3/17/2025

SC Assignment 3

Group Member:

Ahsen Rasheed(leader)

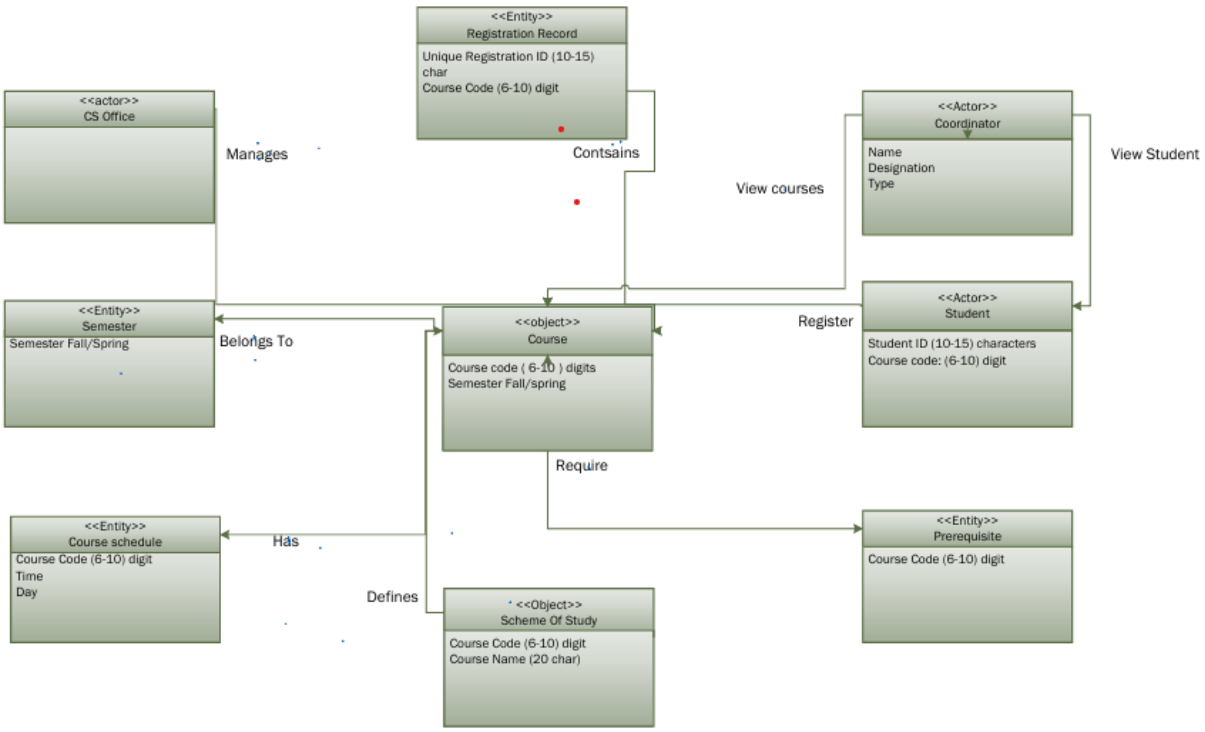
Ahmed Nazir

Yaman Bangash

Submitted To

DR.ONIAZ MAQBOL

Domain Model



Description:

This domain model represents a student course registration and tracking system. It defines key entities such as students, coordinators, courses, semesters, prerequisites, and schedules. The CS Office manages registration records, which track student enrollments while enforcing prerequisite requirements. Coordinators can view student data, and the system ensures courses belong to specific semesters with defined schedules.

Cardinality:

1**. CS Office manages Semesters**

One-to-Many (1:M) → A single CS Office manages multiple Semesters.

2. **Semester belongs to Course**

One-to-Many (1:M) → A single Semester can have multiple Courses, but a Course belongs to only one Semester.

3. **Course requires Prerequisite**

Many-to-Many (M:N) → A Course may have multiple Prerequisites, and a Prerequisite may be required for multiple Courses.

4. **Course has a Course Schedule**

One-to-One (1:1) or One-to-Many (1:M) → A Course may have one or more schedules, but a Schedule belongs to one Course.

5. **Scheme of Study defines Course**

One-to-Many (1:M) → A single Scheme of Study defines multiple Courses.

6. **Student registers for Course**

Many-to-Many (M:M) → A Student can register for multiple Courses, and a Course can have multiple registered Students

7. **Coordinator views Students**

One-to-Many (1:M) → A Coordinator can view multiple Students

8. **Registration Record contains Courses**

One-to-Many (1:M) → A Registration Record contains multiple Courses

Ahsen Rasheed(04072312012)

Use Case: View Grades

ACTORS: Student,Teacher,course coodinator

Preconditions:

1. The student must be registered in the system.

2. The student must have completed at least one course.

3. The grading data must be available in the system.

4. The student must be logged into the system.

Postconditions:

1. The student’s grades for completed semester are displayed.

2. No modifications are made to the grades (read-only access).

3. The system logs the access timestamp for record-keeping

4. If the student has no grades available, an appropriate message is shown.

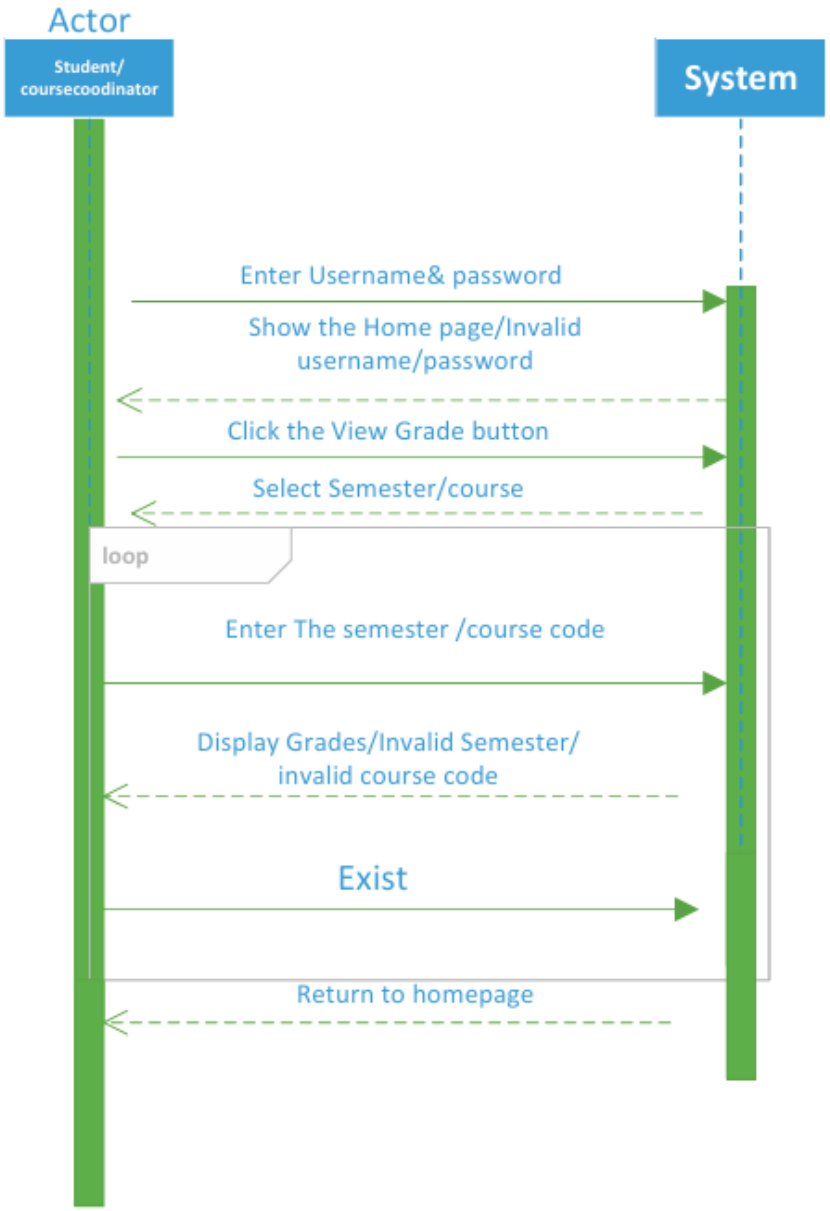
Alternatives:

1.Use Online registration portal to create account for login.

2.Check the Grades from the cs office.

3.course completed but system is not showing then ask it form the cs office.

System Sequence Diagram



Short Description:

This system sequence diagram illustrates the View Grades use case, where a student logs in by entering a username and password. Upon successful authentication, the system displays the home page. The student then clicks the View Grade button, and the system prompts for a semester selection. After the student enters the semester, the system retrieves and displays the grades or notifies if the course is incomplete**.**

It will done until user press the exist button.

Yaman Bangash(04072312021)

Use Case: Semester-Based Enrollment

Actors:

Primary Actor: Student (initiates course registration)

Supporting Actor: CS Office (uploads study scheme, manages course availability, enforces prerequisites)

Preconditions (Essential System and User Requirements):

1. The student must be logged into the system with valid credentials.

2. The CS Office must have uploaded the study scheme, including available courses, semester structure, and prerequisites.

3. The system must have an active semester session (Spring/Fall) to allow enrollment.

4. The student must not have any pending academic holds (e.g., fee dues, probation, or disciplinary actions).

5. The student's previously taken courses and grades must be available for prerequisite verification.

Postconditions (System State After Execution):

1. The student is successfully registered in the selected courses if all conditions are met.

2. The system updates the student's enrolled courses for the selected semester.

3. The system ensures that all prerequisite checks are applied and enrollment is only allowed where requirements are met.

4. The system records the registration details and generates an enrollment summary for the student.

5. The CS Office can view updated enrollment statistics and student course registrations.

6. If enrollment is denied, the system provides a clear reason (e.g., missing prerequisites, maximum credit hour limit reached).

Alternatives:

A1: Student Tries to Enroll Without Meeting Prerequisites

System verifies prerequisites and finds missing course requirements.

System denies enrollment and displays an error:

"Enrollment failed: You must complete [Prerequisite Course] before registering for [Selected Course]."

Student can select another course or contact the CS Office for assistance.

A2: Student Tries to Enroll Beyond Maximum Credit Hour

System checks the total credit hours for the semester.

If the limit is exceeded, system denies additional enrollments and displays

"You cannot enroll in more than 18 credit hours this semester."

Student must drop a course before adding another.

A3: Student Requests to Drop an Enrolled Course

System verifies course drop eligibility (within allowed drop period).

If allowed, course is removed, and records are updated

If not allowed (e.g., deadline passed), system denies the request and provides a reason.

A4: No Courses Available for the Selected Semester

System checks the study scheme and finds no courses available.

System informs the student:

"No courses are available for [Selected Semester]. Please contact the CS Office."

A5: System Fails to Fetch Semester Data

Due to a system error or missing data, the system cannot load semester details.

System displays an error message and logs the issue for technical support.

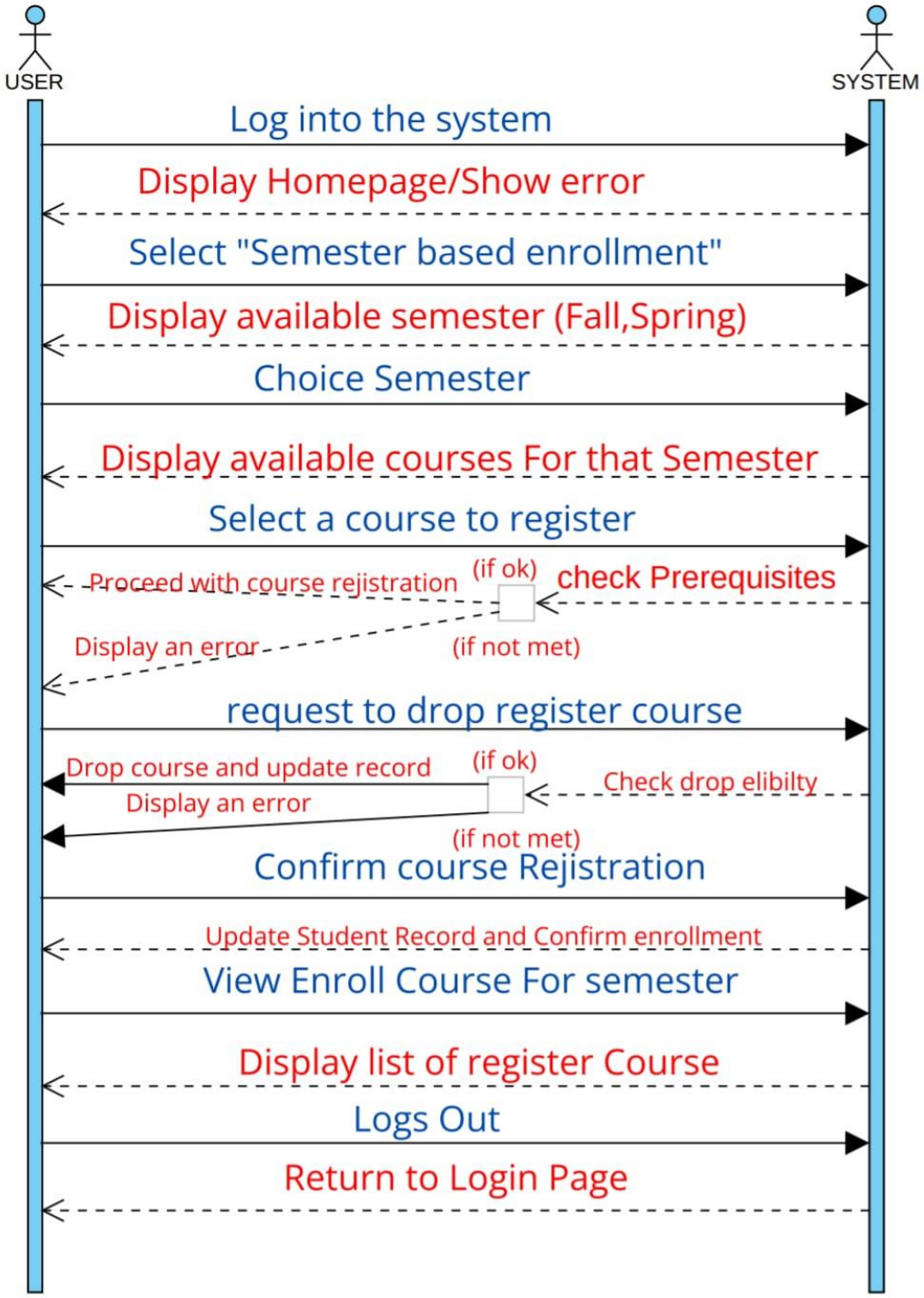
Student is advised to try again later or contact support.

A6: CS Office Updates the Study Scheme Mid-Semester

If the CS Office updates the study scheme, the system refreshes available courses.

Already registered students remain unaffected, unless the CS Office manually removes courses

System Sequence Diagram



Description:

Description of the System Sequence Diagram (SSD) for Semester-Based Enrollment

The Semester-Based Enrollment SSD represents how students interact with the enrollment system to register for courses. The system processes each request and provides feedback accordingly. Below is a step-by-step

1. Login Process:

The student enters their username and password.

The system verifies the credentials. If correct, it displays the home page.

2. Selecting the Semester-Based Enrollment Option:

The student selects the "Semester-Based Enrollment" feature from the menu.

The system displays the available semesters (Spring, Fall)

3. Choosing a Semester:

The student selects a semester (either Spring or Fall)

The system fetches available courses for the selected semester and displays them.

4. Registering for a Course:

The student selects a course to enroll in.

The system checks the prerequisites:

If prerequisites are met, the system allows the student to proceed.

If prerequisites are not met, the system displays an error message.

5. Confirming Course Registration:

The student confirms course registration.

The system updates the student’s records and displays a confirmation message.

6. Dropping a Course (If Needed):

The student requests to drop a registered course.

The system checks if the drop request is valid:

If dropping is allowed, the system removes the course and updates the student’s records.

If dropping is not allowed, the system displays an error message.

7. Viewing Enrolled Courses:

The student requests to view their registered courses for the semester.

The system displays a list of enrolled courses.

8. Logging Out:

The student logs out of the system.

The system ends the session and redirects the user to the login page**.**

Ahmed Nazir(04072312012)

Use Case: Course Registration

Preconditions:

1. The student must be logged into the system.

2. The student must have an active account.

3. Course registration must be open for the current semester.

4. The student should have no pending dues.

5. Pre-requisites are passed/cleared.

Postconditions:

1. The student is successfully registered for the selected courses.

2. The student enrollment record is updated.

3. The student receives a confirmation message that courses have registered successfully.

Alternative Flow:

Step.1: Student is not logged in

If the student is not logged in, the system informs them to log in before proceeding

Step.3: Course Registration is closed

If course registration is closed, the system displays an error message and prevents further action.

Step.4: Student has pending dues

If the student has pending dues, the system restricts registration and asks the student to clear the dues.

Step.6: Clashes between selected courses

If there is a time conflict with another registered course, the system asks the student to choose a different time slot.

System Sequence Diagram

Description:

This System Sequence Diagram (SSD) represents the interaction between a student and the course registration system. It visually maps out the exchange of messages, beginning with the student's login attempt and the system's response. The student navigates to the course registration section, enters the course code, and the system verifies prerequisites, credit limits, and course availability. If any conditions prevent registration, the system notifies the student with an appropriate message. The interaction

concludes with either a successful course registration or an exit to the main menu. The diagram effectively illustrates prerequisite enforcement and validation in the registration process.