

## Iteración 2 SuperAndes

1. Con respecto a la iteración uno se realizaron varios cambios en el modelo del mundo y, por lo tanto, en el modelo de tablas. Los cambios que se hicieron fueron: Primero, se añadió una nueva clase en el modelo del mundo la cual representaba el carrito de compras, para esta, se consideró que tendría un cliente asignado y una fecha de creación (Para poder analizar en cuanto tiempo un carro estaría abandonado). Ahora, con respecto a los productos que puede tener el carrito, se crea una nueva relación en el modelo conceptual y en las tablas ya que esto representaba una relación de muchos a muchos, teniendo en cuenta que en este modelo un producto representa todos los productos de este mismo en el supermercado, lo que permite que un producto pueda estar en más de un carrito a la vez.

Como siguiente serie de cambios, para poder analizar la operación de SuperAndes, se añadió un atributo indicando la sucursal de la cual se está comprando a la factura. Ya que esta clase no se utilizó en la iteración anterior. Además, para mejorar las promociones se eliminó la relación de promociones productos y se añadió un identificador de producto en la Promoción. Como último cambio al modelar las facturas se decidió eliminar la relación de sucursales con clientes ya que no era necesario que estos dos estuvieran relacionados.

2. La agregación de estos nuevos cambios no trae problemas al modelo inicialmente propuesto, ya que los cambios no fueron muy grandes en la lógica ni en el acceso de datos en la aplicación. El modelado de un carrito de compras en realidad permite modelar situaciones no contempladas al caso en que no estuviese.

Con respecto a la lógica que hay que tener en cuenta para esta iteración. Se propone que el carrito de compras sea un elemento que se persista con sus elementos para que no se pierda la información de una compra en proceso, así, cuando un cliente está agregando productos a un carrito estos se persisten para que no haya perdidas en caso de fallas externas y que la base de datos quede coherente. Al momento que el cliente introduce un producto al carrito este se descuenta del estante y se adiciona al carrito, en caso de que se abandone el carrito, simplemente se retira la asociación del cliente con ese carrito. Después de que un carrito esté abandonado la aplicación por si misma recoge los productos que aún están dentro del carrito.

Para que la compra del cliente sea privada y solo él pueda acceder al carrito se le asocia al carrito el cliente, pero además al persistirlo junto con el cliente, indica que solo ese cliente usa el carrito de compras. Por lo tanto, existen más carritos. Un carrito sin cliente es un carrito disponible siempre y cuando se le retiren los productos que han sido abandonados.

Cuando el cliente decida pagar se van a restar las unidades de cada producto que el cliente añadió al carrito, de la base de datos general. El carrito se desvincula del cliente y se genera una factura que muestra los productos comprados.

3. Resultados logrados:

Logramos exitosamente en esta iteración completar los requerimientos de la iteración anterior, así como casi todos los requerimientos funcionales de esta. Con todos los requerimientos solicitados para el carrito de compras y un acercamiento a la orden de pedido al proveedor.

Resultados no logrados:

Para esta entrega no logramos completar los requerimientos de consulta, así como las pruebas de los requerimientos funcionales con diferentes casos. Esto debido a falta de tiempo y que en la iteración anterior nos haya faltado la mayoría del trabajo que hubo que completar en esta iteración.

Objetivos:

Para la siguiente iteración completar los requerimientos faltantes de esta y organizar mejor el tiempo para terminar completamente lo faltante.