

# ACIT 1515 Project

For this project you will be building two apps: a flask server that reads/writes data stored in json files, and a client application that allows users to interact with the server using the requests library.

## Server

Your server must have (only!) three endpoints:

1. Return all transcripts (see the attached JSON files to determine what constitutes a transcript)
2. Return a single transcript by its id
3. Create/edit a transcript

## Client

The client must display a menu and repeatedly prompt the user for (only!) one of the following choices:

1. View All Students
2. Choose Student
3. Add Student
4. Edit Student

If View All Students is chosen, the client must display a list of all students for whom a transcript is available, and must show (in ascending order of id) the transcript id and student first and last names

If Choose Student is chosen, the client must prompt for a student id. If the id the user enters is valid, a new student menu must be repeatedly displayed with the following choices:

1. List Courses (must display all classes contained in the transcript)
2. Check Status (must indicate if student has achieved an average of 50 or greater across all their classes)
3. Back To Main Menu

If Add Student is chosen, the client must prompt for a first name, last name, and list of (minimum) 2 courses the student is enrolled in. An id must be generated for the student that increments the highest existing transcript id by one. This data must be sent to the server and stored.

If Edit Student is chosen, the client must allow the user to change any part of an existing transcript and send the changes to the server to be stored.

See the following page for submission rules

# RULES

- No list comprehensions, inline for loops, or inline conditional statements
- No jsonify library
- No global variables
- No convenience functions, e.g. sorted(), sort(), max(), sum(), next()
- No app.route()
- No generic exceptions
- No os module, pathlib only
- Invalid input must never crash the program
- The program must exit cleanly (no output) when forcibly quit
- No changing of folder names - I will be testing your scripts against my own copy of the data