Group H

CunyZero

Software Requirements Specification

For Online Course Registration

<version 2.0>

Revision History

Date	Version	Description	Author
10/24/2021	<1.0>	Added Introduction and Use-Case reports	Dong Huang Chen, Lun Qu, Zibin Luo, Zhicong Wen
11/24/2021	<2.0>	Added system collaboration class diagram, petri nets, and er diagram	Dong Huang Chen, Lun Qu, Zibin Luo, Zhicong Wen

Table of Contents

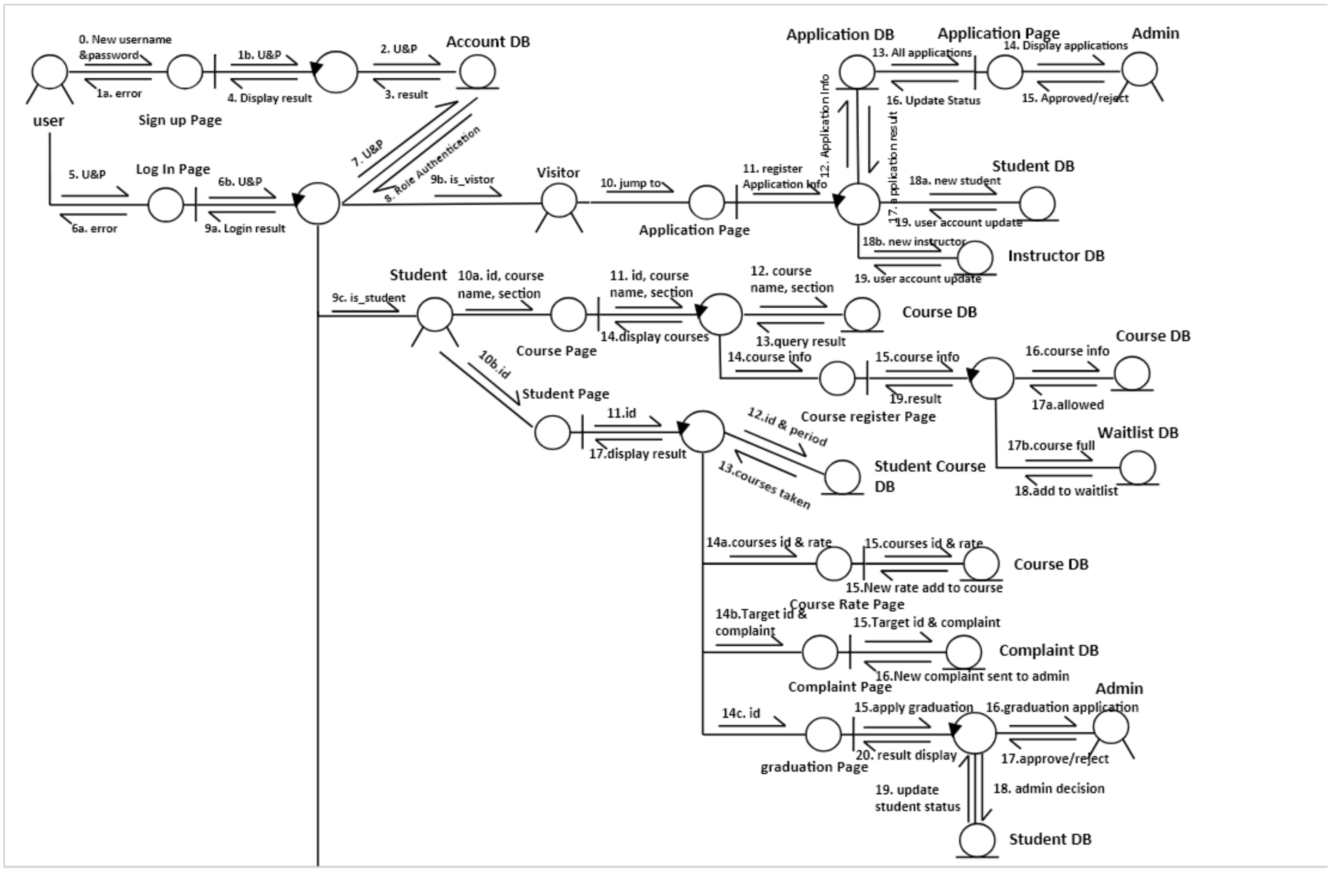
1. Introduction

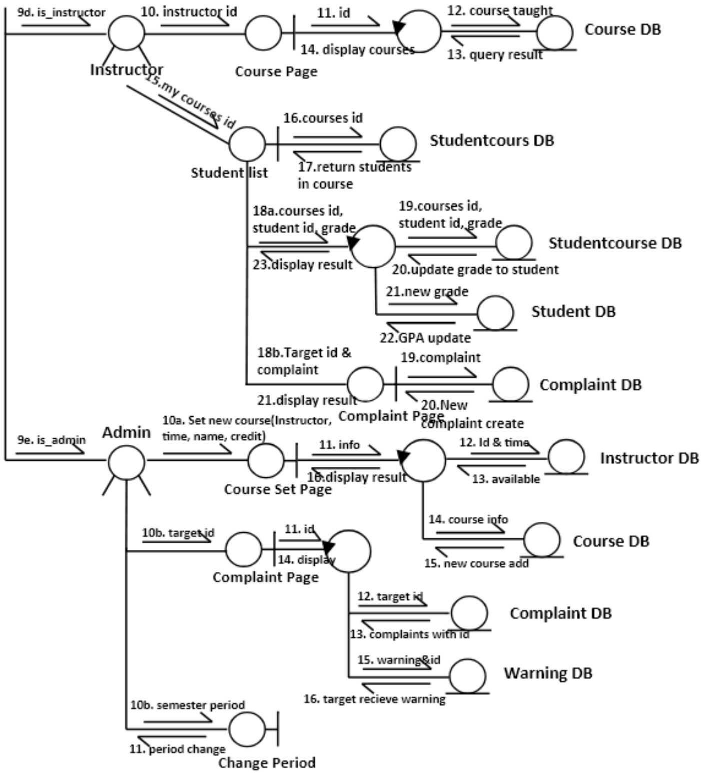
a. Brief

The goal of our project GradSchoolZero is to build a web-based applicaiton for the school academy system. There are four types of users in our project(Admin, Student, Instructor, Visitor). This application will provide rational and interactive interfaces for each type of user to achieve the goal of a course registration system. Some of the features involved are a login system for each user, course registration and dropping for students, course management and registration for instructors, and also course creation by the registrar.

b. Overall collaboration class diagram

Our project can be divided into four major parts by the type of users. When a user login to the system, the system will perform identity verification and distinguish each account by their roles. Different types of accounts can perform corresponding actions which we are going to go further in the user case section.





2. All use cases

a. User case reports and scenarios

The relationship between Guest and System:

- **Every guest of our website will be able to see our home page of the website, which displays the highest/lowest rated class and highest GPA student.**

  Scenarios: At the home page, guests without login see the specific information (course and student)

- **A guest can apply for a new visitor account with a unique username and password.**

  Scenarios: At the register page, guests provide the new username and new password they expected. The system successfully creates a new visitor account with a unique user id, and stores the new account into a database.

Exceptional:

The username provided by the visitor is the same as a username in the database. The UI displays errors and rejects this request.

The username or password in the form submitted by the guest is empty. The UI displays errors and asks guests to fill all fields.

The relationship between Visitor and System:

- **Visitors can browse the listing of instructors and courses in the system.**

  Scenarios: A user login a visitor account can review the instructors and courses that is enrolled in instructor list page or course list page

- **Visitors apply to be students/instructors.**

  Scenarios: User logs in as a visitor, he/she makes an application to register as student/instructor by providing relatives information(such as GPA, program). The request will be stored in the database and wait for the admin to deal with.

  Exceptional: Visitor already has one student/instructor application that is in process. The UI shows an error, "You have an application processing!"

The relationship between admin and system:

- **Admin need to approve or reject the application that is applied by visitor account, including graduate application from student.**

  Scenarios: A user login to an admin account approved a student register application, the visitor account who sent that request will automatically upgrade to a student account, the account will keep the same username and password and a specific employee id will be assigned to them.

  Exceptional: Admin rejects a student register application without a justification. The UI will display an error, "Please provide your justification" and the action is canceled.

- **Admin can review all complaints and take action(such as warning, deregister).**

  Scenarios: Admin receives a complaint and with this complaint as a warning reason sends a warning to the complaint target.

  Exceptional: Admin warns someone without any justification will cause an error message by system.

- **Admin can browse the listing of instructors and courses in the system. And creating new courses.**

  Scenarios: Admin create a course by fill course setting, such as instructor, period, capacity, etc.

  Exceptional: Current period is not in course registration period, the new course is created unsuccessfully.

- **Admin has power to manage the semester period of the system.**

  Scenarios: Admin change the period to course register period, hence students are able to register in the course that admin created in previous period.

The relationship between student and system:

- **Students are able to browse the courses in the system and register in a course.**
- **Students can write ratings and reviews of the current course he takes.**

  Scenarios: During the course running period, students give the rate to the current course he takes. Rating score will affect the overall rate of the course.

  Exceptional: After the course running period, no one can change the rate they are given.

- **Students can drop courses.**
- **Students can make their complaint to the admin about a specific person.**

  Scenarios: Students fill out a complaint form, including the target id, message. The complaint will be stored in the database and reviewed by the admin.

- **Students can apply for graduation**

  Scenarios: Students who meet the graduation requirement apply to graduate. After the admin reviews the student information, he will approve the application and student status will change to graduate.

- **Students will be worn by the system automatically if they have less than 2 courses, gpa between 2-2.25.**
- **Students will be suspended for a semester by system if they have more than 3 warnings.**

The relationship between instructor and system:

- **Instructors can browse the student information who register in their course. And they can complain to the admin.**
- **Instructors need to assign grades to all students in their course in the grade period.**
- **An instructor who accumulated 3 warnings will be suspended by the system.**

The relationship between course and system:

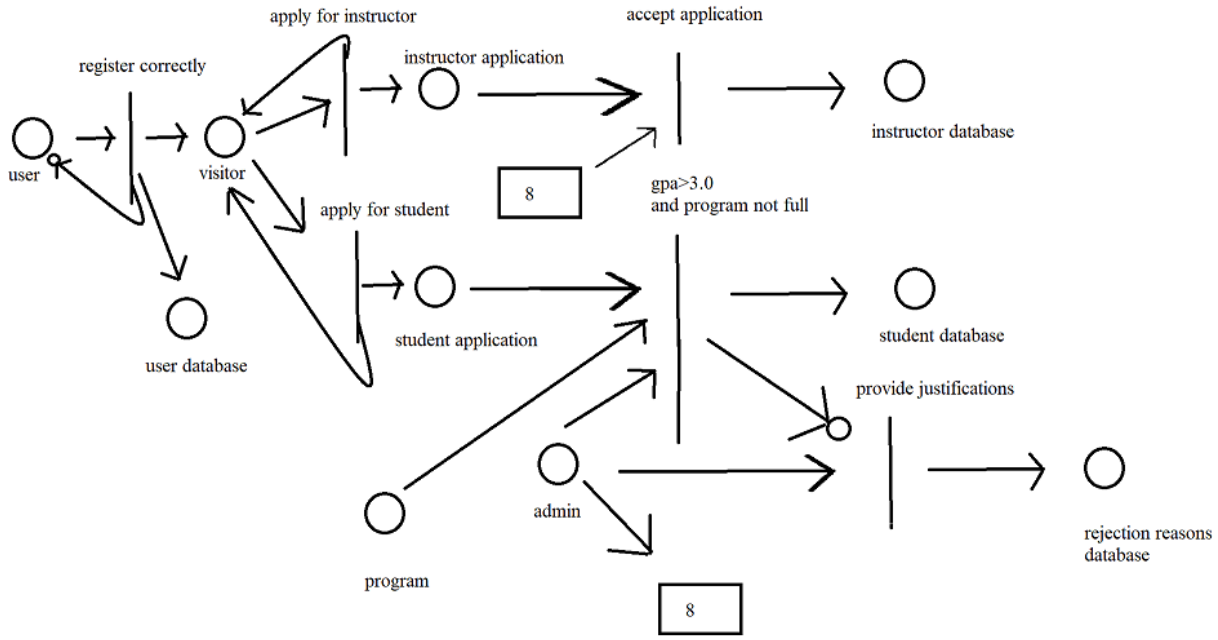
- **A course less than 5 students will be cancelled**

  Scenarios: by the end of the registration period, a course less than 5 students will be automatically canceled by the system. Meanwhile a warning will be sent to the instructor who teaches this course, and the students who register this course will have additional time to re-register to another course.

b. **Use Cases Petri nets and Collaboration Class Diagrams:**

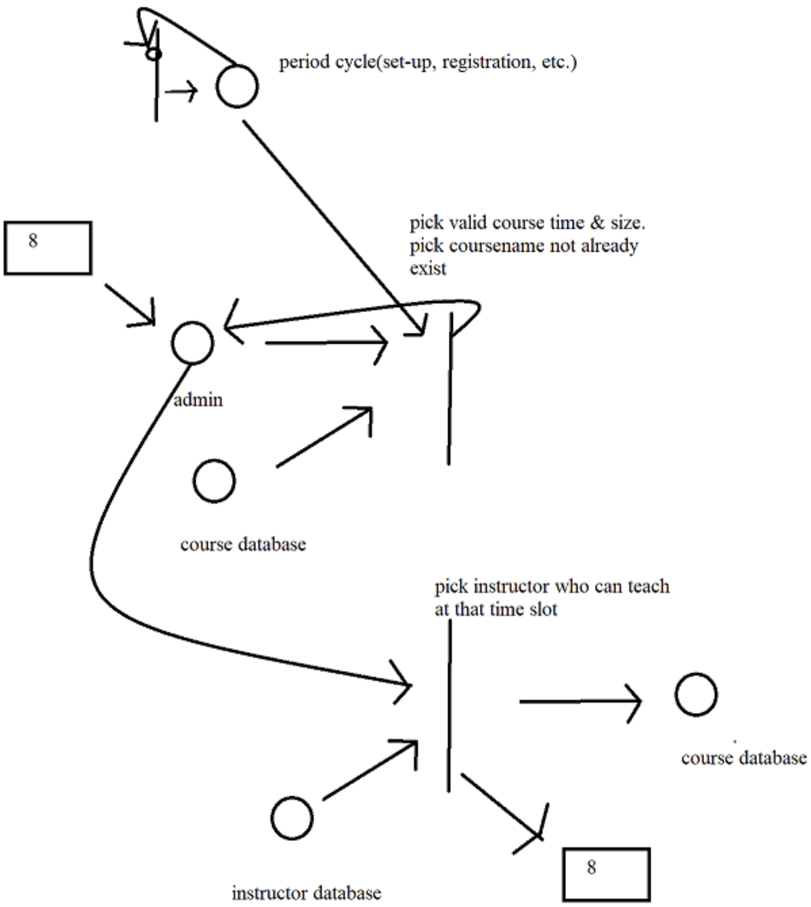
Sign Up as visitor and apply to be student/instructor

Details: Users can sign up as visitors. After they are a visitor, they can apply for instructor or apply for student positions. Visitors who apply for instructor or student need admin to approve the application. If the admin rejects a student who has gpa>3.0 and the program has space, a justification needs to be given.



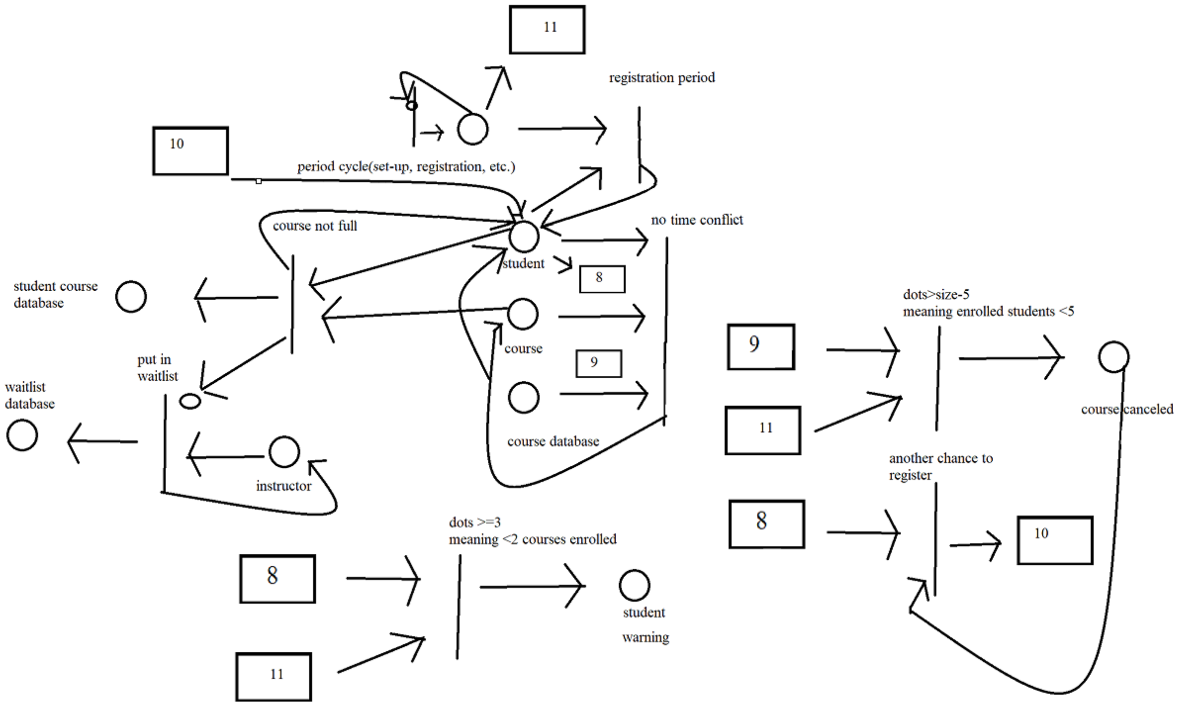
Admin Course creation

Details: When it is the course setup period, the admin can create a course with a coursename, time, and size if the coursename does not exist already. If the instructor they assign to the course is available for that time (not teaching another course and not suspended) then the course is created.


period cycle(set-up, registration, etc.)
pick valid course time & size. pick coursename not already exist
8
admin
course database
pick instructor who can teach at that time slot
course database
instructor database
8

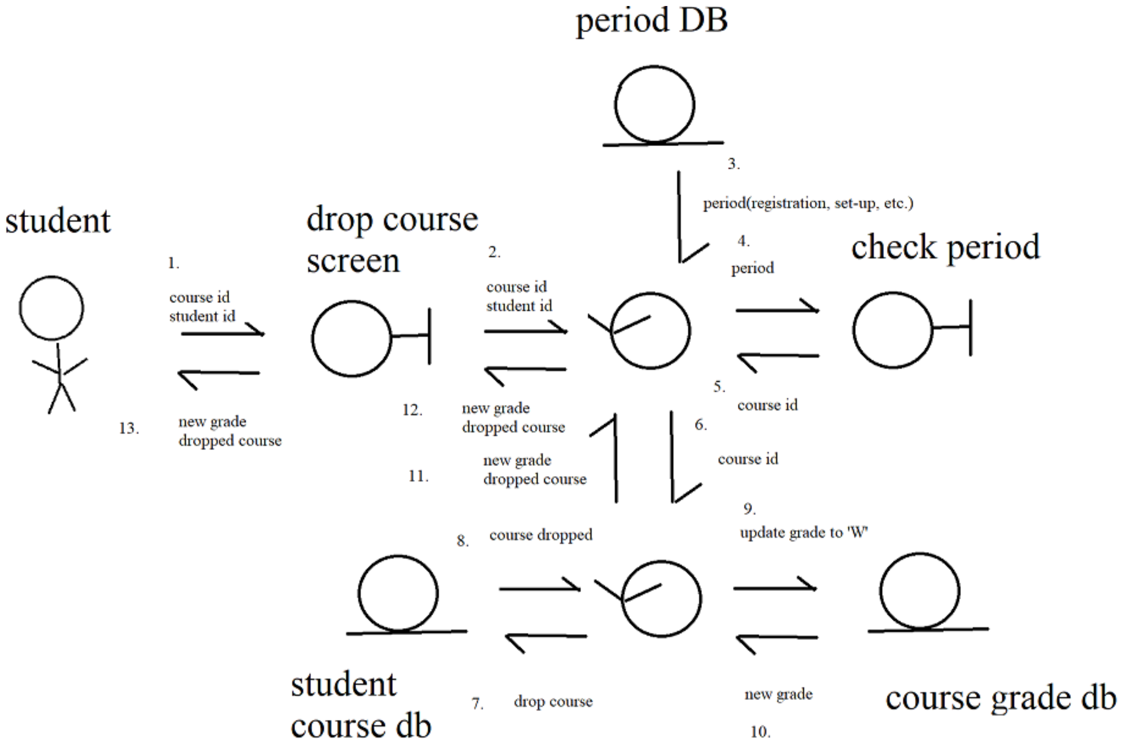
Student course registration

Details: During the course registration period, students can sign up for courses if there is no time conflict and the course is not full. If the course is full, students need the instructor's permission and they will be waitlisted. If the course is cancelled, which happens if it has less than 5 students, students will receive a special opportunity to register again. The student can only register 2-4 courses. Students with less than 2 courses will be warned.



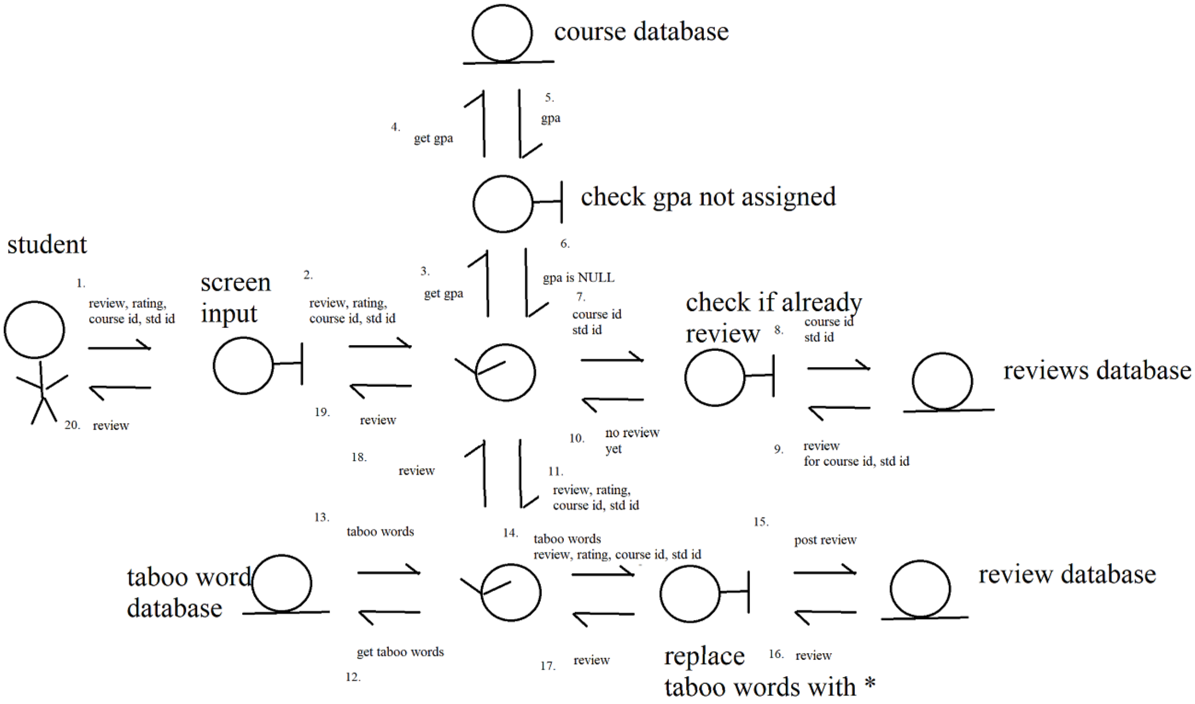
Student dropping course

Details: students can withdraw from a course between registration and grading period with a grade of w.



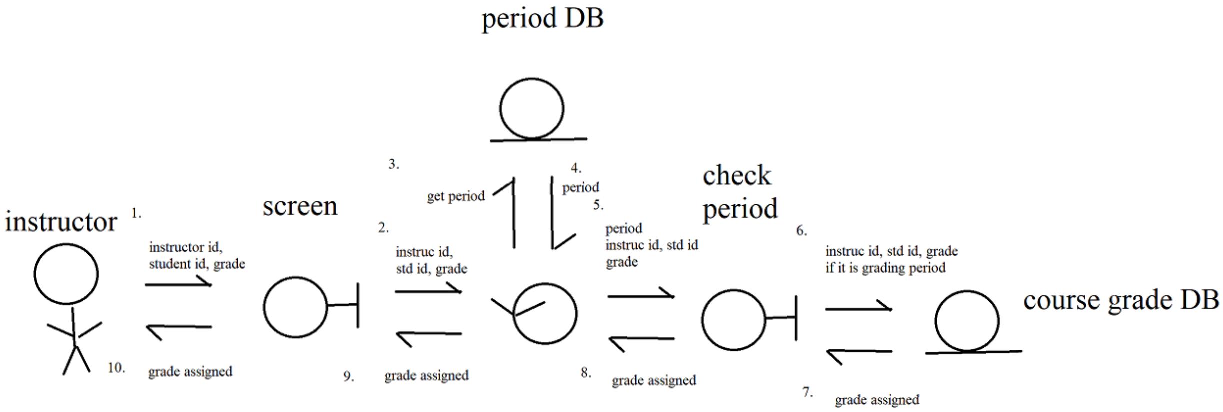
Student writing ratings and reviews

Details: students can write one review and give one rating before they receive a grade from the instructor. If the review has taboo words they will be replaced with *.



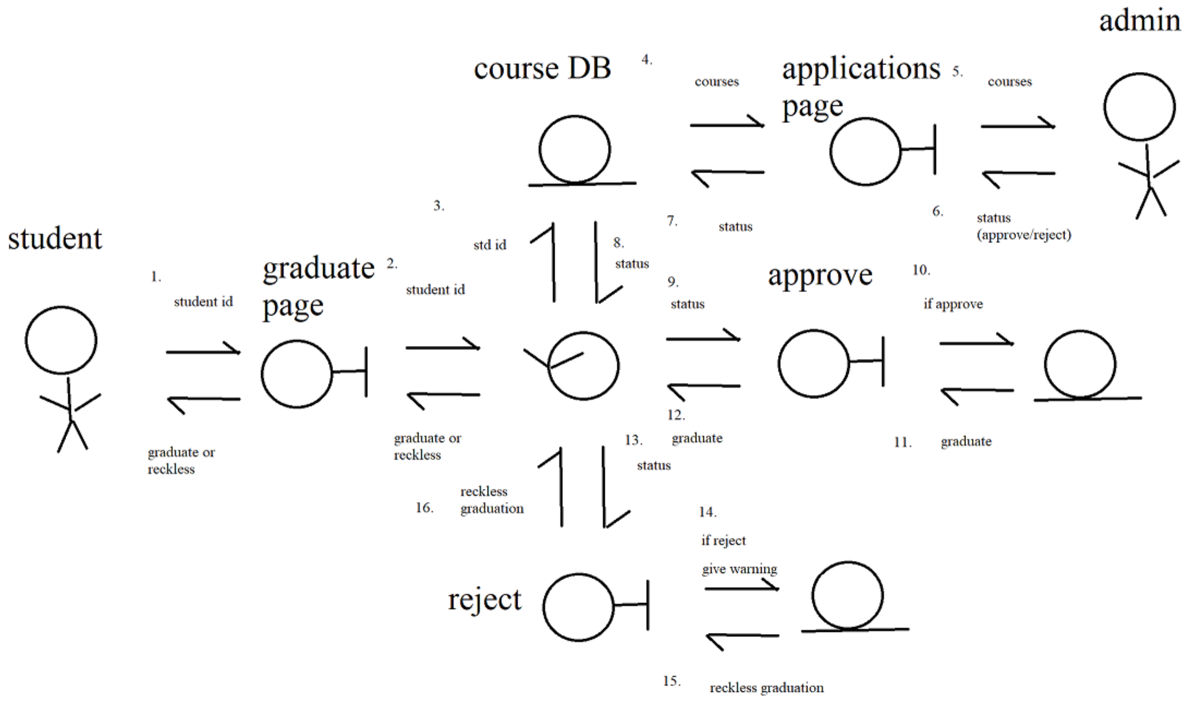
Instructor Assigning grades

Details: Instructors can assign grades to students in their course during the course grading period.


period DB
3. get period
4. period
check period
instructor
1. instructor id, student id, grade
screen
2. instruc id, std id, grade
5. period instruc id, std id grade
6. instruc id, std id, grade if it is grading period
course grade DB
10. grade assigned
9. grade assigned
8. grade assigned
7. grade assigned

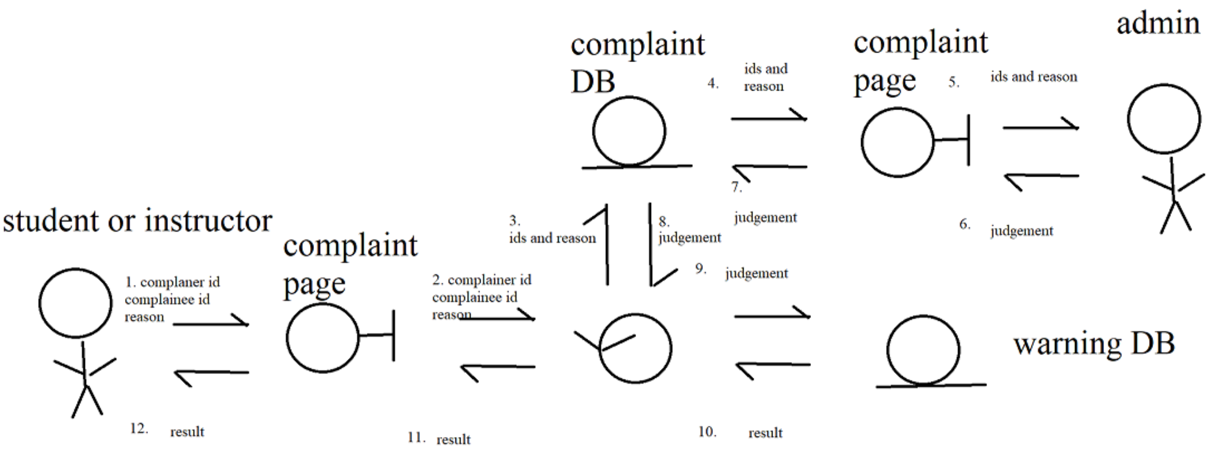
Student Apply for graduation

Details: Students can fill out a graduation application that the admin can approve or reject. If rejected, the student will be warned for reckless graduation. If approved, the student will receive a master degree.

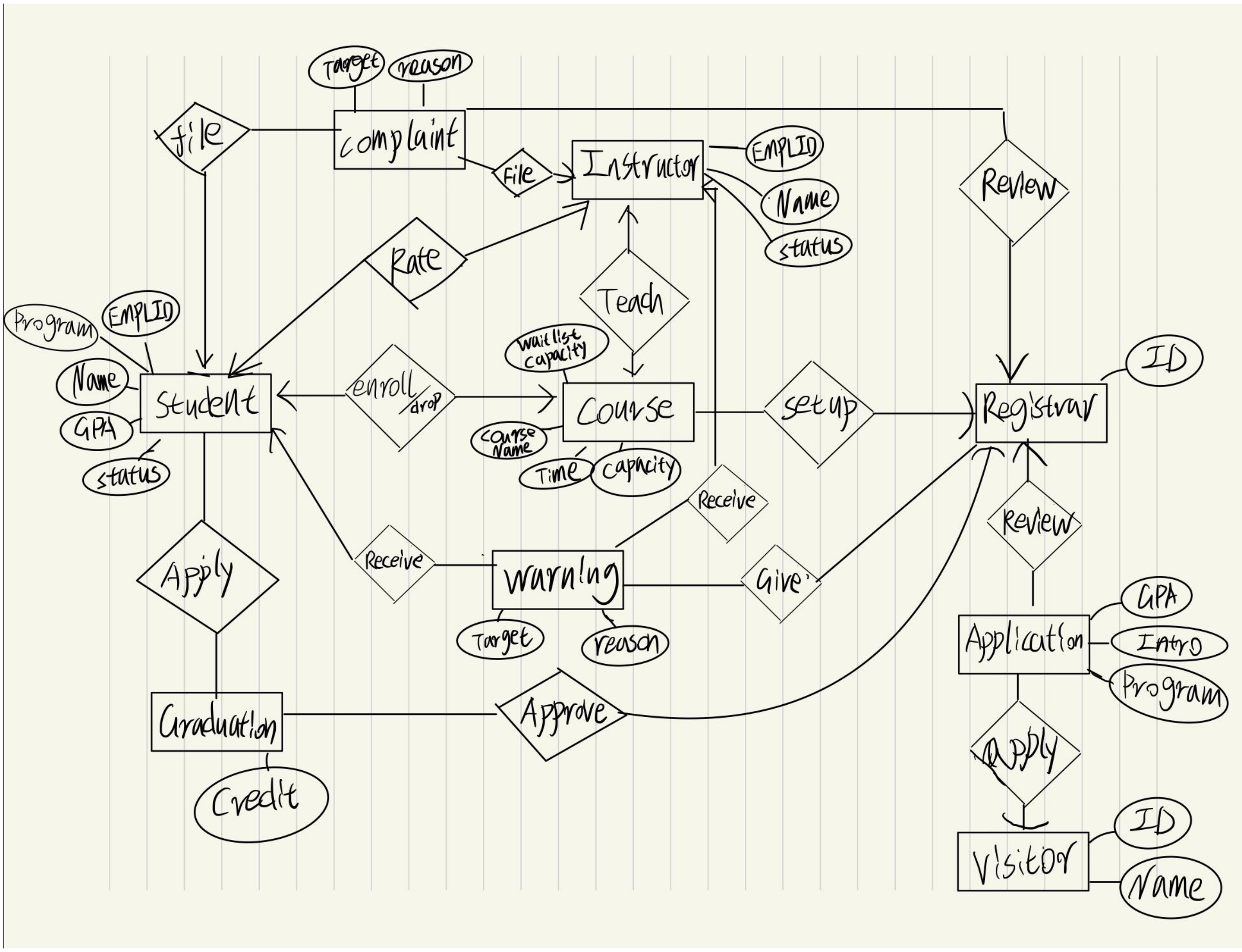


Student and instructor complaints

Details: Instructor/student can file a complaint about an instructor or student. The admin will judge the review and issue the warning accordingly.



3. E-R diagram for the entire system



4. Detailed Design:

1. Login:

```
If login already
    Redirect to homepage
Input password and username
If password and username exist and is in database
    Log the user in/query their information from database
```

2. Logout:

```
If logged in
    If logout button has been clicked
        Logout user/throw away user information
```

3. Register:

```
If not logged in
    Input username and password
    Input password2(confirm password)
    If username is not in database and is valid
        If password is valid and same as password2
            Add new user to database as a visitor
```

4. Change password:

```
If logged in
    Input new password and username
    If password and username is good
        Update database
```

5. Visitor apply for student/instructor:

```
If role is visitor
    If enroll as student/instructor
        Input gpa and program to enroll
        Put in database for admin to view
If role is admin
    Query applications from database
    For each application
        If reject
            If (program not full) and (gpa>3.0) and (apply for student)
                Provide justification and add to database
        If accept
            Generate password and username
            Add into database
```

6. Grade students:

```
If role is instructor
    If grading period
```

```
           Query classes instructor teach
           For each class
               Query all the students
               For each student
                   Assign student a grade for that class
                   Update database
7. File complaint against students/instructor:
   If role is student or instructor
       Select student/instructor to complain about
           Input complaint message
           If instructor input to deregister or warn
           Put into complaints database
   If role is admin
       Query complaints from database
           For each complaint
               If instructor wants to deregister student
                   If accept
                       Kick student from course
                       Update course database
                   Else
                       Warn instructor
                       Update warning database
               Else
                   Warn one side
                   Update warning database
8. Enroll course
   If role is student
       If registration period or student have special permission
           Query courses from database
           Select course to enroll
           If course has not been taken before or if grade<F
               If course to enroll not full
                   If no time conflict
```

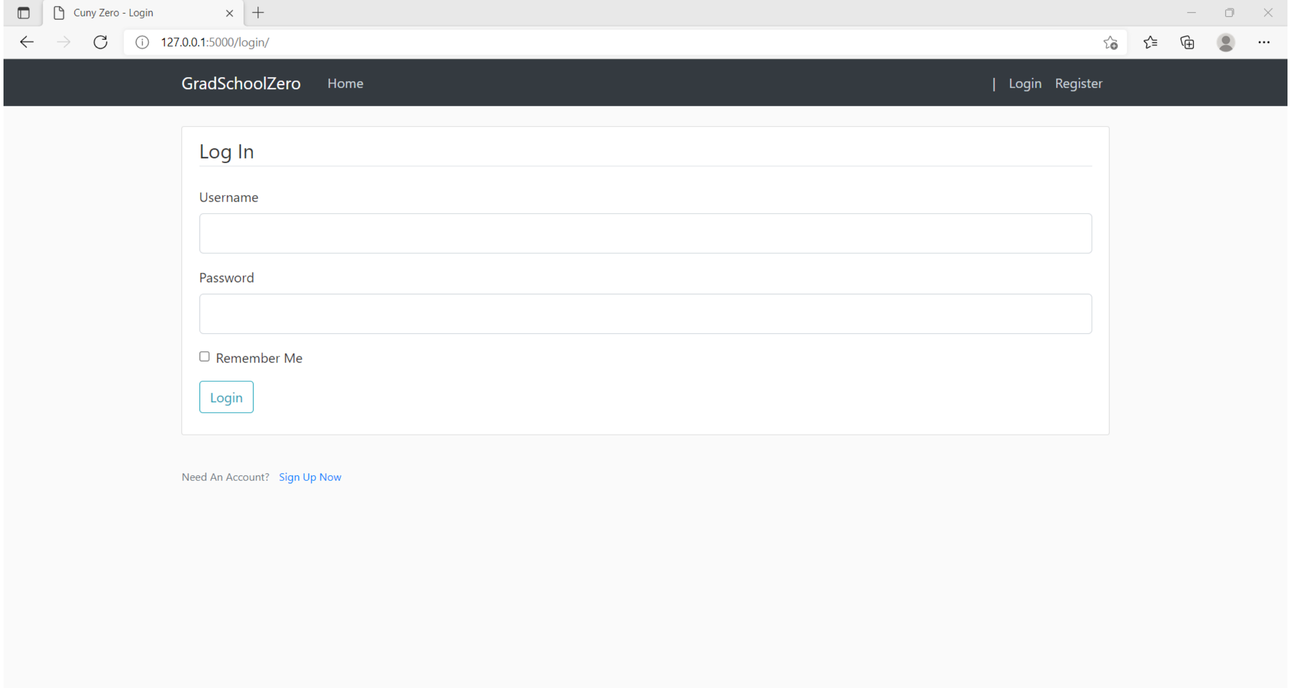
```
                            Enroll course
                            Update course database
                Else
                    Put into waitlist
    If role is instructor
        Query courses taught
        For each course
            Query waitlist
            For each student in waitlist
                If accept
                    Update course database
9.  Drop course:
    If role is student
        If course registration period or running period or grading period
            Query student’s courses
            Select course to drop
            Update grade to ‘W’
            Remove course from student’s course
            Update database
10. Write review:
    If role is student
        Query student’s course
        Select course to post review
        If course grade not posted
            Input rating
            Check rating between 1-5
            Input review
            If review has >=3 taboo words
                Add 2 warning to student
                Update warning database
                return
            Else
```

```
                If taboo words>=1
                    Replace taboo words in review with *
                Add 1 warning to student
                Update review and warning database
            Update avg course rating,
            Avg course rating = avg course rating + (this rating - avg course rating)/(num of ratings + 1)
            Number of ratings ++
            Update rating database
11. Student apply graduation
    If role = student
        If apply for graduation
            If length(query student courses that are finished)<8
                Warn student
                Update warning database
            Else
                Update graduation application in database
    If role = admin
        Query all graduation applications
        For each application
            If accept
                Give master degree
                Update student database
            If reject
                Warn student
                Update warning database
12. Change period
    If role is admin
        Query current period
        Choose to advance to next period
        If change to class running period
            Query courses
            For each course
                If course < 5 students
                    Query students in course
                    For each student
                        Give special permission to register again
                    Cancel course
                    Warn course instructor
                    Update database
            Query students
            For each student
                If course<2
                    Warn student
                    Update warnings database
            Query instructors
            For each instructor
                Query courses they are teaching
                If course canceled and courses = NULL
                    Suspend instructor for 1 semester
                    Update database
    If advance to course set-up period
        Query all courses
            For each course
                Change status to finished
                Update database
        Query all students
            Query all student’s course
            Count = 0
            Semester gpa = 0
            For each course
                If course grade = F and failed course before
                    Delete student from database
                Else If course grade = F
                    Mark student course as failed
                    Update student course in database to allow retake
            If Semester gpa == 0
                Semester gpa = student course gpa
                Count++
            Else
                Semester gpa = semester gpa + (current course gpa - semester gpa)/(count+1)
                count++
        If student overall gpa>3.5 or student semester gpa
            Label as honor student
            Remove warning if there is any
            Update warning database
        If overall gpa>=2 and overall gpa<=2.25
            Warn student about interviewing admin
            Update warnings database
    Query all instructors
    For each instructor
        If suspended 1 semester already
            Unsuspend
            Update database
        Query classes instructor teach
        For each class instructor teach
            If class gpa>3.5 or class gpa<2.5
                If Select to warn instructor
                    Update warning database
                Else
                    Fire instructor
                    Update instructor database
            Query all the students
            For each student
                Query grade
                If grade is null
                    Add 1 warning to instructor
                    Update database
```

Return

System Screens:

Login Page:



Log In

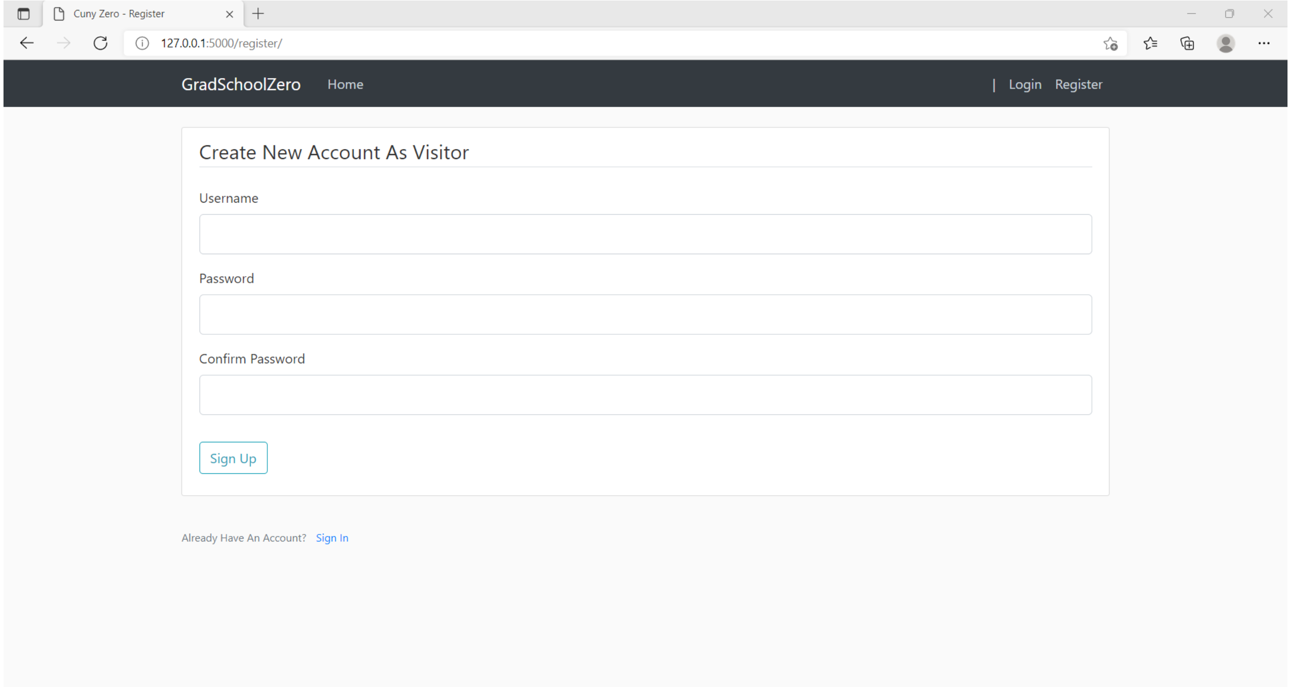
Username

Password

□ Remember Me

Need An Account? Sign Up Now

Register Page:



Create New Account As Visitor

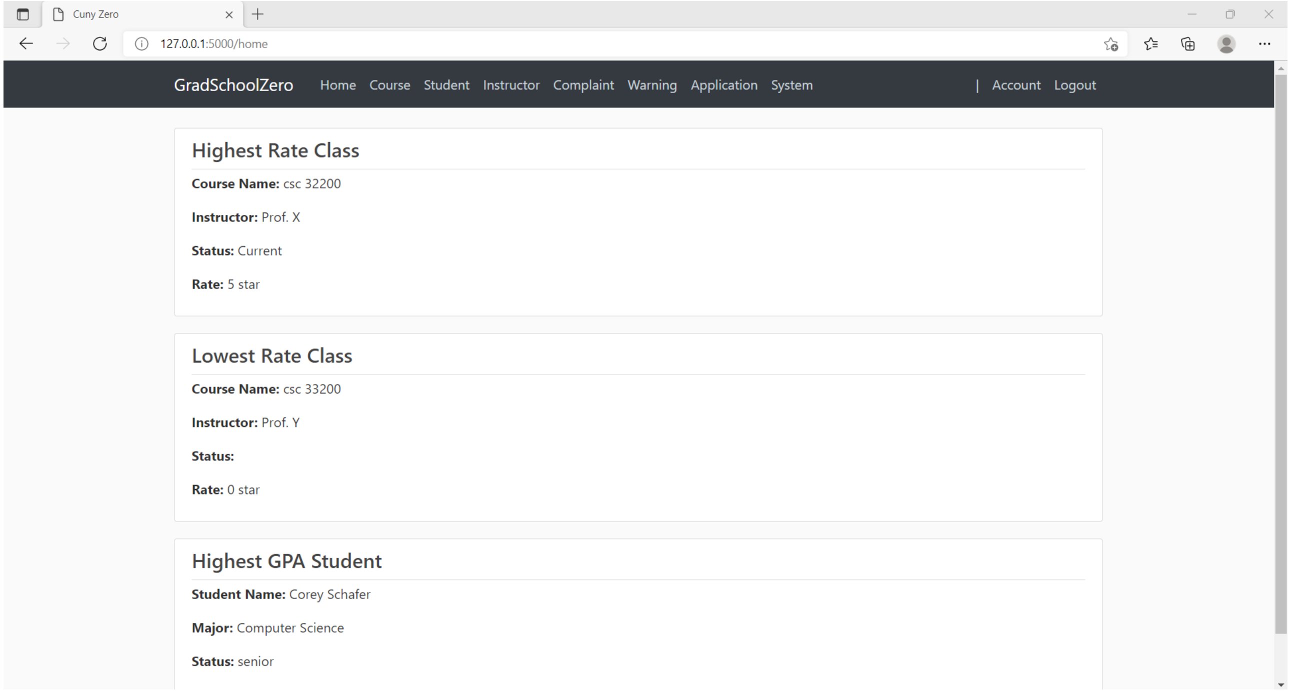
Username

Password

Confirm Password

Already Have An Account? Sign In

Home Page:



Highest Rate Class

Course Name: csc 32200

Instructor: Prof. X

Status: Current

Rate: 5 star

Lowest Rate Class

Course Name: csc 33200

Instructor: Prof. Y

Status:

Rate: 0 star

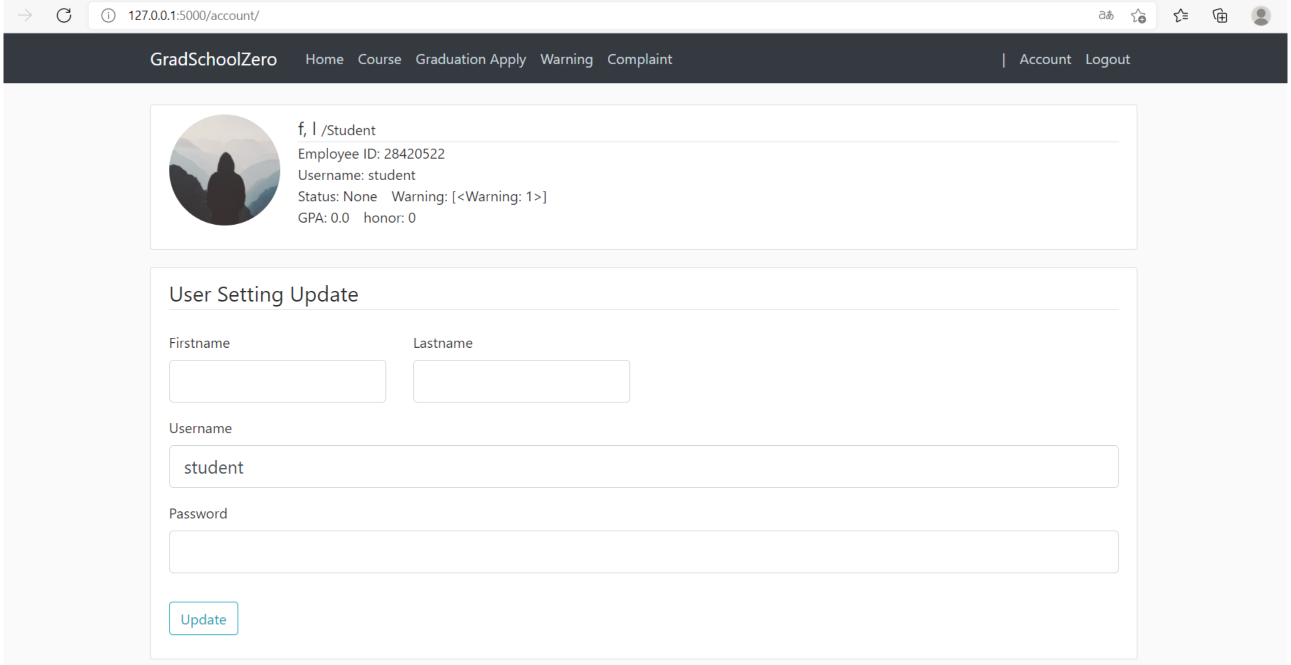
Highest GPA Student

Student Name: Corey Schafer

Major: Computer Science

Status: senior

Account Setting Page:



f, l /Student
Employee ID: 28420522
Username: student
Status: None Warning: [<Warning: 1>]
GPA: 0.0 honor: 0

User Setting Update

Firstname

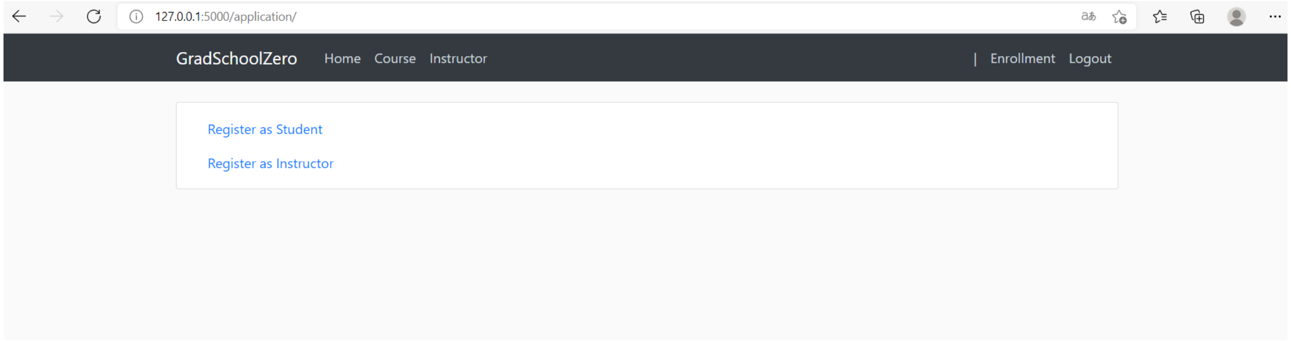
Lastname

Username

student

Password

Visister account register as Student/Instructor:

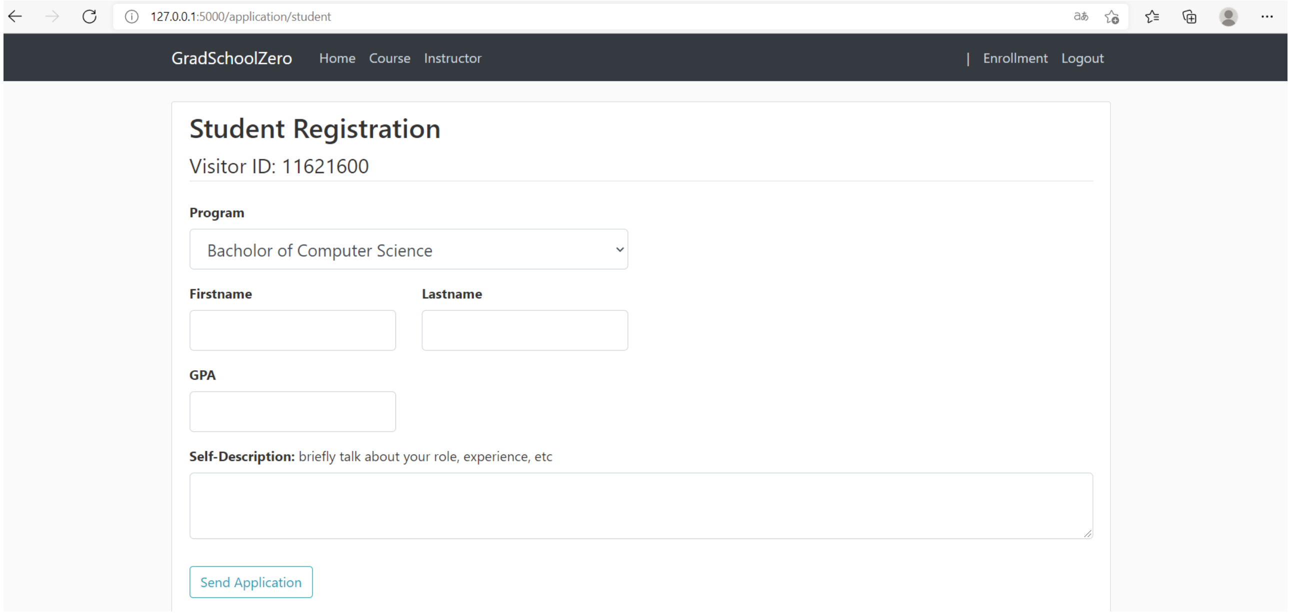


127.0.0.1:5000/application/

GradSchoolZero Home Course Instructor | Enrollment Logout

Register as Student

Register as Instructor



127.0.0.1:5000/application/student

GradSchoolZero Home Course Instructor | Enrollment Logout

Student Registration

Visitor ID: 11621600

Program

Bacholor of Computer Science

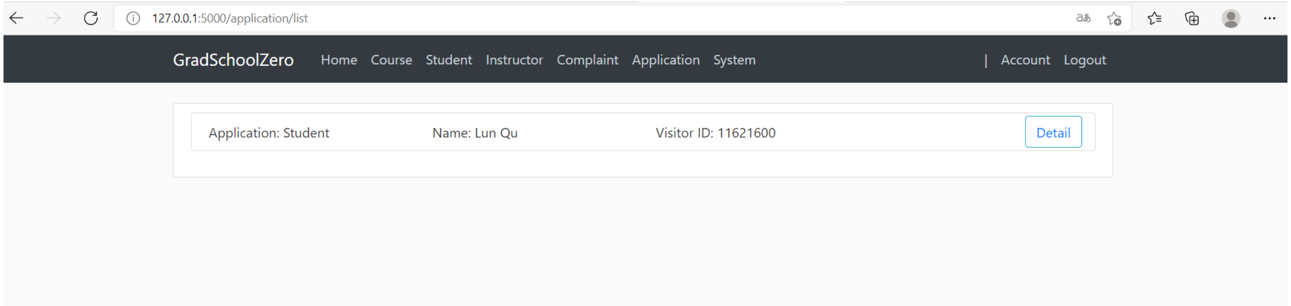
Firstname **Lastname**

GPA

Self-Description: briefly talk about your role, experience, etc

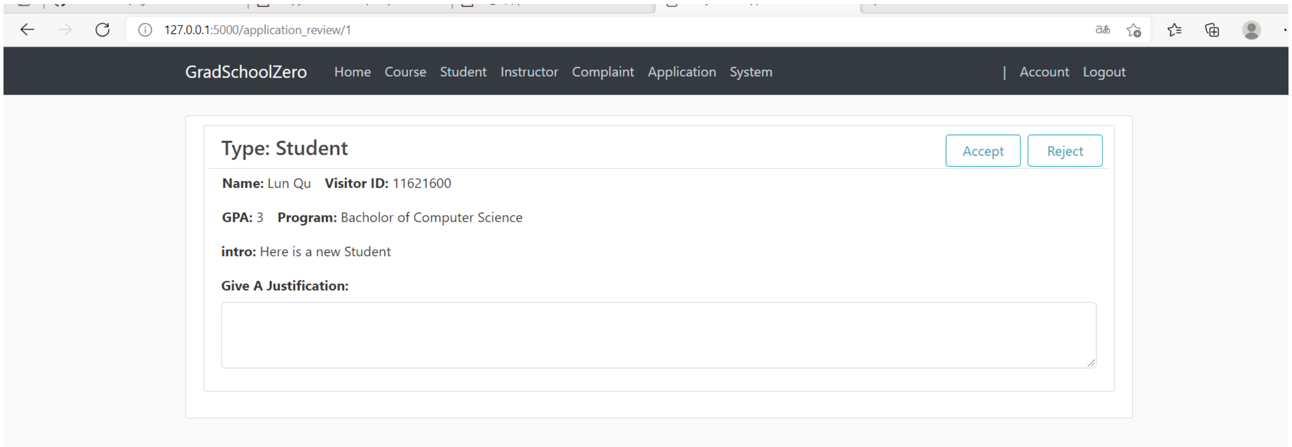
Send Application

Admin deal with application:



127.0.0.1:5000/application/list

Application: Student Name: Lun Qu Visitor ID: 11621600



127.0.0.1:5000/application_review/1

Type: Student

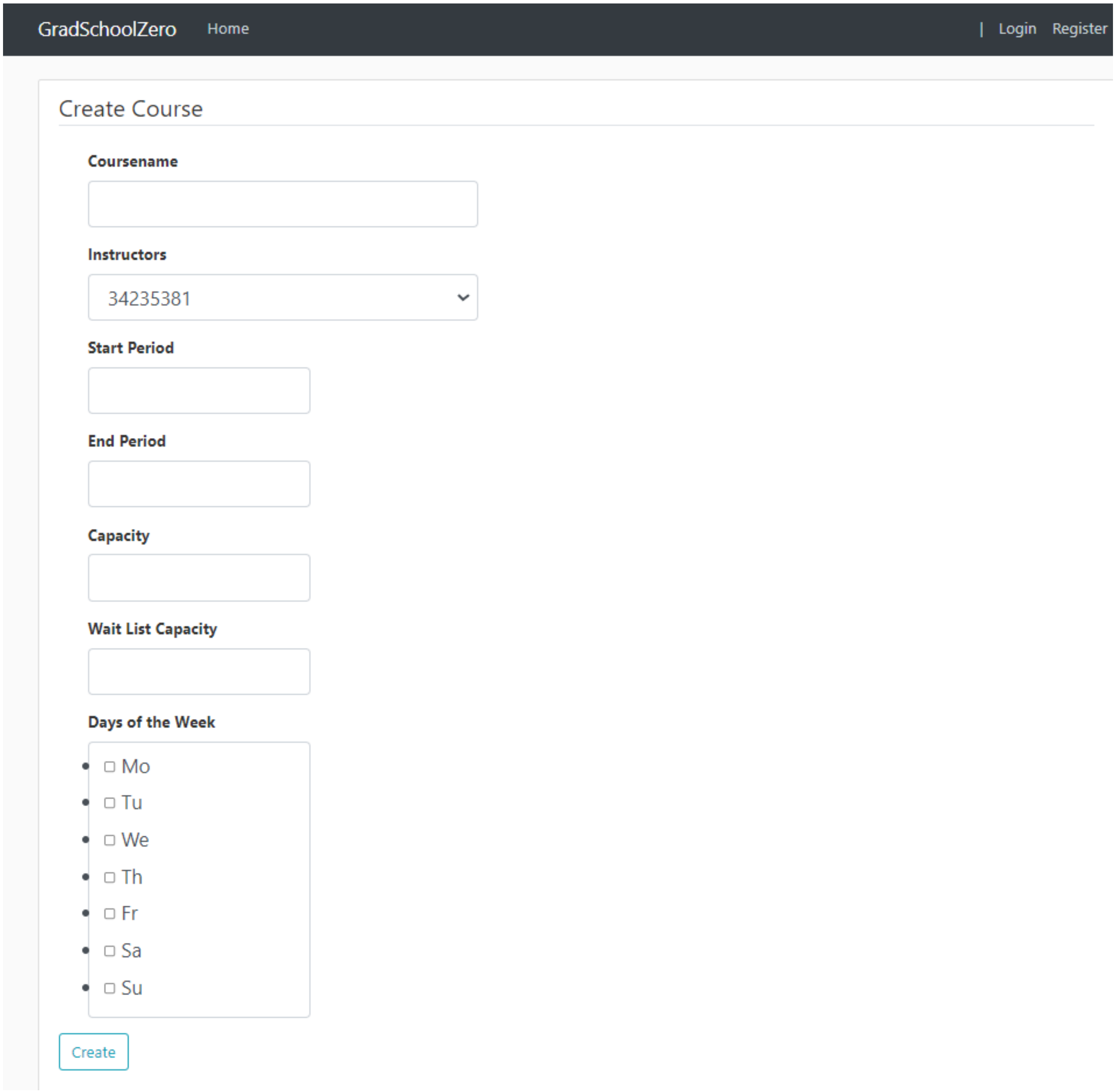
Name: Lun Qu **Visitor ID:** 11621600

GPA: 3 **Program:** Bacholor of Computer Science

intro: Here is a new Student

Give A Justification:

Create Course Page:



Create Course

Coursename

Instructors

34235381

Start Period

End Period

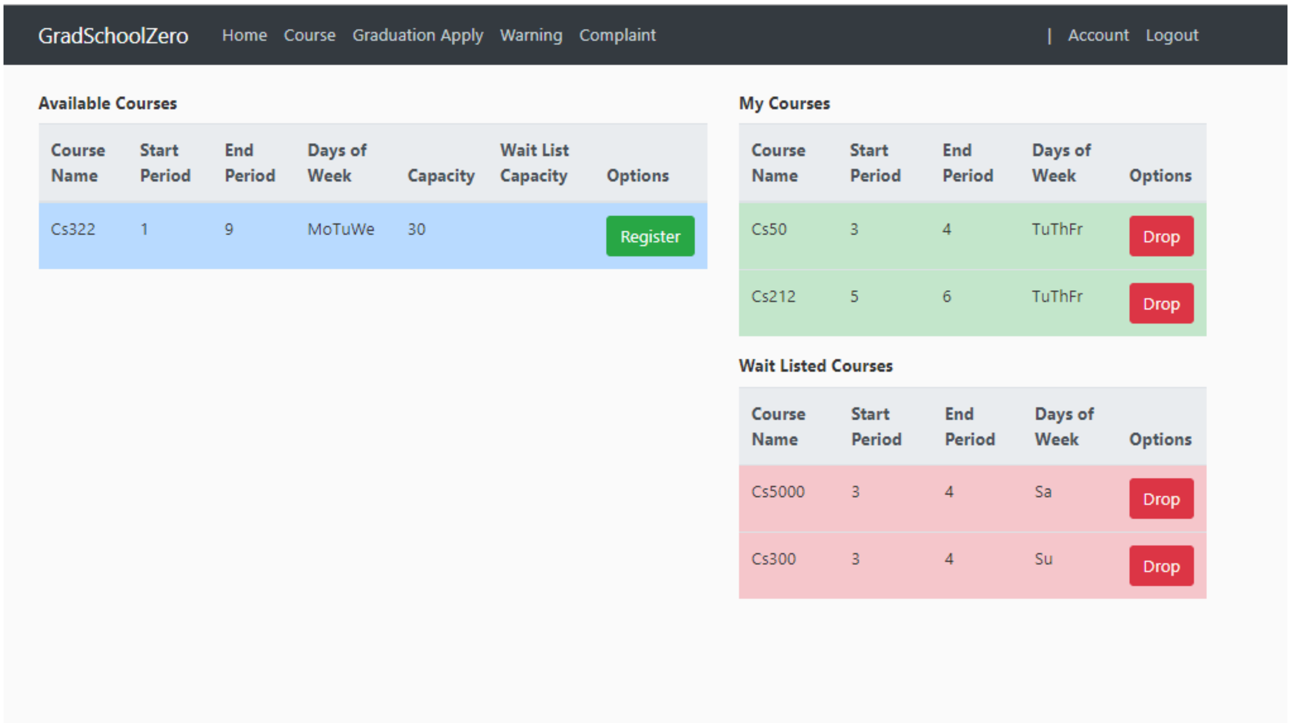
Capacity

Wait List Capacity

Days of the Week

- □ Mo
- □ Tu
- □ We
- □ Th
- □ Fr
- □ Sa
- □ Su

Register Course Page:



Available Courses

Course Name	Start Period	End Period	Days of Week	Capacity	Wait List Capacity	Options
Cs322	1	9	MoTuWe	30		Register

My Courses

Course Name	Start Period	End Period	Days of Week	Options
Cs50	3	4	TuThFr	Drop
Cs212	5	6	TuThFr	Drop

Wait Listed Courses

Course Name	Start Period	End Period	Days of Week	Options
Cs5000	3	4	Sa	Drop
Cs300	3	4	Su	Drop

Memo Group Meeting:

We meet twice a week, starting a couple weeks ago, to discuss what features we work on for the week and to share what we did last week. Overall the group's progress and synergy is good, although one member is a bit behind in the group work, they should be able to catch up pretty easily.

Estimate:

1st week:

- Set-up github

2nd week:

- Finished Login and Register
- Set-up the database schema

3rd week:

- Implemented Visitor application to become a student/instructor
- Implemented Admin page to reject/accept applications

4th week(This Week):

- Implemented course enrollment/dropping

Plan for future:

- Implement complaint and warning pages/system
- Implement instructor grading
- Implement student reviews/ratings
- Include Period cycle(set-up, registration, running, grading) logic with existing work
- Include other task logic with existing work

Date	Duration	
11/04/2021	60 mins	- Set-up github
11/09/2021	90 mins	- Set-up the database schema - Finished setting up page layout - Finished Login and Register
11/16/2021	90 mins	- Implemented Visitor application to become a student/instructor - Implemented Admin page to reject/accept applications
11/18/2021	90 mins	- Implemented course enrollment/dropping
11/23/2021	60 mins	- Implement complaint and warning pages/system

Github:
https://github.com/zwen000/CunyZero