



Backend DevOps - Code Test

Con esta prueba queremos entender tu forma de pensar, es una prueba muy abierta, las dudas o las asunciones también serán valoradas.

El enunciado está pensado sobre un subconjunto super simplificado de uno de los casos de uso de nuestra plataforma. Realiza las asunciones que consideres necesarias.

La prueba está pensada para ser realizada en menos de 6h, pero puedes tomarte el tiempo que necesites.



Enunciado

User story

Desde una de nuestras aplicaciones cliente (Android, iOS y Web), un cliente una vez ha elegido una de las tiendas disponibles en el marketplace debe poder añadir productos al carro y realizar el pedido.

Posteriormente este pedido debe ser asignado a un personal shopper para poder ser comprado en la tienda correspondiente y entregado en la franja horaria selecciona por el cliente.

Acceptance criteria

Use Case 1: Recepción de pedidos.

Necesitamos exponer un servicio web a través de nuestro API REST que sea capaz de gestionar y almacenar los pedidos recibidos, la base de datos donde queremos almacenarlo utiliza un motor MySQL.

El pedido debe contener al menos los siguientes campos:

- Nombre y apellidos del cliente.
- Email.
- Teléfono.
- Dirección de entrega (solo puede existir una por pedido)
- Fecha de compra.
- Fecha de entrega.
- Franja horaria seleccionada para la entrega.
- Productos seleccionados con sus atributos (al menos 5 productos)
 - Nombre
 - Descripción
 - Unidades
 - Precio
 - Tienda de venta
- Importe total

Use Case 2: Dispatching de pedidos

Por otro lado están nuestros shoppers, que mediante su aplicación realizarán la compra de los productos solicitados por los clientes.

Un cliente puede hacer la compra en varios supermercados pero un shopper se encarga de los productos de un supermercado en concreto.

Necesitamos exponer un WS a través de nuestro API REST que reciba como parámetro el ID del shopper y el ID de la tienda donde está comprando y nos devuelva un JSON con la información del pedido asignado y los productos que tiene que comprar.

Job to be done

- ☐ Construir el modelo de datos en MYSQL. Tablas, claves, índices... etc
- ☐ Sentencias SQL necesarias para persistir el pedido en base de datos. Inserts, updates, etc

- ☐ Query o queries SQL para mostrar los productos que debe comprar el shopper
- ☐ Crear un entorno de desarrollo con docker-compose que levante un stack de php, un servidor web (apache o nginx) y una base de datos MySQL.
- ☐ Script (da igual el lenguaje) que levante el docker, cree la base de datos y cargue datos de prueba suficientes como para ver el funcionamiento del modelo.

Evaluable

- Diseño modelado de datos.
- Explicación del modelo y toma de decisiones.
- Escalabilidad en el diseño del modelo.
- Instrucciones para la puesta en marcha del docker.

Entrega

Mediante subida del ejercicio a Bitbucket/Github/... o enviando un email con el código a tech@lolamarket.com