Iteración 2: SuperAndes

Andrés Felipe Hernández León, Jenifer Paola Rodríguez Villamizar Reporte técnico Iteración 2
Universidad de los Andes, Bogotá, Colombia {af.hernandezl, jp.rodriguezv}@uniandes.edu.co
Fecha de presentación: Noviembre 4 de 2018

Tabla de contenido

1 Inti	roducción	1
2 An	álisis	2
2.1		2
3.2	Modelo de datos relacional	4
5 Co	nstrucción de la aplicación	4
5.1	Ajuste de tablas creadas en Oracle	4
5.2	Población de las tablas	4
5.3	Ajustes correspondientes a los nuevos requerimientos:	5
5 4	Pruehas	

1 Introducción

En el siguiente documento se evidenciará el desarrollo de la segunda iteración del curso. Para cada parte de la actividad, se mostrará el trabajo logrado según lo requerido en el documento del proyecto. Adicionalmente, se podrán encontrar anexos a este archivo archivos .sql con las sentencias de los requerimientos en texto plano y el proyecto java.

2 Análisis

2.1 Modelo conceptual SuperAndes

En la figura 1 se muestra el modelo conceptual completo correspondiente al caso de estudio "SuperAndes". Este modelo sufrió varios cambios para esta iteración que se mencionarán más adelante. Si no se puede observar bien la imagen, también esta anexada como se pedía en la carpeta data de nuestro proyecto en java.

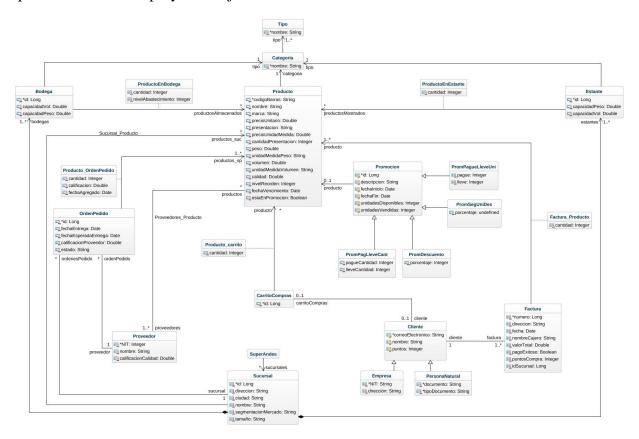


Figura 1. Modelo UML SuperAndes

En primer lugar, el modelo del mundo surge de la combinación de nuestras dos iteraciones, por tanto, hay cosas de ambas iteraciones. Se tomó como base el modelo del mundo de la iteración de Andrés y se complementó con el de Jenifer, así, el principal de los cambios fue en las promociones, donde se plantea una clase de la que hereda cada tipo de promoción para manejar correctamente los distintos tipos de promociones. Por otro lado, se modelo el administrador de la aplicación, quien conoce las distintas Sucursales del programa.

En cambios algo más pequeños, se asignó un id generado por el sistema para las sucursales como identificador único como se sugirió en la sustentación anterior (antes solía ser una llave compuesta entre dirección y ciudad de la sucursal). Siguiendo las otras sugerencias, se borró el vínculo entre el producto y la promoción, el producto ahora no conoce la promoción, pero si tiene un estado para saber si se encuentra en promoción o no y la cardinalidad se invirtió para que las promociones conozcan a sus productos.

3 Diseño de la aplicación

3.1 Modelo de datos relacional

Se presenta el modelo de datos relacional según los estándares del curso. También esta adjunto en un Excel en la documentación del proyecto por si se desea explorar un poco más fácilmente (el Excel se encuentra en "data/Iteracion2/ModeloRelacional").

		CATEGORIA			TI	20			TIPO_PRODU	TO	
		VARCHAR2(100BYTE)			VARCHAR2(100BYTE)		VA	RCHAR2(100 BYTE)	VARCHAR2(100 BYTE	
		nombre			no	ombre			rasProducto	nombreTipo	
		PK, UA				, UA			ODUCTO.codigoBarras, UA		
		CARRITOCO	MPRAS								
NUMBER		VARCHAR2(255 BYT	E)	NUMBER							
id		cliente	114	sucursal	INI NG						
PK, SA		FK_CLIENTE.correo,	UA	FK_SUCURSAL.id, UA, N	IN, NC						
		PRODUCTO_	CARRITO								
IUMBER		NUMBER		VARCHAR2(13BYTE)							
arrito		cantidad		codigoBarrasProducto							
K, FK.CARRITOCOMPF	RAS.id, SA	NN, UA		PK, FK.PRODUCTO.codi	goBarras, SA						
			1	BODEC	SA .						
NUMBER		NUMBER		NUMBER		HAR2(100 BYTE)	NUMBER				
d		acidadVol	capacidadPeso	1	tipo		idSucursal				
K, SA	NN,	UA	NN, UA		FK_CATEGORIA	.NOMBRE, NN, UA	FK_SUCURSAL.id, NN, UA, NC				
				FCTAN	TF						
NUMBER		NUMBER		ESTAN NUMBER	VARCHAR2(100) RVTF)	NUMBER				
d d	cand	acidadVol	capacidadPeso		tipo	, UTIL)	idSucursal		-		
PK, SA	NN,		NN, UA			.NOMBRE, NN, UA	FK SUCURSAL.id, NN, UA, NC				
, 3/1	IVIV,	J.,	, OA		CATEGORIA		Joconomia, NN, OA, NC				
						SUCURSAL					
NUMBER	VAR	CHAR2(255 BYTE)	VARCHAR2(15	O BYTE)	VARCHAR2(200	BYTE)	VARCHAR2(200 BYTE)		NUMBER		
d		ccion	ciudad		nombre		segmentacionMercado		tamanio		
PK, UA	NN,	ND, UA	NN, UA		NN, ND, NC, UA		NN, UA		NN, UA, NC		
					PRODU	JCTOENBODEGA					
	NUMBER			NUMBER(*)			NUMBER(*)		VARCHAR2	13BYTE)	
	· · · · · · · · · · · · · · · · · · ·					1 101					
idBodega			cantidad			nivelAbastecin	niento		codigoBarrasProducto		
PK, FK.BODEGA.id, SA			NN, UA			NN, NC, UA		PK,	FK_PRODUCTO.c	odigoBarras, SA	
				BROBLISTOFNE	OTANITE						
				PRODUCTOENE	STANTE						
	NUMBER			NUMBER(*)		VARCHAR2(13BYTE)					
idEstante			cantidad				• • • • • • • • • • • • • • • • • • • •				
						codigoBarrasProducto					
PK, FK.ESTANT	E.id, SA		NN, UA			PK, FK.PRODU	CTO.codigoBarras, SA				
					SHCLIB	SAL PRODUCTO					
					30C0K						
			NUMBER			VARCHAR2(13	BYTE)				
						codigoBarrasP	roducto				
			lidSucursa								
			idSucursa		NC	_					
				RSAL.id, NN, UA,	NC	_	ICTO.codigoBarras, SA				
	PERSONA	NATURAL			NC	_		EMPRESA			
/ARCHAR2(100B			FK_SUCUI		NC	PK, FK_PRODU	ICTO.codigoBarras, SA		2(255BYTE)		
		VARCHAR2(FK_SUCUI		NC	PK, FK_PRODU VARCHAR		VARCHAR	2(255BYTE)		
locumento		VARCHAR2(1 tipoDocume	FK_SUCUI		NC	PK, FK_PRODU VARCHAR	ICTO.codigoBarras, SA	VARCHAR direction			
documento		VARCHAR2(FK_SUCUI		NC	PK, FK_PRODU VARCHAR	ICTO.codigoBarras, SA	VARCHAR			
documento		VARCHAR2(1 tipoDocume	FK_SUCUI		NC	PK, FK_PRODU VARCHAR	ICTO.codigoBarras, SA	VARCHAR direction			
documento		VARCHAR2(1 tipoDocume	FK_SUCUI		NC	PK, FK_PRODU VARCHAR	ICTO.codigoBarras, SA	VARCHAR direction			
documento		VARCHAR2(1 tipoDocume	FK_SUCUI		NC	VARCHAR nit PK, UA	ICTO.codigoBarras, SA	VARCHAR direction			
documento PK, UA	BYTE)	VARCHAR2(: tipoDocume NN, UA, CK	FK_SUCUI	RSAL.id, NN, UA,	NC	VARCHAR nit PK, UA	CTO.codigoBarras, SA 2(100BYTE)	VARCHAR direction NN, ND, U	JA		
documento PK, UA VARCHAR2(255)	BYTE)	VARCHAR2(: tipoDocume NN, UA, CK	FK_SUCUI	RSAL.id, NN, UA,	NC	VARCHAR nit PK, UA	ICTO.codigoBarras, SA	VARCHAR direction NN, ND, U	2(100 BYTE)		
VARCHAR2(100E documento PK, UA VARCHAR2(255 I correoElectronica PK, UA	BYTE)	VARCHAR2(: tipoDocume NN, UA, CK	FK_SUCUI	RSAL.id, NN, UA,	NC	VARCHAR nit PK, UA CLIENTE VARCHAR empresa	CTO.codigoBarras, SA 2(100BYTE)	VARCHAR direccion NN, ND, U VARCHAR document	2(100 BYTE)		

		D	ROVEEDOR							PROVEEDOR	ES PRODUCTO		
VARCHAR2/	R2(100 BYTE) VARCHAR2(100) NUMBER(*,1)		IMBER/* 1)	_			VARCHAR2(100 BYTE)		VARCHAR2(13 BYTE)				
nit nombre			lificacionCalidad		A .		proveedor		producto				
PK, UA		NN, ND, NC, UA NN, DD, CK			A		PK, FK_PROVEEDO	D sit IIA		PK, FK_PRODUCTO.codigoBarras, UA			
FR, UA	111, 110, 110, 01		, UA INI	N, DD, CK				FK, FK_FKOVEEDO	K.IIII, UA		PK, FK_PKODOCTO.COU	igobairas, OA	
				PRODUCTI	O ORDENPEDIDO								
NUMBER		NUMBER(*)	N	UMBER(*,1)	DATE			VARCHAR2(13 BYT	·E/				
pedido		cantidad			fechaAgregado			producto					
	ENPEDIDO.id, UA			C. NN. UA		NN, UA, NC		PK, FK PRODUCTO.codigoBarras, UA		HA			
TR, TR_ORD	ENT EDIDO.IU, OA	ININ, OA	Ci	, 1414, 0A	INIV, OA, IN	<u> </u>		TR, TR_TRODUCTO	.coalgobal1as,	OA			
							ORDE	NPEDIDO					
NUMBER DATE DATE		ATF	NUMBER(*	NUMBER(*,1)		VARCHAR2(255 BYTE)		VARCHAR2(100 BYTE)		NUMBER			
		fechaEntres		chaEsperadaEntrega		calificacionProveedor		estado			proveedor		idSucursal
PK, SA		UA		N, NC, UA	DD, NN, Ck			NN, CK, UA			FK_PROVEEDOR.nit, NN, NC, UA		FK SUCURSALId, NN, UA, NC
110,011		011	1	1,110,011	100,1111,01			111, 011, 011					11_000011011010711170177110
											ACTURA_PRODUC		
							NUM	BER		NUMBER(*)		VARCHAR2(13BYTE)	
							factu	ra		cantidad		producto	
								C FACTURA.nu	moro CA	NN, UA		PK, FK PRODUCTO.	andiga Parras IIA
							PK, F	C_FACTURA.nu	imero, sa	ININ, UA		PK, FK_PRODUCTO.	codigobarras, UA
							FACTU	IRΔ					
AULINADED	R VARCHAR2(2	CC DVTC\	DATE	VADCUADA/200 DVTE)	NUMBER		_			VARCHAR2(OFF DVTF)	NUMBER	
NOMBER		33 BY IE)		, ,				* * * * * * * * * * * * * * * * * * * *			255 BYTE)		
numero	direccion		fecha	nombreCajero	valorTotal	pagoExitoso	puntosCompra		idSucursal		cliente		
PK, SA				nombreedjero				oscompra					
	INN. NC. UA				NN. DD. NC	NN. NC. DD. CK		<u> </u>		FK SUCURS	AL.id. NN. UA. NC	FK CLIENTE.correo	Electronico, NN, NC, UA
	NN, NC, UA			NN, NC, UA	NN, DD, NC	NN, NC, DD, CK		<u> </u>		FK_SUCURS	AL.id, NN, UA, NC	FK_CLIENTE.correo	Electronico, NN, NC, UA
	NN, NC, UA				NN, DD, NC	, , ,		<u> </u>		FK_SUCURS	AL.id, NN, UA, NC	FK_CLIENTE.correo	Electronico, NN, NC, UA
NUMI	IBER VARCHA	R2(250 BYTE)	NN, NC, UA	NN, NC, UA	PROM_DESCUE	ENTO ATE		DD	VARCHAR2	(13 BYTE)	NUMBER	FK_CLIENTE.correo	Electronico, NN, NC, UA
id	IBER VARCHA	R2(250 BYTE) cripcion u	NN, NC, UA	NN, NC, UA NUMBER(*) unidades/vendidas	PROM_DESCUE DA fecha	ENTO ATE Unicio	NN, E	DATE fechaFin	prod	(13 BYTE) ucto	NUMBER descuento	FK_CLIENTE.correo	Electronico, NN, NC, UA
	IBER VARCHA	R2(250 BYTE)	NN, NC, UA	NN, NC, UA	PROM_DESCUE DA fecha	ENTO ATE	NN, E	DD		(13 BYTE) ucto	NUMBER	FK_CLIENTE.correo	Electronico, NN, NC, UA
id PK,N	IBER VARCHA d des	R2(250 BYTE) cripcion u	NUMBER(*) nidadesDisponibles	NN, NC, UA NUMBER(*) unidadesVendidas NN	PROM_DESCUE DA fecha NN, N	ENTO ATE Inicio IC, UA SCUENTO	NN, E	DATE fechaFin NN, NC, UA	FK (producto.coc	(13 BYTE) ucto ligoBarras) , NN	NUMBER descuento NN, NC, UA	FK_CLIENTE.correo	Electronico, NN, NC, UA
id PK,N NUMI	IBER VARCHA d des NN h	R2(250 BYTE) cripcion u IN,UA	NUMBER(*) nidadesDisponibles NN	NN, NC, UA NUMBER[*] unidadesVendidas NN NUMBER[*]	PROM_DESCUE DA fecha NN, N PROM_SEGUNIDES	ENTO ATE Inicio NC, UA SCUENTO ATE	NN, E	DATE fechaFin NN, NC, UA	Prod FK (producto.coc	(13 BYTE) ucto ligoBarras) , NN	NUMBER descuento NN, NC, UA NUMBER	FK_CLIENTE.correo	Electronico, NN, NC, UA
id PK,N	IBER VARCHA IBER VARCHA IBER VARCHA I des	R2(250 BYTE) cripcion u IN,UA	NUMBER(*) nidadesDisponibles	NN, NC, UA NUMBER[*] unidadesVendidas NN NUMBER[*]	PROM_DESCUE DA fecha NN, N PROM_SEGUNIDES DA fecha	ENTO ATE Inicio IC, UA SCUENTO	NN, E	DATE fechaFin NN, NC, UA	FK (producto.coc	(13 BYTE) ucto ligoBarras) , NN (13 BYTE) ucto	NUMBER descuento NN, NC, UA	FK_CLIENTE.correo	Electronico, NN, NC, UA
id PK,N NUMI id	IBER VARCHA IBER VARCHA IBER VARCHA I des	R2(250 BYTE) cripcion u IN,UA R2(250 BYTE) cripcion u	NUMBER(*) nidadesDisponibles NN NUMBER(*) nidadesDisponibles	NN, NC, UA NUMBER(*) unidades/vendidas NN NUMBER(*) unidades/vendidas	PROM_DESCUE DA fecha NN, N PROM_SEGUNIDES DA fecha	ENTO ATE Inlicito AC, UA SCUENTO ATE Inlicito AC, UA	NN, E	DATE fechaFin NN, NC, UA DATE fechaFin NN, NC, UA	Prod FK (producto.coc VARCHAR2 prod	(13 BYTE) ucto ligoBarras) , NN (13 BYTE) ucto	NUMBER descuento NN, NC, UA NUMBER descuento	FK_CLIENTE.correo	Electronico, NN, NC, UA
id PK,N NUMI id PK,N	BER	R2(250 BYTE) cripcion u IN,UA R2(250 BYTE) cripcion u N, UA	NN, NC, UA NUMBER(*) nidadesDisponibles NN NUMBER(*) nidadesDisponibles NN	NN, NC, UA NUMBER(*) unidades Vendidas NN NUMBER(*) unidades Vendidas NN	PROM_DESCUE DE fecha NN, h PROM_SEGUNIDES DE fecha NN, h	ENTO ATE Initicio KC, UA SCUENTO ATE Initicio KC, UA PR	NN, E	DATE fechaFin NN, NC, UA DATE fechaFin NN, NC, UA	Producto.coc VARCHARZ prod FK (producto.coc	(13 BYTE) ucto ligoBarras) , NN (13 BYTE) ucto ligoBarras) , NN	NUMBER descuento NN, NC, UA NUMBER descuento NN, NC, UA	_	
id PK,h NUMI id PK,h		R2(250 BYTE) cripcion u IN,UA R2(250 BYTE) cripcion u N, UA	NN, NC, UA NUMBER(*) nidadesDisponibles NN NUMBER(*) nidadesDisponibles NN NUMBER(*)	NN, NC, UA NUMBER(*) unidades Vendidas NN NUMBER(*) unidades Vendidas NN NUMBER(*)	PROM_DESCUI DA fecha NN, h PROM_SEGUNIDES Fecha NN, h	ENTO ATE Initio IC, UA SCUENTO ATE Initio Initio ATE FR FR FR FR FR FR FR FR FR F	NN, E	DATE fechaFin NN, NC, UA DATE fechaFin NN, NC, UA DATE fechaFin NN, NC, UA	Producto.coc VARCHAR2 prod FK (producto.coc VARCHAR2	(13 BYTE) ucto ligoBarras) , NN (13 BYTE) ucto ligoBarras) , NN (13 BYTE)	NUMBER descuento NN, NC, UA NUMBER descuento NN, NC, UA NUMBER	NUMBER	NUMBER
id PK,N NUMI id PK,N	BER	R2(250 BYTE) cripcion u IN,UA R2(250 BYTE) cripcion u N, UA	NN, NC, UA NUMBER(*) nidadesDisponibles NN NUMBER(*) nidadesDisponibles NN	NN, NC, UA NUMBER(*) unidades Vendidas NN NUMBER(*) unidades Vendidas NN NUMBER(*)	PROM_DESCUI DJA fecha NN, h PROM_SEGUNIDES DJA fecha NN, h	ENTO ATE Initicio KC, UA SCUENTO ATE Initicio KC, UA PR	NN, E	DATE fechaFin NN, NC, UA DATE fechaFin NN, NC, UA	Producto.coc VARCHARZ prod FK (producto.coc	(13 BYTE) ucto ligoBarras) , NN (13 BYTE) ucto ligoBarras) , NN	NUMBER descuento NN, NC, UA NUMBER descuento NN, NC, UA	_	
id PK,N NUM id PK,N	BER	R2(250 BYTE) uin, UA R2(250 BYTE) uripcion uin, UA R2(250 BYTE) un, UA R2(250 BYTE) uripcion uin, UA	NN, NC, UA NUMBER(*) nidadesDisponibles NN NUMBER(*) nidadesDisponibles NN NUMBER(*) nidadesDisponibles	NN, NC, UA NUMBER(*) unidadesVendidas NN NUMBER(*) unidadesVendidas NN NUMBER(*) unidadesVendidas	PROM_DESCUI DJA fecha NN, h PROM_SEGUNIDES DJA fecha NN, h	ENTO TE Inicio KC, UA ATE Inicio ATE Inicio KC, UA ATE Inicio KC, UA ATE Inicio KC, UA	NN, E	DATE fechafin NN, NC, UA DATE fechafin NN, NC, UA DATE fechafin NN, NC, UA	Prod FK (producto.coc VARCHAR2 prod FK (producto.coc VARCHAR2 prod FX (producto.coc	(13 BYTE) ucto ligoBarras) , NN (13 BYTE) ucto ligoBarras) , NN	NUMBER descuento NN, NC, UA NUMBER descuento NN, NC, UA NUMBER descuento descuento descuento	NUMBER pague	NUMBER Here
NUMI PK,N NUMI id PK,N NUMI id	IBER VARCHA d des IBER VARCHA d des IBER VARCHA NNN NN N	R2(250 BYTE) R2(250 BYTE) R2(250 BYTE) rripcion N, UA R2(250 BYTE) rripcion u R2(250 BYTE) rripcion u R2(250 BYTE)	NN, NC, UA NUMBER(*) nidadesDisponibles NN NUMBER(*) nidadesDisponibles NUMBER(*) nidadesDisponibles NUMBER(*)	NN, NC, UA NUMBER(*) unidades/vendidas NN NUMBER(*) unidades/vendidas NN NUMBER(*) unidades/vendidas NN NN NIMBER(*) unidades/vendidas NN	PROM_DESCUI DE Fecha NN, N PROM_SEGUNDES DE Fecha NN, N DE Fecha NN, N	SHIO STE	NN, E	DATE fechafin NN, NC, UA DATE fechafin NN, NC, UA DATE fechafin NN, NC, UA EVEUNIO DATE fechafin NN, NC, UA EVEUNIO DATE fechafin NN, NC, UA EVECANT	FK (producto.coc VARCHAR2 prod FK (producto.coc VARCHAR2 prod FK (producto.coc	(13 8YTE) ucto ucto iigoBarras) , NN (13 8YTE) ucto ucto iigoBarras) , NN (13 8YTE) ucto iigoBarras) , NN	NUMBER descuento NN, NC, UA NUMBER descuento NN, NC, UA NUMBER descuento NN, NC, UA	NUMBER DRIVE NEV. NC, UA	NUMBER Illere NN, NC, UA
id PK,N NUM id PK,N	VARCHA VARCHA	R2(250 BYTE) cripcion uN,UA R2(250 BYTE) cripcion N, UA R2(250 BYTE) cripcion uN,UA	NN, NC, UA NUMBER(*) nidadesDisponibles NN NUMBER(*) nidadesDisponibles NN NUMBER(*) nidadesDisponibles	NN, NC, UA NUMBER(*) unidades/vendidas NN NUMBER(*) unidades/vendidas unidades/vendidas NN NUMBER(*) NUMBER(*)	PROM_DESCUI DO fecha NN, N, N, PROM_SEGUINDESC Fecha NN, N, N	SHIO STE	NN, E	DATE fechafin NN, NC, UA DATE fechafin NN, NC, UA DATE fechafin NN, NC, UA	Prod FK (producto.coc VARCHAR2 prod FK (producto.coc VARCHAR2 prod FX (producto.coc	(13 BYTE) ucto ligoBarras) , NN	NUMBER descuento NN, NC, UA NUMBER descuento NN, NC, UA NUMBER descuento descuento descuento	NUMBER pague	lleve

Figura 1. Modelo UML SuperAndes

3.2 Creación de las tablas del modelo relacional:

El archivo de scripts se encuentra adjunto como "esquemaSuperAndes.sql" en la carpeta data del proyecto.

4 Requerimientos funcionales modificación y consulta

Se adjuntan los scripts *sql* de los requerimientos de consulta en la carpeta "data/Iteracion2/RFC".

Los requerimientos funcionales se pueden revisar en el proyecto de eclipse SuperAndes por medio de la interfaz interactiva con dos modalidades. Por un lado, si se ingresa como ADMINISTRADOR se puede manejar las sucursales, estantes, bodegas, ordenes de pedido, etc. Si se entra cómo cualquier otro usuario, se puede utilizar el carrito de compras, o en su defecto realizar compras directamente.

5 Construcción de la aplicación

5.1 Ajuste de tablas creadas en Oracle

Se modifica el archivo de la carpeta "data/Crear y poblar tablas/esquemaSuperAndes.sql" según los cambios que se generaron en el modelo conceptual y la unificación de los trabajos realizados anteriormente. Las cosas necesarias y cambios de algunos PK (sucursal) también se realizaron, se agregó un chequeo en el rango de la calificación/calidad de los proveedores, ordenes de pedido y productos. Adicionalmente, se agrega el carrito de compras junto con su clase de asociación.

5.2 Población de las tablas

Se adjunta en la carpeta "data/Crear y poblar tablas/poblarTablas.sql" un script con información de la vida real para llenar las tablas cuidadosamente seleccionada para el correcto funcionamiento de la aplicación. Esto incluye veintidós productos con variadas categorías

(once) y tipos, cuatro sucursales, y varios proveedores para las distintas categorías (y asociados con sus distintos productos).

Adicionalmente se vinculan los productos a las sucursales de acuerdo a la disponibilidad que tienen en estantes y bodegas (para cada categoría y por el peso que pueden almacenar). Así mismo, se crean 20 clientes, 10 personas naturales y 10 empresas.

Por último, se generan órdenes de pedido que cumplen el papel de surtir las diferentes sucursales, por lo que se generan como entregadas y con sus respectivos productos.

5.3 Ajustes correspondientes a los nuevos requerimientos:

Para la nueva iteración se implementó la clase carrito de compras, junto con su clase de asociación a producto ProductoCarritoCompras. El carrito de compras está asociado a un cliente, y guarda los productos que él quiere tomar del estante para después comprar, o dejarlos abandonados. El carrito solamente posee un identificador que permite la correcta asociación a la base de datos y la llave primaria de la sucursal a la que pertenece para poder realizar las transacciones que se describen más adelante. La clase de asociación tiene el id del carrito y el identificador del producto junto con la cantidad de unidades de este.

Para satisfacer las condiciones ACID, hacemos que el carrito persista cada vez que un cliente desea tomar un producto, por esta razón, la transacción inicia sacando los productos de un estante en la sucursal a la que pertenece el carrito, luego, se agregan a la clase productoCarrito y se realiza el commit. De esta manera, los otros clientes no podrán tomar los productos en el carrito de otros clientes, y no se generan unidades extra de productos que no deberían aparecer.

Por otro lado, en cuanto a la búsqueda de carritos abandonados se utilizó la herramienta ScheduledExecutorService que nos permite cada N tiempo realizar un procedimiento. Así, definimos que para que se viera en la aplicación, los carritos abandonados por un cliente (es decir, con un cliente nulo) se recogen cada 6 minutos, realizando las transacciones para devolver los productos a sus estantes y eliminando el carrito de la base de datos. Si en el periodo intermedio el cliente vuelve y no se ha realizado este proceso, por medio de un update puede recuperar el carrito con sus productos.

La venta de la iteración pasada no cambia, pero para la venta de los productos del carrito, cada producto del carrito se pasa a la factura, para así poder realizar el proceso de venta y en ese momento, si se realiza la verificación de inventarios trayendo los productos necesarios al estante de Bodega y si se viola el nivel de abastecimiento, se genera la orden de pedido a su respectivo proveedor.

Por último, en cuanto a las órdenes de pedido, desde la iteración pasada se tenía que una orden de pedido podría tener varios productos del mismo proveedor, por lo cual no se cambió nada en el modelaje para satisfacer este requerimiento. Sin embargo, ahora los pedidos se realizan de manera automática (y manual también como antes).

5.4 Pruebas

Se inicia el desarrollo del demo, se generan todas las transacciones necesarias para implementar CRUD, sin embargo, no se realiza la conexión entre la interfaz y estos métodos. Finalmente, se adjunta en la carpeta "data/ Iteracion2" el Excel con la documentación de las pruebas a realizar para cada requerimiento, casos de terminación exitosa, fallidos y las respuestas esperadas.