DATA LABELING
SYSTEM REQUIREMENTS ANALYSIS DOCUMENT

STAKEHOLDERS

Murat Can Ganiz (Customer)

100217006, Elif Nur Kemiksiz

150117007, Edanur Öztürk

150117028, Reyta Gül Muran

150118020, Emre Eren

150119607, Enes Uzun

150119622, İlayda Oruç

150119664, Nesrin Şimşek

150119809, Hakan Sandıkçı

150119897, Bilal Açıkel

150517059, Özge Saltan

VISION

The project's goal is to simulate our Student Course Registration System department's student course registration system. It will be designed on our department's course-taking policy and procedures. Our system will be created as a simulation that takes the user command-line options and the input file as input. The system performs one semester course registration for all students in one run. The scenario continues until all students' transcripts are completed.

SCOPE

Many students will be able to register courses. First of all, administrator determines the registration term (such as Fall or Spring) in the INPUT.json file. System starts with creating random students by receiving student information from a json file, which consists of random names and surnames. System assigns courses -which are available in the current registration term- to all students according to their semester information, by receiving courses from INPUT.json file. The success status (such as Pass or Fail) of the students is randomly determined by the system. System stores the taken courses data with grades and

success status for each student in the transcript, which is in related student's json file.In some cases, failures may occur. The system also logs failures that occured during the registration process in related student's json file. Also system logs whole failure information for all students in a json file, prints this information to the screen and terminates.

PROBLEM STATEMENT

The main purpose is to develop solutions to some classification problems related to the problems experienced by the student in course selection and the mechanisms in the data sets.

SYSTEM CONSTRAINTS

In this project, it is to make a console application that will run in the Python program.

GLOSSARY OF TERMS (ALPHABETICALLY LISTED)

- Advisor: The instructor who approves the courses taken by the students
- ControlCenter: An installation from which a series of operations is directed
- Course: The course which taken by a student in a current semester
- Dataset: The cluster that contains the information (id, name, gpa, course, etc.) to be used
- Elective: A course that you choose to take as part of your program of study
- Json file: The file which includes meta data
- Lecturer: The instructor who gives the course that is connected to the department
- Sections: Different time frames of courses
- Semester: Includes what year we are in university
- Student: Who selects the courses

- Prerequisites: The course in the previous semester depends on the course that needs to be taken
- in the semester
- Transcript: Includes name, surname, number, lessons taken and letter grades given

FUNCTIONAL REQUIREMENTS

- A list of student names will be read from an input file by the system.
- The system will assign each student a name and surname from the list at random, as well as create and assign a unique student id at random.
- The system will use the input file to generate a list of courses.
- For each student, the system will build a transcript and save the information and transcript in a file.
- Either the selected student or a random student will be registered by the system.
- The student will be registered by the system.
- The registration procedure will be saved as a JSON file by the system.
- The transcript of the student following registration will be included in the output.
- The transcript screens of students can be viewed by users. There are taken courses, grades, YANO, GANO, provided credit, completed credit, offered courses, and registration problems on the transcript.

NON-FUNCTIONAL REQUIREMENTS

- Python was used to efficiently handle object-oriented programming.
- A command-line interface will be available for the system.
- JSON must be used for all output and input files.
- The input file's name will be "Input.json."
- His/her student number will be the name of the produced students' file.

USE CASE: Register a course

Actors: Student, Advisor

1. When the program starts to execute, system checks semester of this registration.
2. System assigns courses to created students related to the student's semester information.
3. System randomly assigns elective courses from elective courses pool.
4. In some cases, (such as full quota, insufficient credit conditions, etc.) the system may not approve some courses.
5. Registration process is printed to the screen.
6. After registration is completed, system terminated.

DOMAIN CLASS DIAGRAM