Use Cases Part 2

Table of Contents

Use Cases **1**

UC1: View Courses 2

UC2: Register Course 3

UC3: Drop a Course 4

UC4: View Timetable 5

UC5: Input Office Hours 6

UC6: Indicate Conflict between Office Hours and Timetable 7

UC7: View Classlist 8

Updated use case diagram for Academic Planner subsystem

UC1: View Courses

Purpose: To view all courses as well as their information in order to have a better understanding of course offerings.

Stakeholder List:

- **Student**: wants to view course information in order to plan their timetable and make informed choices for scheduling their degree.
- **Applicant**: wants to view course information in order to plan their degree and make informed choices.
- **Professor**: wants to view a specific course they may be interested in teaching.
- **System Admin/Tech Support**: wants to view courses and their information to review the data before seeing what needs to be changed or updated if they want to override the information as well as seeing if the changes they made are correct.

Primary Actor: Student

Preconditions: Student must be authenticated into GryphHub.

Postconditions: Student receives a list of courses including the description, prerequisites, course code, number of credits, offering type (distance education or regular), location if applicable, semester offered, seats available, lab time, meeting time, and restrictions.

Basic Flow:

1. The student chooses to view course offerings.
2. The user is given a list of courses in alphabetical order
3. The user is able to enter a search attribute for a course (ex. course name, course code) or filter the course list by type or level.
4. The system searches the course information database for any records that match the search attribute and displayed the filtered list of courses.
5. The users examine and assess the information and use it for their intended purpose.

Alternate Flow (Exceptions):

3a. Invalid search attribute entered.

1. System signals an error with a message prompt and rejects the entry.
2. System prompts the student, applicant, student counselor or System Admin/ Tech to re-enter a valid search attribute.

4a. Search attribute not found.

1. The system displays a message indicating that the course was not found given the current search attribute.
2. The student can modify the search attribute and try again, returning to step 3 of the Basic Flow

—----------

UC2: Register Course

Purpose: To allow a student to register for a course

Stakeholder List:

- **System Admin / Tech Support**: wants to be able to register a student for a course and have access to rectify possible errors.
- **Student**: wants to be able to register a for a course.

Primary Actor: Student

Preconditions: The student must be logged into the system and have access to view courses and the academic planner.

Postconditions: Student has successfully registered for a course or is registered in the waitlist for the course.

Basic Flow:

1. The student views the list of courses available to register for.

2. The student selects a course they want to register for.
3. The student chooses a section of the course they want to register for.
4. The system checks the student's eligibility for the selected section by verifying the pre-requisites and course availability
5. The system adds the course to the student's timetable.
6. The system sends a confirmation of the registration to the student.

Alternate Flow (Exceptions):

3a. All sections for the course are full
    1. The system notifies the student that all sections for the selected course are full
    2. The system adds the student to the waitlist and returns to step 1 of the Basic Flow

4a. The student is not eligible to register for the course:
1. The system notifies the student that they are not eligible to register for the course and provides information on the prerequisites required to be eligible.
2. The student can choose to register for the prerequisites first and return to register for the desired course later.
3. Return to step 2 of the Basic Flow.

—----------

UC3: Drop a Course

Purpose: To allow students the ability to remove courses from their timetable

Stakeholder List:
- **Student**: wants to remove a course from their timetable for various reasons
- **Student Councilor**: wants to be able to remove a student from a course on behalf of a student with permission.
- **System Admin / Tech Support**: wants to be able to register a student for a course and have access to rectify possible errors.

Primary Actor: Student

Preconditions: Student must be registered in at least one course

Postconditions: The student has successfully dropped a course and it has been removed from their timetable.

Basic Flow:

1. Student is able to view their timetable and select the course that they want to drop.
2. Student confirms the act of dropping a selected course
3. The course is removed from the student's timetable.
4. The student is also removed from the class of that particular course, i.e., the classlist is updated.
5. Timetable will update itself with its new changes.

Alternate Flow:

1a. The Drop Deadline has Passed

1 The student is unable to remove a course from timetable

2 The student is prompted to reach out to the student councilor in order to make a change

5.a. Timetable not updated after schedule change

1 Technical issue, the user has the ability to reach out to Student Counselor or System Administrator / Tech support in order to make scheduling changes

—----------

UC4: View Timetable

Purpose: The purpose of this use case is to allow the student to view their timetable in order to have a visual understanding of their schedule.

Primary Actor: Student

Stakeholders List

- **Student counselor**: responsible for helping students with academic planning and scheduling.
- **Professor**: responsible for teaching courses and scheduling office hours.
- **System Admin / Tech Support**: wants to be able to register a student for a course and have access to rectify possible errors.

Preconditions: The student must be authenticated to access the university portal.

Postconditions: The student can view their timetable for the current academic semester.

Basic Flow:

1. The student attempts to view their time table
2. The system creates the timetable for the current academic semester based on student's courses
3. The system displays the timetable for the student, which includes the course name, course code, course instructor, class time, and location.
4. The system notifies the student that the timetable can be viewed and student reviews their timetable

Alternate Flow:

2a. The system is unable to create the timetable

1. The system displays an error message indicating that the timetable could not be created.
2. Return to step 2 of the basic flow

UC5: Input Office Hours

Purpose: For professors to be able to input their office hours

Stakeholder List:

- **System Admin / Tech Support:** To immediately identify and resolve any problems that could pop up during the process.
- **Student:** To be able to view office hours entered by the professors teaching the courses they are currently enrolled in.

Primary Actor: Professors

Preconditions: The professor must be teaching (or plan on teaching) course(s) and has scheduled office hours that are to be uploaded

Postconditions: The office hours are uploaded to the academic planner and students taking the course are able to view it

Basic Flow:

1. The professor accesses the academic planner and the courses they are teaching
2. The professor inputs the Course Id and time for their office hours
3. The professor uploads the office hours

Alternate Flow (Exceptions):

1a. Unable to view the courses they are teaching.

1. Verify that they have already added courses first.
2. Technical issue, the user has the ability to reach out to System Administrator / Tech support.

3. Return to step 1 of Basic Flow

1b. The professor is unable to access the academic planner
1. Technical issue, the user has the ability to reach out to System Administrator / Tech support.
2. Return to step 1 of Basic Flow

2a. The professor entered a course they do not teach
1. System rejects the request and signals error with a message prompt\
2. Return to step 1 of Basic Flow

—----------

UC6: Indicate Conflict between Office Hours and Timetable

Purpose: For the professor to be able to schedule their office hours and avoid conflicts

Stakeholder List:
- **System Admin / Tech Support:** To immediately identify and resolve any problems that could pop up during the process.
- **Professor**: To be able to verify that there are no conflicts in their schedule while they are uploading timings.
- **Student**: To have access to a clean timetable for all the courses they are currently registered for.

Primary Actor: Professor

Preconditions: The professor must be teaching (or plan on teaching) courses and has entered scheduled office hours

Postconditions: The students are now able to see the professors scheduled office hours.

Basic Flow:
1. The Professor accesses the academic planner and is able to view the courses they are teaching.
2. The Professor selects the course they would like to input office hours for
3. The Professor inputs the scheduled hours.
4. The System throws an error and the Professor is required to perform step 3 again.
5. Once there is no conflict, the office hours are updated to the timetable.

Alternate Flow (Exceptions):

1a. The professor isn't able to view their courses

1. Technical issue, the user has the ability to reach out to System Administrator / Tech support.

UC7: View Classlist

Purpose: The purpose of this use case is to allow the professor to view their classlist in order to have a visual understanding of their schedule.

Primary Actor: Professor

Stakeholders List

- **Student counselor**: wants to be able to verify if a student is in a classlist
- **Professor**: wants to view the classlist for a particular course

Preconditions: The student must be authenticated to access the university portal.

Postconditions: The professor can view the classlist for a course

Basic Flow:

1. The professor enters the course ID for which they want to view the classlist.
2. The system retrieves the classlist for the course.
3. The system displays the classlist, which includes the course ID, student name, and student ID
4. The professor reviews the classlist.

Alternate Flow:

1a. Course ID not found.

1. The system displays a message indicating that the classlist was not found given the current search attribute (course ID)
2. The professor can modify the search attribute and try again, returning to step 1 of the Basic Flow