

Iteración 2

Mario Santiago Andrade Vargas, Andrés Felipe Rodríguez Murillo

Universidad de los Andes, Bogotá, Colombia

ms.andrade@uniandes.edu.co

af.rodriguezm1@uniandes.edu.co

Tabla de contenido

1	Análisis	1
2	Diseño de la aplicación	2
5	Requerimientos funcionales de consulta.....	6
6	Bibliografía	9

1 Análisis

Si lo requiere, ajuste el modelo del mundo (modelo conceptual: diagrama de clases UML y su respectivo modelo relacional) propuesto en la iteración 1 que representó el caso de estudio. Indique cuáles clases del modelo del mundo fueron actualizadas o creadas en esta iteración. **(Guía de la iteración)**

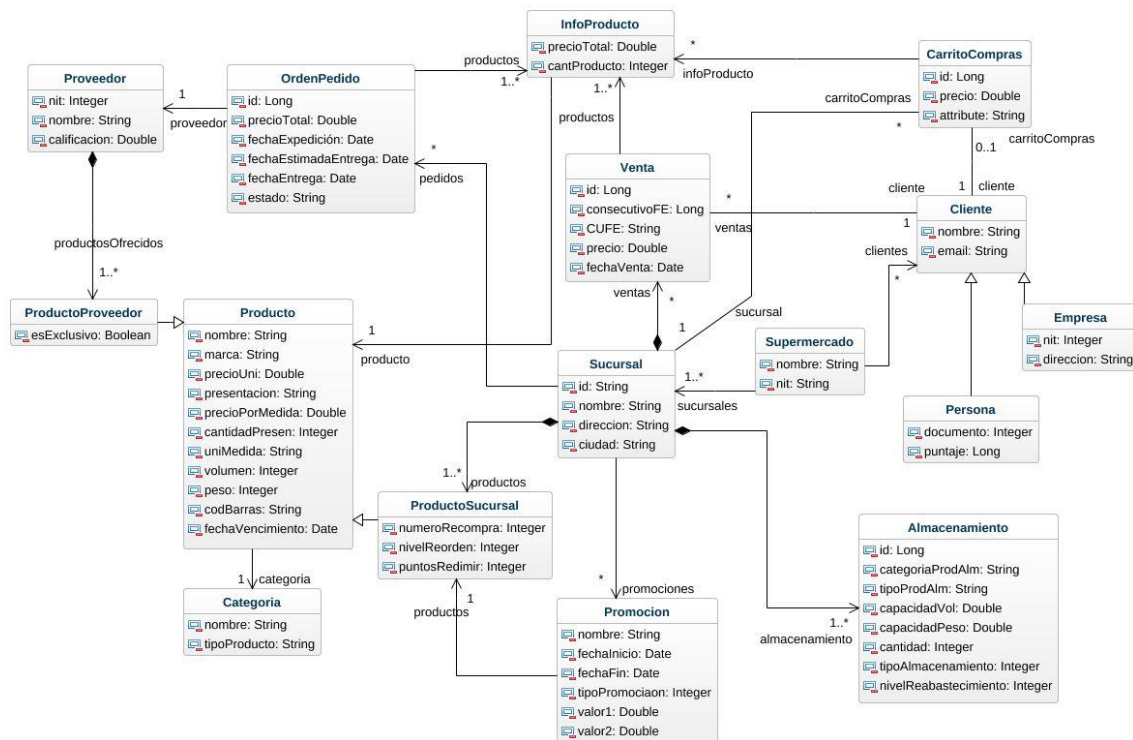


Imagen 1. Modelo conceptual de Iteración 2.

Para la iteración 2, se nos pidió agregar el uso de un carrito de compras, para esto nosotros pensamos en que el carrito de compras sea una instancia momentánea de compra, que una vez el cliente desee confirmarla, se guarda como compra.

1.A Clases modificadas

1.A.1 Clase CarroCompras

Esta clase representa un carro de compras de un cliente, este cuenta con una id para reconocerlo, un cliente y una sucursal. Además, tiene una lista de información de producto, con la cual se sabe el código del producto y la cantidad que lleva del producto. Un cliente puede tener un solo carro de compra máximo, es por esto la cardinalidad de 0 a 1 en la asociación cliente-carro.

1.A.2 Clase Resolucion

Esta clase fue eliminada del modelo debido a que esta no nos ofrecía un gran aporte a la solución del problema que se planteó en la iteración 1.

1.A.3 Tabla Supermercado

Esta tabla recibió un cambio en sus columnas, pues al decidir eliminar la tabla de resoluciones, es necesario eliminar el atributo de *FacturaE*, que nos permitía saber si la empresa estaba o no facturando electrónicamente.

1.A.4 Tabla Ventas y OrdenPedido

Tanto la tabla Ventas como OrdenPedido se les agregó un nuevo atributo que es el atributo del valor de la venta u orden, esto para facilitar el acceso a esta información, la cual en la iteración 1 no la teníamos.

2 Diseño de la aplicación

A partir del diseño existente, analice el impacto que representa la introducción de los nuevos requerimientos y restricciones a nivel del modelo conceptual. Realice los cambios necesarios en su modelo relacional para respetar las reglas de negocio y asegurar la calidad del mismo. Tenga en cuenta los comentarios recibidos en la sustentación de la iteración anterior. Documente el diseño y las decisiones tomadas para crear los elementos de la base de datos que da el respaldo de persistencia a la aplicación, a partir del modelo conceptual. Incluya un listado con las tablas generadas en la base de datos, utilizando los estándares establecidos y buenas prácticas ilustrados en Parranderos-JDO. Este listado, resultado de una consulta SQL, debe incluir el nombre de la tabla, el nombre y el tipo de dato de sus campos, así como los nombres de restricciones de llaves primarias, llaves foráneas y de chequeo (**Guía de la iteración**).

2.A Impacto de los cambios

En la solución propuesta en la iteración 1, nuestro modelo implementaba una clase llamada *InfoProducto* que nos permitía tanto hacer ventas con varios productos, como hacer ordenes de pedido con varios productos. Debido a esta clase, la nueva implementación no nos trae gran impacto, pues es solo crear una clase nueva, llamada *carroCompras*, en la cual manejaremos el

cliente del carrito, la sucursal a la que pertenece el carrito, y su estado junto con un arreglo de *InfoProducto*.

2.A Modelo de datos relacional

2.A.1 Tabla Sucursal

Id	Nombre	Dirección	Ciudad
PK, SA	NN	NN	NN
1	Las Aguas	Calle 2 #2-22	Bogotá
2	Los Llanos	Calle 3 #3-33	Medellín

Tabla 18. Tabla sucursal

2.A.2 Tabla Proveedor

NIT	Nombre	Calificación
PK	PK	
60455642-7	Éxito	5
24585268-8	ElectroAlpes	4.3

Tabla 19. Tabla Proveedor

2.A.3 Tabla Categoría

Id	Nombre
PK, SA	PK
1	Perecedero
2	Aseo

Tabla 20. Tabla Categoría

2.A.4 Tabla Tipo Producto

Id	Id_categoria	Nombre
PK,SA	PK, FK _{Categoría.Id}	ND, NN
1	Perecedero	Carne
2	Aseo	Escoba

Tabla 21. Tabla Tipo Producto

2.A.5 Tabla Producto

Codigo_barras	Id_categoria	Id_tipo_producto	Nombre	Marca	Presentación	Cantidad_presen	Uni_medida	Volumen	Peso
PK	FK _{Categoría.Id} , NN	FK _{Tipo_Producto.Id} , NN	NN	NN	NN	NN	NN	NN	NN
509546052892	1	1	Carne de res	Carnitas	Filetes de carne	100	Gr	10	100
509546052893	2	2	Escoba larga	P&G	Escoba sola	1	Gr	30	50

Tabla 22. Tabla Producto

2.A.6 Tabla Producto Proveedor

Codigo_Barras	Nit_proveedor	Es_exclusivo	Precio_Unitari	Precio_unidad_medida
PK, FK _{Producto.codigo_barras}	PK, FK _{Proveedor.Nit}	NN	NN	NN
509546052892	60455642-7	0	10000	5000
509546052893	60455642-7	1	2000	2000

Tabla 23. Tabla Producto Proveedor

2.A.7 Tabla Producto Sucursal

Codigo_Barras	Id_sucursal	Precio_Unitari	Precio_unidad_medida	Numero_recompra	Nivel_reorden
PK, FKProducto.codigo_barras	PK, FKSucursal.Id	NN	NN	NN	NN
509546052892	1	20000	5000	50	30
509546052893	2	2000	2000	30	50

Tabla 24. Tabla Producto Sucursal

2.A.8 Tabla Productos Redimibles

Codigo_Barras	Id_sucursal	Puntos
PK, FKProducto.codigo_barras	PK, FKSucursal.Id	NN
509546052892	1	3000
509546052893	1	5000

Tabla 25. Tabla Productos Redimibles.

2.A.9 Tabla Promoción

Id	Id_sucursal	Codigo_barras	Nombre	Fecha_inicio	Fecha_Fin	Tipo_Prom	Valor1	Valor2
PK,SA	PK, FKSucursal.Id	NN	NN	NN	NN	NN		
1	1	509546052892	Super prom	20/02/2018	31/12/2018	1	5	3
2	1	509546052893	Lleve más	31/03/2018	05/05/2019	2	0	0

Tabla 26. Tabla Promoción

2.A.10 Tabla Orden Pedido

Id	Id_sucursal	Nit_Prov	Fecha_Exp	Fecha_Est	Fecha_entrega	Estado
PK, SA	PK, FKSucursal.Id	PK, FKProveedor.Nit	NN	NN		NN
1	1	60455642-7	30/02/2018	01/03/2018		Espera
2	1	60455642-7	31/03/2018	05/04/2018		Espera

Tabla 27. Tabla Orden Pedido

2.A.11 Tabla Info Producto Proveedor

Id_orden	Codigo_barras	Cantida_producto
PK, FKOrden_pedido.id	PK, FKProducto_proveedor.codigo_barras	NN
1	509546052892	3
1	509546052893	2

Tabla 28. Tabla Info Producto Proveedor

2.A.12 Tabla Almacenamiento

Id	Id_sucursal	Id_categoria_producto	Id_tipo_producto	Capa_volumen	Capa_peso	Cantidad	Tipo_almacenamiento	Nivel_reactividad
PK, SA	PK, FKSucursal.Id	PK, FKCategoria.Id	PK, FKTipoProducto.Id					
1	1	2	1	30	500	30	1	5
2	1	2	1	50	300	40	2	30

Tabla 29. Tabla Almacenamiento

2.A.13 Tabla Clientes

Email	Nombre
PK	NN
Ejemplo@ejemplo.com	Ejemplo
Ejemplo2@ejemplo2.com	Ejemplo2

Tabla 30. Tabla cliente

2.A.14 Tabla Personas

Email	Documento	Puntaje
PK, FK _{Cliente.email}	PK	NN
Ejemplo@ejemplo.com	24585268	0
Ejemplo2@ejemplo2.com	123456789	5000

Tabla 31. Tabla Persona

2.A.15 Tabla Empresas

Email	Nit	Dirección
PK, FK _{Cliente.email}	PK	NN
otroEjemplo@uniandes.edu.co	13456789-9	Calle 5N #2 - 2
jajajEjemplo@uniandes.edu.co	789456123-9	Carrera 62 - 5

Tabla 32. Tabla Empresa

2.A.16 Tabla Ventas

Id	Id_sucursal	Email_cliente	Consecutivo_FE	CUFE	Fecha_Venta
PK, SA	PK, FK _{Sucursal.Id}	PK, FK _{Clientes.email}			NN
1	1	ejemplo@ejemplo.com			30/09/2018
2	1	Ejemplo2@ejemplo2.com	1	K2nk3wjn...	31/12/2018

Tabla 33. Tabla ventas

2.A.17 Tabla Info Producto Sucursal

Id_venta	Codigo_barras	Cantida_producto
PK, FK _{ventas_pedido.id}	PK, FK _{Producto_sucursal.codigo_Barras}	NN
1	509546052892	3
2	509546052893	2

Tabla 34. Tabla Info Producto Sucursal

2.A.19 Tabla Información

Nit	Nombre
NN	NN
123456789-9	SuperAndes

Tabla 36. Tabla Información

Esta tabla recibió unas modificaciones desde la anterior iteración, eliminamos el campo de facturación electrónica para simplificar las cosas

2.A.20 Tabla Carrito

Id	Email_cliente	Id_Sucursal	Precio	Estado
PK, SA	PK, FK _{Cliente.email}	PK, FK _{Sucursales.id}	NN	NN
1	ejemplo@ejemplo	1	5000	Activo
2	Ejemplo2@ejemplo	3	50000	Abandonado

Tabla 37. Tabla Carrito.

Esta tabla fue agregada al modelo, esta tabla maneja lo que es el carrito de un cliente mientras culmina sus compras.

2.A.21 Tabla InfoProductoCarrito

Id_carrito	Cod_barras	Cantidad	Precio_total
PK, FK _{carrito.id}	PK, FK _{productoSucursal.codigoBarras}	NN	NN
1	136516165563	2	10000

2	321365146213	50	10000
---	--------------	----	-------

Tabla 38. Tabla InfoProductoCarrito.

Esta tabla fue agregada en esta iteración, con el fin de manejar los productos dentro de un carrito de compras.

2.B Listado de tablas.

Incluya un listado ISIS 2304 – Sistemas Transaccionales con las tablas generadas en la base de datos, utilizando los estándares establecidos y buenas prácticas ilustrados en Parranderos-JDO. Este listado, resultado de una consulta SQL, debe incluir el nombre de la tabla, el nombre y el tipo de dato de sus campos, así como los nombres de restricciones de llaves primarias, llaves foráneas y de chequeo.(Guía de iteración).

```

1. SELECT A.table_name AS Tabla,
2. A.column_name AS Nom_Atributo,
3. B.data_type AS Tipo_Dato,
4. A.constraint_name AS Nom_Constra
5. FROM all_cons_columns A
6. INNER JOIN all_tab_columns B
7. ON A.column_name = B.column_name
8. AND A.table_name = B.table_name
9. WHERE A.OWNER = 'ISIS2304A261820'
10. AND A.table_name NOT LIKE 'A_%'
11. AND A.table_name NOT LIKE '%$%'
12. ORDER BY A.table_name,
13. A.column_name;
```

3 Requerimientos funcionales de consulta

3.A.1 MOSTRAR EL DINERO RECOLECTADO POR VENTAS EN CADA SUCURSAL DURANTE UN PERIODO DE TIEMPO Y EN EL AÑO CORRIDO

```

1. SELECT ID_SUCURSAL,
2. SUM(PRECIO) AS ACUMULADO
3. FROM (
4. SELECT d.id_sucursal,
5. d.precio_unitario*c.cantidad_producto as precio
6. FROM info_producto_sucursal C
7. inner join (SELECT a.codigo_barras,
8. a.id_sucursal,
9. a.precio_unitario,
10. b.id
11. FROM PRODUCTO_SUCURSAL A
12. INNER JOIN VENTAS B
13. ON a.id_sucursal = b.id_sucursal
14. WHERE b.fecha_venta BETWEEN TO_DATE('01/01/2018', 'DD/MM/YYYY') AND
TO_DATE('31/12/2018', 'DD/MM/YYYY')) D
15. on c.codigo_barras = d.codigo_barras
16. and c.id_venta = d.id)
17. GROUP BY ID_SUCURSAL;
```

3.A.3 MOSTRAR EL ÍNDICE DE OCUPACIÓN DE CADA UNA DE BODEGAS Y ESTANTES DE UNA SUCURSAL

```

1. SELECT ID_SUCURSAL,
```

```

2.     CANTIDAD,
3.     CAPA_VOL AS CAPACIDAD_VOLUMEN,
4.     CAPA_PESO AS CAPACIDAD_PESO
5. FROM ALMACENAMIENTO;

```

3.A.4 MOSTRAR LOS PRODUCTOS QUE CUMPLEN CON CIERTA CARACTERÍSTICA

```

1. SELECT *
2. FROM PRODUCTO
3. WHERE ID_CATEGORIA = 1;
4.
5. SELECT *
6. FROM PRODUCTO
7. WHERE ID_TIPO_PRODUCTO = 1;
8.
9. SELECT *
10. FROM PRODUCTO
11. WHERE MARCA LIKE '%a%';
12.
13. SELECT *
14. FROM PRODUCTO
15. WHERE Nombre LIKE '%a%';
16.
17. SELECT *
18. FROM PRODUCTO
19. WHERE volumen >= 50;
20.
21. SELECT *
22. FROM PRODUCTO
23. WHERE Peso >= 50;

```

3.A.5 MOSTRAR LAS COMPRAS HECHAS POR SUPERANDES A LOS PROVEEDORES

```

1. SELECT A.ID_SUCURSAL,
2.     b.email,
3.     b.direccion,
4.     a.fecha_exp,
5.     a.fecha_Est,
6.     a.fecha_entrega
7. FROM ORDEN_PEDIDO A
8.     INNER JOIN EMPRESAS B
9.     ON a.nit_proveedor = b.nit;

```

3.A.6 MOSTRAR LAS VENTAS DE SUPERANDES A UN USUARIO DADO, EN UN RANGO DE FECHAS INDICADO

```

1. SELECT *
2. FROM VENTAS
3. WHERE EMAIL_CLIENTE = 'doña@blanca.com'
4.     AND fecha_venta BETWEEN TO_DATE('01/01/2018', 'DD/MM/YYYY') AND TO_DATE('31
    /12/2018', 'DD/MM/YYYY');

```

3.A.7 ANALIZAR LA OPERACIÓN DE SUPERANDES

```

1. SELECT X.ID_SUCURSAL, x.PRECIO_TOT, ADD_MONTHS((SELECT MIN(FECHA_VENTA) FROM VENTAS), A
) AS FECHA FROM (
2. SELECT ID_SUCURSAL, MAX(PRES) as PRECIO_TOT FROM(
3. SELECT ID_SUCURSAL, SUM(PRECIO) As Pres, A
4. FROM(
5. SELECT d.id_sucursal,
6. d.precio_unitario*c.cantidad_producto as precio,
7. FLOOR((D.FECHA_VENTA - (SELECT MIN(FECHA_VENTA) FROM VENTAS)))/(7*4)) AS A
8. FROM info_producto_sucursal C
9. inner join (SELECT a.codigo_barras,
10. a.id_sucursal,
11. a.precio_unitario,
12. b.id,
13. B.FECHA_VENTA
14. FROM PRODUCTO_SUCURSAL A
15. INNER JOIN VENTAS B
16. ON a.id_sucursal = b.id_sucursal) D
17. on c.codigo_barras = d.codigo_barras
18. and c.id_venta = d.id)
19. GROUP BY ID_SUCURSAL, A)
20. GROUP BY ID_SUCURSAL) X INNER JOIN (SELECT ID_SUCURSAL, SUM(PRECIO) As Pres, A
21. FROM(
22. SELECT d.id_sucursal,
23. d.precio_unitario*c.cantidad_producto as precio,
24. FLOOR((D.FECHA_VENTA - (SELECT MIN(FECHA_VENTA) FROM VENTAS)))/(7*4)) AS A
25. FROM info_producto_sucursal C
26. inner join (SELECT a.codigo_barras,
27. a.id_sucursal,
28. a.precio_unitario,
29. b.id,
30. B.FECHA_VENTA
31. FROM PRODUCTO_SUCURSAL A
32. INNER JOIN VENTAS B
33. ON a.id_sucursal = b.id_sucursal) D
34. on c.codigo_barras = d.codigo_barras
35. and c.id_venta = d.id)
36. GROUP BY ID_SUCURSAL, A) Y ON x.PRECIO_TOT = y.pres AND x.id_sucursal = Y.ID_SUCURSAL;
37.
38. SELECT X.ID_SUCURSAL, x.PRECIO_TOT, ADD_MONTHS((SELECT MIN(FECHA_VENTA) FROM VENTAS), A
) AS FECHA FROM (
39. SELECT ID_SUCURSAL, MIN(PRES) as PRECIO_TOT FROM(
40. SELECT ID_SUCURSAL, SUM(PRECIO) As Pres, A
41. FROM(
42. SELECT d.id_sucursal,
43. d.precio_unitario*c.cantidad_producto as precio,
44. FLOOR((D.FECHA_VENTA - (SELECT MIN(FECHA_VENTA) FROM VENTAS)))/(7*4)) AS A
45. FROM info_producto_sucursal C
46. inner join (SELECT a.codigo_barras,
47. a.id_sucursal,
48. a.precio_unitario,
49. b.id,
50. B.FECHA_VENTA
51. FROM PRODUCTO_SUCURSAL A
52. INNER JOIN VENTAS B
53. ON a.id_sucursal = b.id_sucursal) D
54. on c.codigo_barras = d.codigo_barras
55. and c.id_venta = d.id)
56. GROUP BY ID_SUCURSAL, A)
57. GROUP BY ID_SUCURSAL) X INNER JOIN (SELECT ID_SUCURSAL, SUM(PRECIO) As Pres, A
58. FROM(
59. SELECT d.id_sucursal,
60. d.precio_unitario*c.cantidad_producto as precio,
61. FLOOR((D.FECHA_VENTA - (SELECT MIN(FECHA_VENTA) FROM VENTAS)))/(7*4)) AS A

```



```

62. FROM info_producto_sucursal C
63. INNER JOIN (SELECT a.codigo_barras,
64.                  a.id_sucursal,
65.                  a.precio_unitario,
66.                  b.id,
67.                  B.FECHA_VENTA
68.              FROM PRODUCTO_SUCURSAL A
69.              INNER JOIN VENTAS B
70.              ON a.id_sucursal = b.id_sucursal) D
71.      ON c.codigo_barras = d.codigo_barras
72.      AND c.id_venta = d.id)
73. GROUP BY ID_SUCURSAL, A) Y ON x.PRECIO_TOT = y.pres AND x.id_sucursal = Y.ID_SUCURSAL;

```

3.A.8 ENCONTRAR LOS CLIENTES FRECUENTES

```

1. SELECT nombre,
2.      output,
3.      numcompras
4. FROM (SELECT nombre,
5.              numcompras,
6.              Nvl(Substr(fecha, 0, Instr(fecha, '-') + 4), fecha) AS output
7.      FROM (SELECT Count(email) AS numCompras,
8.              nombre,
9.              To_char(fecha_venta, 'MM-YYYY') AS fecha
10.     FROM clientes
11.     INNER JOIN ventas
12.     ON clientes.email = ventas.email_cliente
13.     GROUP BY email,
14.              nombre,
15.              fecha_venta))
16. WHERE numcompras >= 2
17. GROUP BY nombre,
18.      output,
19.      numcompras;

```

4 Bibliografía

4.A Guía de entrega de iteración 2