STUDENT COURSE REGISTRATION SYSTEM

REQUIREMENTS ANALYSIS DOCUMENT

1-Vision

Our initial goal for the Student Course Registration System is to produce a simulation of student course registration for our department (CSE). It will be modeled our department's rules and procedures for enrolling in courses. We are going to build our system as a simulation that takes arguments from a user-supplied command-line input file. The simulation software continues until all of the student transcript documents are generated.

2-Scope

The initial version of the Student Course Registration System will be a simulation. A dataset is going to sent to this system as a JSON file. There are requirements for the courses, and other business logic will be based on the rules and procedures of our department for enrolling in classes. In this round, courses will be offered at random, and students will be chosen at randomly. The system will produce output as a Json file after handling each instance.

3-Functional Requirements

- Costumers can view students' transcript document screen. In this screeen there are grades, courses which are already taken, GPA, CGPA, given credits, completed credits, offered courses and errors during the registration.
- User profile can create students again.

4-Nonfunctional Requirements

- This program written by Java to handle object oriented software programming more effectively.
- The capacity is: Totally 280 students (70 students for each year.)
- Whenever the user wants to use it, they can access and view the system. It's always available.
- It is access to student information and available courses. The student can quickly finish the registration procedure.

5-Problem Description

We will have a student registration system that can operate both step-by-step and as a simulation as a consequence of this project. In each of the eight semesters, we will have an equal number of students. There will be requirements when registering for such students, such as finishing the necessary courses and staying within the course quota. Furthermore, students must also earn a certain amount of credits in order to enroll in final year courses. The user interface will be alerted of the activities made after the registration procedure is complete.

6-Collabrators

Customer: MURAT CAN GANİZ

Developers:

- Tarık Işın
- Yağmur Koçoğlu
- Ayşenur Tüfekçi
- Abdullah Enes Dizer
- Oğuzalp Acıkan
- Mervan Aburşu
- Mert Efe Karaköse
- Muhammet Görkem Gülmemiş
- Fatih Erkam Dinler

7-USE CASE

- **Use Case: Register to Course**

Actor: Student-System

1) Student enters to the system.

2) The system presents the list of the courses which student can choose and take.

3) Student check over listed courses.

4) Student checks the elective courses to be taken.

5) Student selects courses from the list.

6) Student sends the courses that she/he selected to the advisor's approval.

7) System shows the result of the advisor's approval to the student who selected courses.

Alternative Case: Selection Error

4a. In step 4, the system could show error for selection overlapping courses. The system gives permission to the student to re-select courses in the list.

- **Use Case: Advisor's Approval**

Actor: Advisor-System

1) Advisor enters to the system.

2) System shows the student's selection course request to the advisor.

3) Advisor checks course quotas and overlaps.

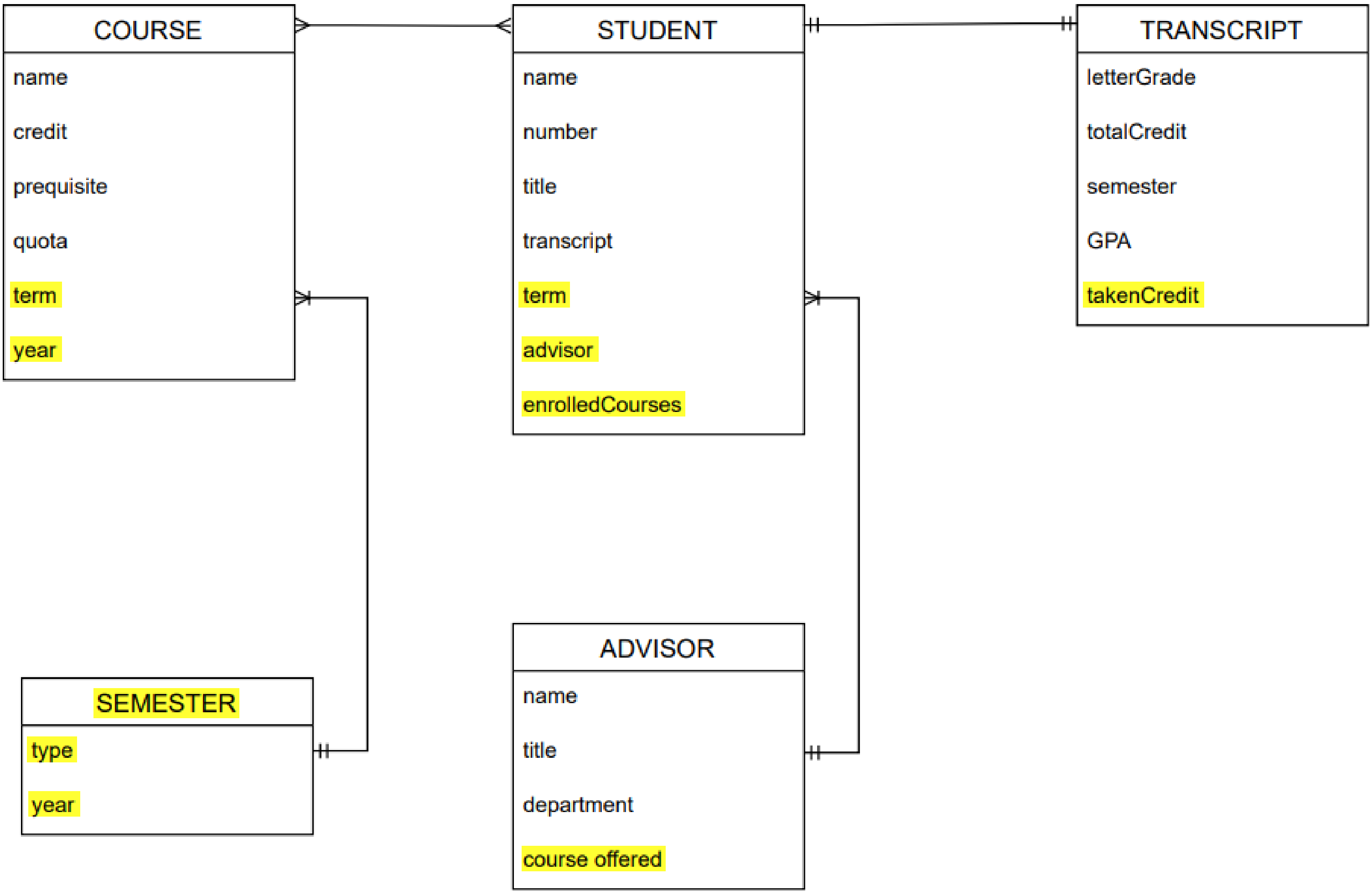
4) Advisor starts to the process off approving course selection request.

5) Advisor approves the courses.

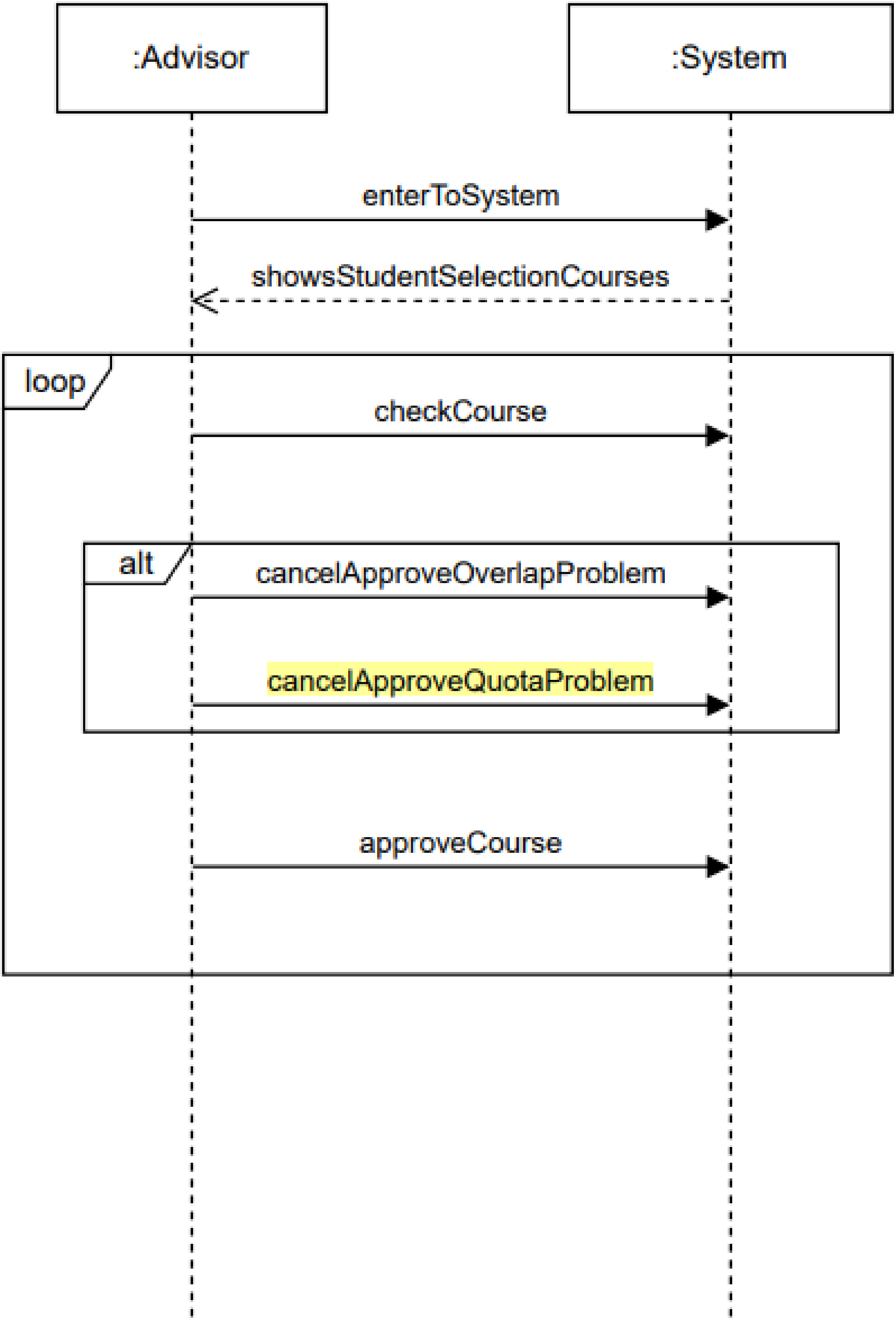
Alternative Case: Disapproval the selection

4a. In step 4, the advisor do not approve because of overlapping courses and quotas.

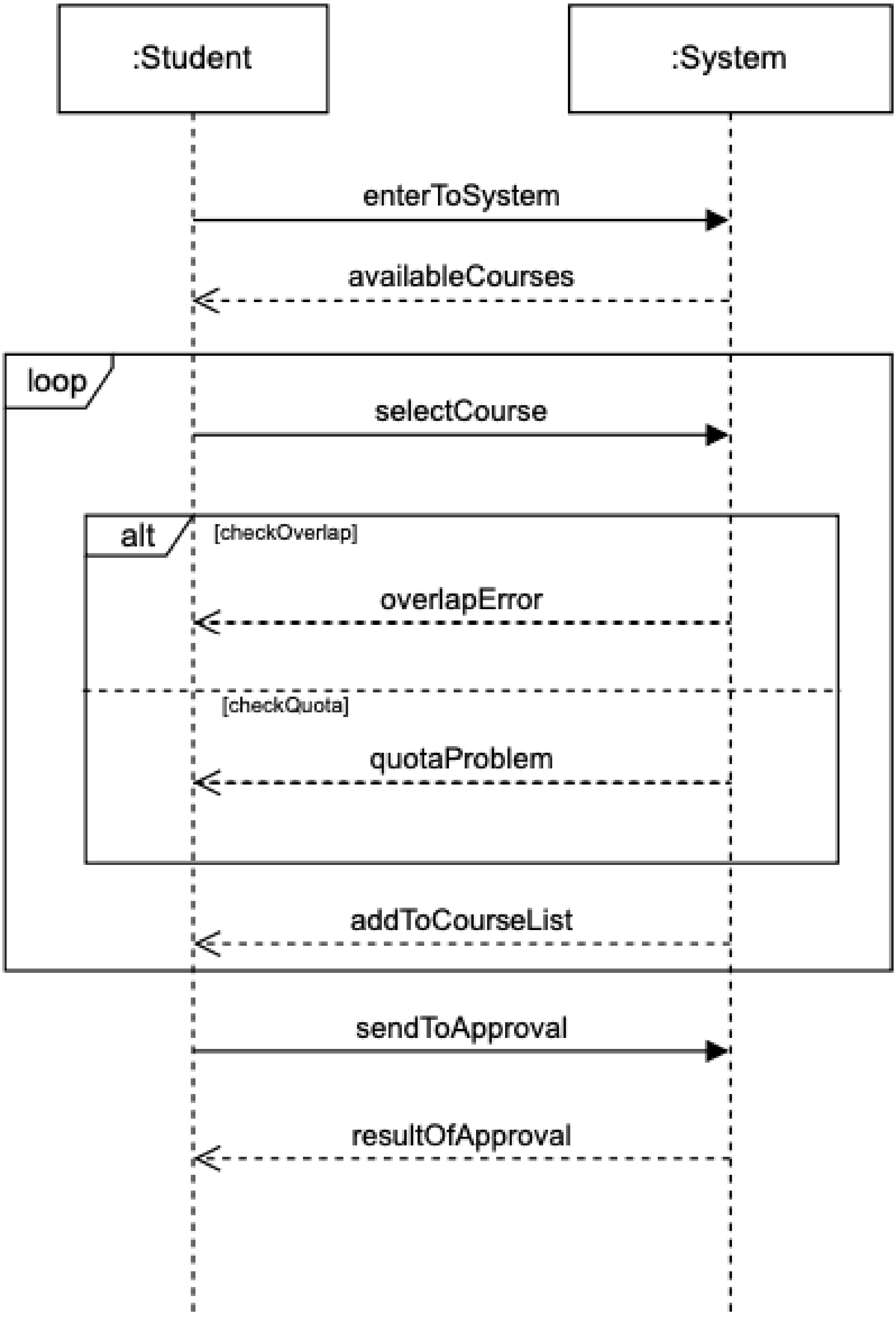
8-Domain Model



9- System Sequence Diagrams(SSD)


:Advisor
:System
enterToSystem
showsStudentSelectionCourses
loop
checkCourse
alt
cancelApproveOverlapProblem
cancelApproveQuotaProblem
approveCourse

Register To Course



10-GLOSSARY

- **Advisor:** She/He is the person who helps the student in the registration stages of the system.
- **CGPA:** CGPA is used to measure the overall academic achievement of a student by awarding A, B, C, D or F grades.
- **Class:** A class defines an object's interface and implementation. It specifies the object's internal representation and defines the operations the object can perform.
- **Course Number:** The numeric part of a course name.
- **Department:** A division of a large organization of university.
- **Enrolled:** When a student is registered for a course.
- **GANO:** The GPA, or Grade Point Average, is a number that indicates how high student scored in his/her courses on average.
- **Inheritance:** A relationship that defines one entity in terms of another. Class inheritance defines a new class in terms of one or more parent classes. The new class inherits its interface and implementation from its parents.
- **Lecturer:** A person who gives lectures, especially as an occupation at a university or college of higher education.
- **Name**: A word or a combination of words by which a student.
- **Offered Course:** Courses that the student is entitled to take in the current semester.
- **Overlap:** Is the case that there is a conflict in the curriculum that the student wants to create in the system.
- **Prequisite:** It contains the conditions that is necessary for the student to register for a course.
- **Quota:** Maximum number of students who can enroll the course.
- **Rank:** A position in the hierarchy of system actors.
- **Semester:** A half-year term in a university.
- **Student Number**: A numerical values assigned to a student upon entering an educational institution and used to identify the student in official documents.
- **Student Process:** Student's calculations in the process.
- **Student:** Person who enrolls in a course.
- **Transcript:** It is a document showing the courses that the student has registered, passed, and attended throughout his/her education life, and the grades he/she has taken.
- **Year:** Indicates the year in which the student entered the university.