

CS480 Computer Science Education (Fall 2023)

Student: Ronaldo Canizales

Canvas LTI Assignment

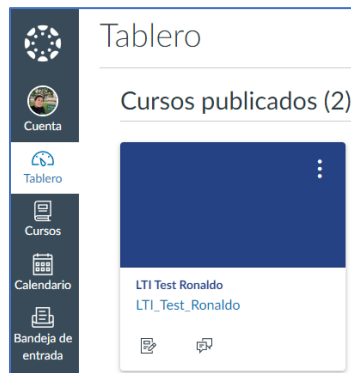
(1) Create a Git repo to post your code and invite me to your repo.

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

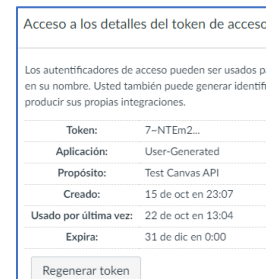
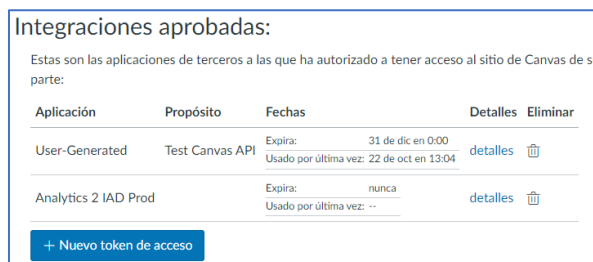
My username meaning in Spanish is “Building Codes,” a word game with my second name, Armando.

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

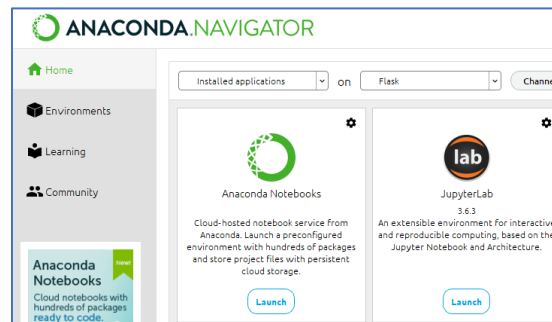
First, I created a <https://canvas.instructure.com/> account and created the course “LTI Test Ronaldo.”



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:

Aplicaciones externas

Las aplicaciones permiten agregar nuevas funciones a Canvas de manera sencilla. Se pueden incorporar a cursos individuales o a todos los cursos de una cuenta. Una vez configuradas, puede enlazarlas a través de los módulos del curso y crear tareas para las herramientas de evaluación.

[Ver algunas herramientas LTI que funcionan muy bien con Canvas.](#)

Nombre	
Admin Analytics	
Canvas Commons	
Google Hangouts Meet LTI	
Learning Solutions Portable Course	
LTI Flask app	

Editar aplicación

Nombre: LTI Flask app

Clave de consumidor: MY_CONSUMER_KEY | Secreto compartido: [Sin cambiar]

Ejecutar URL: http://127.0.0.1:5000/lti/

Dominio: | Nivel de privacidad: Público

Campos personalizados: custom_field_name_01=custom_field_value_01, custom_field_name_02=custom_field_value_02, course_id=\$Canvas.course_id

Descripción: This LTI tool is used as an example POC for developing app using Flask.

Cancelar | Entregar

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

Nombre	Identificador de inicio de sesión	Identificación del SIS	Sección	Rol	Última actividad
101499012256341593644	101499012256341593644		Test01	Profesor	22 de oct en 18:20
106972639705603873715	106972639705603873715		Test01	Estudiante	22 de oct en 12:49

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

Tareas	
Homework Test 01	✓
Homework Test 02	✓
Homework Test 03	✓
Homework Test 04	✓

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

Flask Test

LTI Flask app

Editar aplicación

Título: LTI Flask app

URL: http://127.0.0.1:5000/lti/

Sangría: Sin sangría

☒ Cargar en una pestaña nueva

Cancelar | Actualizar

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> {}".format(teacher.get_profile()['short_name'])
OUTPUT_TEXT += "<b>Course:</b> {}".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> {}".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> {}".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/>

The image shows two side-by-side screenshots. The left screenshot is a Canvas LTI app interface. It has a header 'LTI_Test_Ronaldo > Módulos'. Below it, there's a section 'Flask Test' with a sub-item 'LTI Flask app'. A blue arrow points from 'LTI Flask app' to the right screenshot. Below this, there's a red warning box that says 'Usted está tratando de lanzar contenido inseguro desde un sitio seguro (Canvas). impedir que este contenido se cargue.' and a message 'Es necesario cargar esta herramienta en una ventana nueva del navegador'. At the bottom, there's a button 'Cargar LTI Flask app en una ventana nueva'. The right screenshot is a web browser window showing the output of the LTI app. The address bar shows '127.0.0.1:5000/lti/'. The content is: 'Welcome to my Canvas LTI Assignment. Teacher: Armando Códigos. Course: LTI Test Ronaldo. List of users in the course: Name: Ronaldo Armando Canizales Turcios ING. Email: rcanizales@uca.edu.sv List of assignments per student: +Name: Ronaldo Armando Canizales Turcios ING. -Assignment: Homework Test 01. Submitted: No. -Assignment: Homework Test 02. Submitted: Yes. -Assignment: Homework Test 03. Submitted: Yes. -Assignment: Homework Test 04. Submitted: No.'

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

YouTube URL's: https://youtu.be/vrPIXx_spes (10 minutes)

