### **Prueba para Candidato Backend Semisenior: Desarrollo de una API de Gestión de Citas para Peluquería**

**Objetivo:** Crear una API sencilla pero funcional en Python para gestionar reservas de citas en una peluquería.

#### **Requerimientos Básicos:**

**Clientes y Peluqueros:**

* Crear funcionalidades para añadir nuevos clientes y peluqueros a la base de datos. Los clientes deben tener nombre, teléfono y email, y los peluqueros deben tener nombre y una lista de servicios que ofrecen.
* **Reservas de Citas:**
* Desarrollar una funcionalidad para que los clientes puedan reservar citas, eligiendo un peluquero y uno de sus servicios. La cita debe incluir una fecha y hora específicas.
* **Verificación de Disponibilidad:**
* Asegurarse de que un peluquero no pueda ser reservado para dos citas al mismo tiempo.
* **Notificaciones:**
* Implementar un sistema básico para enviar emails de confirmación de cita.
* **Finalización de Citas:**
* Permitir que los peluqueros marquen las citas como completadas.

#### **Requerimientos de Creatividad:**

**Innovación en la Funcionalidad:**

* Añade una característica única que diferencie tu API de una típica aplicación de gestión de citas. Puede ser algo relacionado con la experiencia del usuario, la eficiencia del servicio, características adicionales para clientes o peluqueros, etc.

#### **Consideraciones Adicionales:**

* **Claridad y Estructura:** El código debe ser fácil de entender y bien organizado.
* **Documentación:** Incluye comentarios o documentación breve que explique cómo funciona tu API y cómo usarla.
* **Pruebas:** Escribe algunas pruebas básicas para demostrar que tu API funciona como se espera.
* **Despliegue (Opcional):** Si puedes, despliega tu API en un servicio gratuito y proporciona un enlace para probarla.

#### **Entrega:**

* Sube tu proyecto a un repositorio de GitHub y comparte el enlace.
* Asegúrate de incluir instrucciones claras sobre cómo instalar y ejecutar tu proyecto.

### **Evaluación:**

**La prueba será evaluada en base a los siguientes criterios:**

**Funcionalidad:**

* La API debe cumplir con todos los requerimientos básicos y funcionar correctamente.
* **Calidad del Código:**
* El código debe ser limpio, bien organizado y seguir las mejores prácticas de desarrollo en Python.
* Se valorará la aplicación correcta de principios como DRY (Don't Repeat Yourself) y KISS (Keep It Simple, Stupid).
* **Innovación y Creatividad:**
* La característica única añadida para mejorar la funcionalidad o la experiencia del usuario.
* **Documentación:**
* Claridad y completitud de la documentación, incluyendo comentarios en el código y un README informativo.
* **Pruebas:**
* Calidad y cobertura de las pruebas escritas, utilizando pytest u otro framework de pruebas en Python.
* **Manejo de Errores:**
* Implementación efectiva del manejo de errores y respuestas adecuadas enviadas al cliente.
* **Despliegue (Opcional):**
* Si se realiza, la facilidad de acceso y funcionamiento de la API en el entorno de despliegue.

**Nota:** Mientras que el despliegue es opcional, se considerará como un valor añadido a la prueba.