MARMARA UNIVERSITY

FACULTY OF ENGINEERING

COMPUTER ENGINEERING DEPARTMENT

CSE2063

Object Oriented Software Design

Java Project – Fall23 – Iteration 1

1. Aksanur KONUK 150120049
2. Duru BAŞTUNALI 150120075
3. Elife KOCABEY 150120054
4. Eren ÇETİN 150120008
5. Nidanur DEMİRHAN 150120042
6. Semih BAĞ 150120070
7. Zeynep YILMAZ 150120066

1. Purpose And Description

In this project, we will try to develop a course registration system. In the first phase of the project, only two different user types will be considered and developed. These user types will be divided into two: student and consultant.

When students log in to the system, they will be able to choose from the courses offered to them. After making a selection, they will be able to send it to the consultant for approval.

When the advisor logs into the system, he will be able to see the students he advises and their information. They will also be able to see course selection requests submitted by their students that are awaiting approval. Advisors will be able to approve or disapprove course selection requests from students.

In short, we will develop a course registration system similar to the real world.

2. Glossary Of Important Concepts

(In ascending order)

1. Adding course: Adding new course to the courses have taken.
2. Advisor: A person who is responsible for adding/dropping courses for students, as well as teaching courses.
3. Cancel button: Allows changes not to be saved.
4. Command-line: It is a text-based interface that provide user to make operation on Course Registration System.
5. Course: It is the integrity of information created by the combination of interconnected information.
6. Course Registration System: It is a system that students can request adding or removing courses, advisors can accept or reject those requests.
7. Course Section: Gives information about course.
8. Dropping course: Dropping course from courses have taken.
9. Grade: Indicator showing how successful a lesson was passed
10. JSON: It is a file format that provide storing datas.
11. Lecturer: A person who is responsible for teaching courses.

12. Login: Accessing Course Registration System with correct combination of username and password.
13. Menu: It is a list of what the user can do in Course Registration System.
14. Password: It is a combination of characters that enable the use course registration system.
15. Prerequisite Course: Before taking that course, other courses related to it must be taken.
16. Save button: Allows changes to be saved.
17. Student: A person who is responsible for completing courses.
18. Transcript: Students' grade of lecture records
19. User: It is a person that try to use Course Registration System. It can be student, lecturer or advisor.
20. UserName: It is a name. The one of the factors that enable the use course registration system.

3. List Of Requirements

3.1 FUNCTIONAL REQUIREMENTS

LOGIN INTERFACE

- A login page must be loaded first when the system is started.
- Users must be able to log in to the system by entering their username and password on the login page.
- When trying to log in with a user profile information that does not exist in the system, an error message should be displayed on the screen and then the reset login interface should be displayed on the screen.
- If the password in the database does not match the password entered by the user, an error message should be printed on the screen and the login interface should be printed on the screen again.
- The program should be able to terminate with the cancel option in the login interface.

MENU

- Each of the headings in the menu content should be considered as a component and should be designed in a way that new menu options can be easily integrated or removed when necessary.
- Users can access the tab they want to select by clicking on the relevant tab number from the keyboard.
- Each user should be able to get the necessary information from the menu tabs created according to the user type.

COURSES LIST

- Students should be able to see in a list which courses are included in the department.
- Students should be able to see which instructor gave the relevant course or other information about the course.

COURSES OFFERED

- Students should be able to learn the list of opened courses from the relevant menu tab.

SELECTED COURSES

- The student type user should be able to choose the courses he/she wants from the courses offered to him/her and create a course registration list.
- Students should be able to send the list of selected courses they have created to their advisor for approval.
- Students should be able to remove any course from the list of selected courses they have already created.
- Students should be able to cancel the list of selected courses that has already been created.
- Students should be able to see the status and content of the approval request they have previously submitted.

ADVISOR INTERFACE

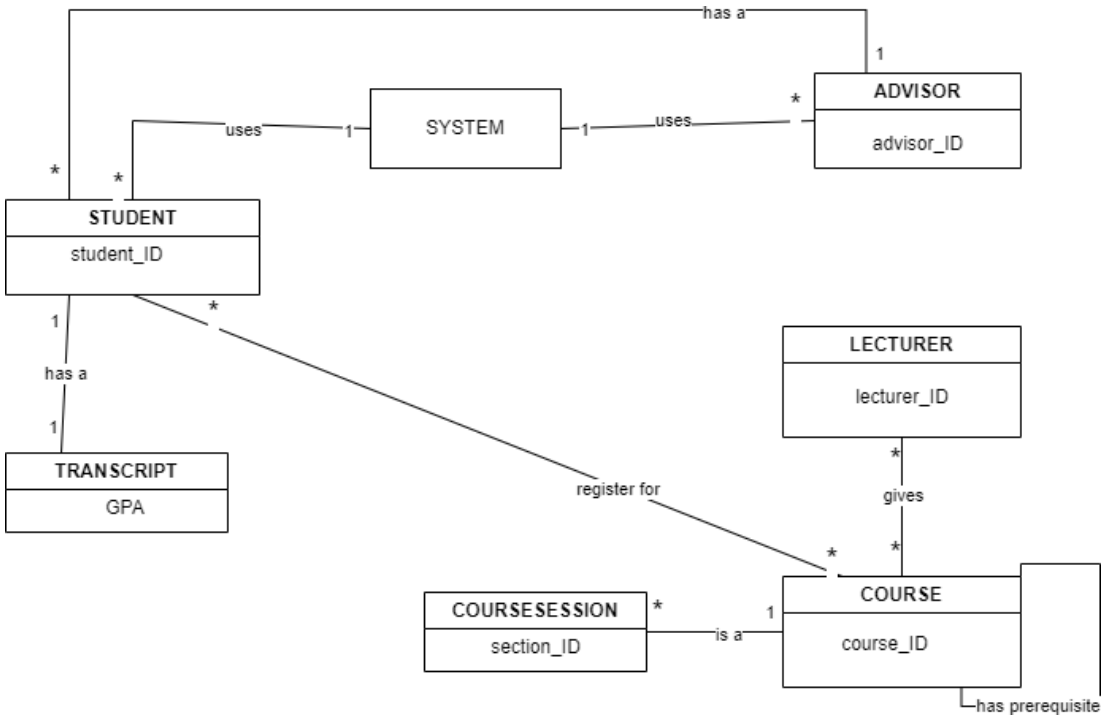
- Advisors should be able to see the students they are interested in in a list.
- Advisors should be able to view course selection approval requests they receive.
- Advisors can approve any approval request

  Advisors should be able to reject any approval request

3.2 NON - FUNCTIONAL REQUIREMENTS

- JSON filing system should be used to store data
- When the system is run, the relevant json files should be read and processed into memory as objects.
- When the program is closed, all changes made to the objects so far should be processed into json files.
- When the user logs in successfully, the user-specific menu interface should be printed on the screen.
- The system should automatically adjust the courses a student can take.
- The courses a student can take should be created with that student's information. While doing this, conditions such as average score and affiliated course status should be taken into consideration.
- Since a graphical user interface will not be developed yet, the menu to be created in the terminal should be simple and understandable.

4. Domain Model



5. Use Cases

Use Case 1: Login System for User
Primary Actor: User
Stakeholders and Interests:

- Student: wants to log in to the system for class registration
- Advisor: wants to log in to the system for student's class registration approval or rejection
- Lecturer: wants to log in to the system to view the classroom content for each of their class

Preconditions: The user should have their username and password saved in files.

Postcondition: Login is confirmed. The system is reached by the user and the main menu is formed based on their user type such as student, advisor, or lecturer.

Main Success Scenario:

1. The user is welcomed by a login screen.

2. The login screen displays an area for the user to enter their username and password.
3. The user will enter their username and password information together.
   1a. No such user exists; System gives a warning as "Incorrect Username/Password".
   2a. The password doesn't match for the given username; System gives a warning as "Incorrect Username/Password".
   2. The login screen displays an are for the user to enter their username and password.
4. If username and password information are both correct and valid for the associated account, the login process occurs and the user will successfully reach their account for further actions.

Use Case 2: Basic Course Registration System (Student Perspective)
Actor: Student
PreConditions: The student is logged into the course registration system.
Postconditions: The student successfully sent the course registration request and the system saved it in the Json file

Main Flow:

1. Student-specific menu content is printed on the screen
2. The student enters an input (selected courses tab order) from the keyboard.

2a- The student entered a value that is not in the menu
2b- Appropriate error message is printed on the screen
1- Student-specific menu content is printed on the screen

3. The list of courses opened on the screen is printed.
4. The student enters the numbers of the courses he or she wants to take from the keyboard.

4a-If a number that should not be entered is entered, the courses indicated by the valid numbers are added to the selection list.
For others, the error message is printed on the screen.
4- The student enters the numbers of the courses he or she wants to take from the keyboard.

5a- The student presses the save button
5b- The student presses the cancel button
5ba- Cancellation confirmation text is printed on the screen
4- The student enters the numbers of the courses he or she wants to take from the keyboard.

6. The student presses the send button for the advisor's approval.
7. The registration request process ends successfully

Use Case 3: Basic Course Registration System (Advisor Perspective)
Actor: Advisor
Preconditions: The advisor is logged into the course registration system.
Postconditions: The advisor has reviewed and approved course selections for their assigned students, and the students are successfully registered for the approved courses.

Main Flow:

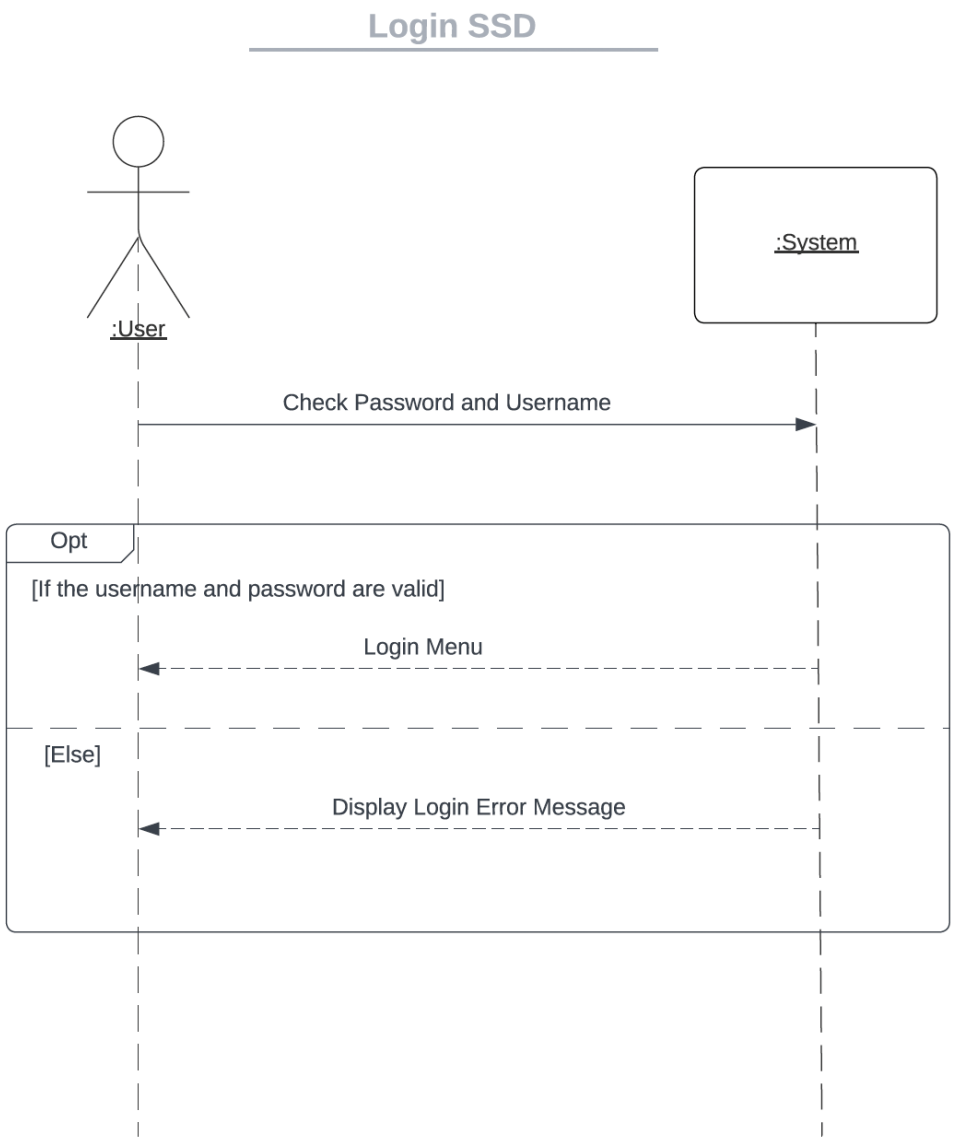
1. The advisor initiates the course registration process for their assigned students.
2. The system presents a list of students for whom the advisor is responsible.
3. The system presents a list of students who have submitted course selections for advisor approval.
4. The advisor selects a student from the list.
5. The system displays the available courses for the selected student.
6. The advisor reviews the courses and selections made by the student.
   1a. If the advisor decides to reject a student's course selections, they provide feedback, and the student can make revisions to their selections.
   2a. If the selections are approved, the system registers the student for the chosen course.

4. The advisor selects a student from the list.

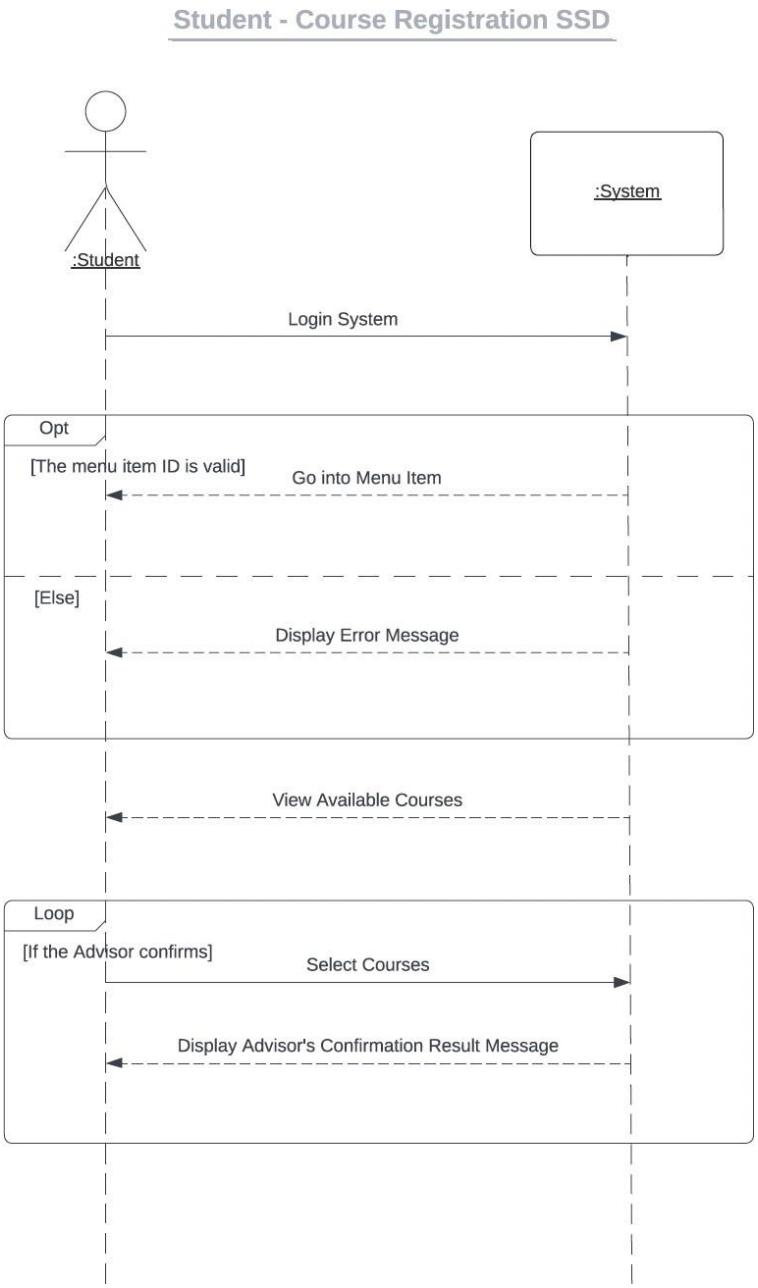
7. Once all students' course selections have been reviewed, the advisor confirms the registration process for approved students.
   1a. If the course registration system encounters technical issues or errors at any point in the main flow, the use case may terminate with an error message.
8. The use case ends and the advisor has successfully completed the course registration for their assigned students.

6. System Sequence Diagrams

Login SSD:



Student - Course Registration SSD:



Advisor - Course Registration SSD