

## **Test backend - Python**

Implementar una API Rest que permita calcular el Máximo Común Divisor (Greatest Common Divisor), utilizando el [algoritmo de Euclides](#). Realizar una implementación propia (no utilizar las implementaciones de bibliotecas existentes, como `math.gcd`).

### **Requisitos:**

- La API debe ser implementada en Python 3.7+, utilizando el framework [Flask](#) (y eventualmente [flask-restful](#)).
- La API debe estar cubierta por tests.
- Ver definición de endpoints más abajo.

### **Puntos a evaluar:**

- Organización del código. Se espera una clara separación en controladores, servicios y cualquier otra capa que se crea necesaria.
- Tests. Unitarios y de integración.
- Uso adecuado de las tecnologías solicitadas.
- Punto extra (no es obligatorio): Dockerización.
- Punto extra (no es obligatorio): Documentación en formato Swagger/OpenAPI 2.0

### **Endpoints a implementar:**

GET /gcd?a=<int>&b=<int>

En caso de éxito: Devuelve 200 OK, y el body de la respuesta es el `gcd(a, b)`

En caso de error en los parámetros (números decimales, negativos, cadenas, basura, parámetros faltantes, etc): Devuelve 400, bad request.

### **Entregables**

- Un archivo zip "*pipet\_prueba\_tecnica.zip*" con el código fuente y un archivo *readme.md* con la documentación e instrucciones para levantarlo.