

“Informe Iteracion 2. Sistemas transaccionales”

Josealejandro. Barbosa Jacome, Nicolas. Hernandez Siachoque

Informe de iteracion 2

Universidad de los Andes, Bogotá, Colombia

{j.barbosaj, n.hernandezs}@uniandes.edu.co

Fecha de presentación: Noviembre 04 de 2018

Tabla de contenido

1	Análisis del modelo	1
2	Diseño de la aplicacion	1
3	Construccion de la aplicación.....	2
4	Modelos solicitados.	2
5	Resultados logrados.	4
6	Resultados no logrados.	5
7	Balance del plan de pruebas.	5
8	Supestos adicionales sobre las reglas de negocio.....	5
9	Bibliografía.....	5

1 Análisis del modelo

Realizamos ajustes al modelo de Nicolás Hernández ya que se realizo la conformacion de un nuevo grupo. Las clases que poseian relacion con producto como `ProductoSucursal`, `ProductoPedido`, `VentaProducto`, `EstanteProducto` y `BodegaProducto` se le añadieron los atributos de cantidad con el fin de permitir tener este valor en la relacion. Se añadió la clase `Carrito` con el fin de manejar la relacion entre los clientes y los productos que van llevando al hacer sus compras. Se eliminaron relaciones redundantes como `LlegadaPedido` a `Sucursal` ya que existia la relacion de `Pedio` a `Sucursal`. Se cambia el diseño de los clientes, ahora `PersonaNatural` y `PersonaJuridica` heredan de `Cliente`. Se eliminan relaciones reduntantes de categorias.

2 Diseño de la aplicacion

Como se presenta en el literal anterior, realizamos diversos cambios respecto al analisis del modelo. Estos cambios fueron implantados al modelo relacional y se añadieron las respectivas restricciones de chequeo, llaves foraneas y nulidad.

El listado de las tablas con sus respectivos atributos se puede encontrar en el script sql `ESTADO_TABLAS.sql` en la carpeta de requerimientos funcionales

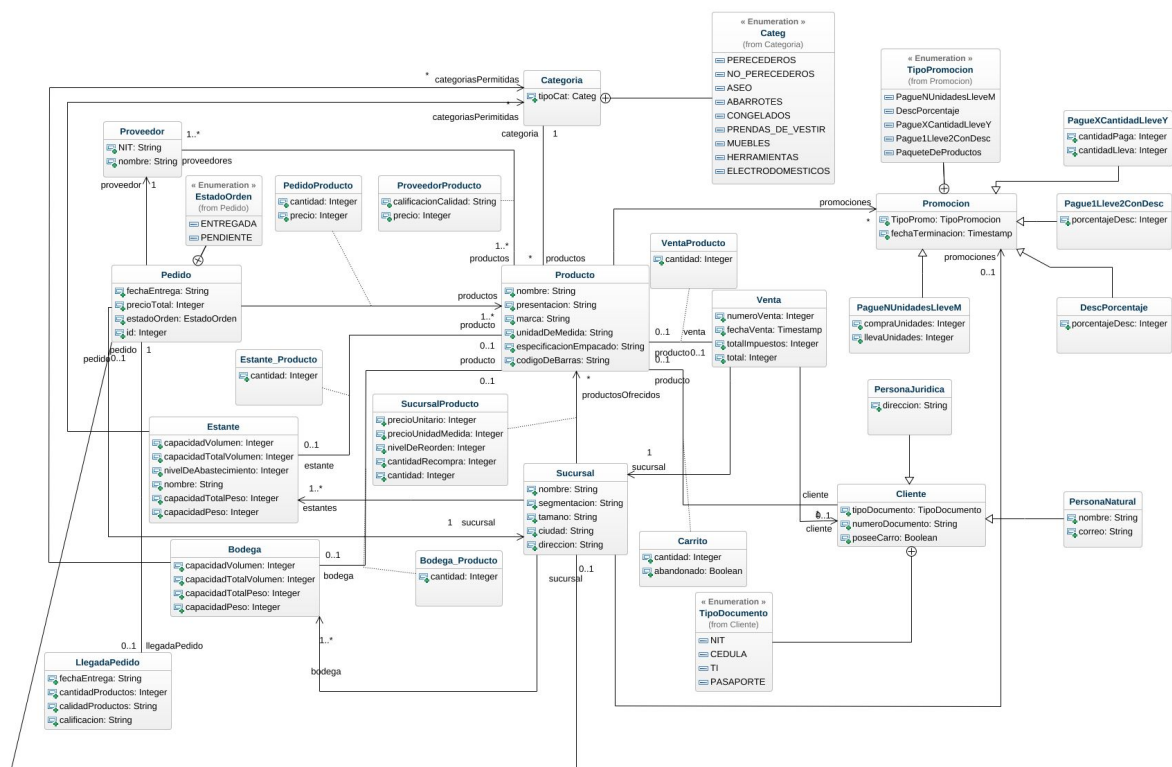
El modelo se encuentra en BCNF y no presenta ninguna anomalia de insercion, borrado o actualizacion.

La logica empleada para el carrito de compras identifica al `Carrito` como la relacion entre un `Cliente` y un `Producto`. Para que un cliente pueda usar un carrito debe solicitarlo antes. Al solicitar un carrito, su atributo de `poseeCarro` cambia de 0 a 1, indicando que el cliente ya puede realizar operaciones con el carro. Añadir carro disminuye de su estante la cantidad

indicada y agrega estos productos a `Carrito`, devolver productos realiza la operación contraria. Un cliente puede abandonar su carrito y esto implica que todos los productos de este cliente en el carrito quedan marcados como `abandonado=1`. Al recoger los productos abandonados, se actualiza la cantidad en el estante y posteriormente se eliminan los productos abandonados de la tabla `Carrito`. Si el cliente desea pagar puede llevar su carrito y automáticamente se registra su venta. Finalmente el cliente termina sin carro.

3 Construcción de la aplicación.

Se implementaron todos los requerimientos, se puede comprobar su funcionalidad con los scripts sql y la aplicación en java. Para realizar pruebas se pueden tomar los valores especificados en el documento de excel `pruebas.xlsx`.



ESQUEMA DE LA BASE DE DATOS

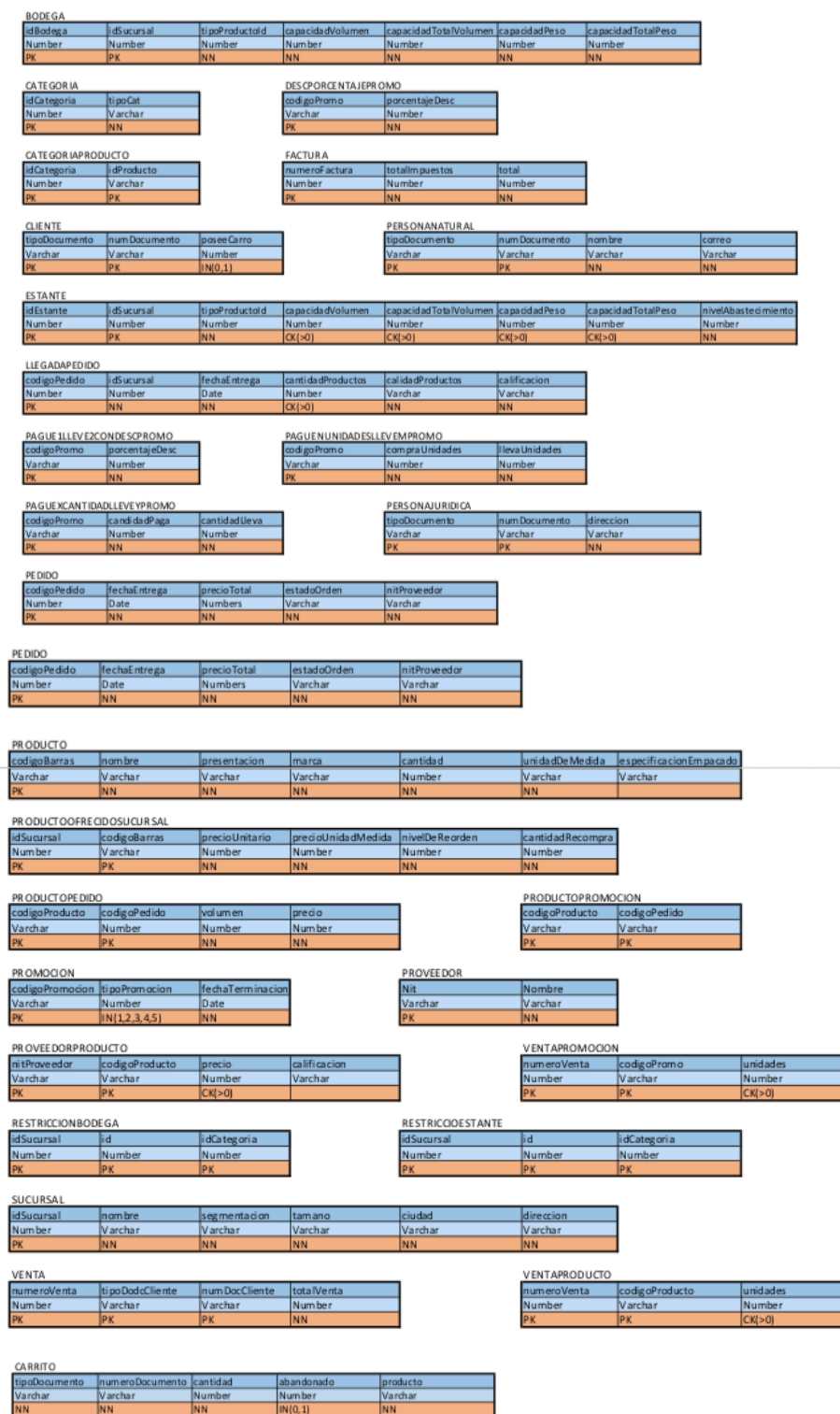


Figura 3. Esquema de la base de datos

5 Resultados logrados.

Logramos realizar el modelo a base de datos de un supermercado, cumplimos con los requerimientos funcionales y no funcionales. La aplicación es transaccional ya que todas las

operaciones se completan exclusivamente cuando se realiza commit, de lo contrario se invalida la accion.

6 Resultados no logrados.

No pudimos realizar los requerimientos que estaban catalogados como bono

7 Balance del plan de pruebas.

Se realizaron las pruebas de insercion desde Java. El resto en sql developer debido a las limitaciones de la aplicación. Hay unas pruebas que requieren el poblado de la base de datos. Estan especificadas en el nombre del metodo.

8 Supestos adicionales sobre las reglas de negocio.

Asumimos que hay un numero ilimitado de carritos en cada sucursal ya que permite añadir carritos a los clientes sin ninguna restriccion ni limite. Asumimos que la promocion de paquete de productos tiene un nuevo codigo de barras.

9 Bibliografía

1. *www.xml.org*. [En línea] [Citado el: 28 de Abril de 2010.] <http://www.xml.org>.
2. The Institution of Engineering and Technology. *A Guide to Technical Report Writing*. [En línea] www.theiet.org/students/resources/technicalreport.cfm.
3. *Universidad de los Andes*. [En línea] [Citado el: 28 de Abril de 2010.] <http://uniandes.edu.co>.
4. IEEE. *Manual de estilo de documentos técnicos*. [En línea] [Citado el: 28 de Abril de 2010.] http://standards.ieee.org/guides/style/2009_Style_Manual.pdf.
5. LNCS Springer Verlag. *Lecture Notes in Computer Science*. [En línea] [Citado el: 28 de Abril de 2010.] <http://www.springer.com/computer/lncs?SGWID=0-164-12-73062-0>.
6. Universidad de los Andes. *CARTILLA DE CITAS: Pautas para citar textos y hacer listas de referencias*. [En línea] Universidad de los Andes. [Citado el: 28 de Abril de 2010.] http://decanaturadeestudiantes.uniandes.edu.co/Documentos/Cartilla_de_citas.pdf.