CS480 Computer Science Education (Fall 2023)
Student: Ronaldo Canizales
Canvas LTI Assignment

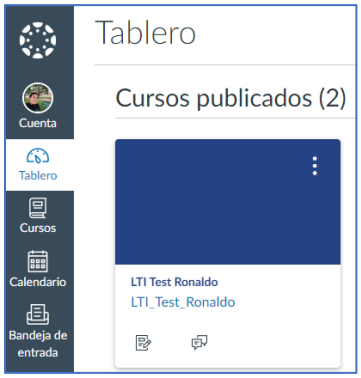
(1) Create a Git repo to post your code and invite me to your repo.

The link to the repository is the following: https://github.com/armandocodigos/Canvas-LTI

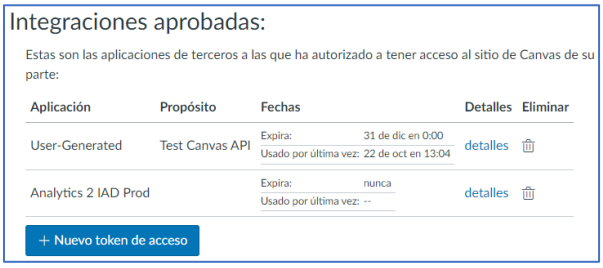
My username meaning in Spanish is “Building Codes,” a word game with my second name, Armando.

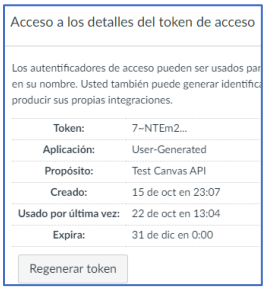
(2) Write a document explaining how you developed your Canvas LTI application.

First, I created a https://canvas.instructure.com/ account and created the course “LTI Test Ronaldo.”

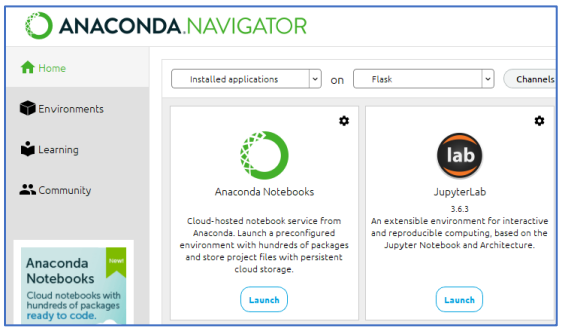


I generated an Access Token in my account’s Approved Integrations to be able to use Canvas API.



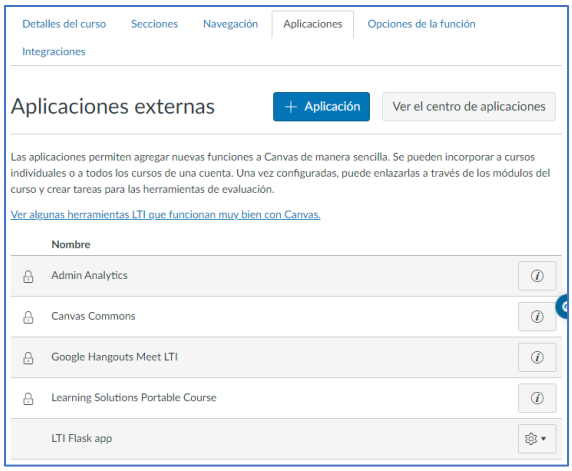


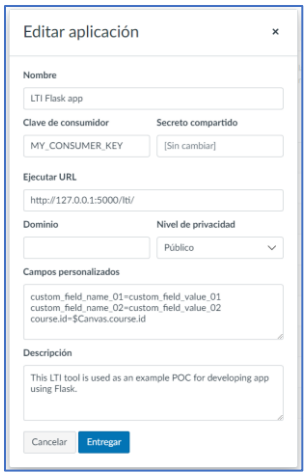
Using Anaconda, I created a new environment for this assignment.



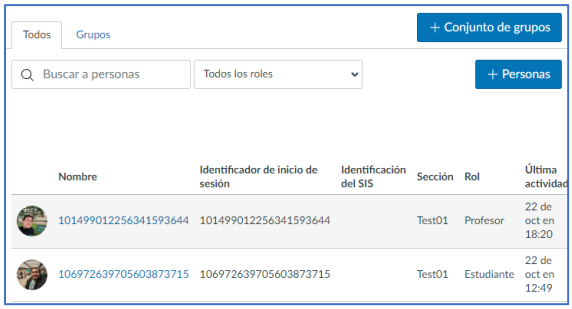
I installed Flask (https://pypi.org/project/Flask/), PyLTI (https://pypi.org/project/PyLTI/), and Canvas API (https://github.com/ucfopen/canvasapi). Then, I cloned the GitHub repository https://github.com/kunal-aga/lti-poc-flask as a starting code for my assignment.

At the External Apps tab of my course Settings, I configured “LTI Flask app” as follows:

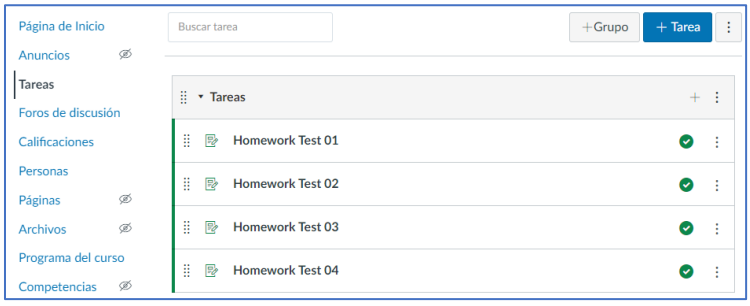




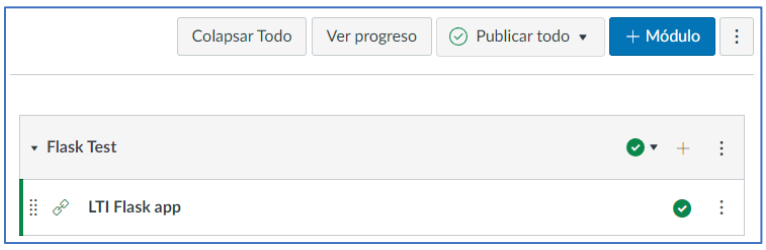
I invited my other Gmail-based account (given by my Salvadoran university) as a student in the course.

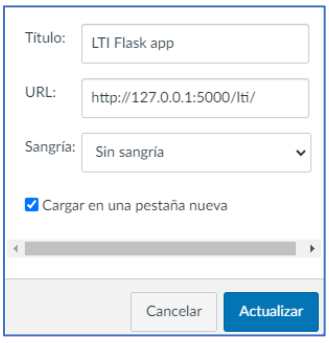


I created four assignments. Then, in my student account, I submitted two of them.



To implement and test my LTI code, I created a “Flask Test” activity inside my course’s Modules.





My code workflow. Inside the "app.py" file, I added the following code to the "index" function:

[a] Set API_URL and API_KEY. Initialize a new Canvas object.

```
API_URL = "https://canvas.instructure.com"
API_KEY = "7~NTEm2g0cHTJWI7ApGGP9TY7rbemxTaRcSVIfs6Dt9hcUqzE1UWEFolnMmqRw2jZD"
canvas = Canvas(API_URL, API_KEY)
```

[b] Retrieve course ID and user ID from custom parameters. Obtain and display the names.

```
course = canvas.get_course(params['custom_course_id'])
teacher = canvas.get_user(params['custom_canvas_user_id'])

OUTPUT_TEXT = "Welcome to my Canvas LTI Assignment.<br>"
OUTPUT_TEXT += "<b>Teacher:</b> {}.<br>".format(teacher.get_profile()['short_name'])
OUTPUT_TEXT += "<b>Course:</b> {}.<br>".format(course.name)
```

[c] Obtain all users enrolled in the course whose role is a student. Display each name and email.

```
OUTPUT_TEXT += "<br><b>List of users in the course:</b><br>".format(params['context_title'])

users = course.get_users(enrollment_type=['student'])
for user in users:
    profile = user.get_profile()
    OUTPUT_TEXT += "<b>Name:</b> {}. ".format(profile['short_name'])
    OUTPUT_TEXT += "<b>Email:</b> {}<br>".format(profile['primary_email'])
```

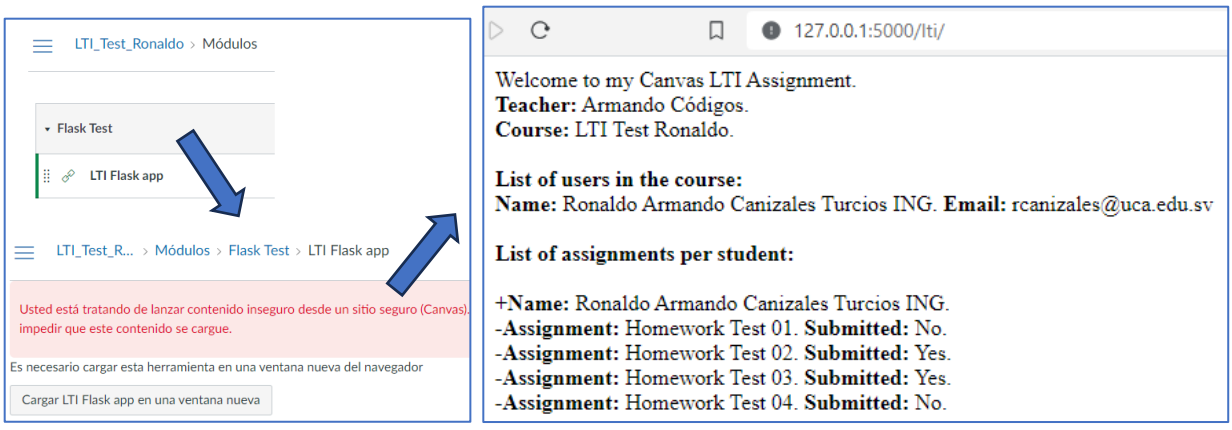
[d] Per student, obtain and display all assignments' information: name and submission status.

```
OUTPUT_TEXT += "<br><b>List of assignments per student:</b><br><br>"

for user in users:
    profile = user.get_profile()
    OUTPUT_TEXT += "<b>+Name:</b> {}.<br>".format(profile['short_name'])

    submissions = user.get_assignments(course)
    submission_list = [sub for sub in submissions]
    for sub in submission_list:
        OUTPUT_TEXT += "<b>-Assignment:</b> {}. ".format(sub.name)
        OUTPUT_TEXT += "<b>Submitted:</b> {}.<br>".format(
                                        "Yes" if sub.has_submitted_submissions else "No")
```

[e] Return the variable "OUTPUT_TEXT" that contains the HTML code to be displayed.

```
return OUTPUT_TEXT
```

Execute the LTI app through the "LTI Flask app" module, which redirects to http://127.0.0.1:5000/lti/



(3) Do a video demonstrating your code and your LTI working.

YouTube URL's: <link>