**Testing Course Manager’s REST APIs using Curl commands:**

Start up the Course Manager application using the “*flask run*” command

Note: Make sure you are in the course\_manager directory and have set the environmental variable FLASK\_APP=course\_manager in order for this to successfully run.

It should be running on your localhost at port 5000. This is the following url: *127.0.0.1:5000*

**NOTE: if running through container runtime, it’s necessary to change the port from 5000 to the port returned from the API call**

1. Curl command to make a GET request to a specific endpoint:

curl http://127.0.0.1:5000

This will contact the base endpoint (*http://127.0.0.1:5000*) and send back an authentication error if you are currently not logged in.

If you are logged in, a message saying the user you are logged in as, will be sent back.

1. Curl command to login as a user and and create a session in Course Manager:

curl -d "username=student" --cookie-jar ./session <http://127.0.0.1:5000/api/login>

The --cookie-jar flag saves the session cookie inside of a designated file ( the ‘session’ file in the example above).

The -d argument is used to add data in this request. By default, this signifies a POST request. It places data in the body of the request in the form “key=value”. In this case we are passing in the “student” value for the “username” key.

NOTE: The ROLE environment variable must be set to the corresponding role of the user you want to login in as. By default, it is set to ‘student’. For example, if you want to change the ROLE value to ‘coordinator’ use the following command:

export ROLE=coordinator

The user you are attempting to login as must already exist in the corresponding table (based on their role) for the database you are trying to contact.

1. Curl command to view the account details for a logged in user:

curl --cookie ./session --cookie-jar ./session <http://127.0.0.1:5000/api/user>

This sends a GET request to this API endpoint and will send back a JSON response of this user’s account information.

The --cookie flag sends the session cookie in the request from the designated file it is stored (‘session’ in our case). This must be sent with every REST API request or else you will receive an authentication error.

The --cookie-jar flag is used to modify the updated session cookie in the designated file (‘session’) from the REST response. In order to track a user’s inactivity the session cookie is updated every time the user makes a REST request, therefore, the stored session cookie must be updated each time.

1. Curl command to add a new user to the database:

curl --cookie ./session -d "username=student\_tester&name=Student Test&role=student" --cookie-jar ./session <http://127.0.0.1:5000/api/user>

The -d argument takes in three key/value pairs in the body of the POST request. These consist of the “username”, “name”, and “role” of this user. The ‘&’ symbol is used to separate the key/value pairs.

Similar to the last command, you must both send the session cookie and then update this session based on the cookie sent back.

You should receive a message in the response that the user was successfully added.

1. Curl command to delete an existing user from the database:

curl -X DELETE --cookie ./session -d "username=student\_tester&role=student" --cookie-jar ./session http://127.0.0.1:5000/api/user

The -X <Method> argument allows you to make specific method requests using curl. In this case we are making a DELETE request.

In this request, we must pass in the username and role of the user in the body of the request using the -d argument.

You should receive a message back that the user was successfully deleted.

1. Curl command to logout as a logged in user:

curl --cookie ./session <http://127.0.0.1:5000/api/logout>

This will logout the user and end their session. You will receive authentication errors if you try to make a REST request to Course Manager endpoints until you log back in as a user using the ‘/api/login’ endpoint.

1. Curl command to add a new course:

curl --cookie ./session -d "name=CSC326&days=MW&start\_time=11:00&end\_time=12:45&instructor\_username=instructor" --cookie-jar ./session http://127.0.0.1:8000/api/course

1. Curl command to add delete an existing course:

curl -X DELETE --cookie ./session -d "course\_name=CSC326" --cookie-jar ./session http://127.0.0.1:5000/api/course

1. Curl command to enroll a student in a course:

curl --cookie ./session -d "course\_name=CSC316&username=student" --cookie-jar ./session http://127.0.0.1:5000/api/mapping

1. Curl command to modify the grade in a course for a student:

curl -X PUT --cookie ./session -d "course\_name=CSC316&username=student&grade=3.5" --cookie-jar ./session http://127.0.0.1:8000/api/mapping/grade

1. Curl command to view the schedule of the logged in user:

curl --cookie ./session --cookie-jar ./session [http://127.0.0.1:5000/api/](http://127.0.0.1:5000/api/user)schedule

**Testing Container Runtime’s REST APIs using Curl commands:**

curl -d "username=student&service=course\_manager" --cookie-jar ./session <http://127.0.0.1:5000/service_request>