

Desafío Falabella Connect

Nombre: Tin Gun Zalabata

Tel: +57 318 5285973

Email: tzalabata15@gmail.com

Ejercicio 1:

Crear un servicio REST para la entrega de patentes (placa automovil) y/o ID, compuesto por:

1. un endpoint donde se ingrese una patente y este retorne el ID asociado a la patente.
2. un endpoint donde se ingrese el ID y este retorne la patente.

El dominio de las patentes corresponde al siguientes: [AAAA000, AAAA001, ..., ZZZZ999], el ID es secuencial, comienza en 1.

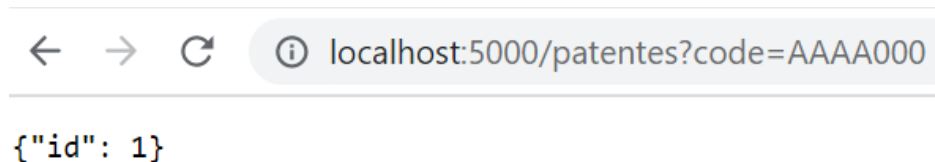
Entonces: AAAA000 tiene ID =1, AAAA001 tiene ID = 2 y sucesivamente.

Se excluye del dominio letra Ñ

ENDPOINTS

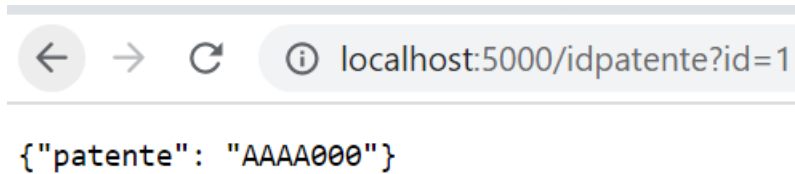
1. /patentes?code=**AAAA000**

En este endpoint denominado patentes, se ingresa una patente a través de la variable code y se retorna un json que contiene el ID asociado a la patente ingresada. Para la patente resaltada en rojo, el json correspondiente retornado por el servicio es {"id": 1}, tal como se aprecia en la siguiente imagen:



2. /idpatente?id=**1**

En este endpoint denominado idpatente, se ingresa un número id y se retorna un json con la patente correspondiente a dicho número. Para el id ingresado (resaltado en rojo), el json retornado por el servicio es {"patente": "AAAA000"}", como puede apreciarse en la siguiente imagen:



DESPLIEGUE

El despliegue se realiza de forma manual utilizando la terminal de comandos. Desde la terminal de comandos se debe ingresar a la carpeta que contiene al proyecto y se ejecutan dos comandos:

- **'docker build -t app_python .'** sin comillas. Este comando se puede ejecutar exitosamente en Windows. Para ejecutarlo en una terminal unix se debe agregar 'sudo' antes del comando.
- **'docker run --rm app_python'** sin comillas. Este comando se puede ejecutar exitosamente en Windows. Para ejecutarlo en una terminal unix se debe agregar 'sudo' antes del comando.

Con esto, se desplegará la aplicación, la cual se puede acceder a través de un navegador ingresando directamente a los servicios programados:

- <http://localhost:5000/patentes?code=AAAA000>
- <http://localhost:5000/idpatente?id=1>

Como se describía antes, para poner a prueba los endpoints, únicamente se deben modificar los datos resaltados en rojo.

HERRAMIENTAS

- Lenguaje de programación: Python
- Framework: flask
- Control de versiones: git, github