

Evaluación técnica – navent.com

1- Se desean modelar los Pedidos realizados por un cliente. Las operaciones que se necesitan son: crear pedidos, modificar pedidos, buscar pedidos por id, y borrar pedidos. Se pide modelar un servicio que implemente estas operaciones (en lenguaje Java) sabiendo que se utiliza una estructura de caches para no acceder a la base de datos en cada operación solicitada a este servicio backend. El servicio es utilizado en un sitio web, con usuarios concurrentes. Tomar como ya implementadas las clases BumexMemcached y PedidosDAO, descritas más adelante."

Datos:

Tabla Pedido:

Campos: idPedido, nombre, monto, descuento

La clase BumexMemcached es un singleton que tiene los siguientes métodos (tomarlos como ya implementados, no es necesario codificarlos):

- void set(String key, Object value)
- Object get(String key)
- void delete(String key)

La clase PedidosDAO tiene los siguientes métodos estáticos que actualizan la tabla Pedido (tomarlos como ya implementados, no es necesario codificarlos):

- void insertOrUpdate(Pedido pedido): inserta un nuevo pedido en la base de datos o modifica un pedido existente (en caso de crear uno nuevo, el pedido pasado como parámetro se completa con el nuevo id).
- void delete(Pedido pedido): elimina el pedido que corresponde al id recibido.
- Pedido select(Integer idPedido): busca un pedido por id.

2- Suponiendo que la tabla Pedidos tiene muchos registros y columnas (algunas de ellas nullable, algunas BLOB / "binary-large-object"), que consideraciones se deberían tener en cuenta para realizar desde un sitio web consultas a la base de datos de manera eficiente? Discuta performance a nivel motor de base de datos, networking, capa applicativa desde donde se realizan las solicitudes, entre otros aspectos que considere relevantes.

3- Implemente en una página HTML con el código javascript correspondiente el formulario para guardar un Pedido a través de una invocación AJAX a la URI **/pedidos/guardar**.

Aplicar las siguientes validaciones a los campos:

- **nombre** es un campo obligatorio y no puede superar los 100 caracteres
- **monto** es un campo obligatorio del tipo integer
- **descuento** es un campo del tipo integer