

Universidad José Antonio Páez

Facultad de Ingeniería

Departamento de Ingeniería en Computación

**Documentación del Código de la API**

**PROFESOR: ALUMNOS:**

María García Acosta Gregory 29.947.116

Daniel Racero 28.465.991

Daniel Almarza 30.195.507

Daniel Rivero 30.334.420

Manuel Peralta 28.022.934

**1. Introducción:**

El presente informe tiene como objetivo proporcionar una documentación detallada de un código que involucra la implementación de una API RESTful para calcular números de la secuencia de Fibonacci utilizando el framework Flask. Además, se incluye una sección que muestra cómo realizar una solicitud a esta API y mostrar el resultado obtenido.

**2. Estructura del Código:**

El código se divide en dos secciones principales: la primera sección se encarga de crear una API RESTful con Flask, mientras que la segunda sección se centra en realizar una solicitud HTTP a dicha API y mostrar el resultado obtenido.

**2.1. Creación de la API RESTful:**

En esta sección, se utiliza el framework Flask para crear una API que puede recibir solicitudes para calcular números de la secuencia de Fibonacci en una posición determinada.

**2.1.1. Dependencias Importadas:**

- `from flask import Flask, jsonify`: Se importa la clase `Flask` para crear la aplicación Flask y la función `jsonify` para convertir objetos en respuestas JSON.

**2.1.2. Creación de la Aplicación Flask:**

- `app = Flask(\_name\_)`: Se crea una instancia de la clase `Flask` con el nombre de la aplicación como parámetro.

**2.1.3. Cálculo de la Secuencia de Fibonacci:**

- `def fibonacci(n)`: Se define una función denominada `fibonacci` que toma un parámetro `n`.

- El algoritmo utilizado para calcular el número de Fibonacci en una posición determinada es el método de programación dinámica.

- Se crea una lista `fib` de tamaño `n+1`, inicializada con ceros.

- Se asignan los valores iniciales de la secuencia de Fibonacci a `fib[0]` y `fib[1]`.

- Se itera desde el índice 2 hasta `n` (inclusive) y se calculan los números de Fibonacci sucesivos mediante la suma de los dos números anteriores.

- Finalmente, se devuelve el número de Fibonacci en la posición `n`.

**2.1.4. Ruta de la API para Calcular un Número de Fibonacci:**

- `@app.route('/fibonacci/<int:n>', methods=['GET'])`: Se define una ruta `/fibonacci/<n>` para la API con el método GET y parámetro `n` especificado como entero.

- `def get\_fibonacci\_number(n)`: Se define una función denominada `get\_fibonacci\_number` que toma un parámetro `n`.

- Se calcula el número de Fibonacci llamando a la función `fibonacci` definida anteriormente con el valor de `n`.

- A continuación, se devuelve el número de Fibonacci en formato JSON utilizando la función `jsonify`.

**2.1.5. Inicio de la Aplicación Flask:**

- `if \_name\_ == '\_main\_':`: Se verifica si el script se está ejecutando directamente.

- `app.run()`: Se inicia la aplicación Flask y se la hace accesible en `http://127.0.0.1:5000/`.

**2.2. Realización de una Solicitud a la API y Mostrar el Resultado:**

Esta sección muestra cómo realizar una solicitud HTTP a la API Flask y mostrar el número de Fibonacci obtenido como resultado.

**2.2.1. Dependencias Importadas:**

- `import requests`: Se importa el módulo `requests` para realizar solicitudes HTTP.

**2.2.2. Solicitud a la API y Presentación del Resultado:**

- Se solicita al usuario que ingrese el valor de `n` para el número de Fibonacci.

- Se construye la URL de la API utilizando el número de Fibonacci deseado.

- Se realiza una solicitud GET a la URL de la API utilizando la función `requests.get()`.

- Se verifica si la solicitud fue exitosa (código de estado 200).

- Si la solicitud fue exitosa, se obtiene el resultado de la respuesta en formato JSON.

- Se muestra el número de Fibonacci obtenido de la respuesta JSON.

- En caso de que la solicitud no haya sido exitosa, se muestra un mensaje de error junto con el código de estado de la respuesta.

**3. Funcionalidad del Código:**

La funcionalidad del código es la siguiente:

- La primera sección del código crea una API RESTful utilizando Flask que permite calcular números de la secuencia de Fibonacci en una posición determinada.

- La segunda sección del código muestra cómo realizar una solicitud HTTP a esta API y obtener el