COURSE REGISTRATION SIMULATION

1-) Description

Scope: The purpose of this project is to implement a system that helps students about their course registration process. While deciding if a course is available for a specific student or not, both the student and course limitations must be checked with caution and doing these as a human is quite cumbersome. That is why we need this simulation.

Problem Statement: This simulation will be considering all of the cases for each student such as prerequisite tree, completed credits of the student, course type (TE, FTE etc.) and many more. Based on these limitations the simulation will decide and prompt the available courses for that student and lets the student (user) register those available courses. It also checks errors in these processes and prints the cause of error to the user.

2-) Requirements

a) Functional requirements

- Course related informations
- Prerequisite trees for both (2015 – 2019) and (2020 – 20..) entry students
- Student related informations
- Limitation/restriction informations of registration system
- Transcript related informations

b) Non-functional requirements

- Student, transcript etc. informations should be kept in JSON files
- Any database should not be used
- System parameters must be read from a parameter.json file
- Proper error or approval messages should be printed.
- The simulation must execute in command prompt

3-) Limitations

- To register a course, student must pass its pre-requisites.
- To register TE courses and graduation project student must be completed the required credits.
- To register FTE courses, student must be either at SPRING semester or must be at his/her graduation state.
- To register a course, that course's quota must not be exceeded.

4-) Use Case: Registration of Courses

Actors: Student, Advisor, Controller

1) The student has a ID, a register date, semester, a transcript, an error and selected courses.

2) The student sends the list of courses he/she wants to register to related controller.

3) Controller send the curriculum, students and courses to the advisor to check course availability.

4) First, Advisor looks at the requested course and checks wheter it is in the curriculum. Then checks the quota of the course, preRequisite courses for that specified course and if it overlaps with another course or not. Based on these checking it decides wheter the student can register or not and if not it keeps the cause of error.

5) Controller approves the registration request of the student based on input taken from the advisor.

6) Approved courses added to students transcript as completed or failed course.

Failure at approving:

Advisor does not approve a course because collision with other course in schedule

The system does not allow a course when student failed pre-req course

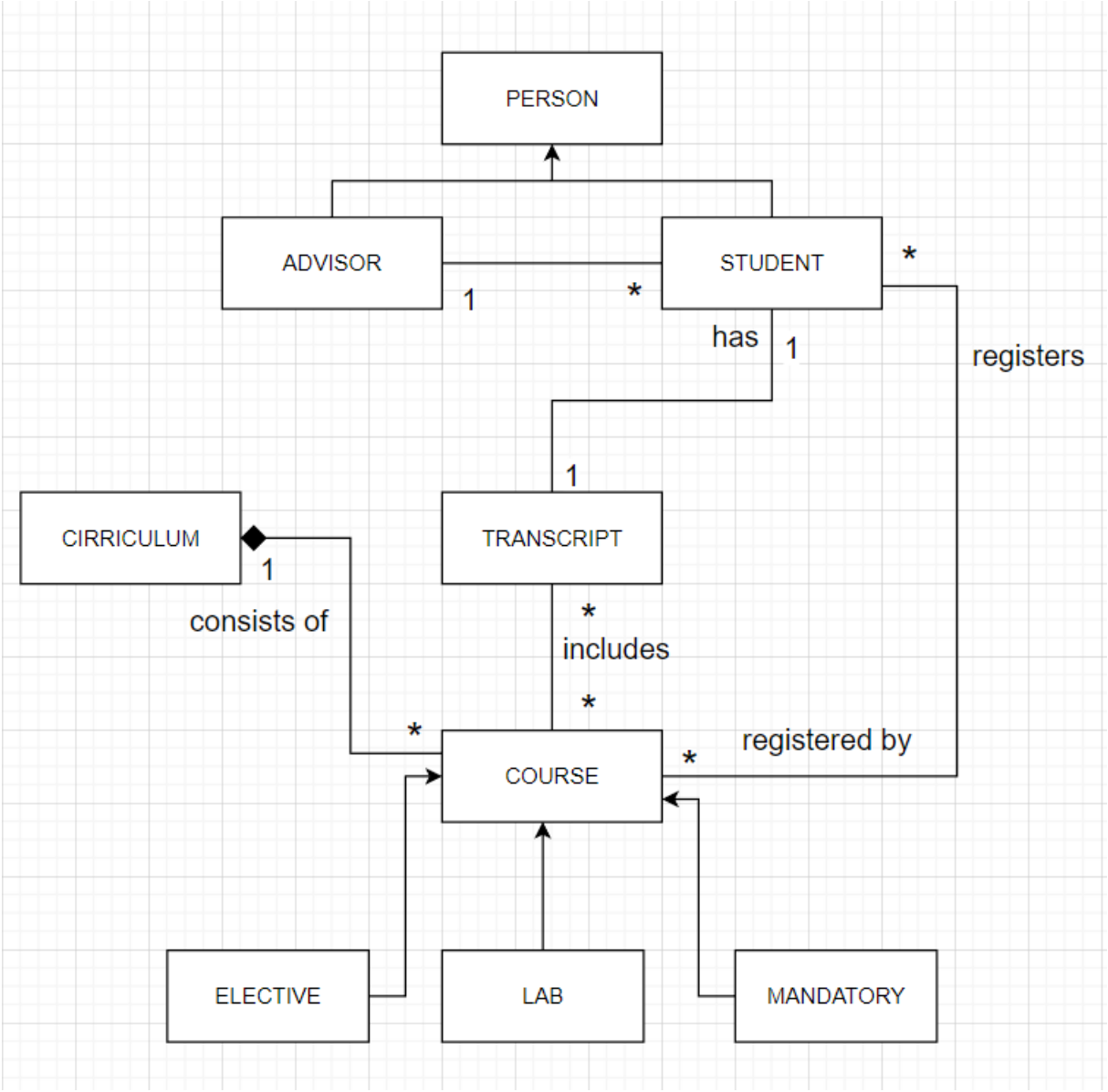
The advisor does not approve a course because student completed credits quota

The advisor does not approve graduation project when student completed credits quota

The advisor does not approve TE if student already took 2 TE and in a semester only 2 TE can be taken

The advisor does not approve FTE because students can't take FTE in FALL semester unless they are graduating this semester

5-) Domain Model



6-) Glossary

SSN: It means to “Social Security Number” that a numerical identifier for citizens.

Gpa: Grade Point Aevrage

UE: It means to “University Elective Courses”.

TE: It means to “Technical Elective Courses”.

NTE: It means to”Non-Technical Elective Courses”.

FTE: It means to “Faculty Technical Electives Courses”.

Advisor: Instructor checking the conditions for the student to enroll in courses.

Collision: The situation where the student tries to take more than one course that coincides with the same time.

7-) System Sequence Diagram