

“Iteración 1 – SuperAndes”

Miguel Hoyos, Juan S. Vaca Tibocha
Universidad de los Andes, Bogotá, Colombia
{ma.hoyosr, js.vacat}@uniandes.edu.co
Fecha de presentación: Octubre 3 de 2018

Tabla de contenido

1 Punto 1	1
2 Punto 2	1
3 Punto 3	1
3.1 Casos de uso	1
3.2 Modelo conceptual propuesto.....	8
4 Punto 4	10
4.1 Modelo Relacional.....	10
4.2 Diseño y las decisiones tomadas para crear los elementos de la base de datos	10
4.3 Crear tablas.....	10
5 CONSTRUCCION DE LA APLICACION, EJECUCION DE PRUEBAS Y ANALISIS DE RESULTADOS	11
5.1 Carga de datos:	11
5.2 Balance de resultados logrados y no logrados.....	11

1 Punto 1

Para este punto, se analizó el caso de estudio entregado y con este se identificaron varios factores a tener en cuenta.

En los roles de usuario, se identificaron tres tipos de usuarios distintos. Por una parte se encuentra el cliente, quien podrá hacer compras siempre y cuando cumpla ciertos requisitos los cuales se hablarán más adelante. Por otra parte, se identificó un usuario operador, quien ofrece los servicios de Superandes y los Proveedores que se encargan de suplir las sucursales.

En adición se identificaron bastantes reglas de negocios que se deben implementar en el código para poder llevar a cabo los requerimientos funcionales del proyecto y que son indispensables para un perfecto desarrollo de los protocolos que tiene que realizar el sistema Superandes.

2 Punto 2

Realización del presente documento.

3 Punto 3

A continuación, se realizara un análisis y modelo conceptual acerca del enunciado de la iteración.

3.1 Casos de uso

Nombre	RF1.
--------	------

Resumen	Registrar Proveedores
Entradas	
Las entradas de este método son todos los atributos necesarios que tiene un proveedor para poder estar registrado en la base de datos.	
Resultados	
Se agregó un nuevo proveedor a la base de datos de superAndes.	
RNF asociados	
Persistencia	

TABLA 1 – RF1

Nombre	RF2
Resumen	Registrar Productos
Entradas	
Las entradas de este método son todos los atributos necesarios que tiene un producto para poder estar registrado en la base de datos.	
Resultados	
Se agregó un nuevo producto a la base de datos de superAndes.	
RNF asociados	
Persistencia	

TABLA 2 – RF2

Nombre	RF3
Resumen	Registrar un cliente
Entradas	
Las entradas de este método son todos los atributos necesarios que tiene un cliente para poder estar registrado en la base de datos.	
Resultados	
Se agregó un nuevo cliente a la base de datos de superAndes.	

RNF asociados
Persistencia

TABLA 3 – RF3

Nombre	RF4
Resumen	Registrar una sucursal
Entradas	
Las entradas de este método son todos los atributos necesarios que tiene una sucursal para poder estar registrado en la base de datos.	
Resultados	
Se agregó una nueva sucursal a la base de datos de superAndes.	
RNF asociados	
Persistencia	

TABLA 4 – RF4

Nombre	RF5
Resumen	Registrar una bodega a una sucursal
Entradas	
Las entradas de este método son todos los atributos necesarios que tiene una bodega para poder estar registrado en la base de datos.	
Resultados	
Se agregó una nueva bodega a la base de datos de superAndes.	
RNF asociados	
Persistencia	

TABLA 5 – RF5

Nombre	RF6
Resumen	Registrar un estante a una sucursal
Entradas	
Las entradas de este método son todos los atributos necesarios que tiene un estante para poder estar registrado en la base	

de datos.
Resultados
Se agregó un nuevo estante a la base de datos de superAndes.
RNF asociados
Persistencia

TABLA 6 – RF6

Nombre	RF7
Resumen	Registrar una promocion
Entradas	
Las entradas de este método son todos los atributos necesarios que tiene una promocion para poder estar registrado en la base de datos.	
Resultados	
Se agregó una nueva promocion a la base de datos de superAndes.	
RNF asociados	
Persistencia	

TABLA 7 – RF7

Nombre	RF8
Resumen	Finalizar una promoción
Entradas	
Las entradas de este método son todos los atributos necesarios que tienen que ser modificados de una promoción activa para poder finalizar la promoción en la base de datos.	
Resultados	
Se eliminó o finalizo una promoción existente de la base de datos de superAndes.	
RNF asociados	
Persistencia	

TABLA 8 – RF8

Nombre	RF9
Resumen	Registrar un pedido de un producto a una sucursal
Entradas	
Las entradas de este método son todos los atributos necesarios que tiene un pedido para poder estar registrado en la base de datos hacia una sucursal.	
Resultados	
Se agregó un nuevo pedido de un producto de una sucursal a la base de datos de superAndes.	
RNF asociados	
Persistencia	

TABLA 9 – RF9

Nombre	RF10
Resumen	Registrar la llegada de un pedido de un producto a una sucursal
Entradas	
Las entradas de este método son todos los atributos necesarios que hay que modificar en una orden para poder decir que ya llegó y que quede registrada correctamente en la base de datos.	
Resultados	
Se registró la llegada a una sucursal de un pedido hecho	
RNF asociados	
Persistencia	

TABLA 10 – RF10

Nombre	RF11
Resumen	Registrar una venta de un producto a una sucursal
Entradas	
Las entradas de este método son todos los atributos necesarios para registrar una compra por parte de un cliente y los que se deben modificar para registrar la actualización de los inventarios en la base de datos.	
Resultados	
Se agregó una nueva venta de la sucursal o una compra de un cliente a la base de datos superAndes.	
RNF asociados	
Persistencia	

TABLA 11 – RF11

Nombre	RFC1
Resumen	MOSTRAR EL DINERO RECOLECTADO POR VENTAS EN CADA SUCURSAL DURANTE UN PERIODO DE TIEMPO Y EN EL AÑO CORRIDO
Entradas	
Se necesitan saber los atributos que determinan el periodo del tiempo y el año corrido.	
Resultados	

Se Mostró el dinero recolectado por ventas en cada sucursal durante un periodo de tiempo y el año corrido
RNF asociados
Persistencia

TABLA 12 – RFC1

Nombre	RFC2
Resumen	Mostrar las 20 promociones más populares
Entradas	
Se necesitan saber las promociones que ha tenido y tiene el sistema.	
Resultados	
Se mostraron las 20 promociones mas populares	
RNF asociados	
Persistencia	

TABLA 13 – RFC2

Nombre	RFC3
Resumen	Mostrar el índice de ocupación de cada una de bodegas y estantes de una sucursal
Entradas	
Se necesita saber la capacidad de las bodegas y la suma total de los productos dentro de la bodega para poder calcular el índice.	
Resultados	
Se muestra el índice de ocupación de las bodegas de las sucursales.	
RNF asociados	
Persistencia	

TABLA 14 – RFC3

Estante: Esta entidad que está en el modelo conceptual UML es persistente puesto que sus datos son guardados en la base de datos.

Orden_De_Compra: Esta entidad que está en el modelo conceptual UML es persistente puesto que sus datos son guardados en la base de datos.

Orden_De_Compra_Producto: Esta entidad que está en el modelo conceptual UML es persistente puesto que sus datos son guardados en la base de datos.

Proveedor: Esta entidad que está en el modelo conceptual UML es persistente puesto que sus datos son guardados en la base de datos.

Factura: Esta entidad que está en el modelo conceptual UML es persistente puesto que sus datos son guardados en la base de datos.

Compra: Esta entidad que está en el modelo conceptual UML es persistente puesto que sus datos son guardados en la base de datos.

Cliente: Esta entidad que está en el modelo conceptual UML es persistente puesto que sus datos son guardados en la base de datos.

Estante_Producto: Esta entidad que está en el modelo conceptual UML es persistente puesto que sus datos son guardados en la base de datos.

Promocion: Esta entidad que está en el modelo conceptual UML es persistente puesto que sus datos son guardados en la base de datos.

Producto_Compra: Esta entidad que está en el modelo conceptual UML es persistente puesto que sus datos son guardados en la base de datos.

Bodega_Producto: Esta entidad que está en el modelo conceptual UML es persistente puesto que sus datos son guardados en la base de datos.

Producto: Esta entidad que está en el modelo conceptual UML es persistente puesto que sus datos son guardados en la base de datos.

Sucursal_Producto: Esta entidad que está en el modelo conceptual UML es persistente puesto que sus datos son guardados en la base de datos.

4 Punto 4

4.1 Modelo Relacional

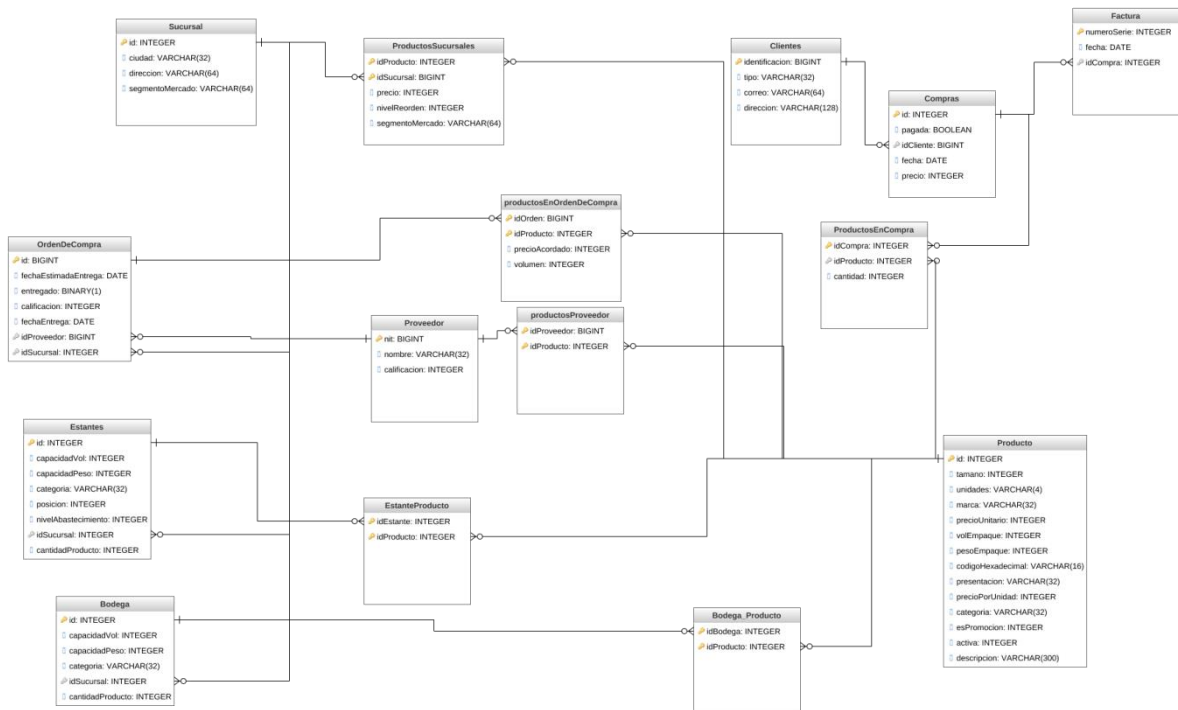


ILUSTRACIÓN 2 - DIAGRAMA RELACIONAL

4.2 Diseño y las decisiones tomadas para crear los elementos de la base de datos

El diseño de nuestra base de datos fue completamente coherente con lo expuesto en el diagrama conceptual, y utilizando la forma de normalización BCNF. Lo anterior, con el fin de garantizar una correcta persistencia de las entidades del diagrama conceptual.

Dentro de las principales decisiones que tomamos al realizar la base de datos fue primero optimizar al máximo el modelo conceptual, buscando que al momento de crear las tablas de la base de datos la normalización no nos generara dificultades en lo que respecta a atributos multivalor y redundancia. Por otro lado, creamos una tabla por cada entidad siguiendo el correcto modelamiento usando los temas vistos en clase para la transformación de diagramas conceptuales a relacionales.

Otra decisión que tomamos con mi compañero antes de proceder el modelamiento en java y la creación de las sentencias de los requerimientos fue realizar distintas pruebas de inserción, actualización y eliminación de datos para verificar que tuviéramos la base de datos en perfecto estado y en BCNF. Con las respectivas pruebas y que realizamos identificamos diferentes fallas como atributos multivalor y dependencias que no nos permitían cumplir con lo requerido para esta entrega es por esto que procedimos a corregir estos, por último, nuestro último objetivo fue realizar una revisión para verificar que con los cambios realizados si respaldáramos la correcta persistencia de todas las entidades planteadas.

4.3 Crear tablas

Para crear las tablas, se utiliza SQL Developer.

5 Construcción de la aplicación, ejecución de pruebas y análisis de resultados

5.1 Carga de datos:

El proceso de carga de datos que seguimos se enumerara el siguiente listado:

1. Plasmamos en un tablero las diferentes tablas que íbamos a crear y a partir de ahí realizamos un análisis para determinar la cantidad de datos que íbamos a introducir y en cuales tablas íbamos a cargar más tuplas.
2. A partir del análisis procedimos a generar archivos .sql y empezamos a crear las tuplas por medio de funciones de Mockaroo que nos permitían variar constantes y valores numéricos a nuestro gusto.
3. Después de haber generado todos los datos y nuestros archivos .sql los guardamos y los descargamos.
4. Realizamos la carga haciendo uso de los scripts .sql en SQL DEVELOPER.
5. Los datos se cargaron de manera correcta logrando el objetivo de cargar una cantidad grade de datos.

5.2 Balance de resultados logrados y no logrados

Logramos realizar la iteración de manera exitosa en un 80% de manera concreta y sencilla. El otro 20 % se encuentra en algunas partes del código las cuales debido a que algunos requerimientos de consulta no logramos que la sentencia nos devolviera un resultado en java. En adición algunas partes del documento como lo fue el extraer el modelo relacional a partir de architecture Enterprise nos fue difícil y por eso no logramos realizar el punto de comparación de modelos relacionales.