

Prueba Técnica (offline):

Plazo: 3 días.

Prueba de Evaluación: Ingeniero de Software Senior

Objetivo: Evaluar la competencia técnica, la capacidad de resolución de problemas y la capacidad de adaptación del candidato a situaciones reales en el contexto del mercado tecnológico.

Secciones:

1. Algoritmos y Estructuras de Datos

○ Pregunta 1 (*Complejidad y Optimización*):

- Genere creativamente un gran conjunto de datos de transacciones de clientes de una plataforma local de comercio electrónico. Cada transacción incluye una marca de tiempo, un ID de cliente y el importe de la compra.

Diseñe un algoritmo para identificar eficientemente a los 10 clientes más frecuentes dentro de un período de tiempo específico. Analice la complejidad temporal y espacial de su solución. Considere la posibilidad de gestionar conjuntos de datos muy grandes que podrían exceder la memoria disponible.

○ Pregunta 2 (*Estructuras de Datos*):

- Un sistema de software necesita gestionar una lista de rutas de transporte público, donde cada ruta tiene un identificador único y un conjunto de paradas.

Diseñe una estructura de datos que permita la recuperación eficiente de rutas en función de una parada determinada, así como la adición o eliminación eficiente de paradas en una ruta. Explique su elección de la estructura de datos y justifique su idoneidad para este escenario.

2. Diseño y Arquitectura del Sistema

○ Escenario:

- Una start up local de reparto de comida a domicilio está experimentando un rápido crecimiento. Necesita rediseñar su sistema backend para mejorar la escalabilidad, la fiabilidad y la facilidad de

mantenimiento. El sistema actual es una aplicación monolítica con un rendimiento limitado. **Diseñe** una arquitectura de sistema distribuida que aborde estos desafíos. Considere factores como el diseño de la base de datos, el diseño de la API, las colas de mensajes y el almacenamiento en caché. Explique sus opciones de diseño y justifique su arquitectura.

3. Codificación y Resolución de Problemas

- Problema:
 - **Desarrolle** un endpoint de API RESTful en un lenguaje de su elección (*Python, Java, Node.js, etc*) que gestione lo siguiente:
 - Reciba un payload JSON con una lista de productos con sus precios y cantidades.
 - Calcule el costo total del pedido, incluyendo el cargo por envío a domicilio.
 - Aplique un descuento basado en el monto total del pedido, si corresponde.
 - Devuelva una respuesta JSON con el costo total y el descuento aplicado, si corresponde.
 - Considere el sistema de "*estrato*" en Colombia y si el cargo por envío podría variar según éste valor.
- Requerimientos:
 - Escriba un código limpio y bien documentado.
 - Incluya pruebas unitarias para verificar la funcionalidad de su código.
 - Demuestre su comprensión de las mejores prácticas para el desarrollo del API.