



**Universidad Autónoma de Yucatán**

**Facultad de Matemáticas**

**Licenciatura en Ingeniería de Software**

**“Proyecto: Coding Test”**

**Servicio Social**

**INTEGRANTES:**

Iván Alberto Aguilar Castillo

Daniel Alejandro Álvarez Ramírez

**Docente:**

Edgar Cambranes Martínez

**Fecha:**

19/06/2020

## Contexto

El proyecto consiste en conectar GitHub con OmegaUp. Se plantea utilizar la API de GitHub para obtener el repositorio de un alumno dentro de una Classroom de GitHub, para así mandar el código del alumno a OmegaUp y obtener la calificación de su programa.

## Opciones para continuar el proyecto

- Continuar desarrollando la API Coding Test.
- Usar la información recopilada y parte del código de Coding Test y trasladarlas a un nuevo proyecto que sea una extensión para Visual Studio Code.
- Seguir el proyecto con el autograding de GitHub Classroom.

## Peticiones utilizadas

### GitHub

La documentación del API de GitHub está completa y se puede consultar aquí:

<https://developer.github.com/v3/>

### OmegaUp

La documentación del API de OmegaUp es más escasa, se recomienda contactar con el equipo de OmegaUp en caso de cualquier inconveniente, sin embargo, aquí están todas las peticiones de OmegaUp que se encontraron y se utilizaron para Coding Test.

Todos los URL del API comienzan con <https://omegaup.com/api>

## POST /user/login

---

### \*Descripción

Iniciar sesión en OmegaUp.

### \*Privilegios

Ninguno.

### \*Parámetros

Parámetro	Tipo	Descripción
usernameOrEmail	String	Nombre de usuario o email de OmegaUp
password	String	Contraseña del usuario

### \*Regresa

Parámetro	Tipo	Descripción
auth_token	String	Token del usuario
status	String	Estatus de la petición
_id	String	

## POST /run/create

---

### \*Descripción

Mandar el código a OmegaUp.

### \*Privilegios

El usuario necesita estar loggeado..

### \*Parámetros

Parámetro	Tipo	Descripción
problem_alias	String	El alias del problema de OmegaUp
language	String	El lenguaje del problema de OmegaUp:  "c11-gcc", "c11-clang", "cpp11-gcc", "cpp11-clang", "cpp17-gcc", "cpp17-clang", "java", "py2", "py3", "rb", "cs", "pas", "hs", "lua"
source	String	El código para enviar
ouat	String	El token del usuario

\*Regresa

Parámetro	Tipo	Descripción
guid	String	
submit_delay	Int	
submission_deadline	Int	
nextSubmissionTimestamp	Int	
status	String	Estatus de la petición
_id	String	

## POST /problem/details

---

### \*Descripción

Obtener los detalles de un problema de OmegaUp, usamos esta petición para obtener el resultado del código recién enviado.

### \*Privilegios

El usuario necesita estar loggeado.

### \*Parámetros

Parámetro	Tipo	Descripción
problem_alias	String	El alias del problema de OmegaUp
ouat	String	El token del usuario

### \*Regresa

Devuelve mucha información, pero la que nos importa para el propósito del proyecto es la siguiente:

Parámetro	Tipo	Descripción
points	Int	Puntos del problema
score	Int	Resultado del alumno, su “calificación”
exists	Boolean	
status	String	Estatus de la petición
_id	String	

## POST /problem/create

---

### \*Descripción

Para crear un problema en OmegaUp

### \*Privilegios

El usuario necesita estar loggeado.

### \*Parámetros

Parámetro	Tipo	Descripción
author_username	String	Nombre del usuario
title	String	Título del problema
problem_alias	String	Alias del problema
source	String	
public	String	Indica si el problema es público o no
validator	Int	
time_limit	String	Indica el tiempo límite del problema
problem_contest	Archivo Zip	El problema a crear, puede usarse el testproblem.zip que ofrece OmegaUp para hacer pruebas:
ouat	String	El token del usuario

Enlace para descargar los zips de ejemplo:

<https://github.com/omegaup/omegaup/tree/master/frontend/tests/resources>

### \*Regresa

Parámetro	Tipo	Descripción
status	String	Estatus de la petición
_id	String	

# Ventajas y desventajas

A continuación se presentan las ventajas y desventajas de las herramientas que se podrían implementar para la evaluación de código.

## API Coding Test

Es un sistema enfocado en la evaluación de los códigos de los estudiantes que está siendo desarrollado por los autores del reporte mismo.

### **Ventajas**

1. Como es un API propio, pueden añadirse y cambiarse funciones, se puede personalizar para adaptarse a las necesidades de los profesores de nuestra facultad.
2. OmegaUp como herramienta para testing, es superior por el momento al autograding, tiene más funciones y permite más personalización en los problemas.
3. OmegaUp tiene soporte para más lenguajes que el autograding.
4. OmegaUp es un sistema terminado, mientras que el autograding sigue en fase de pruebas y aún pueden surgir errores.
5. El sistema está documentado y explicado, se tiene conocimiento de las peticiones más importantes de OmegaUp que utiliza el sistema.
6. La función que conecta a OmegaUp y GitHub para testear un código funciona y está casi completa, falta el detalle de que al usar el api de GitHub pueda funcionar con un repositorio privado.

### **Desventajas**

1. Depende del api de GitHub y de OmegaUp.
2. Poca documentación del api de OmegaUp.
3. Al ser un API enfocada a los concursos, por su propia naturaleza la forma de acceder a ciertos datos (como la calificación del alumno) es más complicada de lo que debería ser para el uso que le queremos dar.
4. Siguiendo lo anterior, para obtener la calificación de un alumno, se necesita esperar cierto tiempo para que OmegaUp compile el ejercicio y devuelva el resultado más actualizado, de no ser así, devolverá el resultado anterior o no devolverá nada si no existen resultados previos (todo esto ya fue programado, sin embargo, es un detalle importante para tener en cuenta).



5. A pesar de que varios de las complicaciones con el api de OmegaUp fueron resueltas (para la función de testear un código), aún hay otras complicaciones para la creación de un problema en OmegaUp.

## **GitHub Classroom**

Es un servicio de GitHub para educación que permite a los maestros distribuir tareas de programación con código inicial, donde los estudiantes pueden trabajar individual o en grupo usando repositorios públicos o privados, y los maestros pueden monitorear la actividad de los estudiantes.

### **Ventajas**

1. Tiene autograding que permite hacer pruebas unitarias del código de los alumnos.
2. Está enlazado con GitHub para acceder a los repositorios de los estudiantes.
3. Es gratuito.

### **Desventajas**

1. Se hace un repositorio por cada tarea que se establece, lo que puede llevar a tener una cantidad muy grande de estos, por lo que una buena organización y buen uso de las herramientas de gestión es necesaria.
2. No están disponibles los cambios del repositorio original a los nuevos repositorios creados por los estudiantes para hacer la tarea.

## **Gradescope**

Es una plataforma en línea para administración de clases. Permite a los maestros dar seguimiento y calificar tareas en general. Además, provee herramientas para tareas de programación como autograding que funciona para todos los lenguajes.

### **Ventajas**

1. Contiene funcionalidades de gestión para tareas y autograding.
2. El autograding es general para todos los lenguajes.
3. Provee estadísticas, como calificación de los alumnos y número de errores cometidos en cierta prueba, que ayudaría en identificar de mejor manera en donde andan fallando los alumnos.
4. Se conecta con los repositorios de los alumnos en GitHub.

### **Desventajas**

1. Su licencia tiene un costo.

## Web-CAT

Es una herramienta de calificación automática para tareas de programación usando las propias pruebas del estudiante. Es un plugin que puede ser instalado en IDEs como NetBeans o Eclipse cuyo uso es mayoritario en los lenguajes Java, C++ y Python.

### **Ventajas**

1. Permite el aprendizaje de pruebas en los estudiantes, dando detalles como cobertura de las pruebas escritas
2. Es Open source y permite la edición del programa de acuerdo a las necesidades particulares.
3. Puede conectarse a Moodle (no está confirmado).

### **Desventajas**

1. Su instalación es más complicada en comparación con el resto de las herramientas aquí mencionadas.
2. Su documentación no está completa.

\*Para ver la instalación del plugin en NetBean [presione aquí](#).

\*\*URL del plugin <http://web-cat.cs.vt.edu/software/netbeans/updates.xml>

## CodeHS

Es una plataforma de aprendizaje que ofrece herramientas para la enseñanza de la ciencia de la computación. Permite hacer pruebas con de unidad para los lenguajes de Java, JavaScript y Python, además de que provee herramientas para dar seguimiento a las calificaciones obtenidas de los estudiantes.

### **Ventajas**

1. Contiene funcionalidades de gestión para tareas y autograding.
2. El autograding con las pruebas de unidad funciona con los lenguajes de Java, JavaScript y Python.
3. Tiene gran soporte para Java, como pruebas que soportan datos primitivos y objetos.
4. Se puede escribir código inicial para las pruebas de unidad que permiten verificar casos de uso más avanzados.
5. Permite monitorear la actividad de los estudiantes.
6. Es gratuito, sin embargo requiere registro de la institución previa y tiene una versión de paga que contiene más herramientas de gestión. [Más detalles aquí](#).

## **Desventajas**

1. No tiene la funcionalidad de las pruebas de unidad para otros de los lenguajes que se utilizan en algunas licenciaturas de la facultad como C, C# o C++.
2. Para crear cursos personalizados, se requiere del plan de pago.