

Use Case UC1: Course Registration

Author

Buğra Ekuklu, 150120016.

Scope

Student Administration Software System

Stakeholders and Interests

- Student: Wants fast and stable registration to courses without errors.
- Student Administration: Wants automatic execution of planned enrollment of students to the opened courses and see failed attempts. This includes enrollment quotas and restrictions to be applied.

Preconditions

Student is authenticated.

Postconditions

Student is enrolled to the course if quota is remaining and there are no restrictions applied, otherwise, if quota was not available, a message should be shown to the student indicating course is full, else, a message should be shown regarding restriction of effect. Central repository should increment the number of enrolled students to the course atomically in case of success. If the attempt was failed, it should be saved to the database, since capacities of the courses might be increased by the administration in future.

Base Flow

1. Student enters to the *course registration panel*.
2. Student enters the *course registration number* (CRN).
3. Student submits the form by pressing '*Enter*' or clicking to the button.
4. System determines if course is available in terms of *quota* and *restrictions*.
5. System assigns the course to the student and increments the number of enrolled students in selected course.
6. System shows a success message indicating student is successfully accepted to the course.

Student repeats steps 2-6 until exiting.

Alternative Flows

2a. Cancel operation:

1. Student cancels the registration process.

2b. Invalid CRN:

1. Student enters an invalid course registration number (CRN).
2. System checks the input and rejects the request since input is malformed.
3. System shows a failure message indicating *CRN* was invalid.

2c. Non-existent CRN:

1. Student enters a non-existent course registration number (CRN).
2. System checks if given course is available and rejects the request since there was no course found.
3. System shows a failure message indicating course was not found.

4a. Enrollment limiting:

1. System finds course is not available due to cap.
2. System logs the registration attempt.
3. System shows a failure message indicating course is full.

4b. Registration restrictions:

1. System finds course is not available due to a restriction.
2. System shows a failure message indicating course could not be assigned.

Special Requirements:

- After successful authentication stage, system should be available to determine the sender of the request, not just the authenticity.
- The system should be scalable in terms of performance in order to serve many students as much as possible.

Open Issues:

- Are there uncapped courses?
- What kinds of restrictions might be applied?
- Will system query external sources (i.e. library service) in order to maintain restrictions?