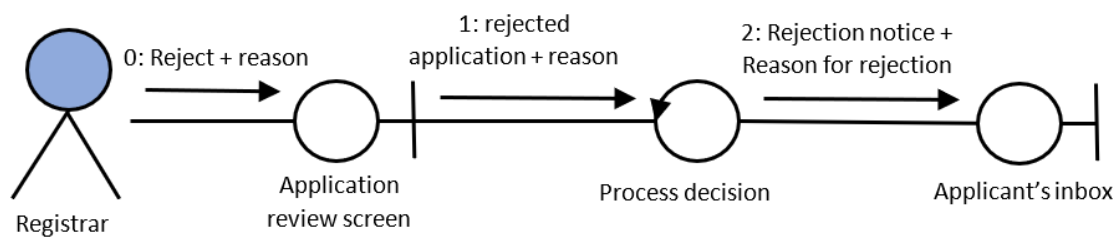
Accept Application:

- Normal scenario(s):
 - Registrar reviews a student/instructor application and accepts. The user is registered into the DB, and a notice is sent to the user with the username and password.



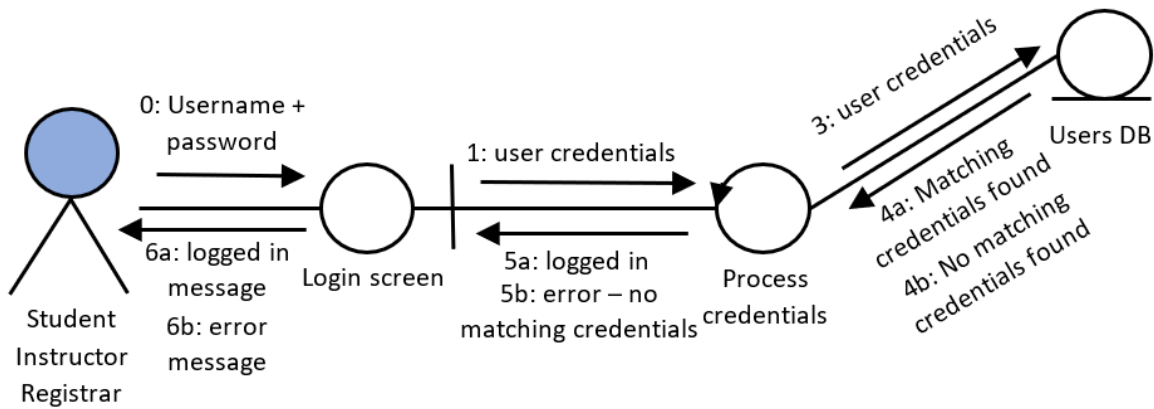
Reject Application:

- Normal scenario(s):
 - Registrar reviews a student/instructor application and decides to reject it. The decision is processed and a notice containing the reason for rejection is sent to the applicant.



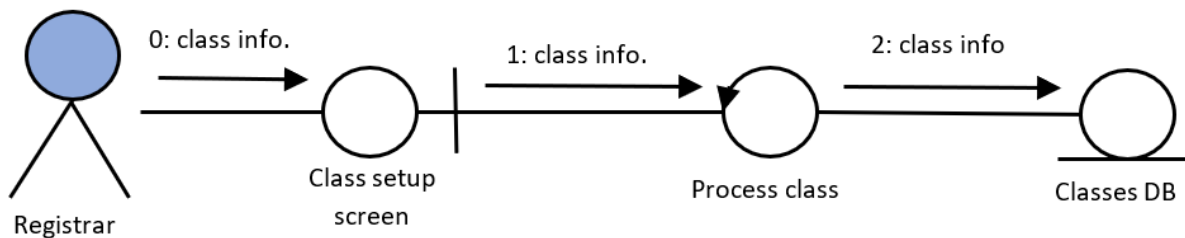
Log In:

- Normal scenario(s):
 - User inputs his/her credentials that are correct, the user is authenticated and logs in successfully.
- Exception(s):
 - Username and/or password don't match/incorrect.



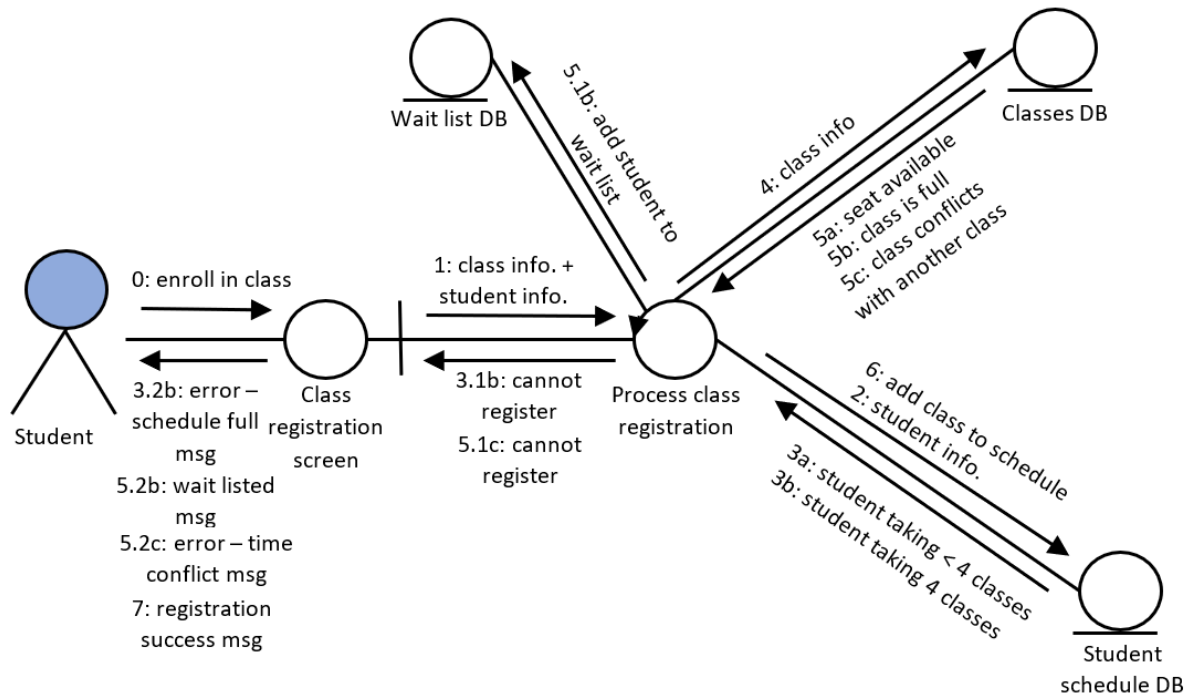
Class Set Up:

- Normal scenario(s):
 - Registrar creates a class with a time, instructor(s), and size.



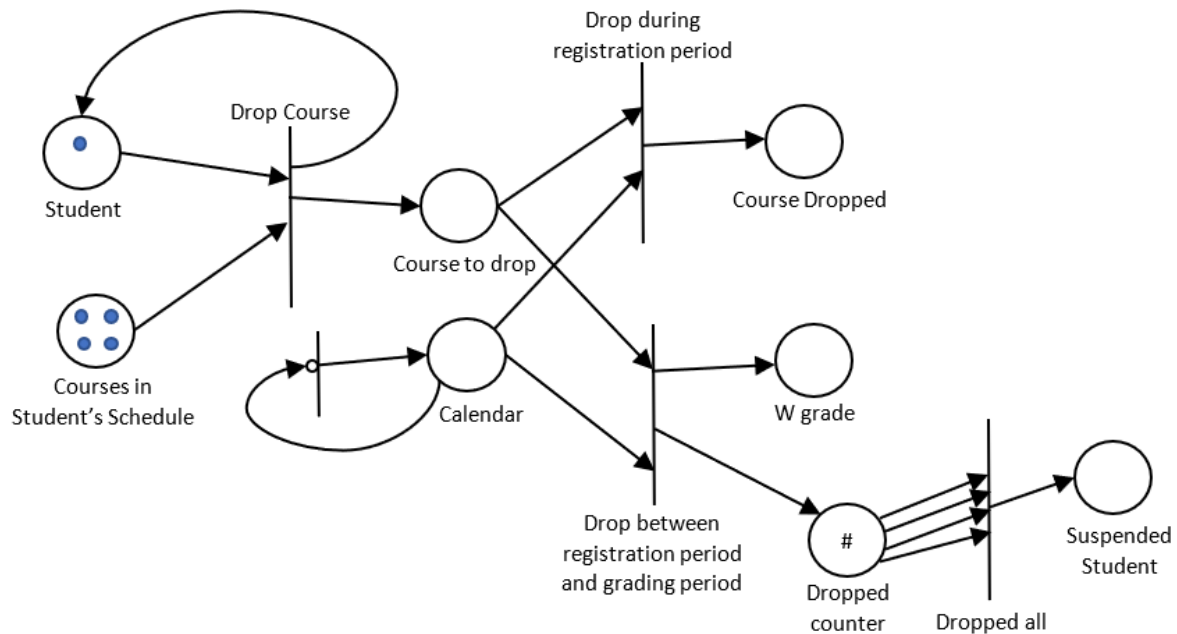
Class Registration:

- Normal scenario(s):
 - A student chooses a class and successfully enrolls in it.
- Exception(s):
 - Student is already taking 4 classes: student cannot register unless a class is dropped.
 - Class conflicts with another class the student is enrolled in: student cannot register into the class unless his/her schedule is changed.
 - Course is full: student is put on waitlist.



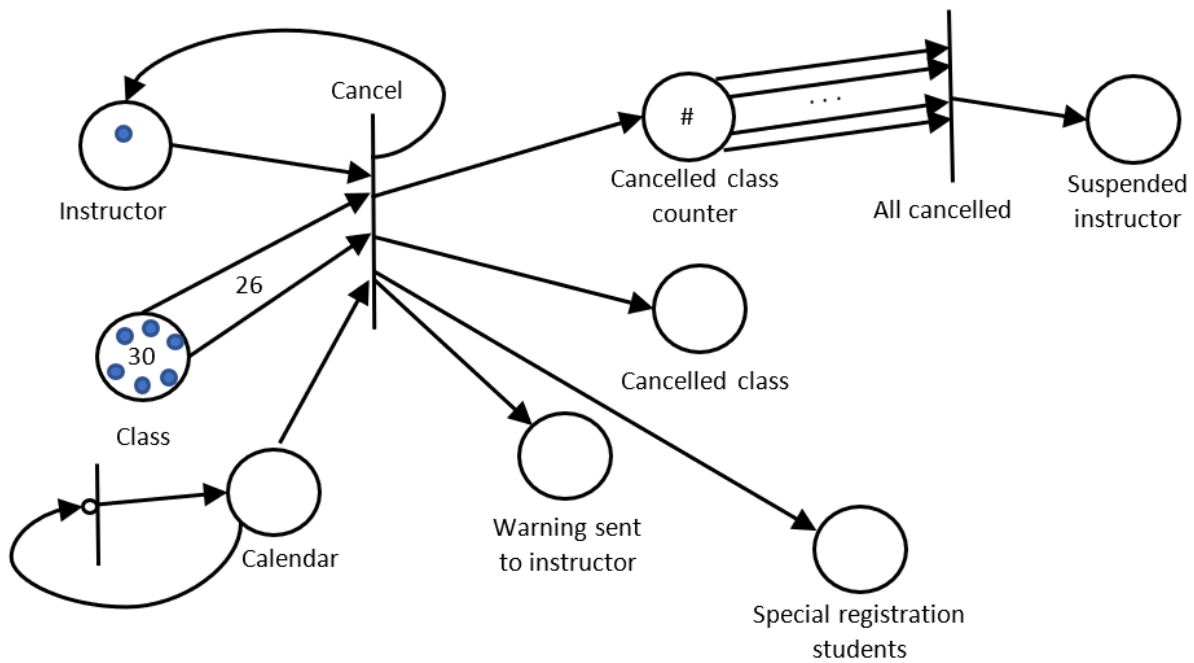
Drop course:

- Normal scenario(s):
 - A student chooses a class he/she is registered in and drops it from his/her schedule.
- Exception(s):
 - Dropped between registration and grading period: receive grade W.
 - Dropped all classes between registration and grading period: suspend student for one semester.



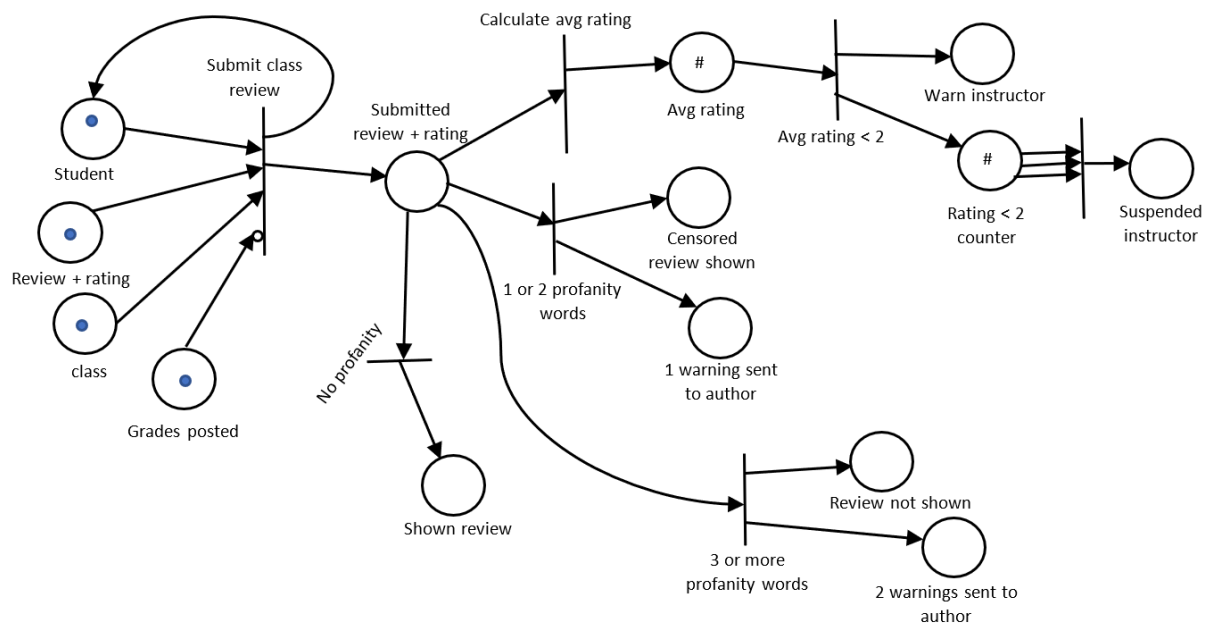
Cancel Class:

- Normal scenario(s):
 - If the class does not meet the minimum student quota by the end of registration period, it will be automatically cancelled, the instructor will be warned, and the students who were enrolled will be given a special registration period.
- Exception(s):
 - If all the classes for an instructor are cancelled, he/she will be suspended for the next semester.



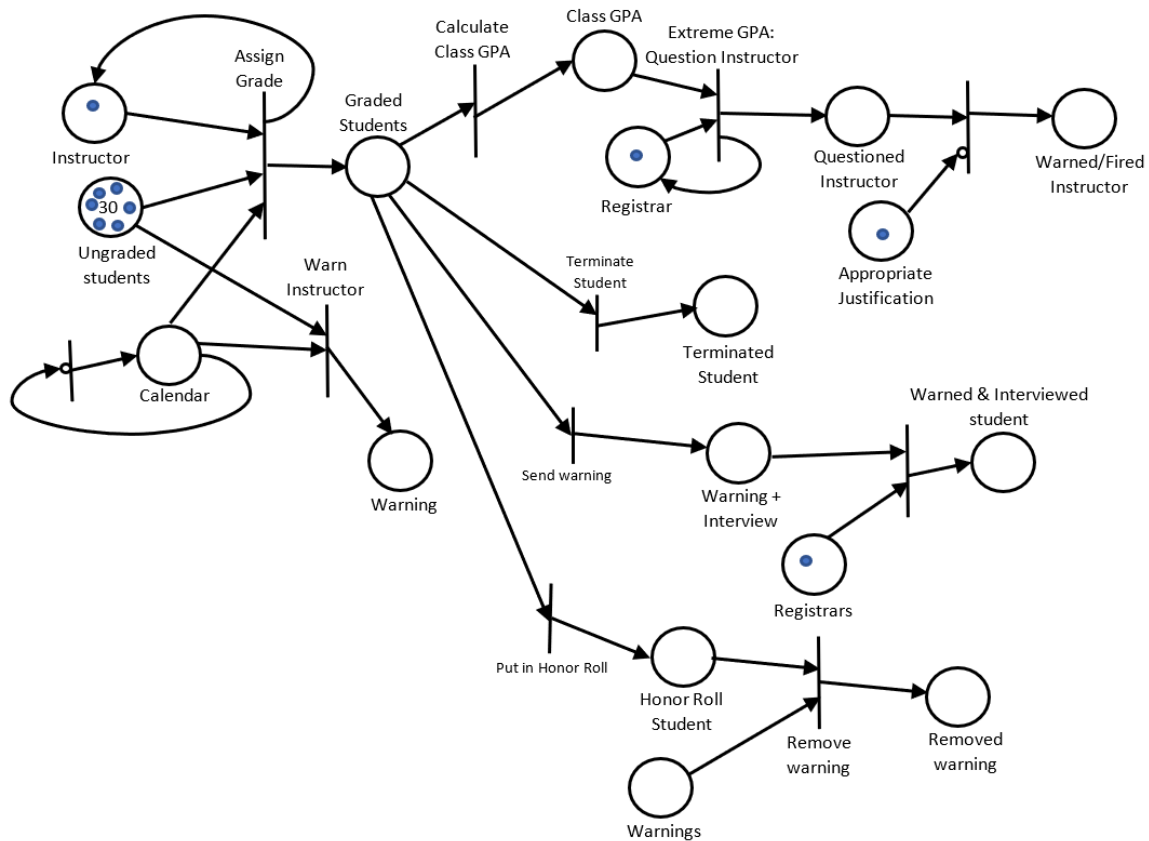
Write Reviews

- Normal scenario(s):
 - Students can write review and rate classes they are taking before grades are posted. These will be displayed on the course information page assuming they are appropriate.
- Exception(s):
 - Instructors of courses that receive a rating of < 2 will be warned. If they accumulate 3 warnings, they will be suspended.
 - If a student's review contains 1 or 2 profanity words, it will be censored and shown. The author will receive 1 warning.
 - If the review contains 3 or more profanity words, it will not be shown, and the author will receive 2 warnings.



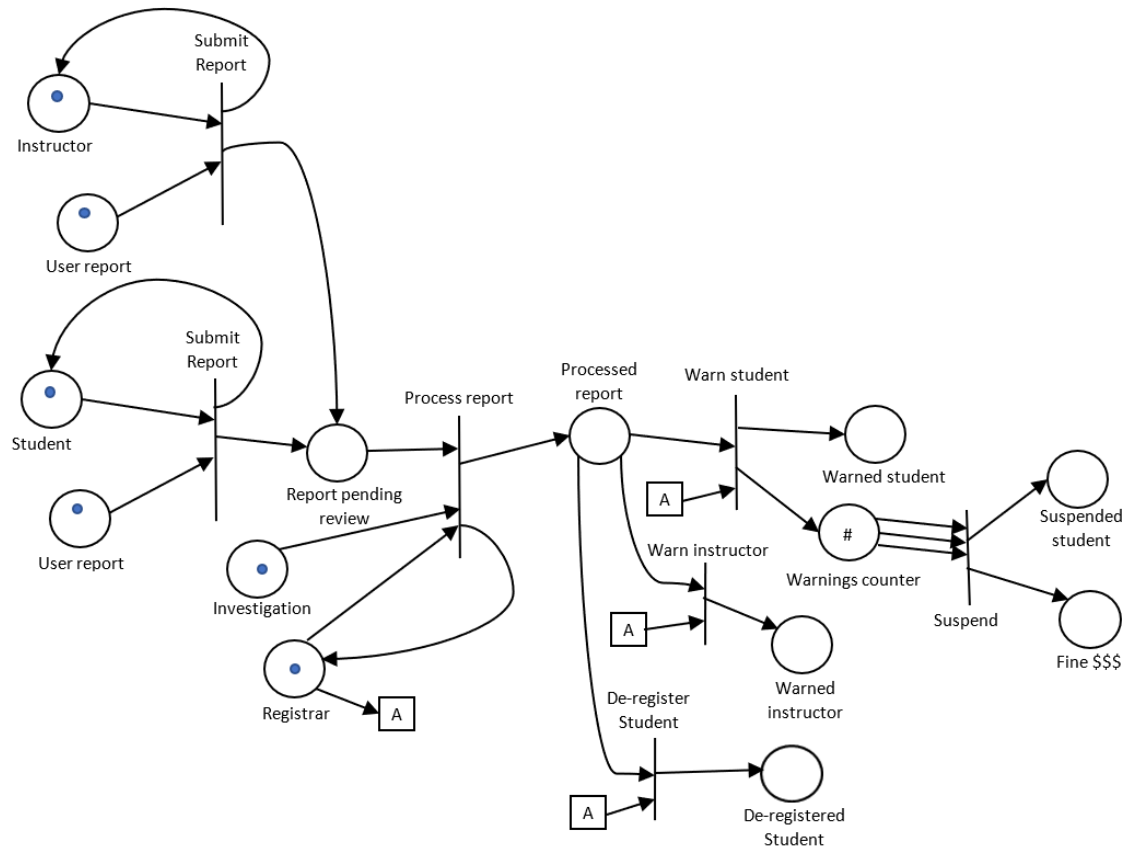
Assign Grades:

- Normal scenario(s):
 - An instructor submits the grades of the students in his/her class during grading period.
- Exception(s):
 - After grading period end: if not all students were assigned a grade, the instructor is warned.
 - If the GPA of the class > 3.5 or < 2.5 , registrar will question the instructor. If the instructor does not provide a justification, he/she will be warned/fired.
 - Students with GPA < 2 or failed the same course twice will be terminated.
 - Students with GPA between 2 and 2.25 will receive a warning with an interview with the registrars.
 - Students with semester GPA > 3.75 or overall GPA (for > 1 semester) > 3.5 will be honor students. This can remove 1 warning for the student if any.



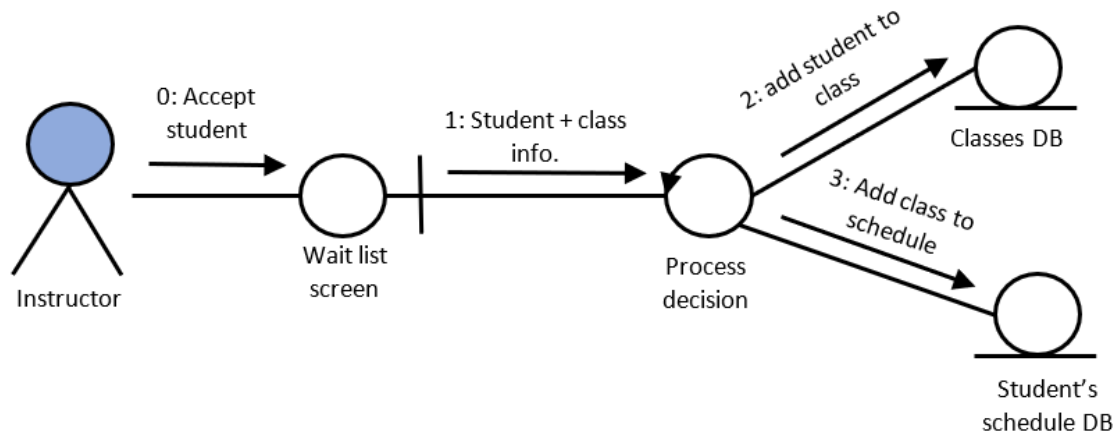
Report User:

- Normal scenario(s):
 - A student or instructor may submit a report about another student or instructor. This will be sent to the registrar who will decide to warn or punish the individual based on investigation.
- Exception(s):
 - If a student receives 3 warnings, he/she will be suspended for 1 semester and must pay a fine.

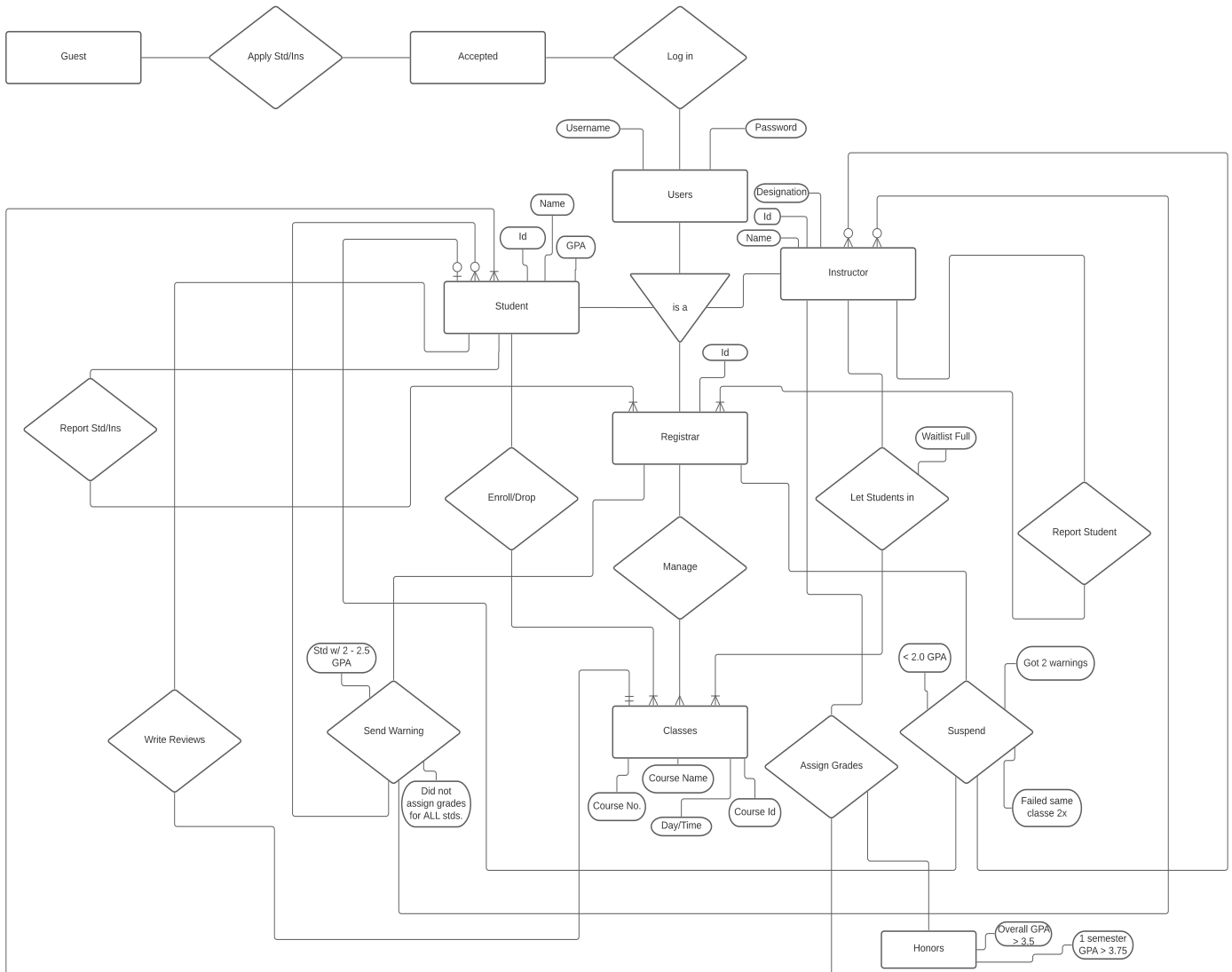


Accept Wait Listed Student:

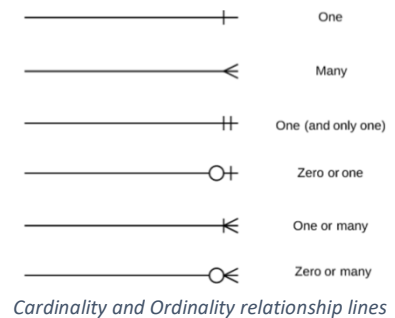
- Normal scenario(s):
 - An instructor reviews the wait list for one his/her classes and decides to accept a waitlisted student.



III. E-R Diagram of the whole system



When a guest applies as a student or instructor, the system assigns the person a username and password. A Registrar, Instructor, or Student can all be users. The Registrar is in charge of class management, which includes opening and closing new classes. If a student enrolled for a class and was placed on a waitlist, the teacher may admit the student if the waitlist is full. Students have the option of



enrolling in or dropping classes. If a student's GPA at the end of the semester is between 2 and 2.5, the registrar will issue a warning to that student. A student with a GPA higher than 3.5 will be included in the honor students list. If the instructor did not assign grades to ALL students, a warning will be issued. The registrar will suspend the student if the GPA is less than 2.0, the student received two warnings, or the student failed the same course twice. If the instructor receives two warnings, he or she will be suspended. A student may submit a report to the registrar regarding the instructor or another student. The instructor can also write a report to the registrar regarding a student. A student can write a review of the classes they are enrolled in.

IV. DETAILED DESIGN

For methods of main functionalities, pseudo-code is used to describe the input/output interactions of the system.

ENROLLMENT SYSTEM

SYSTEM PROMPT Guest to enter informations

GET guest informations

IF (All info have the right syntaxes)

DISPLAY Enrollment Successful.

ELSE

DISPLAY Enrollment Failed. TRY Again

ENDIF

IF (Guest applied as student):

IF (gpa >= 3.0 && student.quota not reached)

APPROVE application

SEND student Account

```
ELSE
SEND Rejection letter with reasons
ENDIF
ELSE
IF (Approved)
SEND Instructor account
ELSE
SEND rejection letter
ENDIF
ENDIF
```

LOGIN PAGE

```
PROMPT User to enter Infos
GET Username
GET Password
IF(User.username == usernameEntered && User.password == Enteredpwd):
Login Successful
IF (User is a student)
Display student.info
Display student.classes
Display student.grade

Elif (User is an instructor)
Display info
Display classes
Display classes.students
Display grading form
ELSE
Login Failed
ENDIF
```


COURSES REGISTRATION.

SYSTEM AUTHENTICATES USER

IF (Course registration period && USER is a student)

FOR course in courses:

DISPLAY Courses with checkboxes

ENDFOR

Prompt Student to select courses

IF (course_selected >= 1 && course.currentCapacity < course.FullCapacity)

Prompt student to register for selected classes

IF (no time conflict)

DISPLAY success message: Courses registered

ELSE:

DISPLAY time conflict error message

ENDIF

ENDIF

ENDIF

DROP COURSES

System authenticates users

IF (Course registration period && User is a student)

FOR course in takenCourses

Display taken courses with checkboxes

ENDFOR

Prompt User to select to be dropped.

IF (course_selected >= 1)

Prompt student to drop selected classes

IF (student dropped for selected courses)

DISPLAY success message: courses dropped

IF (Start date < today.date < end date):

Student.letterGrade = "W"

ELSE:

DISPLAY ERROR message

ENDIF

ENDIF

ENDIF

REVIEW COURSES

System authenticates users

IF (User is a student)

FOR course in takenCourses

Display taken courses with review option

ENDFOR

Prompt User to select to reviewed.

IF (course_selected)

Course review form

Prompt User to fill the review form

IF (form has the right syntaxes)

Display success message: Review posted

Else

Display error messages.

ENDIF

ENDIF

ENDIF

CLASS RUNNING PERIOD

IF (student.takenCourses < 2)

Warn student

ELIF (class.currentCapacity <= 5)

cancel classes

Open special registration period for cancelClasses.students

Suspend class.instructor

ENDIF

GRADING PERIOD

For courses in courses:

IF (No grade assigned)

Warn course.instructor

ELIF (class.GpaAverage > 3.5 OR class.GpaAverage < 2.0)

Demand justification

ENDIF

ENDFOR

Count = 0

IF (student.gpa < 2)

count+= 1

IF (count == 2)

Terminate student

ELIF (student.gpa >2 && student.gpa< 2.5)

SEND warning

ELIF (student.gpa>= 3.5)

student.label = 'HONNOR ROLL'

ENDIF

COMPLAINT SYSTEM

System authenticates user

PROMPT User to fill the complain form

IF(user.complaint is serious)

Investigate

Take actions

ENDIF

IF(students.warningCounts> 3)

Suspend student

Must pay fine

ENDIF

V. SYSTEM SCREENS

CUNY Zero Home About Contact Search Log in Register

City University of New York Graduate School

the City College of New York

Enroll Apply for Graduation Request Transcript Search for Classes

Graduate School Program

Graduate school is a postgraduate program of study that focuses on a certain academic area or professional field of interest. Although traditionally “academic” in nature (focused on the generation of original research in a specific topic), graduate school may also be “professional” in nature (oriented on the development of skills and knowledge for a certain vocation), or a combination of the two.

While the majority of undergraduate degrees allow students to study modules and classes of personal interest, a graduate degree allows for a considerably greater degree of flexibility. Attending extracurricular events and meetings, as well as hearing from guest speakers and lecturers as well as full-time faculty members who interest you, contributes to the diversity and multidisciplinary of graduate school.

November 2021

M	T	W	T	F	S	S
1	2	3	4	5	6	7
8	9	10	11	12	13	14
15	16	17	18	19	20	21
22	23	24	25	26	27	28
29	30	1	2	3	4	5

Honors Reviews Report

The City College of New York
160 Convent Avenue New York, NY 10031
p: 212.650.7000

Facebook Instagram LinkedIn Twitter

Copyright (c) 2021 | City University of New York

Figure 2: Main GUI of the system (1280 x 800)

CUNY Zero
Home
About
Contact
Search
Log in
Register

City University of New York Graduate School
the City College of New York
The City College of New York
Enroll
Apply for Graduation
Request Transcript
Search for Classes

Full Name

Email

DOB

Contact No.

Address

Country

Zip Code

Previous School Name*

Previous School Addr.*

Graduation Date*

GPA*

SUBMIT
Already have an account? Log in

November 2021

M	T	W	T	F	S	S
1	2	3	4	5	6	7
8	9	10	11	12	13	14
15	16	17	18	19	20	21
22	23	24	25	26	27	28
29	30	1	2	3	4	5

Honors
Reviews
Report

The City College of New York
160 Convent Avenue New York, NY 10031
p: 212.650.7000

Copyright (c) 2021 | City University of New York

Figure 3: A student registration GUI screen of the system

VI. MEMOS OF GROUP MEETINGS

We have been holding meetings every week on Fridays to discuss about the design and functionality of the website, tasks to be done by each member and their progress, as well as any necessary written documents that need to be done.

Meeting #	Notes	Date
1	<ul style="list-style-type: none"> Reviewed project specifications Wrote up system outline 	10/01/2021
2	<ul style="list-style-type: none"> Determined what software to use Created to-do list 	10/08/2021
3	<ul style="list-style-type: none"> Reviewed tasks each member oversees 	10/15/2021
4	<ul style="list-style-type: none"> Worked on phase 1 report 	10/22/2021
5	<ul style="list-style-type: none"> Put together basic structure of website 	11/05/2021

6	<ul style="list-style-type: none"> Reviewed progress 	11/12/2021
7	<ul style="list-style-type: none"> Reviewed progress 	11/19/2021
8	<ul style="list-style-type: none"> Worked on phase 2 report 	11/26/2021

VII. ABOUT GITHUB REPOSITORY

<https://github.com/ej0306/csc322.git>