



## **API para Mocks de Servicios REST**

### **Contexto**

En proyectos de desarrollo de software es muy común tener que simular el comportamiento de servicios externos mediante mocks, lo cual facilita el desarrollo y las pruebas de sistemas que dependen de ellos.

El objetivo de este challenge es desarrollar una API que permita definir y gestionar mocks de endpoints. La API deberá devolver respuestas personalizadas según la configuración especificada, teniendo en cuenta la ruta, el método HTTP, y los parámetros enviados en la URL, el cuerpo de la solicitud o los headers, así como variaciones en el Content-Type.

### **Requisitos Funcionales**

#### **1. Configuración Dinámica de Mocks**

- Endpoint de Configuración
  - POST /configure-mock
- Permite registrar nuevas configuraciones para mocks. La configuración deberá incluir:
  - Ruta: URL para acceder al mock (ej. /api/v1/productos)
  - Método HTTP: GET, POST, PUT, DELETE, etc.
  - Parámetros en URL: (ej. fechaInicial=2021-01-29 13:01:00&fechaFinal=2021-01-29 13:06:00)
  - Parámetros en Body: (ej. {"telefono": "45456060"})
  - Encabezados: (ej. Authorization: Bearer xxxxxxxxxxxxxxxxx)
  - Código de Estado HTTP: 200, 404, 500, etc.
  - Contenido de la Respuesta: Datos estáticos o basados en plantillas.
  - Content-Type: (ej. application/json, text/xml)

## **2. Ejecución del Mock**

### **Endpoint Genérico**

La API debe escuchar todas las rutas configuradas. Al recibir una solicitud:

- Buscar la configuración correspondiente.
- Verificar coincidencia del método HTTP y parámetros.
- Devolver la respuesta configurada con el código HTTP y Content-Type.
- Si no hay coincidencia, devolver 404 Not Found.

## **3. Gestión de Configuraciones**

- Ver configuraciones: GET /configure-mock
- Eliminar configuración: DELETE /configure-mock/:id
- (Opcional) Actualización de configuración

## **4. Variaciones y Lógica Condicional**

La configuración debe admitir variaciones en función de los parámetros, por ejemplo:

- Si usuario=admin, retornar una respuesta distinta.
- Se valora el uso de lógica condicional y plantillas dinámicas.

## **Requisitos Técnicos**

### **1. Lenguaje y Framework**

Puedes usar el lenguaje y framework de tu preferencia:

- Node.js + Express
- Python + Flask / FastAPI
- Otro

### **2. Documentación**

Este archivo README.md debe incluir:

- Instrucciones de instalación y ejecución
- Ejemplos de configuración y solicitudes (curl)
- Descripción de la arquitectura y decisiones de diseño

### **3. Buenas Prácticas**

- Manejo adecuado de errores
- Validación de datos
- Código modular y limpio

#### **4. Pruebas Unitarias(Opcionales)**

- Implementar pruebas para demostrar la robustez de tu solución.

#### **5. Entrega**

- Subir el proyecto a un repositorio público (GitHub, etc.)
- Incluir instrucciones claras para probar la aplicación

#### **Bonus (Opcionales)**

- Agrega funcionalidades extra que creas que puedan aportar valor adicional a la solución.

#### **Puntos a Evaluar**

- Diseño y Arquitectura: Claridad en la modularidad y separación de responsabilidades
- Funcionalidad: El API debe comportarse según la configuración definida
- Calidad del Código: Legibilidad, buenas prácticas, patrones adecuados
- Documentación y Pruebas: Claridad y cobertura de pruebas
- Creatividad y Extensibilidad: Funcionalidades extra y facilidad para escalar

## PROMPTS

- Soy un estudiante de ingeniería en ciencias y sistemas de la universidad de San Carlos de Guatemala y quiero conocer las metas, objetivos, alcances, riesgos entre otras, sobre la empresa de Tigo Guatemala en todos los ámbitos tales como la ventas, análisis de datos, data centers, desarrollo de software entre otras actividades que ellos realizan en la empresa.
- Necesito que actúes como un experto en desarrollo de software con experiencia en proyectos de desarrollo empresarial, toma de requerimientos, manejo de base de datos, seguridad, escalabilidad, buenas prácticas y todo los puntos que no te mencione faltantes que conlleva el ciclo de vida del desarrollo de software.
  - Necesito que actúes como un experto en desarrollo de software con experiencia en proyectos de desarrollo empresarial tomando en cuenta el ciclo de vida del desarrollo de software.
  - Necesito que actúes como un experto en desarrollo de software con experiencia en proyectos de desarrollo empresarial, especialmente del lado del backend realizando APIS.
- Con base a la información que me compartiste sobre TIGO Guatemala y que ahora ya eres un experto en desarrollo de software empresarial con muchos años de experiencia, necesito que únicamente analices el documento que te voy a compartir a continuación la cual describe las instrucciones del proyecto a realizar.
- Ahora necesito que solo te enfoques en los 4 puntos de los requisitos funcionales y poder construir el proyecto tomando en cuenta tus sugerencias,arquitectura propuesta y observaciones como una primera versión funcional. Primero necesito un árbol de carpetas de la arquitectura de la solución, quiero que la solución sea sencillo, ordenado y claro de entender , respetando las reglas de codificación y buenas prácticas del lenguaje de programación, framework y las tecnologías necesarias a utilizar, las cuales muestro a continuación:
  - Backend: Node.js + Express
  - Testing: Jest
  - Containerization: Docker + Docker Compose

- Ahora necesito que solo te enfoques en los 4 puntos de los requisitos funcionales. Primero necesito un árbol de carpetas de la arquitectura de la solución, quiero que la solución sea sencillo, ordenado y claro de entender , respetando las reglas de codificación y buenas prácticas del lenguaje de programación, framework y las tecnologías necesarias a utilizar, las cuales muestro a continuación:
  - Backend: Node.js + Express
  - Containerization: Docker + Docker Compose