metin, yazı tipi, logo, simge, sembol içeren bir resim

Açıklama otomatik olarak oluşturuldusimge, sembol, logo, yazı tipi, ticari marka içeren bir resim

Açıklama otomatik olarak oluşturuldu**CSE 3063 PROJECT**

**REQUIREMENT ANALYSIS DOCUMENT (RAD)**

**Iteration 2**

**Group 2**

*150120031 - Şükrü Can Mayda  
150121055 - Alp Büyükköse  
150122610 - Yunus Emre Ocak  
150121065 - Turgut Köroğlu  
150120995 - Seyyid Ali Koldaş  
150121545 - Nurbetül Çakır*

**Project Description**

In this project, we will be creating a course registration system for our department. On the first iteration of the project, the system will accept two types of users: Student and Advisor. Additional roles may be added in next iterations.

When students login to the system, they will be able to choose the courses they will be taking for their current semester. Then, they will send their selection to their assigned advisor to be confirmed. The advisor will be able to see all courses each student has chose and they will approve or disapprove these courses. In the case of approval, the registration process will end successfully. In the case of disapproval, the student will be choosing their courses again based on the limitations.

Furthermore, the students will be able to view their past grades in their transcript, check their current term schedule and learn about the prerequisites. Advisors will also be displaying the students they are held accountable for and get in touch with them via the contact information in their profiles.

Our main goal is to make a clear and understandable, easy-to-use registration system for students and advisors to smooth the academic processes with security and reliability involved.

**Glossary**

* Advisor: Teacher who is responsible for help particular student group which includes adding/dropping courses and help on other subjects.
* Student: Person who is taking courses and responsible with these courses.
* Transcript: Summary of a student's academic performance and progress to date.
* Password: The key of the student's login on registration system.
* Prerequisite Course: The course that is mandatory for specific course.
* Course: The collected information of specific subject.
* Grade Point Average (GPA): The success measure of a student’s current term calculated by the earned grades in their courses
* Cumulative Grade Point Average (CGPA): The success measure of a student’s past terms calculated by the cumulative earned grades in their courses
* Advisor Approval: The advisor confirmation process of courses selected to be taken by a student
* Lecturer: Person who is responsible for giving lectures to students
* Enrollment: Registration to a course
* Login: The process of signing in to a website by a user
* Credit: A unit representing the work load of a course
* Earned Credit: The total credit of courses a student enrolled in the current term
* Received Credit: The total credit of courses a student enrolled and successfully completed in the current term

**Functional Requirements**

**Login:**

* Login page is the first screen when an user wants to enter the system.
* There will be unique IDs assigned to all users.
* Any user can login to the portal by entering their identification number and password.
* If the credential matches with the database’s, the user can display the main page.

**Schedule:**

* Students will be able to see which courses they are taking for the current term, their weekly occurences and classrooms.
* Each individual schedule will be arranged according to the advisor’s approval of courses.

**Student Interface:**

* Students will be able to see their courses, schedule, transcript and advisor information.
* There will be a section for changing personal information.
* Students will be able to register for courses.

**Advisor Interface:**

* Advisors will be able to see their all student’s information except their password.
* There will be a section for changing personal information.
* Advisors will be able to approve or deny requests of students in the registration system.

**Course Registration:**

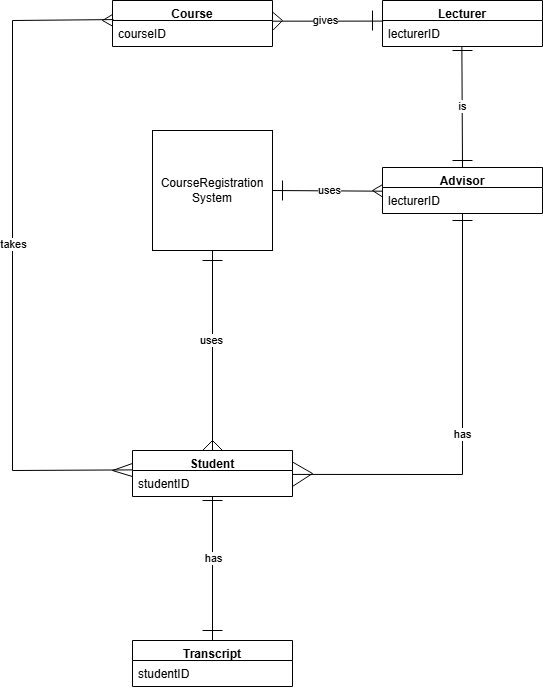
* Students can view and register for available courses based on the rules of prerequisite and senior year courses.
* Advisors can update the courses students can take for each term.
* During registration period, students can add and remove courses for the current semester as long as the courses are compatible with the requirements and quota.
* After the selection is done, students can submit the courses for approval to their advisors.
* Advisors can approve a request to finalize the registration or deny for the student to change its selection.
* If the request is denied, the selection will have to be made again by the student according to the rules.

**Non-Functional Requirements**

System attributes like security, performance, and usability.

* System should respond quickly to the user’s demands during logins, database updates and logouts.
* The changes a user can make will be recorded and the database will be updated.
* Instead of a graphical user interface, a command-line interface will be present, however the interface should still be understandable with texts and order.
* The system will be implemented in Python after first two iterations.
* Necessary data will be stored in separate JSON files.
* Error messages and warnings will be clear and easy to understand for the user.
* Delicate information such as ids and passwords will be stored and handled securely for the user.

**Domain Model**



**Use Cases**

Use Case 1: Student Login

Primary Actor: Student

Stakeholders and Interests:

* Students: Quick and user-friendly access to their accounts to display courses, view schedules or make any other changes
* University IT Department: Guaranteeing that the login system works smoothly and database integrity and security is protected
* Student Affairs’ Office: Managing the content of the system so that students can access the system to check their information
* University Management: Providing a reliable and accessible system backing up the institution’s reputation.

Precondition(s): Student has been registered in the system beforehand and has valid username and password.

Main Flow:

1. Login page interface opens.
2. Student enters their username and password into the blanks.
3. System checks the validity of the info.
4. If the info is correct, student is logged in and directed to the home page.
5. System’s home page is displayed.

Alternative Flow:

* System displays a message indicating some login information is incorrect.
* Student can retry logging in.

Postcondition(s):

* Student is logged in and directed to the home page.

Use Case 2: Advisor Confirmation

Primary Actor: Advisor

Stakeholders and Interests:

* Students: Receiving feedback from advisors on their requests and planning their academic year further
* Advisors: Quick and user-friendly access to their accounts to display students and their requests
* Departments: Ensuring that students register for the correct courses based on the requirements
* Student Affairs’ Office: Monitoring course registrations and confirmations in case any troubles come up
* University Management: Maintaining an efficient registration period backing up the institution’s reputation

Precondition(s):

* Advisor has been registered in the system beforehand and has valid username and password.
* There are waiting registration requests from students.

Main Flow:

1. Advisor logs into the system.
2. Advisor clicks the “Pending Registration Requests” section on the main page.
3. Advisor selects a student to view their request.
4. System displays the request details such as student information, course information and prerequisite condition.
5. Advisor reviews the information and decides whether to approve or reject the request.
6. Advisor clicks the “Approve” button or the “Deny” button.
7. System updates the request’s status accordingly.

Alternative Flow:

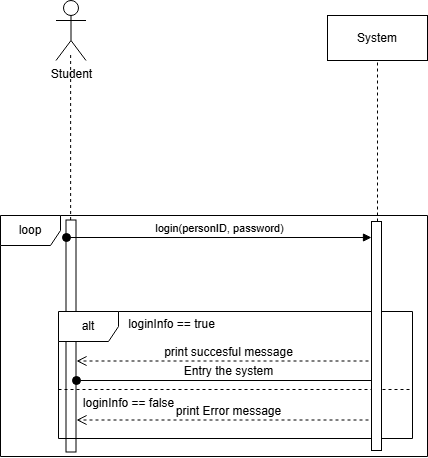
1. Advisor checks the request and chooses to deny it.
2. Systems requests a reason of denial from the advisor.
3. Advisor enters a reason in plain text.
4. System updates the status as “Denied” and includes the reason with it.

Postcondition(s):

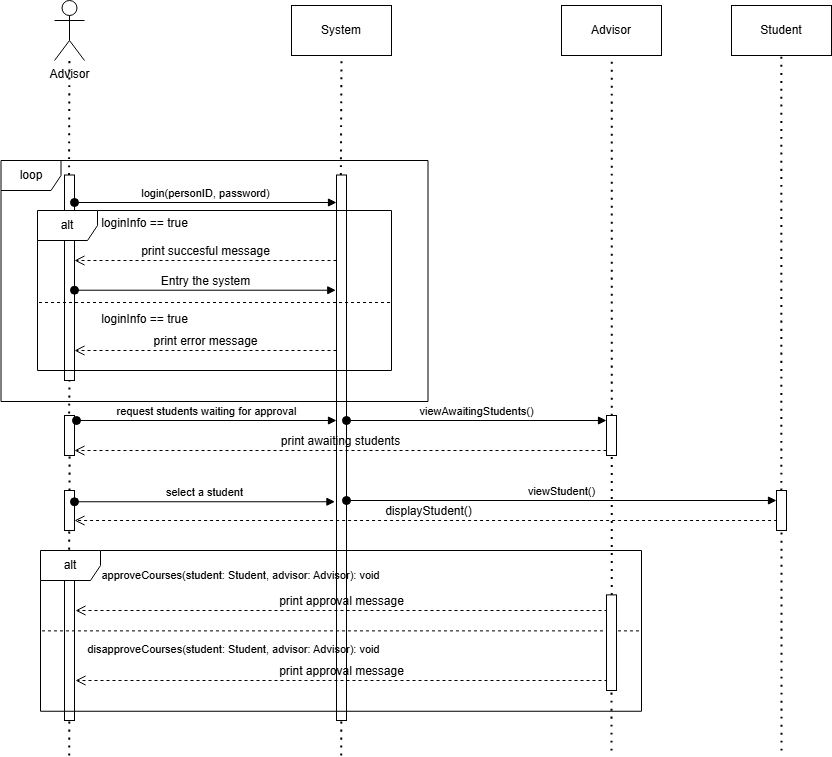
* The student’s registration request is approved or denied.

**System Sequence Diagrams (SSDs)**

**Login Scenario:**

****

**Advisor Confirmation:**

****