

# Prueba Técnica: "Map My World"

---

## Introducción al Desafío

Imagina que estás construyendo el backend para 'Map My World', una aplicación destinada a explorar y revisar diferentes ubicaciones y categorías del mundo, como restaurantes, parques y museos. La meta es ofrecer a los usuarios un mapa interactivo donde puedan descubrir nuevas ubicaciones y ver recomendaciones basadas en categorías específicas. Pero hay un giro: queremos asegurarnos de que las recomendaciones estén siempre actualizadas y sean relevantes.

## Tu Misión

Como parte del equipo de desarrollo de 'Map My World', tu tarea es construir el corazón de nuestra aplicación: una API REST que maneje la lógica para añadir nuevas ubicaciones y categorías, y más importante aún, una característica especial que nos permita mantener nuestras recomendaciones frescas y atractivas para nuestros usuarios.

## Tecnologías

Para la prueba se debe utilizar Python/FastAPI

## Especificaciones Técnicas

### **\*\*Modelos de Datos\*\***

- - **\*\*Ubicaciones (`locations`)\*\***: Cada ubicación tiene una longitud y latitud.
- - **\*\*Categorías (`categories`)\*\***: Cada categoría representa un tipo de lugar para explorar.
- - **\*\*Revisiones de Ubicación-Categoría (`location\_category\_reviewed`)\*\***: Este registro es nuestro control de calidad, asegurándonos de que cada combinación de ubicación y categoría sea revisada regularmente.

### **\*\*Funcionalidades Clave\*\***

- - **\*\*Gestión de Ubicaciones y Categorías\*\***: Permite a los usuarios de nuestra API añadir nuevas ubicaciones y categorías.
- - **\*\*Recomendador de Exploración\*\***: Un endpoint especial que sugiere 10 combinaciones de ubicación-categoría que no han sido revisadas en los últimos 30 días, priorizando las que nunca se han revisado.

## Objetivos

- - Claridad y Estructura: Tu código debe ser claro y estar bien estructurado. Queremos ver cómo organizas tu proyecto y cómo aplicas las buenas prácticas de codificación.
- - Eficiencia y Optimización: Las recomendaciones deben ser generadas de manera eficiente, teniendo en cuenta tanto la velocidad de respuesta como el uso de recursos.
- - Documentación: Una buena API no es solo sobre el código; también es sobre cómo comunicas su uso a otros desarrolladores. Esperamos ver cómo documentas los endpoints y modelos.