Final Project Proposal

MPCS 51410
Haoning Chen

After reading the requirement specification of the course registration system REGIE, I decided to choose the following requirements to implement.

1. User Accounts.

This bounded context handles the **data and account information** of both student and instructor accounts. My idea is that, user accounts can be implemented as a base class, and both student accounts and instructor accounts are the subclasses extending the base class. This part needs to co-operate with user database.

(1) Requirements of **instructor accounts** to be implemented:
    a. Instructor name.
    b. UserID and password of the account.
    c. Department(s)/Division(s) of the instructor.
    d. The courses and the sections of courses that the instructor teaches.

(2) Requirements of **student accounts**:
    a. student name.
    b. UserID and password.
    c. Major(s) of the student.
    d. The courses and the sections of courses that the student currently has selected.
    e. Grades of all the courses the student has taken.
    f. Any restrictions the student currently have.

2. Course Information / Course Data.

This bounded context handles **all the information related to courses**. Each section of courses has its own course number and data entry (object). The Course class needs to correspond with course database.

Requirements to be implemented:
    a. Course name;
    b. Course number (distinct from different sections of courses);
    c. Instructor of the course;
    d. Department/Division that the course belongs to;
    e. Time and classroom location of the course;
    f. Maximum number of students of the course;
    g. The roster that lists the name of students who register the course;
    h. Prerequisites of the course;
    i. Description and features of the course.

3. User Operations.

This bounded context handles all the **operations** that can be made by instructors and students in the registration system.

(1) Requirements of instructors' operations to be implemented:

    a. View course schedules and room locations.

    b. View the courses (sections) the instructor teaches and the rosters of registered students.

    c. Assign grades for students and modify grades.

    d. Modify user account information (password, department, etc.).

    e. Edit course description/ features for each course.

    f. Approve/Deny students' special requests for registration of a course.

    g. Report grades of all the students of a course to the University.

(2) Requirements of students' operations:

    a. Browse/Search available course offering, according to course name, department/division, time, classroom location or instructor name.

    b. Register a course. (Up to three courses are allowed to be registered by a student. Before eventually register the course, system automatically check whether the student have any restrictions, the number of registered courses of this student is fewer than 3, and whether the prerequisite requirements of this course are fulfilled.)

    c. Drop courses that currently are registered by the student.

    d. List all the courses and the schedules of the courses the student currently has selected.

    e. View their course grades.

    f. Print their current transcript.

    g. Ask for consents to instructors for registering some courses.

    h. View restrictions of the student.

    i. Modify their account information.

The following contents are the functions that will not be implemented in my proposal prototype:

1. There are no distinctions on the roles of Professors, Associate Professors, Assistant Professors, Instructors, and Teaching Assistants, etc. All the instructors in the prototype are the same. And each section of courses is taught by exactly one instructor. (I.e., no TA in a course.)

2. There are no distinctions on the roles of full-time students and part-time students. All the students in the prototype are the same.

3. Since there are no TAs in the system, students cannot be instructors simultaneously.

4. Course labs will not be implemented in the prototype.

5. The maximum number of courses a student can register is hard-limited to 3. I.e., course overload informing will not be implemented in the system.

6. Payroll system of instructors will not be implemented.

7. Alternative course recommendation will not be implemented. I.e., if a student attempts to register for a section of a course which is closed (filled the maximum number of students enrolled), the system will not search for other sections of the course as alternatives to the user.

8. A course with fewer than 5 students will not be canceled automatically.

9. All the system maintenance functions (for the role of system administrator) will not be implemented.

All the content above is my proposal for the REGIE prototype.